Computing and Informatics, Vol. 45, 2026, 1208–1235, doi: 10.31577/cai_2026_5_1208

REAL-TIME FALL DETECTION WITH FINE-GRAINED POSTURE SEGMENTATION AND ENVIRONMENTAL CONTEXT AWARENESS

Seif Eddine Mili

Normal Superior School of Constantine
Computer Science Department – LISCO Laboratory
Constantine, Algeria
e-mail: mili.seifeddine@ensc.dz

Fouzi Lezzar*

Abdelhamid Mehri University-Constantine 2
LIRE Laboratory
Constantine, Algeria
e-mail: fouzi.lezzar@univ-constantine2.dz

Nabila Azeri

Abbes Laghrour University
Mathematics and Informatics Department
Khenchela, Algeria
e-mail: azeri.nabila@univ-khenchela.dz

Djamel Benmerzoug

Abdelhamid Mehri University-Constantine 2
LIRE Laboratory
Constantine, Algeria
e-mail: djamel.benmerzoug@univ-constantine2.dz

* Corresponding author

Abstract. Falls are a leading cause of severe injuries and fatalities among the elderly, particularly those living alone. Timely and accurate fall detection is critical to ensuring rapid medical intervention and mitigating associated health risks. Traditional fall detection methods, such as wearable sensors and conventional computer vision techniques, often suffer from high false alarm rates and limited adaptability to real-world environments. These limitations stem from challenges like occlusions, lighting variations, and the inability to reliably distinguish between actual falls and non-harmful postures, such as sitting or lying on a bed. To address these challenges, we propose a novel fall detection system that integrates fine-grained human posture segmentation with environmental context awareness. Our approach leverages an enhanced YOLOv11-Seg model to perform real-time semantic segmentation, enabling precise differentiation between actual falls and similar but non-critical scenarios. By analyzing the spatial relationship between the detected individual and surrounding furniture, our method significantly reduces false alarms while maintaining high detection accuracy. Experimental evaluations conducted across multiple benchmark datasets demonstrate the robustness of our solution, achieving superior performance in terms of accuracy, precision, recall, and F1-score. The proposed system not only outperforms existing state-of-the-art methods but also proves highly effective for real-world fall detection applications, offering a reliable and scalable solution to enhance elderly care and safety.

Keywords: Real-time fall detection, semantic segmentation, deep learning, transfer learning, YOLOv11-Seg, intersection

Mathematics Subject Classification 2010: 68T45, 68T10

1 INTRODUCTION

Falls are a critical indicator of declining physical health, mobility issues, and heightened vulnerability among the elderly. They represent a leading cause of injury-related deaths worldwide, particularly for older adults living alone. According to the World Health Organization (WHO) [1], approximately 30 % of individuals over 65 experience at least one fall annually, with this percentage increasing with age. The National Council on Aging (NCOA) [2] reports that in the United States, an older adult dies from a fall-related injury every 19 minutes, resulting in 27 000 deaths each year. Beyond fatalities, falls frequently lead to severe complications, including fractures, reduced mobility, and loss of independence, profoundly impacting the quality of life of elderly individuals. Moreover, delayed detection and response to a fall can exacerbate health deterioration and even result in death due to prolonged immobility and the absence of immediate medical intervention. Given these alarming statistics, the development of effective and reliable fall detection systems is imperative to ensure timely intervention and mitigate health risks.

Various fall detection systems have been explored in the literature, broadly categorized into wearable-based, ambient sensor-based, and vision-based methods [3]. Wearable sensors, such as accelerometers and gyroscopes, are widely adopted due to their portability and affordability [4]. However, these solutions require continuous user compliance, which is often impractical for elderly individuals. Ambient sensor-based methods, such as radar and microphone-based systems, eliminate the need for user participation but are highly susceptible to environmental noise and occlusions, limiting their reliability in real-world settings [4]. Among these approaches, vision-based deep learning methods have shown significant promise for automated, non-intrusive fall detection. Nevertheless, they face several challenges, including lighting variations, changes in body posture, occlusions, and background clutter [3, 5]. A major limitation of existing vision-based solutions is their inability to distinguish between a person lying on the floor due to a fall and someone resting on a bed or couch [6]. Many models rely exclusively on body posture estimation, overlooking environmental context, which results in high false-positive rates. For instance, a person lying horizontally in an image may be erroneously classified as having fallen, even if they are resting on a mattress. Additionally, occlusions caused by furniture or other objects further complicate detection, as many systems struggle to identify partially visible individuals. These shortcomings highlight the need for a more robust, context-aware fall detection system that integrates both posture recognition and environmental analysis to enhance accuracy and reduce false alarms.

To address these challenges, we propose an innovative system that improves fall detection by analysing the interaction between a person and their environment, particularly furniture. Unlike traditional methods that rely solely on body posture, our approach integrates spatial and contextual cues, enhancing differentiation between actual falls and non-fall scenarios. Our solution leverages a fine-tuned YOLOv11-Seg model, optimized to segment both the human body and surrounding furniture, enabling precise assessment of whether a person is lying on the floor or on a piece of furniture. By incorporating furniture interactions – such as beds or couches – we introduce critical contextual information that refines classification accuracy and reduces false alarms. Additionally, our approach significantly improves occlusion handling in complex environments, ensuring reliable performance even when parts of the body are obscured. This advancement makes our system more robust and accurate compared to existing methods, offering a comprehensive and reliable solution to fall detection. The main contributions of this work are as follows:

- Context-aware fall detection: A deep learning system that integrates human posture estimation with environmental analysis to enhance classification accuracy.
- Intersection-based fall identification: A novel spatial overlap approach that significantly reduces false alarms by leveraging interactions between individuals and their surroundings.

- Hybrid loss function: A tailored loss function designed to improve boundary segmentation accuracy for more precise detection.
- Fine-tuned YOLOv11-Seg model: An optimized model for real-time performance, capable of effectively handling complex scenarios in diverse environments.
- Multi-dataset training approach: Combines three public datasets with locally collected data, employing context-specific annotation techniques to improve model generalization across varied environments.
- Occlusion-aware detection strategy: Ensures reliable fall classification even when individuals are partially obscured by objects, enhancing robustness in challenging conditions.

The remainder of this paper is organized as follows: Section 2 presents a review of related fall detection methods. Section 3 describes the proposed approach, including dataset details, pre-processing steps, and model architecture. Section 4 presents the experimental results, while Section 5 discusses an ablation study, and Section 6 provides a comparative analysis with existing solutions. Finally, Section 7 concludes the paper and suggests future research directions.

2 RELATED WORK

In this section, we present a comprehensive review of recent literature on fall detection and classification, focusing on the application of deep learning models. In [7], a deep learning model combining a Temporal Convolutional Network (TCN) and a Gated Recurrent Unit (GRU) was proposed to extract high-level features for classification, using data collected from an Inertial Measurement Unit (IMU). Another study [8] introduced a battery-free human motion sensing system that recognizes movements via a shoe-embedded triboelectric nanogenerator (TENG), employing a conductive PVA-PEDOT:PSS hydrogel for signal collection. This system was further developed into an AI-based fall detection framework leveraging TENG technology. In [9], a compact Convolutional Neural Network (TinyCNN) with a two-stage feature extraction process was employed and evaluated on the KFall and SisFall public fall detection datasets. In [10], the authors proposed Flow-Pose Net (FP-Net), a two-stream network that integrates optical flow and human pose information for robust and accurate fall detection in videos. A human pose estimator identified body joint locations, feeding into a Graph Convolutional Network (GCN) to extract body appearance features, while a CNN was used for additional feature extraction. Similarly, [11] introduced the Multimodal Spatiotemporal Skeletal Kinematic Gait Feature Fusion (MSTSK-GFF) classifier for video-based fall detection. This method employs a multimodal feature fusion approach, combining a Spatiotemporal Graph Convolutional Network (STGCN) for skeletal modeling and 1D-CNNs for temporal gait modelling to generate two sets of spatiotemporal kinematic gait features from skeleton frame sequences, which are then fused through concatenation for classifica-

