Journal of Computing Theories and Applications ISSN:3024-9104

Research Article

Integrating Convolutional Neural Network and Weighted Moving Average for Enhanced Human Fall Detection Performance

Kyi Pyar *

Faculty of Computer Science, University of Computer Studies, Thaton, Myanmar;
e-mail : ky-ipyar@ucstt.edu.mm
* Corresponding Author : Kyi Pyar

Abstract: This study proposes an approach for human fall classification utilizing a combination of Weighted Moving Average (WMA) and Convolutional Neural Networks (CNN) on the SisFall dataset. Falls among elderly individuals pose a significant public health concern, necessitating effective automated detection systems for timely intervention and assistance. The SisFall dataset, comprising accelerometer data collected during simulated falls and activities of daily living, serves as the basis for training and evaluating the proposed classification system. The proposed method begins by preprocessing accelerometer data using a WMA technique to enhance signal quality and reduce noise. Subsequently, the preprocessed data are fed into a CNN architecture optimized for feature extraction and fall classification. The CNN leverages its ability to automatically learn discriminative features from raw sensor data, enabling robust and accurate classification of fall and non-fall events. Experimental results demonstrate the efficacy of the proposed approach in accurately distinguishing between fall and non-fall activities, achieving high classification performance metrics such as accuracy, precision, recall, and F1-score. Comparative analysis with existing methods showcases the WMA-CNN hybrid approach's superiority in classification accuracy and robustness. Overall, the proposed methodology presents a promising framework for real-time human fall classification using sensor data, offering potential applications in wearable devices, ambient assisted living systems, and healthcare monitoring technologies to enhance safety and well-being among elderly individuals.

Keywords: Accelerometer; Convolutional Neural Network; Gyroscope; Human Fall Classification; Weighted Moving Average.

Received: March, 23rd 2024
Revised: May, 14th 2024
Accepted: May, 15th 2024
Published: May, 16th 2024

1. Introduction

Falls are a common occurrence in daily human life. A fall is an unexpected event that can happen during various activities. According to WHO data, approximately 37.3 million falls occur yearly, many requiring serious medical attention. Additionally, WHO estimates that there are about 646,000 fatal falls annually[1], [2]. Moreover, the consideration of falls is the second major cause of accident death after the injury occurred through the accident of traffic in road. Furthermore, the highest death rates occur among adults over 60 aged people. According to the survey, the highest number of fatal falls occurs in older people. The physical capabilities of human life, like sense, vision, and balance, are weakened when life goes through. These events might occur in older people who are more likely to achieve falls, unfortunately [3], [4]. It has many facts concerned with the health issues occurring unfortunately in older people like stroke, vertigo, hypertension, rheumatic, and headache. Another cluster that has a higher incidence of falls is the children. All events require intensive monitoring to eliminate the issue of serious injury by aiding them suddenly after the fall [5], [6].

The challenges in human fall classification primarily stem from the complexity and variability of human motion patterns and the diverse environmental conditions that can occur. Falls can occur in various ways, including forward, backward, sideways, and falls from different heights. Additionally, age, health conditions, and environmental factors can contribute to the variability in fall patterns. This variability makes it challenging to develop a one-size-fits-all approach for fall detection and classification[7], [8]. Non-fall activities, such as sitting down, lying down, or performing daily activities, can sometimes exhibit motion patterns similar to falls. Distinguishing between genuine falls and non-fall activities with similar motion characteristics poses a significant challenge in fall classification. Falls can occur in diverse environmental conditions, including indoor and outdoor settings, lighting conditions, and terrain. Environmental factors such as cluttered environments, obstacles, and uneven surfaces can impact the accuracy of fall detection algorithms [9], [10].

Traditional methods for fall detection often rely on wearable sensors, surveillance cameras, or environmental sensors to collect data, which is then processed using machine learning algorithms to classify human activities and detect falls. Among these algorithms, Convolutional Neural Networks (CNNs) and Weighted Moving Average (WMA) have gained increasing attention for their potential to enhance the accuracy and reliability of fall classification systems. CNNs have demonstrated remarkable performance in image classification tasks and have been successfully applied to various pattern recognition problems [11]–[13]. By automatically learning hierarchical features from raw data, CNNs can effectively extract discriminative features and capture complex patterns inherent in sensor data collected from wearable devices or surveillance systems. Meanwhile, WMA is a well-established technique in time series analysis, commonly used to smooth out fluctuations and identify underlying trends in data. By assigning weights to recent observations, WMA enables the detection of short-term changes while preserving long-term trends, making it suitable for capturing temporal dynamics in human motion data.

