Multi-Sensor Fusion for Robust Fall Detection and Classification: A Deep Learning Approach with Missing Modality Adaptation

by

Zhaksylyk Chalgimbayev

Submitted to the Department of Data Science
in partial fulfillment of the requirements for the degree of

Master of Science in Data Science

at the

NAZARBAYEV UNIVERSITY

June 2025

Author .....
Department of Data Science
11.05.2025

Certified by .....
Adnan Yazici
Chair, Professor
Thesis Supervisor

Accepted by .....
Yelyzaveta Arkhangelsky
Dean, School of Engineering and Digital Sciences

Multi-Sensor Fusion for Robust Fall Detection and Classification: A Deep Learning Approach with Missing Modality Adaptation

by

Zhaksylyk Chalgimbayev

Submitted to the Department of Data Science
on 11.05.2025, in partial fulfillment of the
requirements for the degree of
Master of Science in Data Science

Abstract

Falls are a major concern for the well-being of elderly and disabled individuals. Timely detection of falls could play a crucial role in preventing severe consequences of these accidents. This study proposes a deep learning-based multi-sensor fusion framework that integrates camera, wearable sensor data as well as other sensors. To allow a uniform CNN-based pipeline for both camera and sensor data, the methodology involves the conversion of 1D sensor data into 2D Recurrence Plot images. In total, four models that utilize either feature-level or input-level fusion strategies were trained and evaluated on the publicly available fall dataset. All the models were trained using a windowing approach with overlapping segments for potential real-world usability. Two of the models were trained following the multi-task learning approach, meaning apart from fusion heads, sensor based independent training was employed. Results of the performed experiments reveal that A2 was a slightly better performing binary and multiclass classification model, with all the important metrics being above 99%. On the other hand, models that were implemented using the multi-task learning approach did not demonstrate a significantly higher resiliency to missing modality scenarios. This study was able to achieve robustness to missing data without significant performance sacrifices. The main aim of this study was to contribute to the development of sensor-agnostic networks that could also be potentially used in real-time scenarios.

Thesis Supervisor: Adnan Yazici
Title: Chair, Professor

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

For the elderly population, falls pose a great threat to their health and independence. According to the World Health Organization, the annual portion of people aged above 65 who experience falls is roughly 28-35% [20]. The statistics for the elderly aged above 70 years increase to the range of 32-42% [20]. Every year, there are about 3 million emergency department visits associated with elderly people falling, and roughly 1 million elderly hospitalizations caused by falls [21]. Moreover, the fact that falls are the most common cause of traumatic brain injuries (TBI) makes the timely detection of such accidents vital [22]. Furthermore, the timely detection of falls is important to prevent situations in which individuals struggle to get up for an extended amount of time. This prolonged state in which an individual remains on the ground could result in unpleasant conditions including dehydration, hypothermia, and could even lead to lethal outcomes [23]. All the above, make the timely detection of falls of paramount importance.

In recent years, the rapid development of sensor technologies combined with the advancements in the artificial intelligence (AI) field, allowed the emergence of various robust Human Activity Recognition (HAR) and fall detection solutions. Different research studies developed frameworks that utilize machine learning and deep learning techniques to accurately classify readings from sensors. These sensors are typically

used to gather data from various modalities, including inertial (e.g., accelerometer, gyroscope, etc.), audio, and video data. One of the most extensively used data modalities for fall detection and prediction is video data [24]. The reason for such high rates of usage could be related to the non-intrusive nature of the sensors that collect video data. In addition, 2D cameras are cost-effective and more prevalent, especially compared to the 3D cameras that capture depth information.

This study aims to develop a robust deep learning based solution that is capable of detecting fall events based on multimodal data acquired from video and inertial sensors. The proposed approach involves several key preprocessing steps to derive uniform 2-dimensional images from each sensor. The derived images will then be fused at the input level or, at the feature level, after the feature extraction operation performed by CNN branches, depending on the network architecture. The approach will also incorporate a fusion head that will classify the feature vector at the final step.

First of all, the study proposes several comprehensive deep learning models that not only use multimodal data for the tasks of fall detection and fall prediction, but also perform well despite the unavailability of certain data modalities during inference.

Objectives of the Study

- Leverage multimodal data to implement a robust deep learning-based fall detection solution

- Apply feature-level as well as input-level data fusion techniques to combine data from different sensors for enhanced model performance

- Incorporate modality dropout technique to ensure model resiliency to missing data modalities

- Compare the developed model(s) with the existing models that used similar approach

The rest of the paper is organized as follows: Section 2 describes the previous work that has been done in this field, as well as general background information. Section 3 explains the proposed approach in fine detail. Section 4 discusses the results of the performed experiments and summarizes key takeaways.

Chapter 2

Related works

HAR and Fall detection tasks have gained increased attention from the research community over the course of the last decade. Numerous research endeavors were carried out to produce innovative approaches to solve HAR and fall detection tasks using different sets of data modalities. Some researches focus on inertial sensor-based classification leveraging data acquired from sensors like accelerometers and gyroscopes [3]. Vision-based approaches are also prevalent due to the amount of information that could be extracted from video/image data [4]. Audio-based [5] and even WiFi-based [6] HAR were proven to be feasible. It is also a common practice, in the context of HAR research, to combine multiple data types in order to boost the robustness of the proposed framework. These frameworks are referred to as "multimodal" approaches to solve HAR-related tasks.

Classical machine learning-based techniques were extensively employed for HAR and fall detection tasks. For instance, Yazici et al. (2023) used the Random Forest algorithm to classify a data stream from inertial sensors (e.g., accelerometer, gyroscope, etc.) into "fall" or "no fall" classes, achieving roughly 99% accuracy on the MHEALTH dataset [7]. Wang et al. (2020) have also utilized Random Forest for fall detection using dynamic and static features of the human body which were extracted using a dual-channel sliding window model [8]. Taghvaei et al. (2017) used a Hidden Markov Model in conjunction with an Autoregressive-Moving-Average model for the fall detection task [9]. In their comparative study, Zerrouki et al. (2016), ex-

plored the performance of several widely used machine learning algorithms in terms of metrics such as average accuracy and Area Under Curve (AUC) values on two distinct fall detection datasets and identified Support Vector Machine (SVM) to be the best-performing model [10].

