AI-Based Fall Detection and Post-Fall Analysis Using Computer Vision and Wearable Sensors

Tomas Marin, Samuel Karson, Mikhail Gromov, Jafar Saniie
Embedded Computing and Signal Processing (ECASP) Research Laboratory (http://ecasp.ece.iit.edu)
Department of Electrical and Computer Engineering
Illinois Institute of Technology, Chicago, IL, U.S.A

***Abstract*—Falls significantly affect the health of older adults and impose a substantial burden on healthcare systems. Traditional fall detection methods, typically based on wearable sensors or simple rule-based algorithms, often suffer from high false alarm rates and low user compliance. This paper introduces a hybrid fall detection framework that addresses the limitations of single-sensor systems as well as the privacy concerns associated with camera-only approaches. The proposed system integrates vision-based pose estimation with wrist-worn inertial data to enhance both accuracy and reliability. YOLOv8 is employed to extract skeletal keypoints from video, while a CNN-LSTM architecture enables accurate activity recognition. Real-time alerts are deployed on Raspberry Pi and ESP32 edge devices. Experimental results show improved detection accuracy and reduced latency, particularly through sensor fusion. Additionally, a finite-state machine supports detailed post-fall analysis, ensuring robust real-time performance on resource-constrained edge hardware. Overall, the proposed multimodal system improves fall detection accuracy and reliability in unstructured environments while providing automated alerts. This scalable approach supports proactive fall detection and post-fall health monitoring, which are critical for medical professionals conducting injury and damage assessment.**

Keywords—**Fall Detection, Computer Vision, YOLOv8, Sensor Fusion, Post-Fall Analysis, LSTM, Edge Computing**

I. INTRODUCTION

Falls are a leading cause of injury and hospitalization among older adults, often resulting in long-term health complications, loss of independence, and increased healthcare costs [1]. The World Health Organization (WHO) reports that one in three people over the age of 65 experiences a fall annually, highlighting the urgent need for reliable fall detection systems. Conventional fall detection methods typically rely on wearable inertial sensors or threshold-based algorithms. However, these approaches face limitations such as high false alarm rates, sensitivity to sensor placement, and low user compliance due to the inconvenience of continuous device use [2]. To address these issues, the integration of artificial intelligence (AI) with multimodal sensing has emerged as a promising solution.

Despite recent advances in AI-driven fall detection, many systems either lack real-time capability, fail in unstructured environments, or do not offer post-fall analysis for risk evaluation. Furthermore, camera-based systems often raise privacy concerns, while single-sensor systems can struggle with accuracy in complex scenarios.

To overcome these challenges, this paper proposes a hybrid fall detection system that combines vision-based pose estimation with inertial sensor data from a wrist-worn device [3]. By leveraging spatial and temporal features through a CNN- LSTM model and incorporating a finite-state machine (FSM) for post-fall behavior analysis, we achieve enhanced detection accuracy and robustness.

This work presents a complete end-to-end solution for intelligent fall detection and response, featuring:

- A vision-based module using YOLOv8 pose estimation to extract skeletal keypoints from real-time video streams.
- A CNN-LSTM classifier trained on sequences of pose data for activity recognition.
- A finite state machine for analyzing post-fall conditions (e.g., immobility vs. recovery).
- Sensor fusion with wristband accelerometer data to increase robustness.
- A real-time implementation on edge devices (Raspberry Pi and ESP32) with automated alert triggering.

The remainder of this paper is structured as follows. Section II reviews current fall detection systems. Section III describes the proposed system architecture. Section IV details model evaluation and real-time testing results. Section V concludes the paper and outlines future directions.

II. RELATED WORKS

Fall detection systems have been extensively studied in recent years, employing a range of techniques from wearable sensor-based approaches to advanced deep learning and hybrid AI architectures.

A. Wearable Sensor-Based Approaches

Traditional systems rely on accelerometers and gyroscopes embedded in wearable devices to detect falls by applying threshold-based or statistical classifiers [2]. While lightweight and power-efficient, these methods are prone to false positives during activities of daily living (ADLs) and require users to consistently wear the device [1]. Recent work has improved accuracy by combining wearable data with physiological metrics such as heart rate or using ensemble machine learning

979-8-3315-6026-3/26/$31.00 ©2026 IEEE

models [2]. However, these improvements come at the cost of increased sensor complexity and reduced user comfort.