The integration of CNNs and WMA presents a promising approach to human fall classification, leveraging the strengths of both methodologies to improve the accuracy and robustness of predictive models. In this paper, we propose to explore the combined use of CNNs and WMA for human fall classification to develop a comprehensive and effective fall detection system. The hypothesis behind combining CNNs and WMA lies in their complementary strengths. While CNNs excel at capturing spatial and temporal patterns in sensor data, WMA provides a mechanism for smoothing out noise and identifying underlying trends. By preprocessing the raw sensor data using WMA to capture temporal dynamics and reduce noise and then feeding the smoothed data into a CNN for further feature extraction and classification, we aim to enhance the accuracy and reliability of our fall classification system. The Sisfall dataset trains the data with the CNN deep learning model. The system implementation is performed for the Sisfall dataset. The contribution of this paper is

1. The convolutional neural network is applied for the human fall classification.
2. WMA was used to remove the noise of sensor data to achieve the most accurate classification results.

The remainder of this document is structured as follows: In Section 2, we explore related literature and background theory pertinent to the subject matter in Section 3. Section 4 discusses the proposed system design, detailing its components and architecture. Following this, Section 5 delineates the methodology employed for performance evaluation, shedding light on the metrics utilized and the experimental setup. Finally, Section 6 encapsulates the key insights drawn from the study and offers concluding remarks on the research endeavor.

2. Related Works

The authors [14] presented a deep-learning approach to automatic human fall detection through frames with the camera. This approach provided the propositions of humans through information to segmentation and body joint locations. The presented ideas were applied to convert multimodal visual presentations to input to FallNet, where the CNN applies multiple and modality-specific layers and uses higher embedding attributes to recognize falls. The human fall dataset contains synthetically created segmentation and human pose data among many camera viewpoints. The performance evaluation of complex public presented about this system provided higher recall and accuracy in fall recognition.

In this study [15], Wi-Sense is introduced as a system for human activity recognition, employing a CNN to discern human actions based on distinctive Wi-Fi channel state information (CSI) patterns that are independent of environmental factors. Initially, Wi-Sense captures CSI data using a standard Wi-Fi network interface card, then employs the CSI ratio method to mitigate noise and phase offset effects, supplemented by principal component analysis to eliminate redundant information. This preprocessing step not only reduces data dimensions but also mitigates environmental influences. Subsequently, processed data spectrograms are computed, revealing environment-independent time-variant micro-Doppler fingerprints corresponding to various activities. These spectrogram images are then utilized for training a CNN model. The effectiveness of the proposed approach is assessed using a dataset of human activities collected from nine participants in an indoor setting, demonstrating an overall accuracy of 97.78% in activity recognition. Additionally, the integration potential of Wi-Sense into existing eHealth infrastructure is discussed, highlighting its relevance within the context of health information systems standards.

The article [16] introduced a novel framework for representing human posture dynamics during falls, termed the 'five-point inverted pendulum model.' It employs an advanced two-branch multi-stage convolutional neural network (M-CNN) to capture and reconstruct the inverted pendulum structure of human posture in complex real-world scenarios. Additionally, the study addresses the temporal continuity of fall events by employing multimedia analytics to track changes in the time-series representation of the human inverted pendulum structure. This approach enables the construction of a spatio-temporal evolution map illustrating the movement patterns of human posture over time. Moreover, by integrating computer vision techniques and multimedia analytics, the study identifies visual characteristics associated with the spatiotemporal evolution of human posture during potentially unstable states. Furthermore, the research investigates two key features of human fall behavior: motion rotational energy and generalized force of motion. Experimental findings in real-world settings demonstrate the method's robustness, broad applicability, and high accuracy in detecting falls.