Following their emergence as a breakthrough in the field of AI, Deep Learning techniques were incorporated to solve tasks from a myriad of domains. The field of HAR also benefited from Deep Learning methods due to their ability to handle large and complex data in its raw form, eliminating the need for extensive data pre-processing steps. In [7], a vanilla Convolutional Neural Network (CNN) was trained on image frames extracted from the DMLSmartActions dataset to identify falls. Their proposed network achieved an accuracy of 86.97% on the test set of the aforementioned dataset. Lu et al. (2018) proposed a 3D CNN-based feature extractor and trained it on a popular Sports-1M dataset before combining it with the Long Short-Term Memory (LSTM) model for temporal information [11]. A similar combination of deep learning models was proposed by Islam et al. (2023), they used a CNN with a special Convolution Block Attention Module (CBAM) for feature extraction purposes from the visual data in combination with a Convolutional LSTM to capture temporal information from multimodal data [12]. In 2020, Ihianle et al. (2020) developed a framework that consisted of CNN and Bidirectional LSTM networks for multimodal HAR multiple wearable sensor data [13].

Xie et al. [14] incorporated input-level fusion to perform binary (fall vs no fall) classification on the UP-FALL dataset [1]. Their pipeline involved the extraction of 2D skeleton data and the subsequent keypoints fusion. However, their model is not adapted for potential real-time usage and most importantly does not utilize multimodal data and relies solely on camera data. In contrast, Cai et al. [15] opted for combining multiple data modalities such as acceleration and human skeleton via feature-level fusion to train a GBDT for a fall detection task. Xu et al. [16] and Yang et al. [17] also applied feature-level fusion to combine manually extracted features and perform fall detection tasks. And obvious downside of these studies is that the necessary features have to be selected by an expert. This study [18] employed graph

convolutional networks alongside a 1D CNN to generate spatiotemporal kinematic gait features. After that, they applied concatenative feature fusion and trained the models to detect falls. The limitation here is that it only relies on a single modality - video. Similarly, [19] only relies on video modality while using decision-level fusion for fall detection. Pian et al. [2] implements fall detection models via input-level fusion by stacking multiple sensor modalities as channels after converting 1D signals into images using Gramian Angular Field method. This study's main limitation is related to the model's usability for potential real-time cases as well as the absence of resiliency to missing data modalities during inference.

This study proposes several fall detection and classification models that are not only multimodal but are also potentially applicable for real-time usage, are capable of classifying with any subset of modalities during inference, and that do utilize both feature-level and input-level fusion strategies. The key differences between our proposed approach and other previous papers are summarized in Table 2.1

Table 2.1: Comparison of Related Works

Study	**Multimodal**	**Real-Time**	**Fusion Type**	**Robustness to Missing Sensors**
[14]	✗	✗	Input Level	✗
[15]	✓	✓	Only Feature Level	✗
[16]	✓	✓	Only Feature Level	✗
[17]	✗	✓	Only Feature Level	✗
[18]	✗	✓	Only Feature Level	✗
[19]	✗	✗	Only Decision Level	✗
[4]	✓	✗	Input Level	✗
Our Study	✓	✓	Input + Feature Level	✓

Chapter 3

Methodology

3.1 Dataset

This study uses a publicly available dataset called UP-FALL [1]. The dataset includes readings from multiple data modalities and is designed for fall detection/classification and HAR tasks. The dataset collection process involved recording 17 subjects performing 11 activities in 3 trials on multiple wearable and environmental sensors.

The data modalities present in the dataset belong to three main categories: vision-based, wearable-inertial, and environmental data. Infrared sensors and 2D cameras make up the vision-based sensors, while the wearable-inertial data was collected using accelerometers, gyroscopes and luminosity sensors. The latter group of sensors was attached to the subjects' ankles, belts, necks, right pockets, and wrists. Additionally, EEG sensors were worn by the subjects for brain activity data while performing the activities. The detailed figure with the sensor configuration and experimental setup can be seen in Figure 3-1

With regards to the activities, there are 5 types of falls, such as "Falling forwards using hands", "Falling forwards using knees", etc. The remaining 6 activities are different Activities of Daily Living (ADLs) such as "Walking", "Standing", etc. The full list of activities and their duration in seconds can be seen in Table 3.1.

The data belonging to subjects number 5 and 9 were excluded from the dataset due to corrupt and missing data. Furthermore, the dataset was partitioned into

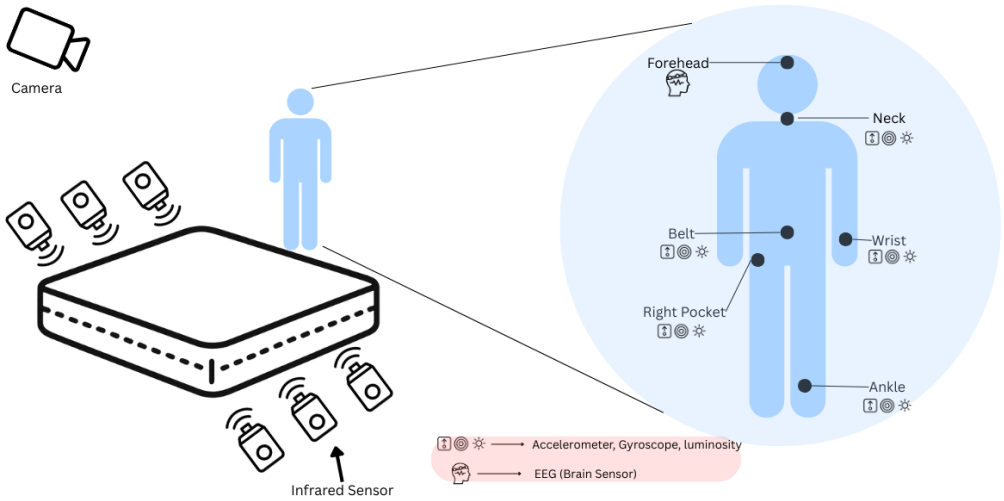

Figure 3-1: Dataset collection setup

train, test, and validation subsets according to the following strategy: trial 3 data of all the subjects was only used for inference to assess how the models will perform on completely unseen data. The data from trials 1 and 2 were used for training and validation, with validation consisting of 2 random subjects' data, while training included the trial 1 and 2 data of the remaining subjects.

Additionally, the best performing models were further tested following the leave-one-subject-out (LOSO) strategy, in which all the trial data available for N-1 subjects is used for training, and all the trials of the remaining 1 subject are used purely for testing. The process is performed iteratively while isolating every single subject for testing, and the average of performance metrics was recorded to analyze the performance of the models on completely unseen individuals.

3.2 Data Preprocessing

3.2.1 Video Data Preprocessing

Originally, the dataset stores video data as a sequence of frames taken at an 18Hz frame rate. For the benefits related to storage and computational cost reductions, as

Table 3.1: List of Activities in the UP-FALL Dataset

Activity No.	**Activity Name**	**Duration (s)**
1	Falling forward using hands	10
2	Falling forward using knees	10
3	Falling backward	10
4	Falling sideways	10
5	Falling sitting in empty chair	10
6	Walking	60
7	Standing	60
8	Sitting	60
9	Picking Up an Object	10
10	Jumping	30
11	Lying Down	60

well as motion information capturing, the "Summed Difference Image" technique was employed. The selected approach involves the calculation of the absolute difference between consecutive frames. As a next step, the addition of the differences is performed over a specified window (for this study, 10s windows were chosen). The given preprocessing technique helps to focus on motion patterns across the frames while effectively eliminating any noise and static parts of the video.

