Edge AI-Based Fall Detection with Standard RGB Cameras

Original
Edge AI-Based Fall Detection with Standard RGB Cameras / Proietti, M.; Piergallini, E.; Visi, A.; Dragoni, A. F.. - (2025), pp. 771-775. (4th IEEE International Conference on Metrology for eXtended Reality, Artificial Intelligence and Neural Engineering, MetroXRAINE 2025 Ancona, IT 22 - 24 October 2025) [10.1109/MetroXRAINE66377.2025.11340414].

Availability:
This version is available at: 11566/355415 since: 2026-04-09T20:02:30Z

Publisher:
Institute of Electrical and Electronics Engineers Inc.

Published
DOI:10.1109/MetroXRAINE66377.2025.11340414

(Article begins on next page)

Edge AI-Based Fall Detection with Standard RGB Cameras

1st Melissa Proietti
Dipartimento di Ingegneria dell'Informazione
Università Politecnica delle Marche
Ancona, Italy

2nd Enrico Piergallini
Dipartimento di Ingegneria dell'Informazione
Università Politecnica delle Marche
Ancona, Italy
s1118929@studenti.univpm.it

3rd Andrea Visi
Dipartimento di Ingegneria dell'Informazione
Università Politecnica delle Marche
Ancona, Italy
s1118766@studenti.univpm.it

4th Aldo Franco Dragoni
Dipartimento di Ingegneria dell'Informazione
Università Politecnica delle Marche
Ancona, Italy
a.f.dragoni@staff.univpm.it

***Abstract*—Falls among elderly individuals represent a significant public health concern due to their frequency and severe consequences, such as injuries, reduced independence and increased healthcare demands. Current fall detection methods primarily rely on wearable sensors, which face usability and acceptance challenges, particularly among cognitively impaired users. This paper proposes a fully non-invasive fall detection solution based on vision technology and edge computing, using standard RGB cameras and low-power hardware. The system leverages *YOLOv8*, a lightweight and efficient convolutional neural network, to perform real-time on-device inference without requiring user cooperation. Experimental evaluations conducted in real-world residential care environments demonstrated high accuracy and reliability with rapid response times. The proposed approach provides a practical, cost-effective and user-friendly alternative to traditional wearable-based systems, suitable for a variety of healthcare and residential settings.**

***Index Terms*—Fall Detection, Elderly Care, RGB Cameras, Edge Computing, Computer Vision, YOLO, Deep Learning, Assistive Technology**

I. Introduction

According to the World Health Organization (WHO) [1], 28% to 35% of adults aged 65 and over experience falls annually, increasing to 32%–42% for individuals over 70 years old. Elderly residents of long-term care facilities are particularly at risk, with 30%–50% experiencing falls annually and 40% of these experiencing repeated incidents. Falls significantly impact public health, causing 20%–30% of moderate-to-severe injuries, 10%–15% of emergency room visits and over 50% of hospitalizations for patients above 65. Common injuries include femur fractures, head trauma and upper limb injuries, often leading to post-fall syndromes characterized by dependency, immobility, depression and reduced daily activity. Falls also account for 40% of all injury-related deaths among the elderly.

Various technologies have been developed to detect falls, often involving wearable sensors. However, elderly individuals, especially those with dementia, frequently forget or refuse to wear these devices. To address this limitation, a cost-effective and non-intrusive system based on a Raspberry Pi and a camera is proposed. This device can be wall-mounted or ceiling-mounted to provide continuous 24-hour monitoring, leveraging vision-based algorithms to detect falls in real-time. The objective is to verify whether this approach achieves performance comparable to traditional wearable sensor-based systems.

The main challenge in fall detection is accurately distinguishing between genuine falls and other activities that resemble falling, such as bending down, lying on a bed, or sitting on the floor. This task, although seemingly straightforward, is highly complex for automated video-based systems and requires detailed analysis of human poses and movements. Effective fall detection involves the strategic placement of cameras on walls or ceilings to maximize coverage, combined with sophisticated algorithms that analyze captured events. Commonly, multi-camera systems use majority voting to determine falls when using an odd number of cameras, or confidence scores when the number is even. Typically, such systems prioritize sensitivity, accepting false alarms rather than missing actual falls, to ensure rapid intervention.