This study built upon our previous research focusing on designing and implementing a Fall Detection System (FDS) utilizing an inertial measurement unit worn at the waist [17]. The dataset used for analysis is sourced from SisFall, a publicly available repository containing a variety of Activities of Daily Living and fall instances. Initially, we conducted preprocessing and feature extraction procedures on the dataset, followed by applying five distinct Machine Learning algorithms, facilitating a comparative analysis. Notably, ensemble learning techniques such as Random Forest and Gradient Boosting exhibited superior performance, demonstrating Sensitivity and Specificity values nearing 99%. Our primary contribution lies in developing a multi-class classification framework for fall detection, coupled with examining the impact of sensor sampling rates on FDS efficacy. In our multi-class classification approach, falls are categorized into three distinct phases: pre-fall, impact, and post-fall, a novel extension that requires thoughtful consideration. Through extensive experimentation with sampling rates ranging from 1 to 200 Hz, we observed nuanced effects on system performance. While higher sampling rates generally enhance detection accuracy, our findings suggest that a sampling rate of 50 Hz typically suffices for reliable fall detection.

3. Background Theory

3.1. Weighted Moving Average(WMA)

A WMA is a moving average that places a higher emphasis on recent data points in a time series data set [18]. This is achieved by assigning weights to each data point in the series, with the most recent data points assigned the highest weights. The WMA is then calculated by multiplying each data point by its assigned weight, summing the products, and dividing by the sum of the weights. A WMA is a statistical technique used to smooth out a time series data set by giving different weights to different periods in the data set. Unlike a simple moving average, which assigns equal weights to all periods, a WMA assigns higher weights to more recent periods and lower weights to older periods. It is represented in Equation (1).

$$M = \frac{\sum_{t=1}^{n} W_t * V_t}{\sum_{t=1}^{n} W_t} \tag{1}$$

Where M is the average value, V is the actual value, W is the weighting factor, and n is the number of periods in the weighting group.

3.2. Convolutional Neural Network (CNN)

A CNN is a deep, feedforward neural network specifically developed to address the processing demands of complex data[19]. The architecture of CNNs consists of various layers, including convolutional layers, pooling layers, and fully connected layers. These layers are organized sequentially, with the input data fed forward through the network. The final layer typically produces a set of predicted labels or a probability distribution representing possible labels. A CNN comprises essential components, including an input layer, an output layer, and hidden layers encompassing various specialized layers such as pooling, convolutional, normalization, and fully connected layers. The input layer represents input data, such as text documents or images as vectors, initializing the neural network's processing. The pivotal building block within a CNN is the convolutional layer, which generates feature activation maps from the input layer by applying multiple filters. These filters traverse the input layer's height and width, computing dot products between the input layer and filter entries, producing two-dimensional activation maps. These activation maps are subsequently passed to pooling layers for further processing. The stride parameter determines the movement of filters across the input layer, typically set to one.

Additionally, the Rectified Linear Unit (ReLU) activation function is commonly employed within convolutional layers to rectify negative values by converting them to zero. Each convolutional layer within the CNN is characterized by various parameters, including the kernel size, zero padding, stride, input size, and the map stack. Finally, the input signal undergoes processing through the activation function (ReLU), ensuring that the dimensions of the input and output data remain consistent throughout the network's architecture. Equation (2) is used for the ReLU activation function.

$$f(x) = \max(0, x) \tag{2}$$

The pooling layer is utilized in CNNs to reduce the spatial dimensions of the output obtained from the convolutional layers. This serves two main purposes: to decrease the network's computational complexity and help mitigate overfitting. Various types of pooling layers, such as max pooling and average pooling, select the maximum or average value from a group of pixels. The Pooling Layer serves as an intermediary component within CNNs, tasked with diminishing the spatial dimensions of the input data. Unlike the convolutional layer, the pooling layer does not incorporate weights in its operations. Instead, it applies a filter across all input data, facilitating the extraction of essential features without introducing additional weights.

One common pooling technique is maximum pooling, which involves the selection of the pixel with the highest value as the filter traverses the input. This selected pixel is then forwarded to the output array, thereby retaining the most prominent features from the input data while discarding less relevant information. On the other hand, average pooling calculates the average value within the receptive field and sends it to the output array as the filter moves across the input. While this layer provides several benefits to the CNN, it may lead to information loss. However, its advantages, such as noise feature reduction, efficiency improvement, and overfitting prevention, outweigh this limitation.