For every sliding window of 10 seconds or 180 frames, the Summed Difference Image is calculated as:

$$SDI = \sum_{i=1}^{N-1} |I_{i+1} - I_i|$$

where I_i is the normalized video frame at index i, and N is the total number of frames in a given window.

The image normalization is performed using the min-max technique:

$$I_{norm} = \frac{I - I_{min}}{I_{max} - I_{min}} \times 255$$

As a final step, the obtained images are rescaled to 128x128, which helps keep the computational complexity low, compared to some of the previous research papers [2] which opted for higher resolution options. This step is performed using the OpenCV library, which provides comprehensive downscaling options that ensure key detail

preservation.

An example image (prior to downscaling) as a result of this preprocessing pipeline could be seen in Figure 3-2

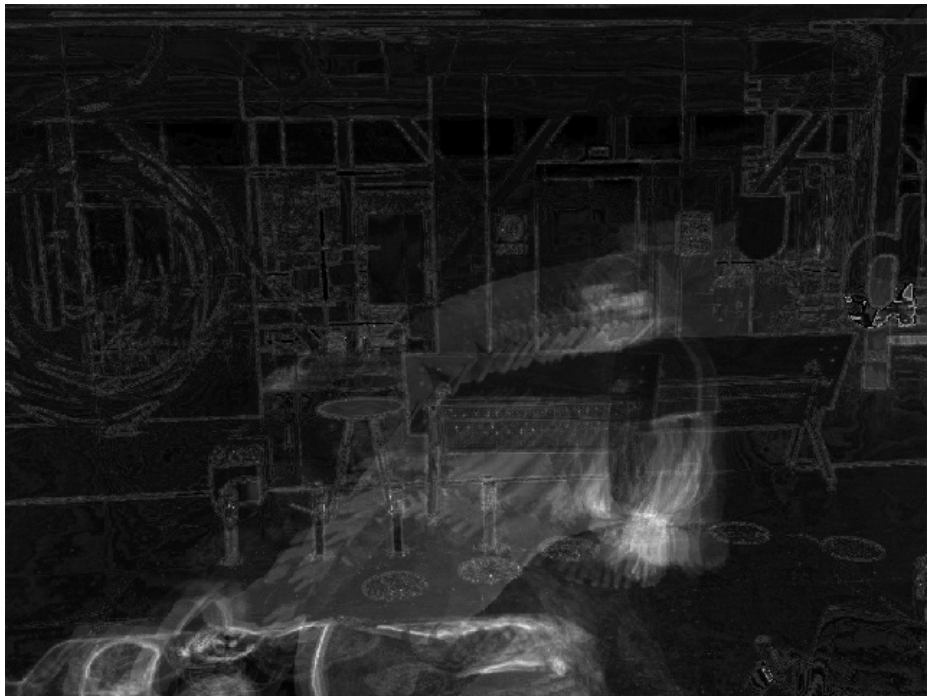

Figure 3-2: Example of a summed difference image computed over a 10-second window.

3.2.2 Sensor Data Preprocessing

The raw data collected from the Inertial Measurement Units (IMUs), infrared sensors, and a brain sensor also underwent certain preprocessing steps. First of all, the 1-dimensional signal magnitudes were transformed into 2-dimensional Recurrence Plot (RP) images, which allows time-series data to be suitable for CNNs.

The Recurrence Plots technique is crucial for capturing the self-similarity of the time-series data along the time axis and most importantly, it highlights important dynamical structures of the data.

It could be defined with the given formula:

$$R_{i,j} = \begin{cases} 1, & \text{if } \|x_i - x_j\| < \epsilon \\ 0, & \text{otherwise} \end{cases}$$

where $x(t)$ is a time-series data and ϵ is a threshold.

The RP image generation was done with the help of the pyts library. Furthermore, the same normalization, resolution, and windowing techniques as in video data preparation were used for the sensor data preparation. An example image (prior to downscaling) as a result of this preprocessing pipeline could be seen in Figure 3-3

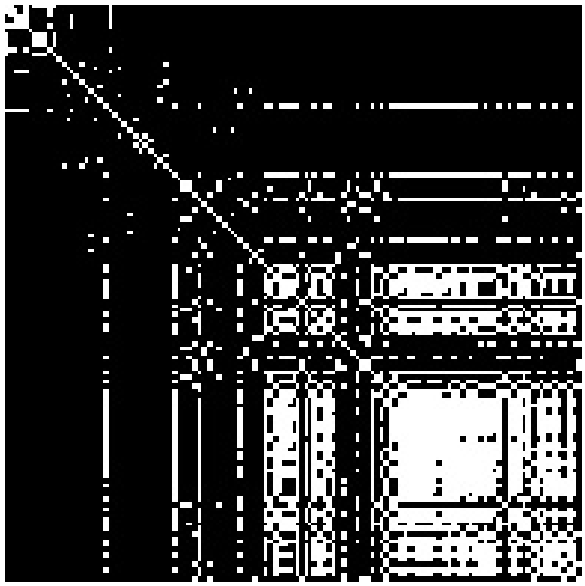

Figure 3-3: Example of a recurrence plot generated from accelerometer data.

3.3 Proposed Multi-Sensor Model Architectures

This study proposes multiple robust CNN-based architectures that utilize data from all the available modalities of the dataset to perform fall detection as well as classification tasks. These architectures consume 2D images (Summed Difference Images and Recurrence Plot images) since such consistency of input shapes across the modalities greatly facilitates feature fusion for both feature-level and input-level fusion strate-

gies. In other words, instead of multiple 1D and 2D networks, it allows the utilization of a single CNN infrastructure and treating different modalities as simply different CNN branches.

3.3.1 Overview of Approaches

This study proposes 4 architectures in total, which can be divided into 2 approaches: (A) Single fused head and (B) multi-task heads. For each of the aforementioned approaches, 2 different architectures were created: (1) a branch per sub-sensor and (2) a location-based multi-channel branch.

Approach A: Single Fused Head

In this approach, feature-level fusion via concatenation is applied to the features extracted by each of the CNN branches before being fed to a classification head. During training, a single loss function is used for the entire network: cross-entropy loss for binary classification tasks (A1 and A2), and focal loss with class weights for multiclass classification tasks (A1 and A2).

The diagram of the "One Branch per Sub-Sensor" network implemented using Approach A could be seen in Figure 3-4. In essence, there is a dedicated separate branch for every single modality, which allows fully independent sensor representations and easier handling of the missing data modalities during inference.