B. Vision-Based Deep Learning Models

Vision-based systems eliminate the need for physical contact by analyzing video data to detect falls. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) networks, have been used to extract spatial and temporal features, respectively [4], [5].

Pose estimation methods, such as those based on YOLOv8 and AlphaPose, extract skeletal keypoints from video streams, allowing the system to focus on human body configurations rather than full-frame data [6]. FD-YOLO, a recent variant optimized for fall detection, integrates attention modules to boost robustness in complex scenes [7].

C. Hybrid Models and Sensor Fusion

Hybrid systems leverage the strengths of both vision and wearable sensors, improving reliability and minimizing false detections. Lin et al. [8] demonstrated an edge AI-based framework that fused pose and sensor data for real-time fall detection with low latency. Similarly, recent works combine CNNs for feature extraction with LSTMs for temporal modeling, enabling robust classification of fall sequences [5].

Sensor fusion techniques enhance performance in noisy environments and offer redundancy in case of occlusion or signal loss in one modality. These approaches are increasingly popular for real-world deployments where environmental variability is high.

D. Pose Estimation via YOLOv8

Human skeleton keypoints are extracted from RGB video using YOLOv8 pose estimation. The model is deployed on the edge device for real-time inference and generates a sequence of 2D joint coordinates per frame. This data is saved in NumPy format for fast processing. Fig. 1 shows the Pose Estimation via YOLOv8.

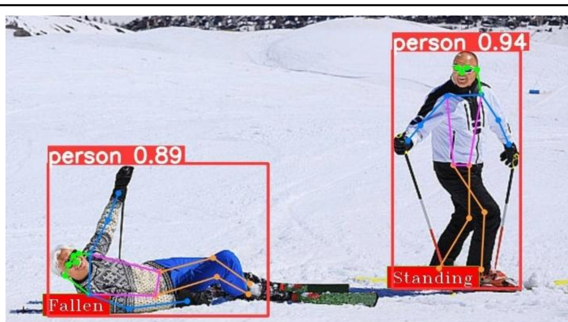

Fig. 1. YOLOv8 Pose model architecture used to estimate 2D joint coordinates in real time. Source: zhuanlan.zhihu.com [9].

III. Fall Detection System Architecture

The proposed fall detection framework combines computer vision, AI-based post-fall analysis, and sensor fusion to achieve accurate, real-time monitoring. The architecture is divided into three main stages: video-based pose extraction, machine learning-based fall prediction, and system-level coordination with wearable sensors.

A. Overview

The system operates on edge devices (e.g., Raspberry Pi) and integrates camera-based pose estimation with inertial data from a wearable wristwatch. Fig. 2 presents a high-level pipeline of the fall detection framework.

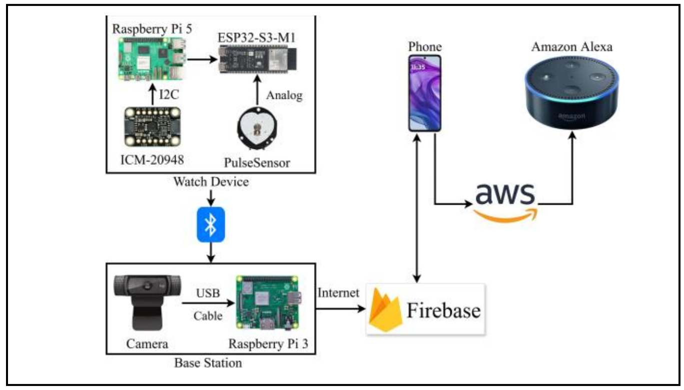

Fig. 2. Overview of the hybrid fall detection system.

B. Fall Classification via CNN + LSTM Network

A hybrid deep learning architecture is presented for fall detection, integrating a Convolutional Neural Network (CNN) with a Long Short-Term Memory (LSTM) network (see Fig. 3). This model is designed for spatio-temporal feature learning from video sequences to classify actions into four states: Activities of Daily Living (ADL), Falling, Lying, and Recovering. The CNN, based on the ResNet18 architecture, extracts rich spatial features from individual RGB frames, while the LSTM module models the temporal dependencies across the sequence of these features.

The ResNet18 CNN backbone is initiated with a 7×7 convolutional layer with a stride of 2 and padding of 3, followed by batch normalization, a ReLU activation function, and a 3×3 max pooling layer with a stride of 2. This initial stage downsamples the 224×224×3 input frames to a 56×56×64 feature map. The main body of the ResNet18 model consists of four stages, each comprising two basic residual blocks. These blocks utilize identity skip connections to mitigate the vanishing gradient problem, facilitating the training of deep networks. Following the residual blocks, a global average pooling layer compresses the spatial dimensions, yielding a 512-dimensional feature vector for each frame.