The fully-connected layer in a CNN resembles the layers found in traditional neural networks. It takes the output from the convolutional and pooling layers and utilizes it to make predictions about the input data. During training, the network learns the optimal weights for the fully connected layers based on the specific task. The fully connected layer in a CNN serves a crucial role akin to the output layer in a multilayer perceptron (MLP). Its primary responsibility is consolidating information gleaned from the final feature maps and producing the final classification output. In this layer, each neuron from the preceding layer is interconnected with every neuron in the current layer. This comprehensive connectivity ensures that all extracted features are effectively integrated to make informed decisions during the classification process. Essentially, the fully connected layer is the ultimate processing stage within the CNN architecture, functioning analogously to fully connected layers in traditional artificial neural networks. The input to this layer is a flattened column vector derived from the previous layers, transforming feature vectors through backpropagation during training over multiple epochs. In the final element of the fully connected layer, an activation function is applied to generate class label predictions, representing the ultimate output of the CNN. Activation functions such as Sigmoid or SoftMax are commonly utilized for this purpose. The role of this final element is to process the aggregated information from the preceding layers and produce meaningful predictions regarding the classification of the input data. The process involves reducing the data dimensionality at the pooling layer to a single dimension and establishing connections with every neuron in the fully connected layer. This comprehensive connectivity allows for integrating relevant features extracted by the convolutional layers, enabling the network to make informed decisions during the classification task. The activation function employed in the final layer is crucial for classification tasks. In this context, the sigmoid activation function is often utilized to map the network's output to a probability distribution, facilitating the classification of input data into distinct classes. This activation function is characterized by Equation (3).

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{3}$$

For the sigmoid activation function, the output is between 0 and 1. If the output value is near 0, the output classification label is 'Fall'; otherwise, it is 'Not Fall'.

4. Proposed Method

This system aims to develop the human fall classification system using a CNN model. This system classifies the fall or non-fall from the motion sensors' data. We extract automatically learned features from the accelerometer and gyroscope sensor data and provide them to a classification algorithm in this system. Figure 1 describes the design of the system.