The diagram of the "Location-Based Channel Stacking" network implemented using Approach A can be seen in Figure 3-5. In this network, both input-level and feature-level fusion techniques are being used. Instead of dedicating a separate CNN branch for every single data modality, sensors that are attached to the same body part of the subject are fused via channel stacking and fed to the shared CNN branch. For instance, there is only a single branch reserved for the "Ankle" sensors, and it receives 3-channel input, which is composed of: AnkleAcceleration, AnkleAngularVelocity, and AnkleLuminosity images stacked together as 3 different channels. Lastly, feature concatenation is performed at the final fused head. As a result, the network has fewer

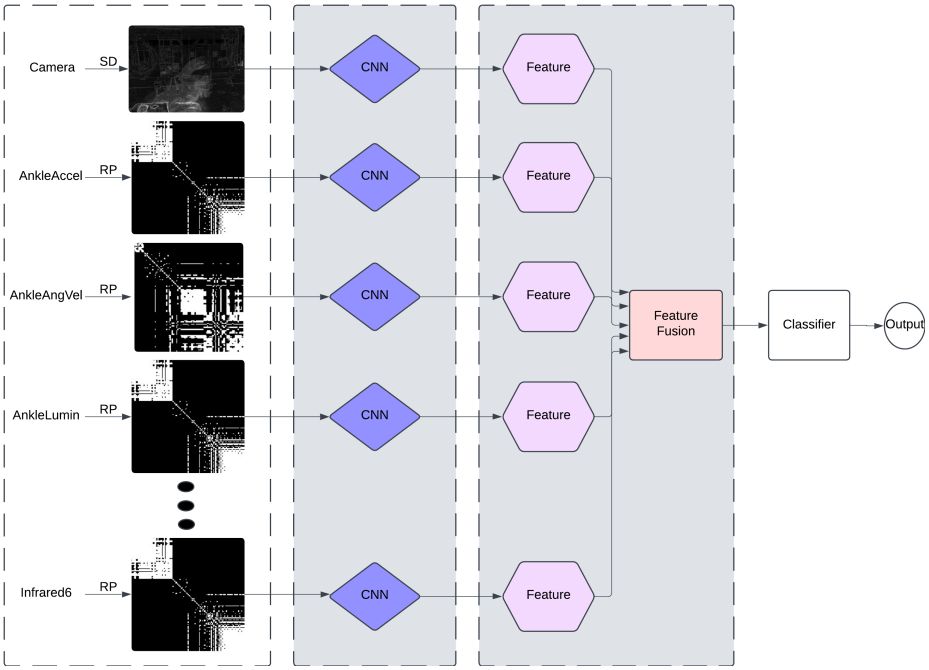

Figure 3-4: Approach A: Multi-Branch Single Fused Head (One Branch per Sub-Sensor)

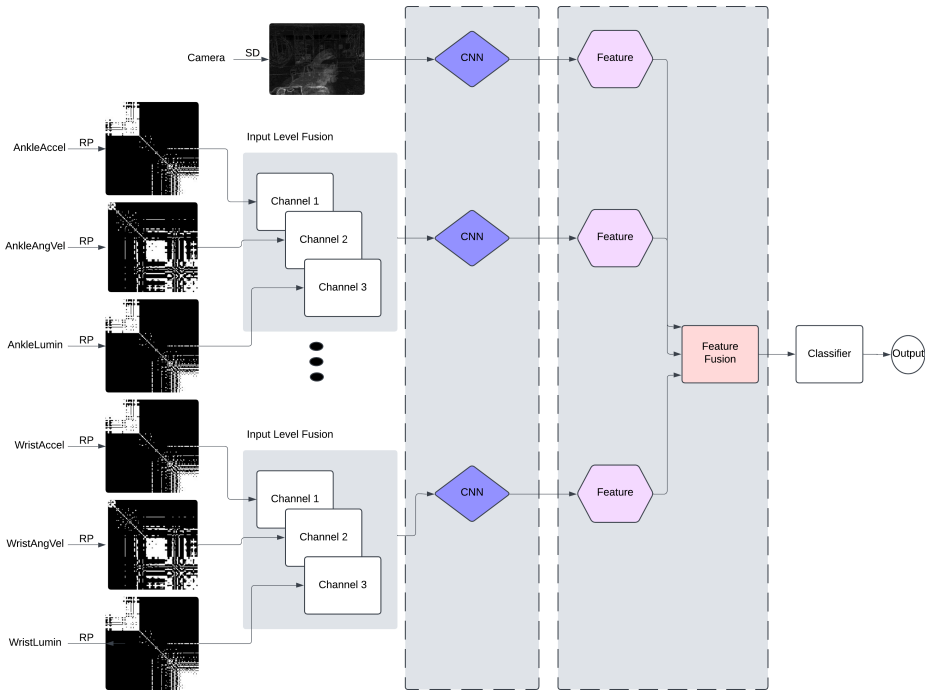

Figure 3-5: Approach A: Multi-Branch Single Fused Head (Grouped by Top-Level Folder)

total branches and is lighter, while sacrificing the level of robustness of the previous network.

Approach B: Multi-Task (Multi-Output)

In comparison to Approach A, in this multi-task approach, apart from the fused head, each CNN branch has a classification head of its own. All these classification heads of the networks that were implemented using this approach were optimized using the following loss function:

$$\text{Total Loss} = \sum_{\text{branch}} w_{\text{single}} \cdot L_{\text{branch}} + w_{\text{fused}} \cdot L_{\text{fused}}$$

where L_{branch} is the loss (cross entropy or focal) associated with each individual branch of the network while L_{fused} is the loss that belongs to the final classification head. The remaining 2 variables, w_{single} and w_{fused} represent the weights attached to the individual branches and the fused head, respectively. This allows the model to control the impacts of separate modality branches and the fused head on the final prediction.