This study adopts a supervised learning approach utilizing neural networks. During training, the network minimizes a cost function that quantifies discrepancies between its predictions and provided labels. Misclassifications trigger adjustments to the neural connections through backpropagation. The training dataset included diverse labeled video scenarios, enabling the neural network to generalize effectively and robustly recognize falls.

A. State of the art

State-of-the-art fall detection technologies, as highlighted by Zhang et al. [2], have increasingly focused on computer vision-based approaches due to their non-invasive nature and high monitoring accuracy. Vision-based methods can generally be categorized into three main classes: systems using a single RGB camera, multi-camera systems utilizing 3D reconstruction and approaches leveraging depth-sensing cameras.

The first category involves systems based on a single RGB camera, which offer a practical and cost-effective solution. Mirmahboub et al. [3] developed a system that extracts the silhouette of a person through background subtraction. From this silhouette, relevant features are derived and processed by a Support Vector Machine (SVM) classifier to detect falls. More advanced techniques, such as the one proposed by Foroughi et al. [4], combine Integrated Time Motion Images (ITMI) with Principal Component Analysis (PCA) and a neural network for classification. Other notable works by Zhang et al. [5] and Feng et al. [6] employ different strategies, such as tracking the centroid of the human figure, using statistical models to evaluate motion, or representing human posture with geometric shapes such as ellipses to capture fall patterns.

The second category includes calibrated multi-camera systems that allow for 3D reconstruction of the observed environment. Auvinet et al. [7] employ such systems to analyze vertical volume distribution to detect falls, while Thome et al. [8] introduce a multi-view approach based on Layered Hidden Markov Models (LHMM), combining low-level feature extraction from different viewpoints with a fusion process for final decision-making. Zweng et al. [9] propose a method where each camera independently estimates fall probability, removing the need for prior calibration. These probabilities are then merged to provide a more robust fall detection mechanism.

The third approach utilizes depth-sensing technologies like Microsoft Kinect. Diraco et al. [10] use a Time-of-Flight sensor to monitor the distance of the human centroid from the ground, triggering an alert when the person remains immobile near the floor for a predefined duration. Leone et al. [11] define temporal rules based on centroid height and immobility. Zhang et al. [12] integrate multiple dynamic features, such as head height and velocity, within a Bayesian framework. Zhen-Peng et al. [13] enhance this method by extracting 3D joint trajectories of the head using pose estimation and classifying fall events with a pre-trained SVM.

Pose estimation techniques, as emphasized by Huang et al. [14] and Chang et al. [15], play a crucial role in modern fall detection systems. These approaches use tools like OpenPose to extract body joint coordinates from video sequences. The extracted pose data is then fed into Convolutional Neural Networks (CNNs) for effective classification. Such techniques enable detailed analysis of body posture and facilitate early detection of anomalous behaviors.

CNNs have become the backbone of many contemporary fall detection solutions. With their hierarchical structure, CNNs process visual data through convolutional layers, pooling layers and fully connected layers. These networks learn increasingly abstract representations of input data, ranging from basic edges and textures to complex patterns such as posture configurations or fall-like motions. This allows CNN-based systems to reliably distinguish fall events from normal activities, even in scenarios involving spatial variations or partial occlusions, making them an essential component of robust and reliable detection frameworks.

II. Materials and methods

The implementation focuses on enabling real-time detection using low-cost hardware, demonstrating the applicability of Edge AI in healthcare monitoring contexts.

A. YOLOv8 architecture

The core of the system is the YOLOv8 [16] ("*You Only Look Once*", version 8) model, an object detection algorithm developed by Ultralytics in 2023. This version introduces several improvements over previous versions, including an anchor-free design and a modular architecture divided into three main components:

- **Backbone:** based on *CSPDarknet53*, it extracts features from input images through a series of convolutional layers;
- **Neck:** utilizes a combination of Feature Pyramid Network (*FPN*) and Path Aggregation Network (*PANet*) to merge multi-scale features and improve generalization;
- **Head:** responsible for producing the final bounding boxes and class probabilities using refined feature maps.