The sequence of 512-dimensional feature vectors is then passed to a two-layer LSTM network. This temporal module, with a hidden size of 128, is responsible for learning the dynamic patterns and contextual relationships between consecutive frames. The final hidden state of the LSTM, which encapsulates the entire spatio-temporal context of the sequence, is then fed into a fully connected layer. This classifier, with four output neurons, produces the final logits for the four aforementioned fall-related classes.

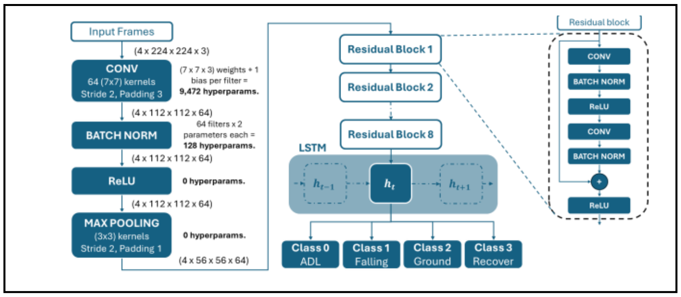

Fig. 3. CNN + LSTM architecture diagram.

C. *Post-Fall Finite-State Machine (FSM)*

A deterministic FSM governs transitions between states. It accounts for motion thresholds and time-based rules to switch between phases, such as “fall detected” and “immobile on ground.” The logic also ensures stability by preventing rapid switching due to noise. Fig. 4 shows the state diagram.

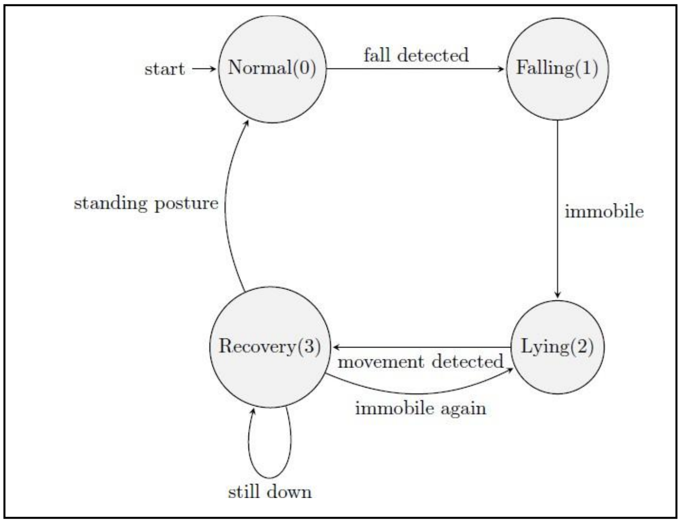

Fig. 4. Finite-state machine used for post-fall state tracking. Transitions are governed by pose stability, motion thresholds, and time constraints.

D. *Sensor Fusion with Wrist-Worn Device* [3]

The system processes IMU sensor data from a Bluetooth-connected wristwatch to detect sudden acceleration. When high acceleration is detected, an alarm signal is transmitted via serial connection to the fall prediction module. This hybrid approach improves robustness against false positives by confirming visual events with physical impact data.

E. *Real-Time Alert and Data Transmission*

Upon confirmed fall detection, the system saves keyframes, logs prediction data, and triggers a notification pipeline that sends a JSON payload to an ESP32, which further relays the message to Firebase and Alexa-integrated systems. This ensures prompt alerts for caregivers and enables remote monitoring.

IV. Experiments and Results

The training of the combined CNN-LSTM model is an iterative process aimed at minimizing a defined loss function, such as cross-entropy loss, which measures the discrepancy between the model’s predicted class probabilities and the true labels. This minimization is achieved through the use of backpropagation, the fundamental algorithm for training neural networks. During backpropagation, the gradient of the loss function with respect to each of the model’s parameters (weights and biases) is computed.

These gradients are then used by an optimization algorithm to update the model’s parameters. Key training hyperparameters in this process are:

- Learning Rate: This parameter controls the magnitude of the weight updates based on the calculated gradients. An appropriately chosen learning rate is critical for achieving convergence; a high value can cause optimization to overshoot the minimum, while a low value can result in extremely slow convergence. A dynamic learning rate schedule, where the learning rate is progressively reduced during training, is commonly employed to improve performance.