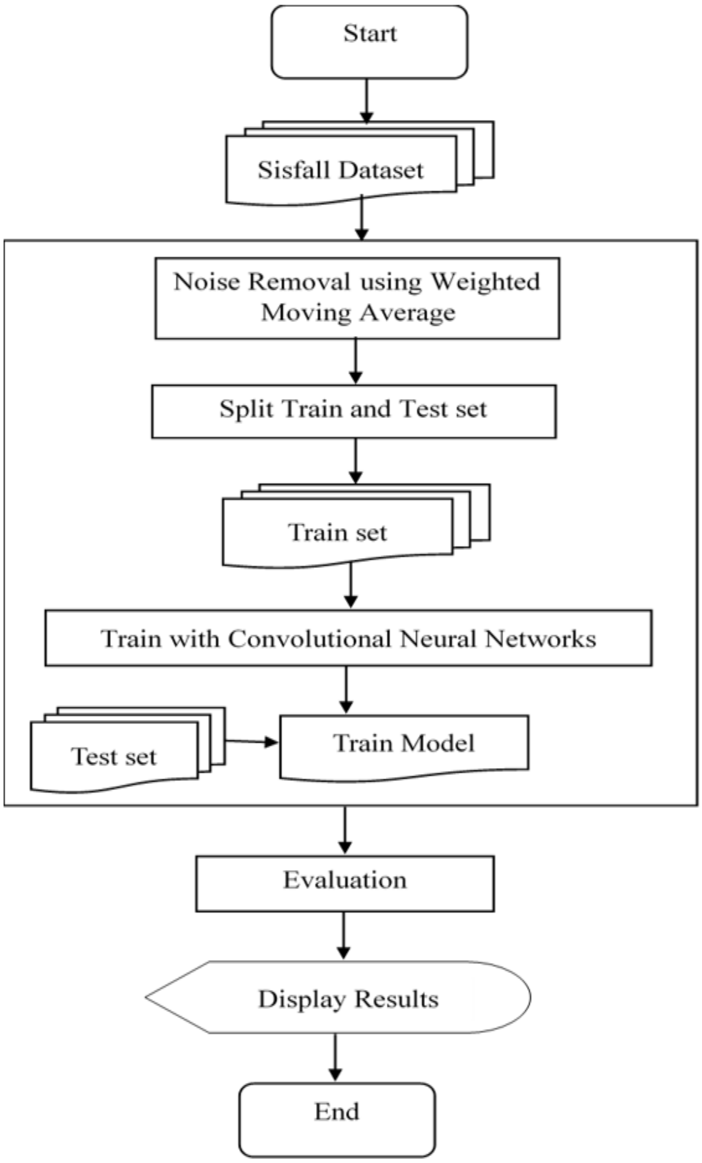

Figure 1. System Design

4.1. Data Collection

In this system, the dataset for fall/not fall detection, the SisFall dataset, is employed [20]. The SisFall dataset is a widely used benchmark dataset in human fall detection and classification. It was developed by researchers at the University of Granada, Spain, to facilitate research and development in fall detection algorithms. The dataset consists of accelerometer and gyroscope data collected from wearable sensors worn by participants performing various activities, including falls and activities of daily living. The data was collected from 15 healthy volunteers, including males and females, aged between 23 and 55. Participants wore a smartphone containing a built-in accelerometer and gyroscope sensors while performing activities in a controlled indoor environment. The dataset includes data from four different types of activities:

Falls: Participants simulated different types of falls, including forward falls, backward falls, and lateral falls.

Activities of Daily Living (ADLs): Participants performed a range of common daily activities, such as walking, sitting, standing, lying down, and picking up objects from the floor.

The dataset includes data from the accelerometer and gyroscope sensors, capturing linear acceleration along three axes (x, y, z) and angular velocity around three axes (x, y, z), respectively. Magnetometer data may also be included in some dataset versions, providing information about sensor orientation. The SisFall dataset typically contains two classes: falls and activities of daily living (ADLs). Falls encompass various types of simulated falls, while ADLs include a range of common daily activities. The SisFall dataset contains 1048576 fall types and 1048576 not fall types. All the activities are recorded at a sampling rate Fs = 200 Hz using a wearable device mounted at the waist of the participant, having three motion sensors i.e., two accelerometers and one gyroscope. These motion sensor data contain noise, so a WMA is applied to remove noise. After noise removal, this data is classified into fall and not fall using a CNN. Figure 2 shows different features in the dataset. The definition of 'Fall' refers to the interval between 'Not Fall' status that changes to 'Lying' status. Regardless of whether the action taken is sitting or walking, this interval belongs to the 'Fall' class. Any status that does not fall into 'Fall,' belongs to the 'Not Fall' class.

accelerometer			gyroscope			accelerometer		
ADXL345_x	**ADXL345_y**	**ADXL345_z**	**ITG3200_x**	**ITG3200_y**	**ITG3200_z**	**MMA8451Q_x**	**MMA8451Q_y**	**MMA8451Q_z**
5	-234	-82	37	4	-7	9	-959	-319
2	-234	-87	35	4	-7	8	-964	-319
6	-234	-84	35	3	-7	11	-962	-323
5	-234	-85	34	4	-8	7	-961	-323
7	-237	-83	34	4	-7	9	-959	-323
-2	-238	-77	35	5	-7	8	-968	-328
7	-237	-83	35	4	-8	7	-963	-322
4	-233	-84	36	4	-7	4	-963	-324
4	-233	-79	35	5	-6	8	-963	-325

Figure 2. Sample different features in dataset.

4.2. Noise Removal

. To ensure data quality, noise is removed from the dataset using a WMA technique. This dataset typically contains accelerometer and gyroscope data recorded during various activities, including falls. The relevant sensor data is extracted from the dataset. This usually includes three-axis accelerometer data and, optionally, gyroscope data. The sensor data is analyzed to understand the types of noise present. This can include sensor inaccuracies, environmental vibrations, and motion artifacts. The noise characteristics in the dataset, such as frequency, amplitude, and duration, are identified. Understanding these characteristics helps in designing an effective noise removal strategy. A suitable window size is chosen for the moving average filter. The window size determines the number of data points included in each calculation. The window size is 3, and each weighted average will be calculated based on the current value, one previous value, and one next value. A set of weights is designed to correspond to each data point within the window. Typically, weights decrease as the distance from the current point increases, giving more importance to recent data. Iterate over the sensor data, applying the moving average filter to each data point. This involves calculating the weighted average of the data points within the window for each timestamp. Update the sensor data with the filtered values, replacing the original noisy data.