tion. To address data limitations and computational efficiency, [12] applied transfer learning by fine-tuning a pre-trained model on a large-scale image dataset, achieving a test accuracy of 98.15 % for fall detection. In [13], the authors developed ARFDNet, an efficient activity recognition and fall detection system. It uses a pose estimation network to extract skeletal features from raw RGB video, processing skeleton coordinates through a CNN, followed by GRUs and fully connected layers for classification. To tackle high false-positive rates, [14] proposed a multimodal CNN with various topologies, trained on RGB images and accelerometer data, and evaluated on the UR Fall and UP-Fall datasets. In [15], fall detection was performed by analysing human body geometry across video frames, using pose estimation to compute angles and distances between vectors formed by the head-centroid and the body's centre hip, relative to the horizontal axis, to create distinctive image features for classification. Additionally, [16] proposed a fall detection method using input-level data fusion, integrating 1D time-series data from wearable devices and 2D visual data from cameras to enhance information complementarity. In [17], the authors proposed a privacy-preserving, camera-based framework for fall detection and post-fall posture classification, employing a Multi-Scale Skip Connection Segmentation Network (MSSkip) to extract human silhouettes, and the Xception network for post-fall classification on a custom dataset, enhancing elderly monitoring while addressing privacy concerns. A real-time indoor fall detection method using computer vision was introduced in [18], integrating geometric features from dual ellipses – representing the head and torso – derived via a Gaussian Mixture Model (GMM), alongside a shallow CNN to improve detection accuracy. By extracting and fusing features such as axis ratios, orientation angles, and vertical velocities into a time-series motion feature, the authors effectively distinguished falls from similar activities. In [19], a segmentation pipeline was proposed to semi-automatically isolate individuals and objects in images, demonstrating that object-specific contextual transformations, such as Gaussian blur on backgrounds, enhance fall detection accuracy and generalization across ResNet-18, EfficientNetV2-S, and Swin-Small models. Finally, [20] utilized a microphone to record reflected signals, capturing the Doppler shift resulting from a fall event.

3 PROPOSED APPROACH

Current human fall detection systems face a significant limitation: beyond struggling with occlusion detection, they often fail to distinguish between a person lying on the floor and one resting on a bed, sofa, or other furniture. This distinction is critical for reducing false alarms and improving the contextual analysis of human postures. To address this challenge, we propose an approach based on intersection analysis between the person and surrounding furniture, calculating the spatial overlap between the human body and objects in the scene. To optimize performance and enhance accuracy, we first classify postures into lying and non-lying (e.g., standing, sitting) categories. This initial classification prevents unnecessary intersection calculations

for every detected person in the camera frame, significantly reducing computational load. By focusing solely on lying individuals, our method improves efficiency, making real-time execution more feasible. To further enhance reliability, our algorithm incorporates a Frame Count Rule, analysing a sequence of T consecutive frames to detect sustained lying postures. A fall is confirmed only if the intersection score remains below a predefined threshold in at least 80 % of these frames. This temporal filtering mitigates the misclassification of transient actions, such as bending down, as falls. To minimize errors, we intentionally avoid requiring 100 % frame agreement, allowing for occasional missed detections in some frames. By prioritizing posture consistency over transient fall dynamics and enforcing an 80 % agreement threshold, our approach effectively reduces false positives while maintaining high precision. This balance ensures robustness, making the method well-suited for real-time applications in healthcare and surveillance. We implement and compare three distinct scenarios (Figure 1) in terms of accuracy, computational efficiency, and practical relevance for real-time fall detection:

Scenario 1: Compute the intersection between the fully segmented human body and segmented furniture to determine contact areas.

Scenario 2: Segment individual body parts (head, upper limbs, and lower limbs), calculate the intersection of each part with segmented furniture for lying individuals, weight their contributions, and aggregate the results.

Scenario 3: Estimate intersection using bounding boxes of the person and furniture, leveraging Intersection over Union (IoU) for an approximate evaluation.



a) Original image b) Scenario 1 c) Scenario 2 d) Scenario 3

Figure 1. Representation of the three tested scenarios

3.1 Dataset

In this study, we utilized two public datasets – the Fall Detection Dataset [21] and Le2i [22] – for training, validation, and testing, supplemented by a third locally developed dataset for additional evaluation. The Fall Detection Dataset comprises RGB images resized to 320×240 pixels, originally recorded at 640×480 resolution using a single Kinect sensor mounted approximately 2.4 meters high. It contains 14 419 images, though not all were suitable for training. The dataset was captured in

various indoor environments from different angles and includes multiple individuals of diverse body sizes. The Le2i Fall Dataset consists of 124 RGB videos, including 94 fall sequences and 30 Activities of Daily Living (ADL) videos, recorded in diverse simulated environments. It captures falls from multiple perspectives – forward, backwards, and sideways – starting from various initial positions, such as standing and sitting. Additionally, it incorporates varied lighting conditions and occlusions to enhance realism. The videos were recorded using a single camera positioned at 2.4 meters high, with a resolution of 320×240 pixels and a frame rate of 25 fps. To integrate this dataset with the Fall Detection Dataset, we extracted 3 020 frames, alongside reserving 27 videos for testing. By combining the Le2i and Fall Detection datasets, we obtained a broader range of postures and fall scenarios, improving the robustness of the dataset for training fall detection models. To further validate our model, we developed a local dataset comprising 24 videos of both fall events and normal daily activities (ADL). This dataset includes four individuals of varying ages and body sizes, recorded in four distinct locations under diverse lighting conditions, ensuring greater diversity and enhancing generalization for real-world applications.

3.2 Dataset Preparation

The preprocessing phase was critical to ensure data consistency and class balance. We applied duplicate elimination and class-specific data augmentation strategies to enhance model performance. The dataset consists of four main classes:

Class 1 (Lying): Data augmentation was applied to increase variability and improve model generalization.

Class 2 (Not Lying): No additional preprocessing was performed beyond standard normalization, as this class already contained sufficient instances.

Class 3 (Furniture): Duplicate instances were removed to eliminate redundancy, followed by data augmentation to enhance robustness.

Body Part Classes: These include head, upper limbs, and lower limbs, enabling finer-grained segmentation for improved analysis. Data cleaning and augmentation techniques were applied to these classes.

Classes 1, 2, and 3 are utilized across all three scenarios, while the body part classes are used exclusively in the second scenario. The "Furniture" class encompasses three types of objects – beds, couches, and mattresses. Grouping these objects into a single class offers several advantages. First, it simplifies the model's classification task by reducing the number of categories, thereby improving generalization and minimizing the risk of misclassification among similar objects. Second, from a fall detection perspective, distinguishing between these objects is unnecessary, as their primary role is to support a lying person. This unified classification ensures more robust and efficient intersection analysis while maintaining the focus on differentiating between lying on furniture and lying on the floor. Additionally, this grouping optimizes training efficiency and enhances real-time performance by

reducing the number of object classes the model must process. Data augmentation techniques were applied to improve model generalization and robustness. To enhance the model's ability to handle diverse fall orientations and varying camera viewpoints, we incorporated rotation-based data augmentation techniques, including 90° clockwise and counterclockwise rotations, as well as upside-down flips. These transformations simulate how falls may appear from different angles – particularly in real-world deployments involving ceiling-mounted or wall-mounted cameras. These augmentations were applied only to images without furniture, as flipping or applying a 90° clockwise rotation to a person lying on a bed would not correspond to a plausible real-world situation. This approach allowed us to effectively synthesize new lying postures from standing positions (especially for ceiling-mounted cameras), helping to address the class imbalance between lying and non-lying instances present in the original dataset. This strategy was applied to improve model generalization and robustness. It included:

Rotations: 90° clockwise, counterclockwise, and upside down, as well as random rotations between −15° and +15°.