YOLOv8 directly predicts the center and dimensions of bounding boxes without relying on predefined anchor boxes, offering improved flexibility when detecting objects of varying sizes.

B. Edge computing with Raspberry Pi

The fall detection system presented in this work is implemented using a low-cost, easily installable embedded device: the *Raspberry Pi 3 Model B+*. This compact and versatile platform, originally developed for educational purposes, has become widely adopted in various fields due to its flexibility and compatibility with different operating systems. Its small size and support for computer vision tasks make it particularly suitable for edge applications like real-time monitoring scenarios.

C. Image acquisition with USB cameras

Input data was collected using *standard USB webcams*, which offer a cost-effective and easily integrable solution. These devices are compatible with Linux-based systems such as *Raspbian*[1] and require minimal configuration, thus enabling a rapid and reliable setup.

D. Dataset collection and augmentation

To train the model on diverse fall scenarios, multiple public datasets were used:

- **Le2i Fall Detection Dataset** [17]: includes videos from four different indoor environments (home, office, lecture room, coffee room);

[1]Raspbian is a Debian-based operating system optimized for the Raspberry Pi hardware. It provides a lightweight, stable environment and comes preloaded with educational tools and programming utilities.

- **UR Fall Detection Dataset** [18]: provides sequences of falls and daily activities in both RGB and depth formats;
- **FPDS_v2 Dataset** [19]: contains over $6,000$ labeled images of fall and non-fall events under varied lighting, occlusion and pose conditions.

To increase dataset variability and support better generalization, a data augmentation pipeline was applied. Transformations such as rotations, brightness adjustments, flips and zooms were used to simulate real-world variability in lighting and viewpoint.

E. Annotation with CVAT

Annotation was performed using the Computer Vision Annotation Tool (*CVAT*) [20], an open-source web-based tool for image and video labeling. Each object of interest was marked with a bounding box and labeled as either `fall` or `nofall`. Annotations were exported in YOLO format to facilitate seamless integration with the training pipeline.

F. Training procedure

The training dataset was organized following YOLO conventions, with separate folders for images and labels, split into `train` and `val` subsets.

The YOLOv8 model was fine-tuned starting from a pre-trained checkpoint using the Ultralytics Python package. The training was performed with the following hyperparameters:

- **Epochs:** 100;
- **Batch size:** 16;
- **Learning rate:** 0.0001;
- **Intersection over Union (IoU) threshold:** 0.5;
- **Optimizer:** Adam;
- **Early stopping:** triggered after 20 consecutive epochs without validation improvement.

The objective was to minimize the loss function, which quantifies the difference between predicted bounding boxes and ground truth annotations, while avoiding overfitting and ensuring the model's ability to generalize to unseen scenarios.

III. Results

The proposed fall detection system, based on YOLOv8 and deployed on a Raspberry Pi 3 Model B, achieved solid performance across both the training and validation phases. As shown in Figure 1, the loss functions (`box_loss`, `cls_loss` and `dfl_loss`) exhibited a steep decrease in the early epochs, followed by a stable plateau, suggesting efficient convergence and good generalization capability. The alignment between the training and validation curves further indicates that the model avoided overfitting, maintaining consistency when exposed to unseen data.

Quantitatively, the model achieved robust performance metrics. The precision reached approximately 93% for both `fall` and `nofall` classes, while the recall for the `fall` class peaked at 96.37%, highlighting the system's ability to detect actual fall events with minimal false negatives. The global accuracy stood at 93.38%, while the mean Average Precision at 0.5 IoU (mAP@50) remained steady around 0.8 across

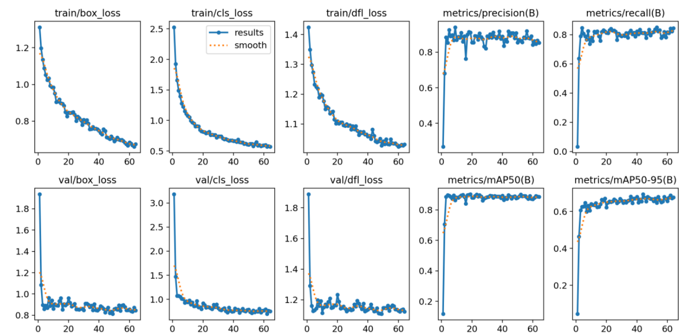

Fig. 1. Evolution of `box_loss`, `cls_loss` and `dfl_loss` during training and validation.

epochs, reinforcing the model's stability. Furthermore, the F1-score, which balances precision and recall, peaked at 0.86 for a confidence threshold of 0.486.