4.3. Classification

Subsequently, the preprocessed data is classified into fall and not fall categories using a CNN model. Our CNN architecture is meticulously crafted to address the critical task of human fall classification. Drawing upon a blend of theoretical insights and iterative experimentation, we meticulously design our model's convolutional, pooling, and fully connected layers. These layers are pivotal in extracting salient features from input data and performing accurate classification. The model undergoes compilation, employing the Adam optimizer with a learning rate of 0.01 and a batch size of 64 to enhance its performance and efficiency. Table 1 describes the proposed CNN model design.

Table 1. CNN Model Design.

Layer	Parameter
Convolutional layer	Filters=512, kernel_size=2, activation='relu', input_shape=(shape,1)
Average pooling layer	Pool_size=2
Convolutional layer	Filters=256, kernel_size=2, activation='relu'
Average pooling layer	Pool_size=2
Convolutional layer	Filters=128, kernel_size=2, activation='relu'
Average pooling layer	Pool_size=2
Convolutional layer	Filters=64, kernel_size=2, activation='relu'
Average pooling layer	Pool_size=2
Flattened layer	None
Dense layer	Units=256 and activation='relu'
Dropout layer	0.5
Dense layer	Units=1 and activation='sigmoid'

The CNN model uses Keras, a high-level neural network API on TensorFlow. The architecture of the model involves transforming the SisFall dataset into a 1-dimensional convolutional architecture. Since the dataset comprises non-numeric and numeric features, converting them into a numeric matrix suitable for input into the CNN is essential. This conversion is facilitated using techniques like one-hot encoding, which is particularly useful for handling categorical features and avoids issues related to categorical conversion to integers. The CNN architecture consists of 512 kernels with a dimension of 1x2 and includes four convolutional layers. An average pooling layer and a sigmoid activation function follow each convolutional layer. The pooling size is set to 2 for all pooling layers, except for the first one, which is set to 2. The sigmoid activation function is employed at the fully-connected layer of the network.

5. Implementation and Discussion

In this system, the task involves classifying fall versus non-fall instances utilizing the SisFall dataset obtained from Kaggle. This dataset is specifically curated through the collection of motion sensor data. A WMA technique is employed to remove any existing noise to enhance the dataset's quality. Subsequently, the cleaned dataset is classified using CNN. The CNN model is developed and trained using Keras, which operates on the TensorFlow framework. The dataset is divided into training and testing sets in a 75%-25% ratio to ensure a comprehensive evaluation. Specifically, 75% of the records are randomly selected for training, while the remaining 25% are reserved for testing purposes. The performance of the proposed system is evaluated using key metrics such as accuracy, recall, f-measure, and precision. These metrics provide valuable insights into the system's effectiveness in accurately classifying fall and not-fall instances.

Furthermore, comparative analysis is conducted to assess the performance of the proposed model against alternative approaches. Figure 3 presents a comparative overview of the performance results obtained from the proposed model employing WMA and CNN and employing only CNN. Additionally, Figure 4 illustrates a graphical comparison of the accuracy achieved by the proposed model using these different methodologies.