Shear transformations: horizontal and vertical shearing up to ±15°.

Brightness and exposure adjustments: Variations between −25 % and +25 %.

Noise injection: Modifying up to 5 % of pixels.

Cutout augmentation: Introducing random occlusions using five boxes, each covering 17 % of the image.

Mosaic augmentation: Enhancing object diversity by merging multiple images into a single training sample.

The labelling process was conducted twice: once for the first scenario (full-body segmentation) and again for the second scenario (body-part segmentation, including head, upper limbs, and lower limbs). This dual-labelling approach ensured that ground truth annotations aligned with each experimental setup. For the third scenario, we utilized the model trained for the second scenario. As we leveraged YOLOv11-Seg [23], which provides both segmentation masks and bounding boxes for the fine-tuned classes, we could directly use its outputs without additional labelling efforts.

Class	Lying	Not lying	Furniture	Head	Upper limb	Lower limb	Total
Before preprocessing	6 556	10 297	14 419	15 538	15 457	15 679	77 946
After preprocessing	8 730	10 297	9 364	9 583	9 572	9 627	57 173

Table 1. Dataset overview before and after preprocessing

Table 1 reports the number of instances rather than the number of images, as a single image may contain multiple objects from different classes. This distinction

is crucial in object detection and segmentation tasks, where each instance (e.g., person, furniture) is independently labelled and processed. The final dataset consists of 23 853 images (57 173 instances), divided into three subsets: training, validation, and testing. The training set comprises approximately 70 % of the dataset and was used for model training. The validation set accounts for 20 % of the dataset and was utilized for model validation and hyperparameter tuning. Finally, the test set represents 10 % of the dataset and includes 51 videos, serving to evaluate the model's performance after training and validation. It is important to note that the data allocated for validation and testing were not subjected to data augmentation, ensuring an unbiased evaluation of the model's performance in real-world scenarios.

3.3 YOLOv11-Seg Model Architecture and Fine-Tuning

We employed YOLOv11-Seg as the core segmentation model for our solution, leveraging its advanced architecture for precise object and posture identification. We selected YOLOv11-Seg for its speed and efficiency, making it an ideal choice for real-time applications [23]. Its enhanced segmentation capabilities enabled accurate differentiation between the human body, body parts, and surrounding furniture, ensuring robust fall detection. The model was trained twice:

1. For scenarios 1 and 3, using the classes "Lying," "Not Lying," and "Furniture."
2. For scenario 2, using all classes, including additional body part classes such as head, upper limbs, and lower limbs.

3.3.1 Fine-Tuned YOLOv11-Seg for Scenarios 1 and 3

The fine-tuned YOLOv11-Seg model (Table 2) demonstrates significant advancements in fall detection through targeted architectural modifications and optimized training strategies. The backbone employs a CSPDarknet architecture with five stages (64–512 channels), where the first three stages are frozen to leverage pretrained feature representations while allowing deeper layers to adapt to task-specific data, thereby balancing generalization and specialization. To refine domain-specific features, a Task-Specific Adapter was introduced, comprising two lightweight convolutional layers with 128 channels and a channel attention mechanism, enhancing the model's ability to distinguish between lying and non-lying states. Multi-scale feature fusion is achieved through a Path Aggregation Network (PANet) neck, which utilizes three feature pyramid levels, each with 256 channels, ensuring robust object detection across varying scales. The Detection Head, modified with three convolutional layers and 256 channels, predicts bounding boxes for three classes – Lying, Not-Lying, and Furniture – using three anchor boxes per scale, enabling precise localization and classification. Additionally, the Segmentation Head, based on a Fully Convolutional Network (FCN) architecture with four convolutional layers and 256 channels, generates class-specific instance segmentation masks at a resolution of 128×96, facilitating accurate localization of body parts and furniture. To

further improve spatial awareness, a Position-Sensitive Attention mechanism was incorporated, utilizing a single convolutional layer with 128 channels to generate a spatial attention map of size 32×24, focusing on positional relationships critical for differentiating object orientations. An auxiliary head, consisting of two fully connected layers with 128 neurons and softmax activation, provides supplementary classification supervision during training, enhancing the model's ability to accurately detect lying postures. Post-processing techniques, including Non-Maximum Suppression (NMS) with an IoU threshold of 0.5 and mask confidence thresholding at 0.3, ensure robust and reliable predictions by filtering redundant detections and refining mask quality. Training was conducted using the AdamW optimizer with a weight decay of 0.0001, a cosine decay learning rate scheduler initialized at 0.001, and an early stopping mechanism to prevent overfitting and ensure convergence. These enhancements collectively improve the precision, robustness, and real-time applicability of the YOLOv11-Seg model, making it highly effective for detecting lying postures and distinguishing falls from other activities in dynamic environments.

3.3.2 Fine-Tuned YOLOv11-Seg for Scenario 2

For Scenario 2, we adapt the fine-tuned YOLOv11-Seg architecture from the previous section, which focused on whole-body segmentation, by introducing detailed body part segmentation for lying and non-lying states while retaining furniture segmentation. The key differences lie in the network outputs, feature refinement, and training supervision to accommodate the new classes, including body parts. The segmentation head was modified to output masks for body part classes instead of full-body segmentation, enabling finer localization of the head, upper limbs, and lower limbs. Similarly, the detection head was adjusted to predict bounding boxes for these additional classes. To enhance part differentiation, the task-specific adapter was improved with additional convolutional layers, refining feature extraction. A position-sensitive attention mechanism was introduced to improve orientation awareness, capturing spatial relationships between body parts across different postures. Additionally, an the auxiliary is used to supervise the body part classes during training (excluding furniture) and removed during inference. These enhancements enable more precise segmentation of body parts, improving the model's ability to analyse human posture while preserving the efficiency of the original architecture.

3.4 Intersection Calculation

This section introduces the algorithms used to calculate the intersection area across the three defined scenarios.

The Scenario1-Intersection algorithm quantifies the interaction between a person and furniture using two key metrics: Intersection over Union (IoU) and Person Coverage, to accurately differentiate between incidental contact and true lying postures. IoU, a widely used object detection metric, is computed by measuring the

Component	Nature	Parameters	Frozen
Input	–	320×240 RGB image	Not trainable
Backbone	Convolutional layers	5 stages, 64–512 channels (first three stages frozen)	First three stages frozen
Task-specific adapter	Convolutional + Attention	2 convolutional layers, 128 channels; Channel attention mechanism	No
Neck	Convolutional layers	3 levels of feature pyramids; 256 channels per level	No
Auxiliary detection head	Fully connected + Convolutional	3 convolutional layers, 256 channels; 3 anchor boxes per scale	No
Segmentation head	Convolutional layers	4 convolutional layers, 256 channels; Output mask resolution: 128×96	No
Position-sensitive attention	Attention mechanism	1 convolutional layer, 128 channels; Spatial attention map size: 32×24	No
Auxiliary head	Fully connected + Convolutional	2 fully connected layers, 128 neurons; Softmax activation	No
Post-processing	Algorithmic (Non-trainable)	NMS IoU threshold: 0.5; Mask confidence threshold: 0.3	Not trainable

Table 2. Fine-tuned YOLOv11-Seg model components

pixel-wise intersection and union of the person and furniture masks. While a high IoU indicates significant overlap, it does not necessarily confirm that the person is lying down. To refine this assessment, Person Coverage is introduced, defined as the proportion of the person's area overlapping with the furniture. This ensures that minimal contact does not lead to misclassification. A high Person Coverage suggests that a substantial portion of the person is lying on the furniture, whereas a low value indicates a leaning or incidental posture. By combining IoU and Person Coverage, the algorithm distinguishes between different interaction scenarios:

Fully lying down: High IoU, high coverage.