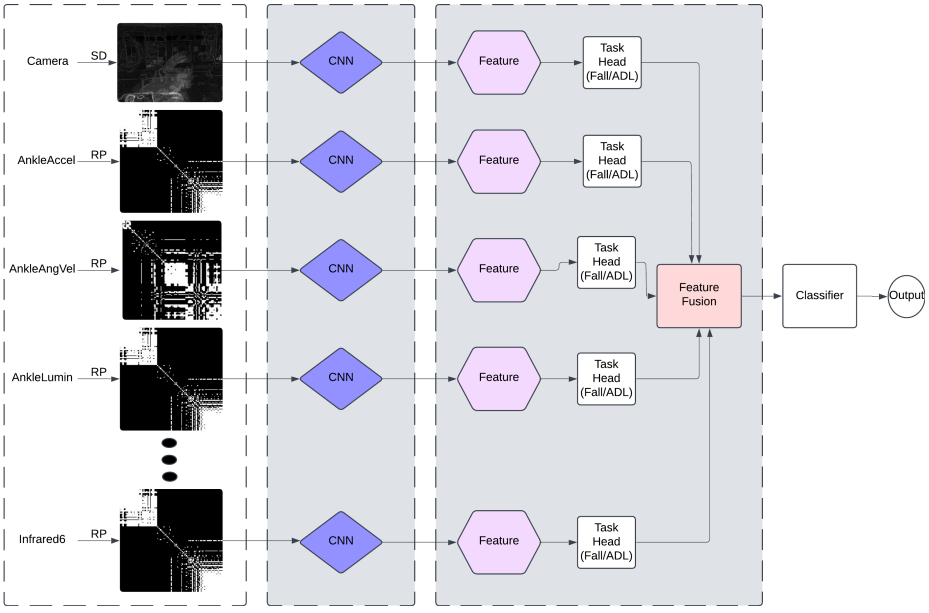

Figure 3-6: Approach B: Multi-Task Model with Separate Branches for Each Sub-Sensor

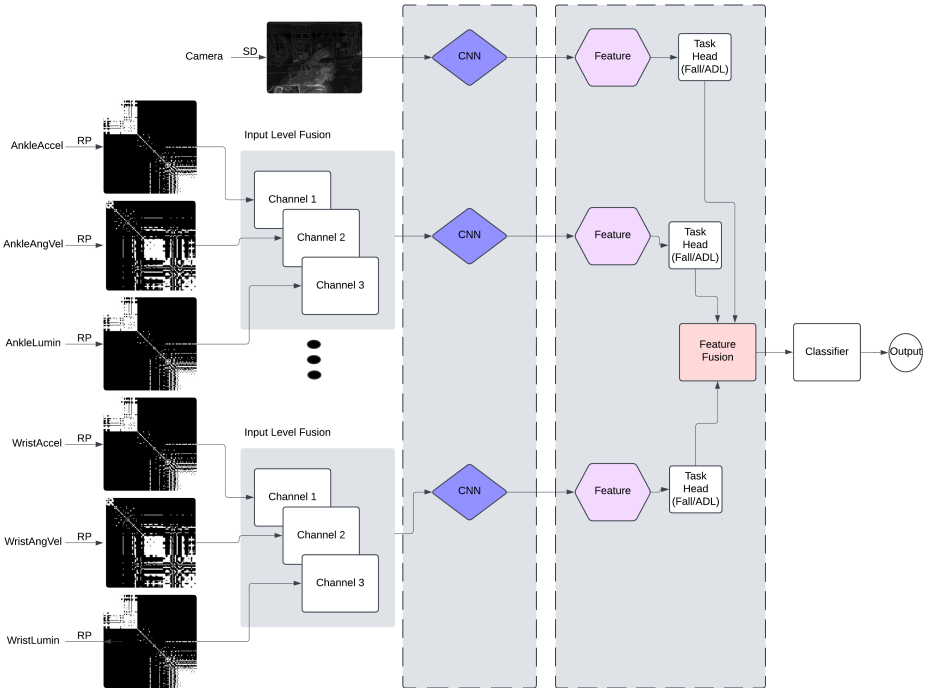

Figure 3-7: Approach B: Multi-Task Model with Grouped Sensors (Feature-Level Fusion)

Separate classification heads allow each branch to classify on its own, thereby allowing the network to be more robust to missing data modalities

Similar to Approach A, "One Branch per Sub-Sensor" and "Location-Based Channel Stacking" networks were implemented for Approach B. The respective network architecture diagrams can be seen in Figures 3-6 and Figure 3-7. "One Branch per Sub-Sensor" network of this approach was trained so that any sensor modality alone could still produce an output. "Location-Based Channel Stacking" network is lighter but still robust in cases where certain locations are entirely missing.

Missing Modality Robustness

In order to make the models robust to any subset of missing modalities, the modality dropout technique was used during training. In other words, during training, to ensure that the model can classify with partial data, certain sensors were artificially cut, and zero images were fed in their stead. In addition, Approach B ensures that each branch is self-sufficient and can classify on its own.

3.3.2 Training and Testing Phases

All the models underwent the hyperparameter tuning phase in which parameters such as learning rate, dropout probability, the weighting factor for single-sensor vs. fused losses (w_{single} vs. w_{fused}) , number of epochs, etc., were tuned based on the model's performance on the validation set. The most optimal configurations were assessed based on validation accuracy.

The models are then retrained with the most optimal hyperparameter configurations on the training + validation sets before being evaluated on a newly seen test set to measure generalization. Models were assessed on both binary and multiclass classification tasks. Moreover, tests were performed to assess model performance when a certain subset of sensors is missing. In addition, LOSO experiments were carried out for the best-performing models to further assess the models' generalizability to unseen subjects. Lastly, the models were also evaluated in terms of their weights and potential applicability for real-time settings.

This consistent 2D pipeline allows handling of all the data modalities uniformly with the same CNN networks, thereby removing the necessity for trying to connect separate 1D networks or classical ML models with 2D CNN, which are essential for video data processing.

Chapter 4

Results

In this section, the results of the experiments and each proposed model's performance metrics will be analyzed in terms of both binary classification and multiclass classification tasks. The performance metrics that were used include accuracy, recall, precision, and F1-scores. Additionally, models were tested for their robustness to missing data modalities/sensors during inference time.

All the proposed models were implemented using the PyTorch library and were trained on Google Colab's NVIDIA Tesla K80 with 12GB of VRAM (Video Random-Access Memory). As was described in the Methodology, trial 3 data in its entirety was used for testing, while the split between training and validation sets was based on subjects.

4.0.1 Baseline Results

To justify multimodal fusion and assess its benefits in terms of performance metrics, it is necessary to get a reference point and obtain baseline metrics. It has been done by first getting both binary and multiclass classification results using only the video data. Then, the results for the same classification tasks have been obtained using only the time-series sensors. To obtain the results, separate CNN networks were trained and tuned based on the hyperparameter tuning approach described in the methodology section.

Binary Classification Baselines

The Table 4.1 summarizes the most important metrics for the binary classification task using camera-only and time-series only data. It can be seen that the recall is the metric that has the lowest value for both modality groups, which indicates the models missed some fall events, which in turn is quite concerning.

Table 4.1: Binary Classification Baseline Results

Modality	**Accuracy**	**Precision**	**Recall**	**F1-score**
Camera-only	0.9397	0.9667	0.8067	0.8629
Time-series only	0.9543	0.9458	0.8751	0.9058

Multiclass Classification Baselines

Baselines for the 11-class classification task are shown in Table 4.2. It is evident from the table that the camera-only baseline model outperformed its time-series-only counterpart across all the metrics by a noticeable margin. It could be due to better spatial queues present in the visual data that made the difference by allowing the model to better differentiate between different kinds of falls. Nevertheless, the results are not the greatest and therefore motivate the usage of multimodal data in a complementary way.