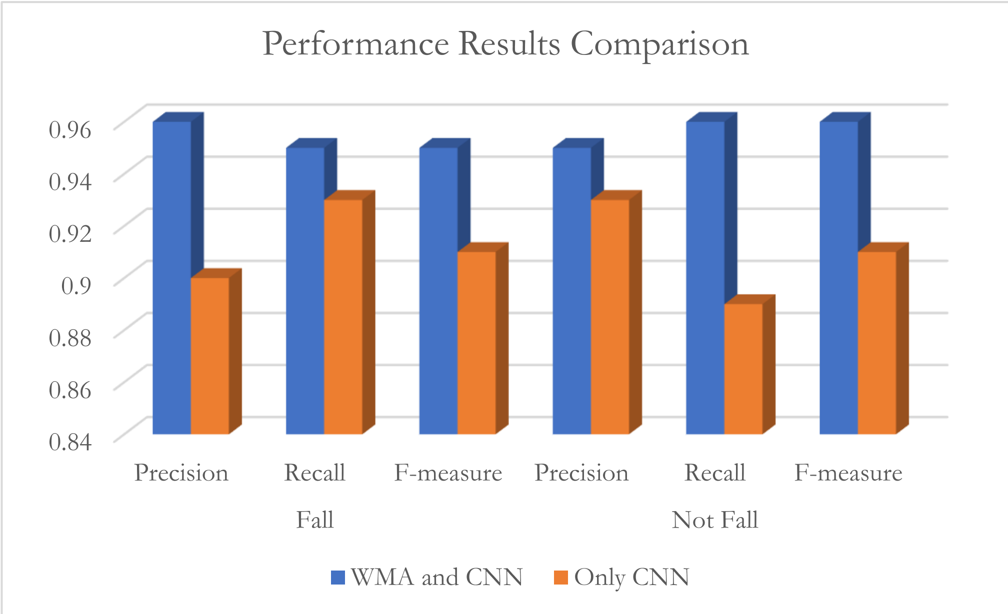

Figure 3. Performance Results

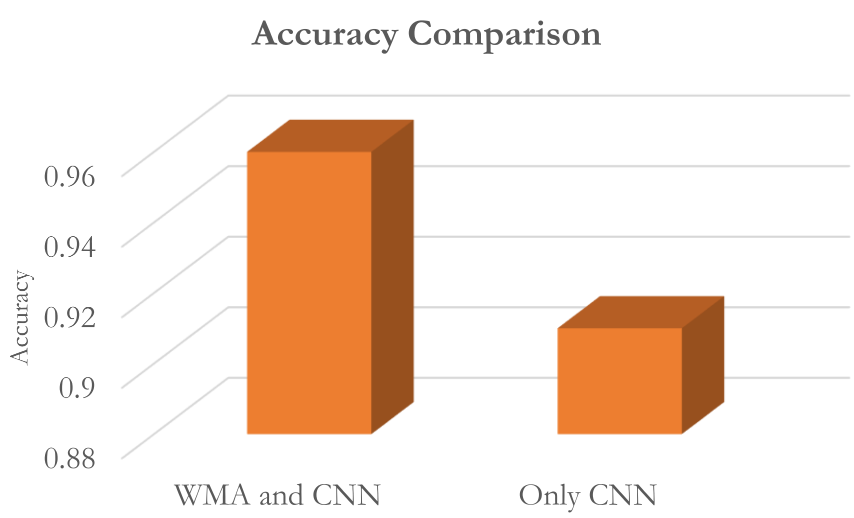

Figure 4. System Accuracy

6. Conclusions

This paper proposes a fall classification method using wearable sensor data and a supervised learning approach based on a deep neural network. The proposed network has been devised by deriving a time series from wearable sensor data and feeding it to a deep CNN to learn multi-level features from wearable sensor time series data automatically. The results will demonstrate that the proposed fall classification method with accuracy, f-measure, precision, and recall indicates the capacity to detect a fall better. According to the evaluation results, this proposed system achieves an accuracy of 96% with WMA+CNN and 76% with only CNN.

Funding: This research received no external funding.

Conflicts of Interest: The author declare no conflict of interest.