Leaning: Medium IoU, medium coverage.

Slight contact: High IoU, low coverage.

No interaction: Low IoU, low coverage [21].

Algorithm 1 Scenario1-Intersection

```
Input: person-mask, furniture-mask
Output: iou, person-coverage
Step 1: Compute Intersection and Union
total-intersection ← Count(person-mask AND furniture-mask)
total-union ← Count(person-mask OR furniture-mask)
Step 2: Compute IoU (Intersection over Union)
if total-union > 0 then
    iou ← total-intersection / total-union
else
    iou ← 0
end if
Step 3: Compute Person Coverage
total-person-area ← TotalPixels(person-mask)
if total-person-area > 0 then
    person-coverage ← total-intersection/total-person-area
else
    person-coverage ← 0
end if
Step 4: Return IoU and Coverage
return (iou, person-coverage)
```

This differentiation is crucial for fall detection systems, where the goal is to determine whether a person has genuinely fallen onto furniture (e.g., a bed or couch) or is merely in contact with it.

The Scenario2-Intersection Algorithm enhances fall detection by systematically evaluating the spatial overlap between a detected person and furniture. It assigns importance weights to body regions based on their relevance to lying postures: head (0.5), upper limbs (0.4), and lower limbs (0.1), emphasizing the head and upper limbs as key fall indicators. The algorithm computes the binary intersection between each body part mask and the furniture mask, determining a coverage ratio by dividing the intersection area by the total area of each body part. To refine accuracy, a Gaussian-weighted adjustment dynamically modifies body part weights based on their Euclidean distance from the centre of mass, using an exponential decay factor $\alpha = 0.1$. This adaptation ensures resilience to variations in body positioning. The final normalized intersection score (ranging from 0 to 1) aggregates the adjusted coverage scores, where higher values indicate a greater likelihood that the person is lying on furniture rather than the floor. This method significantly improves precision by prioritizing key body parts and reducing false positives caused by limb variability.

Gaussian adjustments and noise filtering enhance robustness against occlusions (e.g., blankets), ensuring reliable posture estimation. Additionally, context-aware weight adjustments refine the system's ability to distinguish between lying and

leaning postures, addressing common misclassifications in traditional approaches. These enhancements make the algorithm highly suitable for real-time healthcare and surveillance applications, where accurate and timely fall detection is critical.

Algorithm 2 Scenario2-Intersection

Require: person-mask, furniture-mask, body-parts-masks
Define base weights:
head-weight $\leftarrow 0.5$
upper-limbs-weight $\leftarrow 0.4$
lower-limbs-weight $\leftarrow 0.1$
$\alpha \leftarrow 0.1$
min-coverage-threshold $\leftarrow 0.1$
Calculate the center of mass (CoM):
$(x_c, y_c) \leftarrow$ CenterOfMass(person-mask)
Initialize variables:
weighted-score $\leftarrow 0$
total-adjusted-weight $\leftarrow 0$
for each body-part in {head, upper-limbs, lower-limbs} **do**
    part-mask $\leftarrow$ body-parts-masks[body-part]
    part-intersection $\leftarrow$ Count(part-mask AND furniture-mask)
    if TotalPixels(part-mask) > 0 **then**
        part-coverage $\leftarrow$ part-intersection / TotalPixels(part-mask)
    else
        part-coverage $\leftarrow 0$
    end if
    if part-coverage $<$ min-coverage-threshold **then**
        part-coverage $\leftarrow 0$
    end if
    Compute centroid (x_p, y_p) of part-mask
    $d \leftarrow \sqrt{(x_p - x_c)^2 + (y_p - y_c)^2}$
    $G(d) \leftarrow \exp(-\alpha \cdot d^2)$
    adjusted-weight $\leftarrow$ base-weight[body-part] $\cdot\, G(d)$
    weighted-score $\leftarrow$ weighted-score + (part-coverage $\times$ adjusted-weight)
    total-adjusted-weight $\leftarrow$ total-adjusted-weight + adjusted-weight
end for
Normalize the score:
if total-adjusted-weight > 0 **then**
    normalized-score $\leftarrow$ weighted-score / total-adjusted-weight
else
    normalized-score $\leftarrow 0$
end if
Return normalized-score

The Scenario3 algorithm calculates the overlap score between the bounding boxes of a person and furniture in an image. It first computes the intersection of the two bounding boxes to determine the overlapping area. If no overlap exists, the intersection area is set to zero. The union of the bounding boxes is then calculated by summing their areas and subtracting the intersection to avoid double counting. Finally, the Intersection over Union (IoU) is computed as the ratio of the intersection area to the union area, providing a normalized measure (ranging from 0 to 1) of the spatial overlap between the person and furniture. This metric is widely adopted in object detection to evaluate spatial relationships between detected objects.

Algorithm 3 Scenario3-Intersection

Require: Bp: Bounding box of the person (x_p, y_p, w_p, h_p)
Require: Bf: Bounding box of the furniture (x_f, y_f, w_f, h_f)
Calculate Intersection Area:
$x_{\text{left}} \leftarrow \max(x_p, x_f)$
$y_{\text{top}} \leftarrow \max(y_p, y_f)$
$x_{\text{right}} \leftarrow \min(x_p + w_p, x_f + w_f)$
$y_{\text{bottom}} \leftarrow \min(y_p + h_p, y_f + h_f)$
if $x_{\text{right}} > x_{\text{left}}$ **and** $y_{\text{bottom}} > y_{\text{top}}$ **then**
    Intersection $\leftarrow (x_{\text{right}} - x_{\text{left}}) \times (y_{\text{bottom}} - y_{\text{top}})$
else
    Intersection $\leftarrow 0$
end if
Calculate Union Area:
$\text{Area}_p \leftarrow w_p \times h_p$
$\text{Area}_f \leftarrow w_f \times h_f$
Union $\leftarrow \text{Area}_p + \text{Area}_f - \text{Intersection}$
Compute IoU:
if Union > 0 **then**
    IoU $\leftarrow \frac{\text{Intersection}}{\text{Union}}$
else
    IoU $\leftarrow 0$
end if
Return IoU

3.5 Loss Function

For this project, we implemented a hybrid loss function tailored to our specific objectives, combining Dice loss [24] and Focal loss [25] for segmentation, alongside Complete Intersection-over-Union (CIoU) loss for bounding box regression. Dice loss optimizes the overlap between predicted and ground truth masks, enhancing segmentation accuracy. Focal loss focuses on hard-to-segment regions by reducing the

impact of easily classified pixels, thereby improving the model's ability to segment challenging areas. Meanwhile, CIoU loss refines bounding box regression by incorporating aspects such as aspect ratio consistency and distance between centroids, ensuring precise object localization. This synergistic approach enhances segmentation precision, minimizes classification errors, and improves spatial accuracy. The Dice loss is defined as:

$$\text{Dice loss} = 1 - \frac{1}{C}\sum_{c=1}^{C}\left(\frac{2\sum_i(\text{gt}_{c,i}\cdot\text{pr}_{c,i})}{\sum_i \text{gt}_{c,i} + \sum_i \text{pr}_{c,i} + \varepsilon}\right). \tag{1}$$