A particularly important component of this study was the real-world validation phase, which aimed to evaluate the system's practical viability in operational environments. The trained model was deployed inside an elderly care facility, where typical fall scenarios were identified in collaboration with the local healthcare staff. These scenarios reflected a variety of realistic conditions such as falls during ambulation, from beds and in shared spaces like corridors.

To recreate these events under controlled and representative circumstances, three Raspberry Pi units, each connected to a USB webcam, were deployed. A commonly used model for these tests was the *Logitech C920 Pro HD*, a widely available webcam known for its reliability and image quality. It features Full HD 1080p video recording at 30 fps, automatic light correction, and a 78° field of view, making it well-suited for indoor monitoring scenarios with varying lighting conditions and angles. The camera specifications are based on the Logitech C920 Pro HD Webcam technical documentation [21].

The system's responsiveness and accuracy during these tests were highly satisfactory. Fall events were consistently detected in under two seconds, meeting real-time inference requirements. Importantly, the model was able to maintain its detection capabilities even in challenging configurations involving partial occlusions or non-frontal camera angles. Some examples of such detections are presented in Figure 2, which show actual fall episodes captured from different viewpoints during the testing phase.

From a broader perspective, the successful deployment of the model in a real-life care facility demonstrates the feasibility of employing deep learning-based fall detection on low-power edge devices. Compared to wearable sensors, which may be forgotten, incorrectly used, or outright rejected by older adults, a camera-based, contactless system provides continuous monitoring without requiring user compliance.

Moreover, in sensitive environments such as residential care homes, the system's design deliberately favors sensitivity over specificity: false positives are tolerated to ensure that true

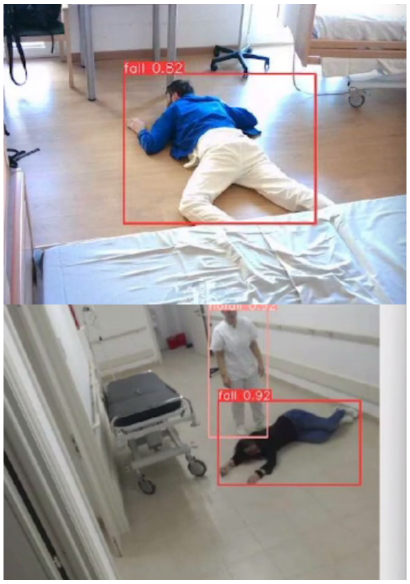

Fig. 2. Examples of detected fall events captured by the deployed system during field testing.

fall events are not missed. This trade-off is appropriate and desirable, as it enables faster staff response and reduces the risk of serious consequences resulting from undetected falls.

In conclusion, the proposed system not only shows promising technical results but also meets practical constraints and user needs, making it a compelling and affordable solution for fall detection in real-world eldercare settings.

IV. Future developments and conclusions

This paper presents a novel non-invasive fall detection system based on edge computing and computer vision techniques. Experimental results demonstrate that the YOLOv8n model, when deployed on low-cost Raspberry Pi devices, achieves high accuracy and real-time performance in detecting falls among elderly individuals. The system addresses the key limitations of wearable-based solutions, providing continuous monitoring without requiring user cooperation.

Looking towards future enhancements, the integration of distributed consensus algorithms presents a promising direction for improving system reliability and accuracy in multi-camera deployments.

A. Distributed consensus for multi-camera fall detection

The multi-camera fall detection system presented in this work inherently requires coordination among multiple autonomous units to reach a consensus on whether a fall has occurred. This scenario presents an ideal application for distributed consensus algorithms, which can enhance the reliability and fault tolerance of the detection system.