Table 4.2: Multiclass Classification Baseline Results

Modality	**Accuracy**	**Precision**	**Recall**	**F1-score**
Camera-only	0.7692	0.7140	0.7692	0.7313
Time-series only	0.6237	0.6075	0.6237	0.6071

4.0.2 Binary Classification Results (Fall vs. No-Fall)

The performance of each proposed model on the binary classification task is summarized in Table 4.3.

Overall, the results were solid across different models with all the key metrics above 99%. In terms of the F1-score, the A2 (location-based fused head) model came

Table 4.3: Binary Classification Results of Proposed Models

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**
A1 (Sub-sensor)	0.9959	0.9870	0.9975	0.9922
A2 (Location-based)	0.9958	0.9959	0.9958	0.9959
B1 (Sub-sensor Multi-task)	0.9958	0.9921	0.9921	0.9921
B2 (Location-based Multi-task)	0.9938	0.9856	0.9909	0.9882
SOTA [2]	0.9986	0.9984	0.9986	0.9985

the closest to the state-of-the-art score. Both of the multi-task models (B1 and B2) were inferior to the Approach A models in terms of F1 by a slight amount. This could indicate a slight advantage of fused classification when it comes to producing a bit more optimal global representations for this particular task. It is also worth noting that the recall scores, which are crucial for real-world scenarios, were 99%+ for all the models, which in turn indicates the low count of false negatives and a low number of potentially missed fall events.

The confusion matrix of the best-performing model (A2) is depicted in Figure 4-1.

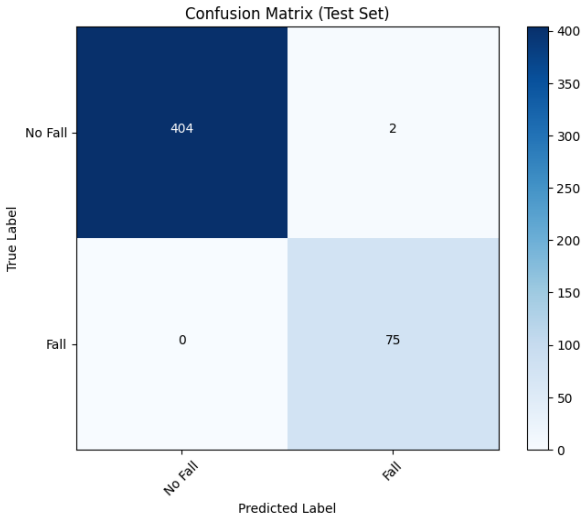

Figure 4-1: Confusion matrix for binary classification using Approach A.2.

The training loss plot over 10 epochs for the best-performing model (A2) is depicted in Figure 4-2

The detailed classification report of the A2 model can be seen in Table 4.4

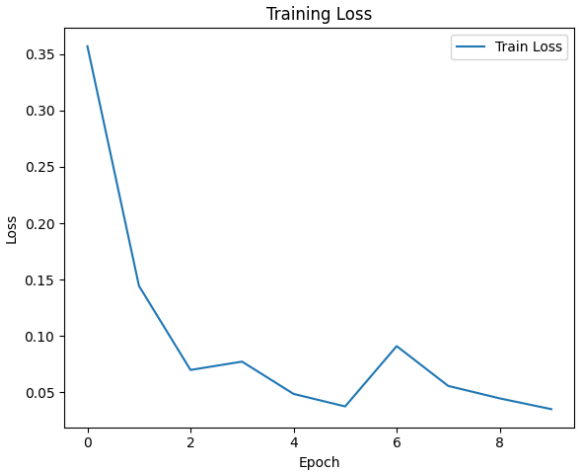

Figure 4-2: Training loss curve for A2 model over 10 epochs

Table 4.4: Detailed classification report of the A2 model (Location-based, Binary Classification).

Class	**Precision**	**Recall**	**F1-score**
No Fall	0.99	0.99	0.99
Fall	0.97	0.99	0.98

The models were also tested for their robustness to missing data during inference. Generally, the trends and performance metrics were similar for all the models. A table summarizing the results of the A2 model can be seen in Table 4.5.

The first key observation that could be made is that the camera is the most crucial sensor for decision making of the models, as the removal of this sensor resulted in roughly a 3% drop in key metrics. Also, the multi-task models (B1 and B2) did not demonstrate significantly better robustness in comparison to the single fused head models (A1 and A2). The models did not experience significant performance degradation when certain inertial sensors were not available. For comparison, the performance of the B2 model with different missing modalities can be seen in Table 4.5

Table 4.5: Performance of A2 model (Location-based, Binary) under different missing modalities.

Missing Modality	**Accuracy**	**F1-score**
None (Full Data)	0.996	0.992
Camera	0.971	0.948
Ankle	0.995	0.992
Belt	0.994	0.983
Neck	0.996	0.992
Right Pocket	0.994	0.991
Wrist	0.995	0.994
Brain Sensor	0.996	0.992
Infrared	0.994	0.993

Table 4.6: Performance of B2 Model (Location-based Multi-task, Binary) under different missing modalities.

Missing Modality	**Accuracy**	**F1-score**
None (Full Data)	0.994	0.988
Camera	0.980	0.976
Ankle	0.993	0.986
Belt	0.993	0.983
Neck	0.992	0.985
Right Pocket	0.993	0.986
Wrist	0.992	0.984
Brain Sensor	0.994	0.988
Infrared	0.994	0.988

4.0.3 Multi-Class Activity Classification Results

The performances of different models in a multiclass classification task are summarized in Table 4.7

It can be seen that the accuracy for the given task is generally lower than in the binary classification task. The key takeaways include the fact that the A2 (Location-based) model did the best among the proposed models, with an accuracy of 87.5% and an F1-score of 87.4%. As with the binary case, the A1 model is slightly behind the A2 model in terms of key metrics, potentially showing the slight benefit of incorporating input-level fusion. Similarly, B1 and B2 models demonstrated lesser numbers, indicating the possible adverse impact of the multi-task approach due to

Table 4.7: Multiclass classification performance comparison across all models.

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**
A1 (Sub-sensor)	0.861	0.878	0.861	0.858
A2 (Location-based)	0.875	0.875	0.875	0.874
B1 (Sub-sensor Multi-task)	0.832	0.835	0.832	0.830
B2 (Location-based Multi-task)	0.863	0.865	0.863	0.861
SOTA [2]	0.898	0.901	0.898	0.897

increased complexity and diluted learning signal.