In this formulation, C represents the number of classes (3 for the first model and 6 for the second), $\text{gt}_{c,i}$ denotes the ground truth (0 or 1) for class c at pixel i, $\text{pr}_{c,i}$ is the predicted probability, and ε ensures numerical stability by preventing division by zero. The Focal loss is expressed as:

$$\text{Focal loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C}\left[\alpha\cdot\text{gt}_{c,i}\cdot(1-\text{pr}_{c,i})^{\gamma}\cdot\log(\text{pr}_{c,i})\right] \tag{2}$$

with N as the total pixels, α (0.25 for scenarios 1 and 3, and 0.35 for scenario 2) weighting the minority classes, and $\gamma = 2$ focusing on difficult pixels like boundaries. The total hybrid loss is:

$$\text{L} = \lambda_1\cdot\text{Dice loss} + \lambda_2\cdot\text{Focal loss}, \tag{3}$$

where $\lambda_1 = 1.0$ prioritizes overlap accuracy and $\lambda_2 = 0.5$ refines edges, empirically tuned to enhance mask quality for fall detection. For bounding box regression, we use the Complete IoU (CIoU) loss, defined as:

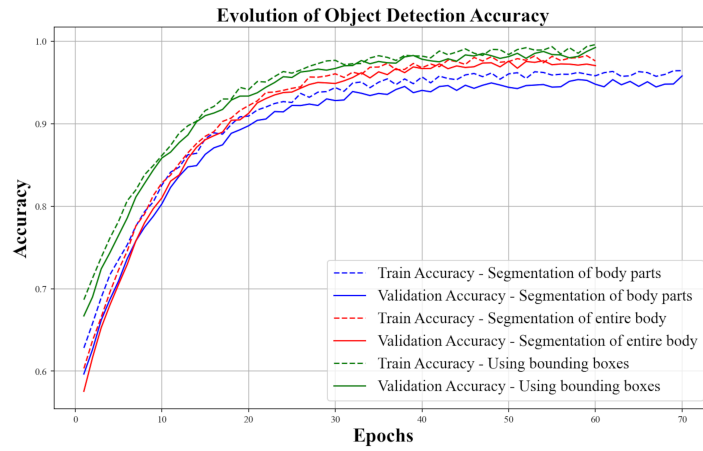
$$\text{L}_{\text{Bbox}} = 1 - \left(\text{IoU} - \frac{p^2(b, b_{\text{gt}})}{c^2} - \alpha\cdot v\right). \tag{4}$$

CIoU loss improves localization accuracy by integrating IoU, which quantifies the overlap between the predicted bounding box b and the ground truth b_{gt}, where a higher IoU indicates better localization [26]. It also includes a centre distance penalty term $\rho^2(b, b_{\text{gt}})$, which normalizes the squared Euclidean distance between their centres using c^2 (the diagonal of the smallest enclosing box) to enhance convergence and alignment [27]. Additionally, an adaptive weighting factor α balances the aspect ratio penalty v, ensuring realistic bounding box proportions [27]. The optimal parameters, $\alpha = 0.5$ and a centre distance penalty coefficient of 1, contribute to precise localization, which is crucial for fall detection.

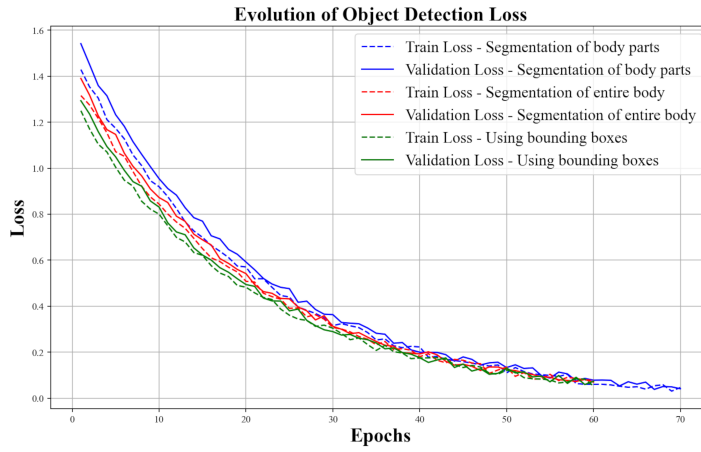
4 EXPERIMENTAL RESULTS

The proposed models were evaluated in an experimental environment equipped with an 11$^{\text{th}}$ Gen Intel(R) Core(TM) i5 processor and an RTX 3050 graphics card. The

training process for YOLOv11-Seg was conducted for both segmentation models, with an initial epoch count of 120 and a batch size of 16. However, early stopping was applied due to the absence of significant performance improvements beyond the 55^{th} epoch for Model 1 and the 65^{th} epoch for Model 2. Consequently, the final number of epochs was set to 60 for Model 1 and 70 for Model 2 to ensure stability and convergence. The performance and loss graphs for both scenarios are presented in Figure 2.



a)



b)

Figure 2. Train and validation accuracy and loss of YOLOv11-Seg for the two models

For object detection (without fall detection), Model 1 demonstrated exceptional accuracy with bounding boxes, achieving a training accuracy of 99.6 % and a vali-

dation accuracy of 99.42 %. The loss function consistently decreased during training before stabilizing (Figure 2). When applied to whole-body segmentation, the same model exhibited a similar trend, attaining a training accuracy of 97.61 % and a validation accuracy of 97.32 %, with the loss function steadily declining before stabilization. Model 2, focused on body part segmentation, showed slightly lower but still competitive performance, with a training accuracy of 96.44 % and a validation accuracy of 96.20 %. Its loss function also decreased steadily before stabilizing. However, this model required a longer training duration, converging after 70 epochs compared to 60 epochs for Model 1.

Model	Scenario	Accuracy	Precision	Recall	F1-Score
Model 1	Segment the whole body	97.21 %	95.71 %	95.96 %	95.83 %
Model 2	Segment body parts	95.89 %	95.91 %	95.89 %	95.90 %
Model 1	Using bounding boxes	99.29 %	98.99 %	98.88 %	98.93 %

Table 3. Performance metrics of YOLOv11-Seg for object detection step

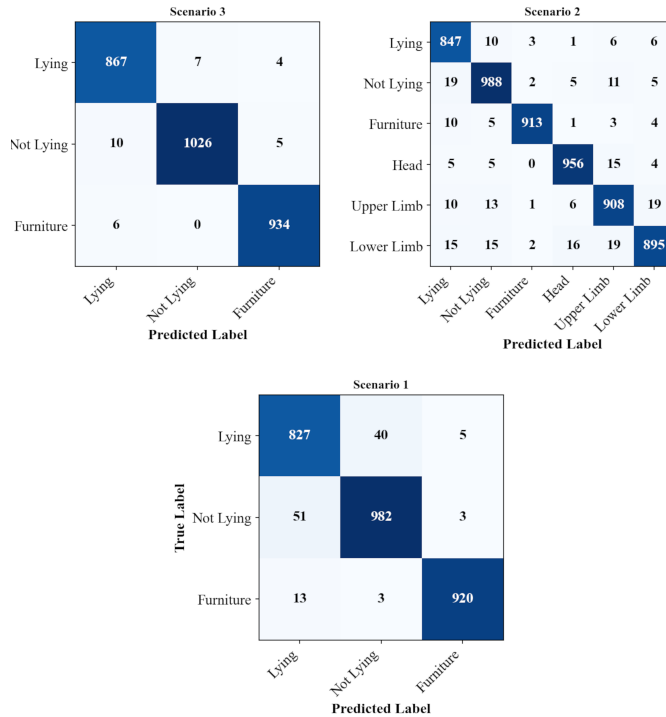


Figure 3. Confusion matrices of object detection step

The results presented in Table 3 and Figure 3 evaluate the three object detection techniques on the test set, demonstrating their ability to accurately detect and classify the following classes: Lying, Not lying, Furniture, Head, Upper limb, and Lower limb. The segmentation-based approaches (whole-body segmentation and body part segmentation) and the bounding box-based detection method were assessed using precision, recall, F1-score, and accuracy metrics. The bounding box-based detection method achieves the highest accuracy (99.29 %) with consistently high scores across all three primary classes (Lying, Not lying, and Furniture), indicating its effectiveness in object localization. The whole-body segmentation technique provides a more detailed understanding of object shapes and maintains a high accuracy of 97.21 %, showcasing its robustness in capturing spatial details.