- Batch Size: This hyperparameter determines the number of training samples processed in each forward and backward pass. A larger batch size provides a more stable estimate of the gradient but requires greater memory and can lead to less generalization. Conversely, a smaller batch size introduces more noise into the gradient updates.

- Epochs: An epoch represents one complete pass of the entire training dataset through the model. The number of epochs determines the total number of training cycles and is a critical factor in preventing both underfitting (too few epochs) and overfitting (too many epochs).

A. *Evaluation Metrics*

To assess the model’s performance, we compute standard classification metrics: accuracy, precision, recall, F1-score, and confusion matrix. These metrics are defined as follows [3]:

- Accuracy: Proportion of correctly classified frames.

- Precision: Proportion of detected fall frames that were true positives.

- Recall: Proportion of actual fall frames that were correctly detected.

- F1-Score: Harmonic mean of precision and recall.

B. *Model Training Results*

The classifier was trained on sequences of pose vectors extracted from a curated dataset of fall and non-fall events. Figs. 5 and 6 display the training and validation accuracy and loss. Table I shows the performance of the model evaluated on a separate test set. The confusion matrix in Fig. 7 reveals that most misclassifications occur between adjacent temporal states (e.g., from state 1 to 2 or 2 to 3), which is expected due to their transitional nature.

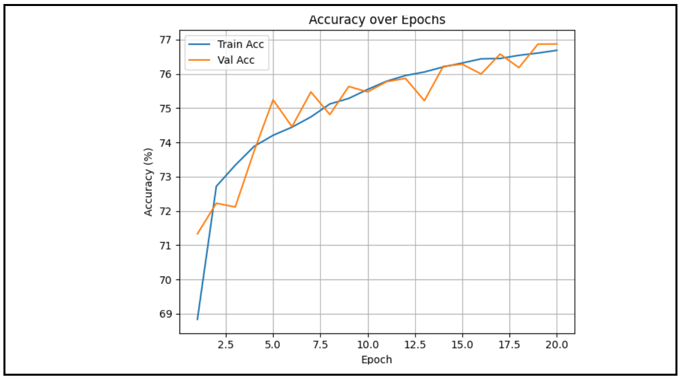

Fig. 5. Training and validation accuracy curves.

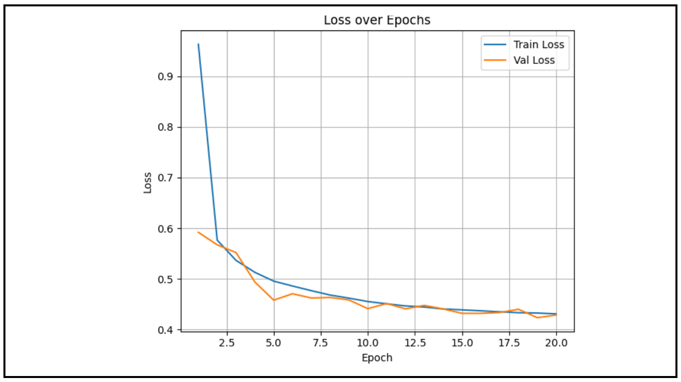

Fig. 6. Training and validation loss curves.

TABLE I. MODEL PERFORMANCE ON TEST SET

Class	Precision	Recall	F1-Score	Support
ADL (0)	0.95	0.96	0.96	324
Fall (1)	0.98	0.91	0.92	98
Lying (2)	0.88	0.85	0.86	76
Recovering (3)	0.91	0.92	0.92	82
Avg/Total	0.92	0.91	0.91	580

C. Sensor Fusion Comparison

We compared the standalone pose-based classifier with a fusion model that integrates acceleration data from the wristband [3]. Table II highlights the performance improvements over all four metrics: accuracy, precision, recall, and F1-score. Sensor fusion reduced false positives, particularly in edge cases where pose estimation alone misclassified unusual ADL movements as falls.