Also, it is evident from the Figures 4.8 and 4-3 that the models showed a struggle to correctly differentiate between different types of falls (classes 1-5)

Table 4.8: Per-class precision, recall, and F1-score of the A2 (Location-based) model on the multiclass classification task.

Class	**Precision**	**Recall**	**F1-score**
Activity1	0.722	0.867	0.788
Activity2	0.636	0.467	0.539
Activity3	0.667	0.533	0.593
Activity4	0.625	0.667	0.645
Activity5	0.722	0.867	0.788
Activity6	1.000	1.000	1.000
Activity7	0.861	0.841	0.851
Activity8	0.830	0.886	0.857
Activity9	0.938	1.000	0.968
Activity10	0.957	1.000	0.978
Activity11	0.933	0.854	0.892

The training loss curve for the best-performing multiclass model can be seen in Figure 4-4

For the multiclass models, the variance in performance metrics was similar when missing certain subsets of modalities. The Table 4.9 shows the performance of the A2 model while missing some modalities. The most notable conclusion from the experiments is that the models hugely depend on visual data to make adequate 11-class classification. When the camera data is unavailable during inference, the metrics drop by about 30% which signifies the critical role of the camera data. Also, when removing modalities like Ankle, Belt, or Right Pocket, there is a slight degradation

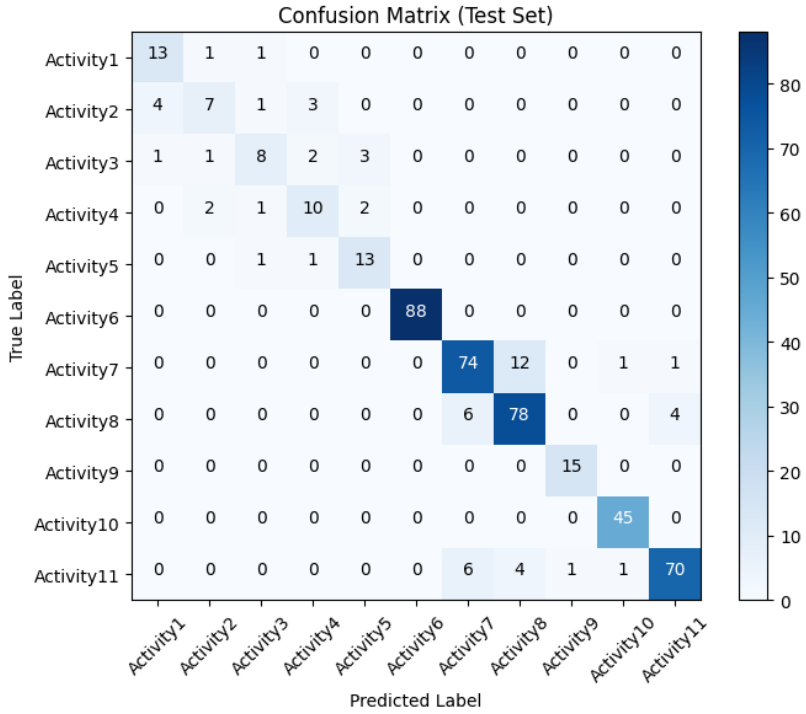

Figure 4-3: Confusion matrix for multi-class classification using Approach A.2.

in accuracy by about 1-2% which is tolerable. Removing more time-series modalities leads to a maximum of about 5% drop.

The multi-task models (B1 and B2) did not provide an outstanding advantage in terms of robustness for the multiclass classification, and the performance drops when missing certain modalities were similar to that of the A1 and A2 models. The summary of the robustness test results for the B2 model can be seen in Table 4.10

4.0.4 Leave-One-Subject-Out results

As discussed before, to further evaluate the generalizability, the best-performing model for both binary and multiclass tasks was re-assessed using the LOSO strategy. The A2 model was trained on all the trials of the N-1 subjects and evaluated on the remaining subject's trial data. Each subject was isolated iteratively, and the results were averaged to get balanced results. The Table 4.11 depicts the averaged key metrics obtained from this experiment. The results for both the binary and multiclass tasks bear a striking resemblance to the results that were obtained using the

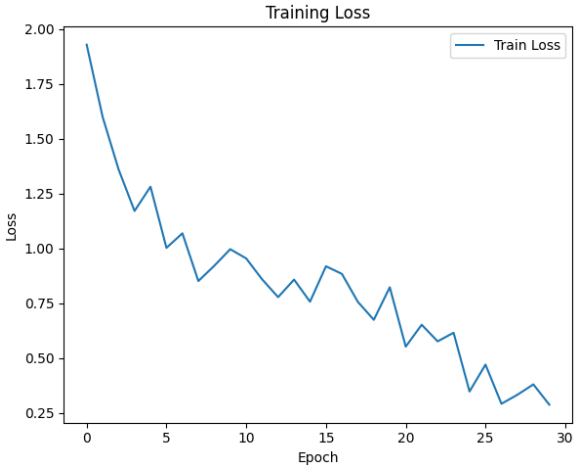

Figure 4-4: Training loss curve for A2 model over 30 epochs

Table 4.9: Performance of A2 (Location-based) model under different missing modality scenarios for the multiclass task.

Missing Modality	**Accuracy**	**F1-score**
None (Full Data)	0.875	0.874
Camera	0.578	0.572
Ankle	0.868	0.827
Belt	0.869	0.867
Neck	0.875	0.874
Right Pocket	0.863	0.860
Wrist	0.875	0.867
Brain Sensor	0.875	0.874
Belt, RightPocket, Infrared	0.842	0.838
Camera, Wrist, Brain	0.516	0.480

trial split strategy. This confirms that the proposed models are perfectly capable of generalizing to completely unseen subjects while not compromising performance.

4.0.5 Model Complexity and Weight analysis

In terms of the models' complexity or weight, our best-performing model A2 has a total of around 2.75 million parameters. Consequently, it is estimated to be around 11 MB in terms of weight, which makes it suitable for edge deployment, especially if it is further pruned or quantized.

Table 4.10: Performance of B2 model under different missing modality scenarios for the multiclass task.