The body part segmentation approach, while achieving a slightly lower accuracy of 95.89 % due to the increased complexity of detecting additional classes (Head, Upper limb, and Lower limb), remains competitive. Despite these variations, all three methods demonstrate high accuracy, underscoring their effectiveness in object detection tasks.

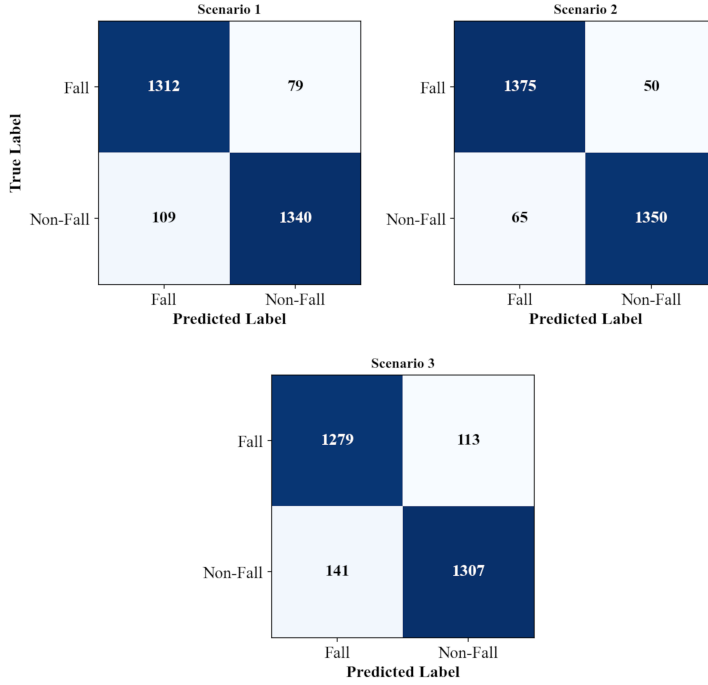


Figure 4. Confusion matrices of the three scenarios on the test set

Table 4 and Figure 4 present the evaluation results on images from the test set, highlighting distinct strengths and weaknesses in fall and non-fall detection across

Scenario	Acc.	Fall			No Fall			FPS
		Prec.	Recall	F1	Prec.	Recall	F1	
1	93.38 %	92.33 %	94.32 %	93.31 %	94.43 %	92.48 %	93.44 %	13
2	95.95 %	95.49 %	96.49 %	95.99 %	96.43 %	95.41 %	95.91 %	11
3	91.06 %	90.07 %	91.88 %	90.97 %	92.04 %	90.26 %	91.14 %	18

Table 4. Performance comparison of different scenarios in fall detection

the three scenarios. Scenario 2 (body part segmentation) achieves the highest overall accuracy (95.95 %) and excels in fall detection with an F1-score of 95.99 %, demonstrating its superior ability to differentiate between a person lying on the floor and on furniture by analysing individual body parts. This detailed segmentation makes it the most robust against occlusions and complex postures. However, its higher computational cost results in the lowest frame rate (11 fps), limiting its real-time performance. Scenario 1 (whole-body segmentation) strikes a balance between accuracy (93.38 %) and speed (13 fps), effectively detecting falls with an F1-score of 93.31 %. Although it may struggle with occlusions, it remains a strong candidate for real-time applications requiring both accuracy and efficiency. Scenario 3 (bounding box-based detection), despite achieving the highest frame rate (18 fps), struggles most with fall detection, with an F1-score of 90.97 % and an accuracy of 91.06 %. This indicates that bounding boxes alone lack the spatial precision needed to distinguish falls from other postures, as strong object localization alone is insufficient for accurate fall detection. However, it performs reasonably well in identifying non-fall situations, achieving an F1-score of 91.14 %. Ultimately, Scenario 2 is the most reliable for accurately detecting fall events, while Scenario 1 offers a practical compromise between detection capability and real-time feasibility. Scenario 3 prioritizes speed over precision, making it better suited for applications where high frame rates are critical. Nevertheless, even at lower FPS, the system remains well within the requirements for fall detection, as falls typically occur over several frames. Notably, studies such as [28] have demonstrated that processing even a single frame per second can yield effective results, achieving an accuracy of 96.9 %.

Scenario	Accuracy	Correct Detection		Incorrect Detection	
		Fall videos	ADL videos	Fall videos	ADL videos
Scenario 1	90.20 %	22	24	3	2
Scenario 2	96.08 %	24	25	2	0
Scenario 3	86.27 %	21	23	5	2

Table 5. YOLOv11-Seg model evaluation performance on test set (Videos set)

To further evaluate the effectiveness of our fall detection system, we tested the three proposed scenarios using 51 test videos that included both fall events and Activities of Daily Living (ADL). The performance of each scenario was assessed by measuring correctly detected falls, correctly identified ADL videos, and incorrect detections (false positives and false negatives). The results are summarized in

Table 5. The results indicate that Scenario 2 (Body Part Segmentation) achieves the highest accuracy (96.08 %), correctly detecting the most fall and ADL videos while exhibiting the lowest number of incorrect detections (2 false negatives and no false positives). This suggests that fine-grained anatomical segmentation significantly enhances the system's ability to differentiate between falls and non-fall activities, reducing the risk of misclassification. In contrast, Scenario 3 (Bounding Boxes) demonstrates the lowest accuracy (86.27 %) with the highest number of false negatives (5), indicating its inability to capture detailed body posture variations, leading to frequent misclassification of falls as non-fall activities. Scenario 1 (Whole Body Segmentation) offers a balanced trade-off between accuracy (90.20 %) and computational efficiency but still results in some misclassifications (3 false negatives and 2 false positives), highlighting its limitations in distinguishing complex postural transitions. These findings emphasize the importance of fine anatomical segmentation for enhancing fall detection accuracy while minimizing false positives and false negatives.

5 ABLATION STUDY

In this section, we conduct an ablation study to analyse the impact of different components within our proposed approach on the achieved results. We evaluate four configurations: excluding Dice loss and Focal loss, excluding occlusion scenarios during testing, and removing intersection algorithms (Table 6). Using only Dice loss leads to a noticeable decline in performance across all scenarios, with a stronger impact on fall recall. In contrast, using only Focal loss results in a smaller performance drop, particularly improving fall recall compared to Dice loss alone. This confirms that their combination is essential for optimizing both fall and non-fall detection.