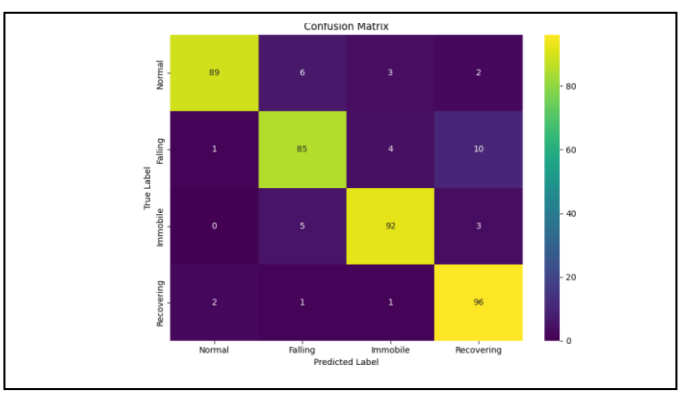

Fig. 7. Confusion matrix for the multi-class post-fall classifier.

TABLE II. PERFORMANCE WITH VS. WITHOUT SENSOR FUSION

Metric	Without Fusion	With Fusion
Accuracy	91.5%	94.7%
Precision	89.3%	93.8%
Recall	88.9%	94.0%
F1-Score	89.1%	93.9%

D. Real-Time Testing

The system was deployed on a Raspberry Pi 4B and tested in a controlled environment with scripted fall events. Fall detection latency remained under 300 ms on average. An example of fall detection is illustrated in Fig. 8.

Fig. 8. Camera detecting a fall in a video.

V. CONCLUSIONS

This work presented a hybrid fall detection and post-fall analysis system that integrates vision-based pose estimation using YOLOv8 with a machine learning classifier and optional sensor fusion via a wearable device. The system achieves accurate, real-time classification of falls and subsequent post-fall states, addressing a critical need in elderly healthcare and ambient assisted living environments.

The pose-based LSTM classifier, combined with a finite-state machine (FSM), demonstrated robust performance in multi-phase fall analysis. Moreover, the integration of inertial data from a wrist-worn device further improved reliability, reducing false positives and enhancing temporal accuracy. The system was successfully deployed at the network edge, demonstrating its practicality for real-time monitoring.

Further advancements of the fall detection system may include:

- Generalization: Training the model on more diverse datasets, including real-world falls, to enhance robustness across populations, environments, and camera views.
- Privacy-aware vision: Investigating privacy-preserving pose estimation approaches, such as skeleton-only representations or edge obfuscation techniques.
- Fall prediction: Extending the system to include fall prediction capabilities by analyzing gait patterns and pre-fall anomalies using Transformer-based temporal models.
- Voice and alert integration: Expanding integration with Alexa or mobile apps to provide caregivers with real-time alerts and contextual information.

- Multi-person detection: Improving detection in crowded scenes using tracking and identity association techniques to focus on the individual at risk.

The proposed system serves as a step towards intelligent, multimodal monitoring systems that can support elderly individuals living independently and safely.

References

[1] B. Lane and E. Cole, "AI-based fall detection and prevention systems for elderly patients," 2025.

[2] Y. Li et al., "A decade of progress in wearable sensors for fall detection (2015–2024): A network-based visualization review," Sensors, vol. 25, no. 7, p. 2205, 2025.

[3] S. Karson et al., "RiskWatch: Wearable fall detection and healthcare accountability device," in 2025 IEEE International Conference on Electro Information Technology (eIT), Valparaiso, IN, USA, 2025, pp. 1–6, doi: 10.1109/eIT64391.2025.11103628.

[4] F. X. Gaya-Morey, C. Manresa-Yee, and J. M. Buades-Rubio, "Deep learning for computer vision based activity recognition and fall detection of the elderly: A systematic review," Appl. Intell., vol. 54, no. 19, pp. 8982–9007, 2024.

[5] J. Wu, J. Wang, A. Zhan, and C. Wu, "Fall detection with CNN-casual LSTM network," Information, vol. 12, no. 10, p. 403, 2021.

[6] L. Liu, Y. Sun, Y. Li, and Y. Liu, "A hybrid human fall detection method based on modified YOLOv8s and AlphaPose," Sci. Rep., vol. 15, no. 1, p. 2636, 2025.

[7] H. Hwang, D. Kim, and H. Kim, "FD-YOLO: A YOLO network optimized for fall detection," Appl. Sci., vol. 15, no. 1, p. 453, 2025.

[8] B.-S. Lin et al., "Fall detection system with artificial intelligence-based edge computing," IEEE Access, vol. 10, pp. 4328–4339, 2022.

[9] Ctrl CV, "YOLOv8 pose 跌倒检测" (YOLOv8 pose fall detection), Zhihu, Feb. 5, 2024. [Online]. Available: https://zhuanlan.zhihu.com/p/680683994 [Accessed: Feb. 5, 2025].