References

[1] R. Tanwar, N. Nandal, M. Zamani, and A. A. Manaf, "Pathway of Trends and Technologies in Fall Detection: A Systematic Review," *Healthcare*, vol. 10, no. 1, p. 172, Jan. 2022, doi: 10.3390/healthcare10010172.
[2] A. Bhattacharya and R. Vaughan, "Deep Learning Radar Design for Breathing and Fall Detection," *IEEE Sens. J.*, vol. 20, no. 9, pp. 5072–5085, May 2020, doi: 10.1109/JSEN.2020.2967100.
[3] K. Saho, S. Hayashi, M. Tsuyama, L. Meng, and M. Masugi, "Machine Learning-Based Classification of Human Behaviors and Falls in Restroom via Dual Doppler Radar Measurements," *Sensors*, vol. 22, no. 5, p. 1721, Feb. 2022, doi: 10.3390/s22051721.
[4] L. Wang, M. Peng, and Q. Zhou, "Pre-Impact Fall Detection Based on Multi-Source CNN Ensemble," *IEEE Sens. J.*, vol. 20, no. 10, pp. 5442–5451, May 2020, doi: 10.1109/JSEN.2020.2970452.
[5] C.-B. Lin, Z. Dong, W.-K. Kuan, and Y.-F. Huang, "A Framework for Fall Detection Based on OpenPose Skeleton and LSTM/GRU Models," *Appl. Sci.*, vol. 11, no. 1, p. 329, Dec. 2020, doi: 10.3390/app11010329.
[6] Y. Luo, Z. Ou, T. Wan, and J.-M. Guo, "FastNet: Fast high-resolution network for human pose estimation," *Image Vis. Comput.*, vol. 119, p. 104390, Mar. 2022, doi: 10.1016/j.imavis.2022.104390.
[7] M. T. Pourazad *et al.*, "A Non-Intrusive Deep Learning Based Fall Detection Scheme Using Video Cameras," in *2020 International Conference on Information Networking (ICOIN)*, Jan. 2020, pp. 443–446. doi: 10.1109/ICOIN48656.2020.9016455.
[8] G. Sun and Z. Wang, "Fall detection algorithm for the elderly based on human posture estimation," in *2020 Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)*, Apr. 2020, pp. 172–176. doi: 10.1109/IPEC49694.2020.9114962.
[9] X. Wang, J. Ellul, and G. Azzopardi, "Elderly Fall Detection Systems: A Literature Survey," *Front. Robot. AI*, vol. 7, Jun. 2020, doi: 10.3389/frobt.2020.00071.
[10] H. Liu *et al.*, "Fast Human Pose Estimation in Compressed Videos," *IEEE Trans. Multimed.*, vol. 25, pp. 1390–1400, 2023, doi: 10.1109/TMM.2022.3141888.
[11] E. Casilari, R. Lora-Rivera, and F. García-Lagos, "A Study on the Application of Convolutional Neural Networks to Fall Detection Evaluated with Multiple Public Datasets," *Sensors*, vol. 20, no. 5, p. 1466, Mar. 2020, doi: 10.3390/s20051466.
[12] H. Sadreazami, M. Bolic, and S. Rajan, "Fall Detection Using Standoff Radar-Based Sensing and Deep Convolutional Neural Network," *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 67, no. 1, pp. 197–201, Jan. 2020, doi: 10.1109/TCSII.2019.2904498.
[13] K. P. Kamble, S. S. Sontakke, H. Donadkar, R. Poshattiwar, and A. Ananth, "Fall Alert: A Novel Approach to Detect Fall Using Base as a YOLO Object Detection," in *Advanced Machine Learning Technologies and Applications*, 2021, pp. 15–24. doi: 10.1007/978-981-15-3383-9_2.
[14] G. P. Reddy and M. K. Geetha, "Video based fall detection using deep convolutional neural network," *Eur. J. Mol. Clin. Med.*, vol. 7, no. 11, 2020.
[15] M. Muaaz, A. Chelli, M. W. Gerdes, and M. Pätzold, "Wi-Sense: a passive human activity recognition system using Wi-Fi and convolutional neural network and its integration in health information systems," *Ann. Telecommun.*, vol. 77, no. 3–4, pp. 163–175, Apr. 2022, doi: 10.1007/s12243-021-00865-9.
[16] J. Zhang, C. Wu, and Y. Wang, "Human Fall Detection Based on Body Posture Spatio-Temporal Evolution," *Sensors*, vol. 20, no. 3, p. 946, Feb. 2020, doi: 10.3390/s20030946.
[17] N. Zurbuchen, A. Wilde, and P. Bruegger, "A Machine Learning Multi-Class Approach for Fall Detection Systems Based on Wearable Sensors with a Study on Sampling Rates Selection," *Sensors*, vol. 21, no. 3, p. 938, Jan. 2021, doi: 10.3390/s21030938.
[18] CFI Team, "Weighted Moving Average (WMA)," *corporatefinanceinstitute.com*. https://corporatefinanceinstitute.com/resources/equities/weighted-moving-average-wma/ (accessed Mar. 23, 2024).
[19] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.
[20] https://www.kaggle.com/datasets/moonesmobaraki/sisfall-dataset-with-lableadl-or-fall