Ablation factor	Sc.	Acc.	Fall			No Fall		
			Prec.	Rec.	F1	Rec.	Prec.	F1
Dice loss	1	90.33	89.75	90.71	90.23	90.91	89.97	90.44
only	2	93.64	93.40	94.06	93.73	93.88	93.21	93.54
($\lambda_2 = 0$)	3	88.03	87.28	88.53	87.90	88.77	87.54	88.15
Focal loss	1	92.08	91.20	92.83	92.01	92.96	91.35	92.15
only	2	94.73	94.44	95.10	94.77	95.02	94.35	94.68
($\lambda_1 = 0$)	3	89.77	88.73	90.65	89.68	90.81	88.93	89.86
Without	1	94.72	93.08	96.42	94.72	96.42	93.08	94.72
consideration	2	96.84	96.19	97.54	96.86	97.51	96.14	96.82
of occlusions	3	92.81	91.64	93.93	92.77	93.99	91.72	92.84
Without fall	1	88.93	82.86	99.66	90.49	99.50	76.92	86.76
intersection	2	91.13	85.55	99.87	92.16	99.82	81.59	89.79
algorithms	3	86.96	79.94	99.50	88.65	99.29	73.78	84.65

Table 6. YOLOv11-Seg model evaluation performance on test set (Videos set)

When occluded images are excluded from the test set (i.e., without considering occlusions), performance improves across all scenarios, particularly in Scenario 1 (+1.34 % in accuracy) and Scenario 3 (+1.75 % in accuracy). This indicates that the techniques used in these two scenarios are less effective at handling occlusions compared to those in Scenario 2. Scenario 2 shows the smallest accuracy increase (+0.89 %), as it was already efficient at detecting occlusions in the baseline, reinforcing the advantage of body part segmentation in managing such challenges.

Removing fall detection algorithms and identifying lying postures as fall situations results in a significant drop in precision, as many non-fall lying postures are misclassified as falls. Fall recall increases dramatically (e.g., 99.5 % in Scenario 3) due to this overestimation, but the fall F1-Score decreases, demonstrating that explicit fall detection mechanisms (i.e., the three proposed algorithms) are critical to minimizing false positives. Scenario 2 provides the most balanced performance, suggesting that body part segmentation better distinguishes between intentional lying and accidental falls. In this configuration, we replaced intersection algorithms with simpler detection techniques based on the posture of the detected person in Scenarios 1 and 3, and the position of body parts in Scenario 2.

This ablation study highlights that body part segmentation (Scenario 2) offers the best trade-off between fall and non-fall detection, while bounding boxes (Scenario 3) struggle most with misclassifications. The results confirm that hybrid loss functions and explicit fall detection mechanisms are essential for accurate and reliable fall detection in real-world scenarios. Additionally, this study demonstrates that the techniques proposed in Scenario 2 enhance the fall detection process, particularly in complex situations often overlooked in the literature, such as occlusions and distinguishing between different lying positions.

Scenario	Acc.	Fall			No Fall			FPS
		Prec.	Rec.	F1	Prec.	Rec.	F1	
YOLOv11-Seg (Sc. 1)	93.38	92.33	94.32	93.31	94.43	92.48	93.44	13
U-Net++ (Sc. 1)	95.05	92.68	95.00	93.83	96.67	95.08	95.87	7.5
YOLOv11-Seg (Sc. 2)	95.95	95.49	96.49	95.99	96.43	95.41	95.91	11
U-Net++ (Sc. 2)	96.97	95.89	97.22	96.55	97.83	96.77	97.30	6

Table 7. Performance comparison of YOLOv11-Seg and U-Net++

To further evaluate the effectiveness of our proposed YOLOv11-Seg-based fall detection system, we compared its performance in Table 7 against an alternative architecture – U-Net++ [29] – a model widely adopted in semantic segmentation tasks and known for its strong performance in segmenting complex structures such as organs and body parts. U-Net++, an advanced variant of the original U-Net architecture, introduces nested skip connections and deep supervision, which significantly enhance boundary precision and contextual understanding – particularly valuable

when segmenting fine-grained body parts. These characteristics make U-Net++ especially suitable for applications requiring detailed posture analysis, such as distinguishing between lying on furniture and actual falls. Both models were trained on the same combined dataset using identical preprocessing, data augmentation strategies, and class definitions, ensuring a fair and direct comparison. The comparison between YOLOv11-Seg and U-Net++ across Scenarios 1 and 2 highlights a clear trade-off between segmentation accuracy and real-time performance. Scenario 3, which relies on bounding box intersection (IoU) for fall detection, cannot be implemented using U-Net or U-Net++ architectures. These models are designed specifically for semantic segmentation and do not generate bounding boxes as part of their output. Unlike YOLOv11-Seg, which provides both segmentation masks and object detection outputs, U-Net-based architectures focus solely on pixel-wise classification, making them incompatible with pipeline stages that require region proposals or bounding box regression. Therefore, any system based on U-Net must rely on full or fine-grained segmentation outputs, limiting its use to Scenarios 1 and 2. In Scenario 1 (Whole-body segmentation), YOLOv11-Seg achieves an accuracy of 93.38 %, with a fall F1-score of 93.31 % and a no-fall F1-score of 93.44 %, operating at a real-time inference speed of 13 fps. When U-Net++ is applied to the same scenario, it demonstrates improved classification performance, achieving an accuracy of 95.05 % and a no-fall F1-score of 95.87 %, indicating better contextual understanding and boundary precision – particularly in distinguishing lying on furniture from actual falls. However, its inference speed drops significantly to 7.5 fps, making it less suitable for real-time deployment. In Scenario 2 (Body part segmentation), which introduces finer-grained analysis through individual body part detection, YOLOv11-Seg improves its performance to an accuracy of 95.95 %, with fall and no-fall F1-scores of 95.99 % and 95.91 %, respectively, while maintaining a frame rate of 11 fps. U-Net++, benefiting from its advanced architecture and nested skip connections, further enhances these results, reaching an accuracy of 96.97 % and a no-fall F1-score of 97.30 %, demonstrating superior ability to model complex postures and occlusions. However, its computational demands increase significantly, limiting its inference speed to only 6 fps. These findings highlight the trade-off between segmentation accuracy and computational efficiency, reinforcing the practical advantage of YOLOv11-Seg in real-world fall detection systems where timely response, environmental context awareness, and architectural versatility are essential.

6 COMPARATIVE STUDY

Our proposed approach, leveraging YOLOv11-Seg for fall detection using RGB camera input, demonstrates superior robustness compared to existing methods by addressing critical real-world challenges often overlooked in the literature (Table 8). While state-of-the-art models such as [6, 11, 13, 15, 16, 17] achieve high accuracy (up to 99.93 %) and precision (99.01 % in [6]), they fail to address occlusions and

the critical distinction between a person lying on a bed (resting) versus the floor (falling) – key factors for practical fall detection.

Ref.	Input type	Dataset	Occlusion handling	Context awareness	Classifier	Model performance
[6]	RGB camera	IASLAB, URFD	No	Yes	Proposed	Recall: 83.37 %
[7]	IMU	MobiAct, Sisfall	No	No	TCN + GRU	Accuracy: 99.5 %
[13]	RGB camera	UP-Fall, ADLF	No	No	CNN, GRU, ANN	Accuracy: 96.7 %
[15]	RGB camera	Le2iFD, URFD	No	No	TCN, LSTM	Accuracy: 97.02 %
[16]	RGB camera	UP-Fall	No	No	CNN	Accuracy: 99.93 %
[17]	RGB camera	UP-Fall	No	No	ConvLSTM	F1-score: 97.68 %
[30]	RGB camera	MultiCam, Le2i	Yes	No	ANN	Recall: 89.92 %
[31]	MCU	UMAFALL, FallAllD	No	No	CNN	Accuracy: 97.91 %
Our approach	RGB camera	Le2i, FDD, local	Yes	Yes	YOLOv11-Seg	Accuracy: 95.95 %

Table 8. Comparison of fall detection methods

Studies like [7, 11, 13, 15, 16, 31] which rely on LSTM, CNNs, STGCNs, and TCNs, focus solely on detecting lying postures without contextual differentiation, rendering them prone to misclassifications in complex scenes. Similarly, [6] achieves high precision but suffers from lower recall (83.37 %) due to its reliance on a simplistic height-based approach to determine whether a person is lying on a bed or the floor. This method is highly sensitive to the person's distance from the camera, leading to misclassifications when depth variations affect perceived height. Consequently, its reliability is compromised, limiting its ability to detect all fall events accurately. Likewise, [30], while handling occlusions, fails to distinguish falls from resting states, resulting in a modest recall (89.91 %).