In the proposed architecture, each camera-equipped Raspberry Pi operates as an independent node within a distributed system, similar to the agents described in Dragoni and Puliti's work on distributed belief revision [22]. As noted in their research, ”to properly ’solve’ a contradiction it is necessary to keep information about *who* said it or, in general, about from *where* that information came” [22]. This principle directly applies to the multi-camera system presented in this study, where each node is required to maintain information regarding the origin and reliability of fall detection events.

Furthermore, the RAFT consensus algorithm [23] provides a robust foundation for coordinating decisions across the network of camera nodes. In the implemented architecture, each node is capable of proposing a fall detection event when its YOLOv8 model identifies a potential fall with high confidence. The RAFT leader election mechanism ensures that a designated node coordinates the consensus process by collecting votes from the others based on their individual detections. This approach proves particularly effective in scenarios where cameras have different viewing angles or where partial occlusions impact the visibility of specific scenes.

Drawing from Dragoni's distributed belief revision framework [24], the basic RAFT implementation can be enhanced by incorporating credibility metrics for each camera node. The credibility assigned to a node's detection can be dynamically adjusted based on factors such as:

- Historical accuracy of the node's detections;
- Current environmental conditions (lighting, occlusions);
- Consistency with detections from neighboring cameras;
- Network latency and communication reliability.

As demonstrated in distributed belief revision systems, ”the reliability of the source affects the credibility of the information and vice-versa” [24]. This bidirectional relationship is crucial within the proposed system, in which the reliability score assigned to each camera influences the weighting of its fall detection proposals during the consensus process.

The integration of RAFT with belief revision principles could offer several advantages for fall detection:

1) Fault tolerance: the system can continue operating even if one or more cameras fail or provide unreliable data;
2) Dynamic adaptation: camera credibility scores adjust over time based on performance, allowing the system to identify and compensate for degraded sensors;
3) Conflict resolution: when cameras disagree on a fall event, the consensus mechanism considers both the number of agreeing nodes and their individual credibility scores;
4) Reduced false positives: by requiring consensus among multiple cameras with verified credibility, the system minimizes false alarms while maintaining high sensitivity to actual falls.

B. Conclusions

This work contributes to the field of elderly care technology by demonstrating the feasibility of a practical, cost-effective fall detection system that operates entirely at the edge. The main contributions of the proposed approach include:

- Successful implementation of YOLOv8 on resource-constrained hardware, achieving inference times suitable for real-time monitoring;

- Validation in real-world residential care environments, demonstrating robustness to varying lighting conditions and complex scenarios;
- A fully non-invasive approach that respects user privacy while maintaining high detection accuracy;
- Scalable architecture that can be easily deployed across multiple rooms and facilities.

The experimental results confirm that the proposed system effectively distinguishes between actual falls and activities of daily living, with rapid response times that enable timely intervention. By eliminating the need for wearable devices, this approach addresses a critical gap in current fall detection technologies, particularly for users with cognitive impairments who may forget or refuse to wear monitoring devices.

As the global population continues to age, the demand for reliable, unobtrusive monitoring solutions will only increase. Edge-based computer vision systems, as demonstrated in this study, offer a viable answer to this challenge, by balancing performance — essential factors for scalable deployment. Future research will focus on implementing the distributed consensus mechanisms discussed above and conducting long-term studies to evaluate system performance across diverse populations and environments.

REFERENCES

[1] World Health Organization, Ageing and Life Course Unit, “WHO global report on falls prevention in older age,” World Health Organization, 2008.

[2] Z. Zhang, C. Conly, and V. Athitsos, “A survey on vision-based fall detection,” in Proc. 8th ACM Int. Conf. on Pervasive Technologies Related to Assistive Environments, 2015, pp. 1-7.

[3] B. Mirmahboub, S. Samavi, N. Karimi, and S. Shirani, “Automatic monocular system for human fall detection based on variations in silhouette area,” IEEE Trans. on Biomedical Engineering, vol. 60, no. 2, pp. 427-436, 2012.