Missing Modality	**Accuracy**	**F1-score**
None (Full Data)	0.863	0.861
Camera	0.541	0.521
Ankle	0.861	0.859
Belt	0.862	0.858
Neck	0.858	0.852
Right Pocket	0.859	0.858
Wrist	0.860	0.859
Brain Sensor	0.861	0.860
Belt, RightPocket, Infrared	0.840	0.836
Camera, Wrist, Brain	0.502	0.410

Table 4.11: Performance of A2 (Location-based) model on Binary and Multiclass Classification Tasks

Task	**Accuracy**	**Precision**	**Recall**	**F1-score**
Binary	0.992	0.991	0.994	0.992
Multiclass	0.861	0.868	0.861	0.867

Table 4.12: Comparison of Model Complexity, Performance, and Robustness

Model	**Binary F1 (%)**	**#Params**	**Size (MB)**	**Edge Suitability**	**Real-time Applicable**	**Robust to Missing Modalities**
Ours (A2)	99.6	∼2.75M	∼11	Good	Yes	**Yes**
CNN (SOTA, 2023)[2]	99.8	∼1.5M	∼6	Good	No	No
Uniformer (2024)[25]	94.1	∼20M	∼80	Limited	Yes	No

The previous study [2] that was the first to apply input level fusion to a fall detection task using Gramian Angular Fields, achieved a slightly higher accuracy for the binary classification problem (99.86% vs 99.58%). With regards to the multiclass classification problem, their best accuracy result was around 89.8% which is close to our best result of 87.5%. However, our models offer potential real-world usability due to our implementation approach and relatively low number of parameters. Additionally, compared to the previous study [2], our models are resilient to missing data modalities and could operate with a different subset of available modalities in general.

Chapter 5

Conclusion

This study proposed a deep learning implementation of a multi-sensor fusion approach that is capable of fall detection and classification. The methodology involved preprocessing of raw 1-dimensional sensor readings into 2-dimensional images using Recurrence Plots. This allowed easier fusion with the camera data, which in turn was converted into Summed Difference images. Furthermore, four separate networks were implemented, 2 of which used single fused classification heads while the remaining 2 models employed multi-task classification approach. The models used both feature level fusion and input level fusion, which have not been widely used for fall detection tasks by previous research. Results showed that the performance metrics of our models were close to matching the results of the state-of-the-art models. This was shown by both split-based and leave-one-subject-out testing strategies. The best-performing model was also compared to the latest research results in terms of its applicability to real-world tasks on an edge device. The A2 model showed that its relatively smaller size, combined with its implementation approach, robustness to missing data modalities during inference, and competitive classification results, allows it to be potentially used in a real setting. Future work could involve further optimization of the models to an edge device with limited computing resources, by applying model optimization strategies as quantization and pruning.

Bibliography

Bibliography

[1] Martínez-Villaseñor, Lourdes, et al. "UP-fall detection dataset: A multimodal approach." Sensors 19.9 (2019): 1988.

[2] Qi, Pian, Diletta Chiaro, and Francesco Piccialli. "FL-FD: Federated learning-based fall detection with multimodal data fusion." Information fusion 99 (2023): 101890.

[3] Wang, Shaobing, and Jiang Wu. "Patch-transformer network: a wearable- sensor-based fall detection method." Sensors 23.14 (2023): 6360.

[4] Wensel, James, Hayat Ullah, and Arslan Munir. "Vit-ret: Vision and recurrent transformer neural networks for human activity recognition in videos." IEEE Access (2023).

[5] T. Giannakopoulos and S. Konstantopoulos, Daily Activity Recognition Based on Meta-Classification of Lowlevel Audio Events. Set´ubal, Por- tugal: ScitePress, May 2017, doi: 10.5220/0006372502200227.

[6] Yan, Huan, et al. "WiAct: A passive WiFi-based human activity recog- nition system." IEEE Sensors Journal 20.1 (2019): 296-305.

[7] Yazici, Adnan, et al. "A smart e-health framework for monitoring the health of the elderly and disabled." Internet of Things 24 (2023): 100971.

[8] Wang, Bo-Hua, et al. "Fall detection based on dual-channel feature integration." IEEE Access 8 (2020): 103443-103453.

[9] Taghvaei, Sajjad, Mohammad Hasan Jahanandish, and Kazuhiro Kosuge. "Autoregressive-moving-average hidden Markov model for vision-based fall prediction—An application for walker robot." Assistive technology 29.1 (2017): 19-27.

[10] Zerrouki, Nabil, et al. "Fall detection using supervised machine learning algorithms: A comparative study." 2016 8th international conference on modelling, identification and control (ICMIC). IEEE, 2016.

[11] Lu, Na, et al. "Deep learning for fall detection: Three-dimensional CNN combined with LSTM on video kinematic data." IEEE journal of biomedical and health informatics 23.1 (2018): 314-323.

[12] Islam, Md Milon, et al. "Multi-level feature fusion for multimodal human activity recognition in Internet of Healthcare Things." Information Fusion 94 (2023): 17-31.

[13] Ihianle, Isibor Kennedy, et al. "A deep learning approach for human activities recognition from multimodal sensing devices." IEEE Access 8 (2020): 179028-179038.

[14] Xie, Leiyu, et al. "Skeleton-based fall events classification with data fusion." 2021 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI). IEEE, 2021.

[15] Cai, Wen-Yu, et al. "GBDT-Based Fall Detection with Comprehensive Data from Posture Sensor and Human Skeleton Extraction." Journal of healthcare engineering 2020.1 (2020): 8887340.

[16] Xu, Tao, Haifeng Se, and Jiahui Liu. "A fusion fall detection algorithm combining threshold-based method and convolutional neural network." Microprocessors and Microsystems 82 (2021): 103828.

[17] Yang, Yi, et al. "Fall detection system based on infrared array sensor and multi-dimensional feature fusion." Measurement 192 (2022): 110870.

[18] Amsaprabhaa, M. "Multimodal spatiotemporal skeletal kinematic gait feature fusion for vision-based fall detection." Expert Systems with Applications 212 (2023): 118681.

[19] De, Anurag, et al. "Fall detection method based on spatio-temporal feature fusion using combined two-channel classification." Multimedia Tools and Applications 81.18 (2022): 26081-26100.

[20] S.-H. Park, Tools for assessing fall risk in the elderly: A systematic review and meta-analysis, Aging Clin. Exp. Res. 30 (1) (2018) 1–16.

[21] Centers for Disease Control and Prevention, National Center for Injury Prevention and Control. Web–based Injury Statistics Query and Reporting System (WISQARS) [online]. Accessed October 20, 2024.

[22] Centers for Disease Control and Prevention (2021). Surveillance Report of Traumatic Brain Injury-related Hospitalizations and Deaths by Age Group, Sex, and Mechanism of Injury—United States, 2016 and 2017. Centers for Disease Control and Prevention, U.S. Department of Health and Human Services.

[23] R. Broadley, J. Klenk, S. Thies, L. Kenney, and M. Granat, "Methods for the real-world evaluation of fall detection technology: A scoping review," Sensors, vol. 18, no. 7, p. 2060, Jun. 2018.

[24] J. Gutiérrez, V. Rodríguez, and S. Martin, "Comprehensive review of vision-based fall detection systems," Sensors, vol. 21, no. 3, p. 947, Feb. 2021.

[25] Núñez-Marcos, Adrián, and Ignacio Arganda-Carreras. "Transformer-based fall detection in videos." Engineering Applications of Artificial Intelligence 132 (2024): 107937.