[4] H. Foroughi, A. Naseri, A. Saberi, and H. Sadoghi Yazdi, “An eigenspace-based approach for human fall detection using ITMI and neural networks,” in Proc. 9th Int. Conf. on Signal Processing, IEEE, 2008.

[5] Z. Zhang, E. Becker, R. Arora, and V. Athitsos, “Experiments with computer vision methods for fall detection,” in Proc. 3rd Int. Conf. on Pervasive Technologies Related to Assistive Environments, ACM, 2010.

[6] W. Feng, R. Liu, and M. Zhu, “Fall detection for elderly person care in a vision-based home surveillance environment using a monocular camera,” Signal, Image and Video Processing, vol. 8, pp. 1129–1138, 2014.

[7] E. Auvinet et al., “Fall detection using body volume reconstruction and vertical repartition analysis,” in Image and Signal Processing (ICISP 2010), Lecture Notes in Computer Science, vol. 6134, Springer, 2010.

[8] N. Thome, S. Miguet, and S. Ambellouis, “A real-time, multiview fall detection system: A LHMM-based approach,” IEEE Trans. on Circuits and Systems for Video Technology, vol. 18, no. 11, pp. 1522–1532, 2008.

[9] A. Zweng, S. Zambanini, and M. Kampel, “Introducing a statistical behavior model into camera-based fall detection,” in Advances in Visual Computing: ISVC 2010, Springer, 2010.

[10] G. Diraco, A. Leone, and P. Siciliano, “An active vision system for fall detection and posture recognition in elderly healthcare,” in DATE 2010, IEEE, 2010.

[11] A. Leone, G. Diraco, and P. Siciliano, “Detecting falls with 3D range camera in ambient assisted living applications,” Medical Engineering & Physics, vol. 33, no. 6, pp. 770–781, 2011.

[12] Z. Zhang, W. Liu, V. Metsis, and V. Athitsos, “A viewpoint-independent statistical method for fall detection,” in Proc. 21st Int. Conf. on Pattern Recognition (ICPR), IEEE, 2012.

[13] Z.-P. Bian, J. Hou, L.-P. Chau, and N. Magnenat-Thalmann, “Fall detection based on body part tracking using a depth camera,” IEEE Journal of Biomedical and Health Informatics, vol. 19, no. 2, pp. 430–439, 2014.

[14] Z. Huang, Y. Liu, Y. Fang, and B. K. P. Horn, “Video-based fall detection for seniors with human pose estimation,” in 2018 Int. Conf. on Universal Village (UV), IEEE, 2018.

[15] W.-J. Chang, C.-H. Hsu, and L.-B. Chen, “A pose estimation-based fall detection methodology using artificial intelligence edge computing,” IEEE Access, vol. 9, pp. 129965–129976, 2021.

[16] Ultralytics YOLOv8. Available: https://github.com/ultralytics/ultralytics

[17] Le2i Dataset. Available: https://imvia.u-bourgogne.fr/en/database/fall-detection-dataset-2.html

[18] UR Fall Detection Dataset. Available: http://fenix.ur.edu.pl/ mkepski/ds/uf.html

[19] E-FPDS or FPDS_v2 Dataset. Available: https://gram.web.uah.es/data/datasets/fpds/index.html

[20] Computer Vision Annotation Tool (CVAT). Available: https://www.cvat.ai/

[21] Logitech, “C920 Pro HD Webcam Technical Specifications,” Logitech International S.A., 2024. Available: https://www.logitech.com/it-it/shop/p/c920-pro-hd-webcam.960-001055

[22] A. F. Dragoni and P. Puliti, “Distributed belief revision versus distributed truth maintenance,” in Proc. 6th IEEE Int. Conf. on Tools with A.I., New Orleans, 1994, pp. 499-505.

[23] D. Ongaro and J. Ousterhout, “In search of an understandable consensus algorithm,” in Proc. USENIX Annual Technical Conference, 2014, pp. 305-319.

[24] A. F. Dragoni, “Distributed decision support systems under limited degrees of competence: A simulation study,” Decision Support Systems, vol. 20, pp. 17-34, 1997.