In contrast, our method integrates fine-grained whole-body and furniture segmentation, enabling precise scene understanding and robust performance even in the presence of occlusions. Trained and tested on diverse datasets (Le2i, FDD, and a local dataset) – unlike the more limited datasets used in prior works (e.g., UP-Fall in [16] and [17]) – our approach enhances generalizability. Although our accuracy of 95.95 % is slightly below some benchmarks, this trade-off reflects the increased robustness and practical applicability of our system. By effectively addressing scenarios that previous methods fail to handle, our approach is better suited for deployment in uncontrolled environments.

7 CONCLUSION

This paper presents a fall detection system that uses an RGB camera to identify fall incidents while addressing challenges such as occlusions and distinguishing between individuals lying on the floor versus a bed. Unlike conventional methods that rely solely on body posture, our approach integrates segmentation and intersection analysis to accurately classify a person on the floor as a fall and on a bed as a non-fall. We fine-tuned a YOLOv11-based segmentation model using a hybrid loss function combining Dice loss and Focal loss to improve segmentation accuracy. The system detects both the person and surrounding furniture, then calculates their intersection to determine the likelihood of a fall. To enhance performance, we combined multiple datasets, yielding 23 853 pre-processed images (containing 57 173 instances) and 51 videos, with an evaluation set of 2 385 images and 51 videos. Our model achieved strong performance, with an accuracy of 95.95 % and a fall F1-score of 95.99 %, highlighting the effectiveness of segmentation-based intersection analysis in reducing false positives.

While our approach markedly improves detection reliability, future work will focus on deploying the model on edge devices for real-time processing, integrating multi-camera setups to address occlusions, incorporating 3D pose estimation and depth sensing for better fall differentiation, and adding temporal motion analysis to enhance robustness. Additional directions include adaptive learning for real-time model updates based on feedback, multimodal fusion with thermal or depth sensors for improved detection in low-light conditions, energy-efficient AI models for mobile applications, and a hierarchical alert system to prioritize critical falls while minimizing false alarms. By addressing these aspects, our system can evolve into a more intelligent, scalable, and efficient fall detection solution for elderly care and healthcare monitoring.

REFERENCES

[1] World Health Organization: Falls. 2021, `www.who.int/news-room/fact-sheets/detail/falls`.

[2] NCOA – National Council on Aging. `https://www.ncoa.org/`.

[3] Newaz, N. T.—Hanada, E.: The Methods of Fall Detection: A Literature Review. Sensors, Vol. 23, 2023, No. 11, Art. No. 5212, doi: 10.3390/s23115212.

[4] Ramachandran, A.—Karuppiah, A.: A Survey on Recent Advances in Wearable Fall Detection Systems. BioMed Research International, Vol. 2020, 2020, No. 1, Art. No. 2167160, doi: 10.1155/2020/2167160.

[5] Gutiérrez, J.—Rodríguez, V.—Martin, S.: Comprehensive Review of Vision-Based Fall Detection Systems. Sensors, Vol. 21, 2021, No. 3, Art. No. 947, doi: 10.3390/s21030947.

[6] Lafuente-Arroyo, S.—Martín-Martín, P.—Iglesias-Iglesias, C.—Maldonado-Bascón, S.—Acevedo-Rodríguez, F. J.: RGB Camera-Based

Fallen Person Detection System Embedded on a Mobile Platform. Expert Systems with Applications, Vol. 197, 2022, Art. No. 116715, doi: 10.1016/j.eswa.2022.116715.

[7] Li, Y.—Zuo, Z.—Pan, J.: Sensor-Based Fall Detection Using a Combination Model of a Temporal Convolutional Network and a Gated Recurrent Unit. Future Generation Computer Systems, Vol. 139, 2023, pp. 53–63, doi: 10.1016/j.future.2022.09.011.

[8] Wang, S.—Gao, J.—Lu, F.—Wang, F.—You, Z.—Huang, M.—Fang, W.—Liu, X.—Li, Y.—Liu, Y.: Human Motion Recognition by a Shoes-Floor Triboelectric Nanogenerator and Its Application in Fall Detection. Nano Energy, Vol. 108, 2023, Art. No. 108230, doi: 10.1016/j.nanoen.2023.108230.

[9] Yu, X.—Park, S.—Kim, D.—Kim, E.—Kim, J.—Kim, W.—An, Y.—Xiong, S.: A Practical Wearable Fall Detection System Based on Tiny Convolutional Neural Networks. Biomedical Signal Processing and Control, Vol. 86, Part C, 2023, Art. No. 105325, doi: 10.1016/j.bspc.2023.105325.

[10] Fei, K.—Wang, C.—Zhang, J.—Liu, Y.—Xie, X.—Tu, Z.: Flow-Pose Net: An Effective Two-Stream Network for Fall Detection. The Visual Computer, Vol. 39, 2023, No. 6, pp. 2305–2320, doi: 10.1007/s00371-022-02416-2.

[11] Amsaprabhaa, M.—Jane, Y. N.—Nehemiah, H. K.: Multimodal Spatiotemporal Skeletal Kinematic Gait Feature Fusion for Vision-Based Fall Detection. Expert Systems with Applications, Vol. 212, 2023, Art. No. 118681, doi: 10.1016/j.eswa.2022.118681.

[12] Patel, A. N.—Murugan, R.—Maddikunta, P. K. R.—Yenduri, G.—Jhaveri, R. H.—Zhu, Y.—Gadekallu, T. R.: AI-Powered Trustable and Explainable Fall Detection System Using Transfer Learning. Image and Vision Computing, Vol. 149, 2024, Art. No. 105164, doi: 10.1016/j.imavis.2024.105164.

[13] Yadav, S. K.—Luthra, A.—Tiwari, K.—Pandey, H. M.—Akbar, S. A.: ARFDNet: An Efficient Activity Recognition & Fall Detection System Using Latent Feature Pooling. Knowledge-Based Systems, Vol. 239, 2022, Art. No. 107948, doi: 10.1016/j.knosys.2021.107948.

[14] Galvão, Y. M.—Ferreira, J.—Albuquerque, V. A.—Barros, P.—Fernandes, B. J. T.: A Multimodal Approach Using Deep Learning for Fall Detection. Expert Systems with Applications, Vol. 168, 2021, Art. No. 114226, doi: 10.1016/j.eswa.2020.114226.

[15] Beddiar, D. R.—Oussalah, M.—Nini, B.: Fall Detection Using Body Geometry and Human Pose Estimation in Video Sequences. Journal of Visual Communication and Image Representation, Vol. 82, 2022, Art. No. 103407, doi: 10.1016/j.jvcir.2021.103407.

[16] Qi, P.—Chiaro, D.—Piccialli, F.: FL-FD: Federated Learning-Based Fall Detection with Multimodal Data Fusion. Information Fusion, Vol. 99, 2023, Art. No. 101890, doi: 10.1016/j.inffus.2023.101890.

[17] Mobsite, S.—Alaoui, N.—Boulmalf, M.—Ghogho, M.: Semantic Segmentation-Based System for Fall Detection and Post-Fall Posture Classification. Engineering Applications of Artificial Intelligence, Vol. 117, 2023, Art. No. 105616, doi: 10.1016/j.engappai.2022.105616.

[18] Yao, C.—Hu, J.—Min, W.—Deng, Z.—Zou, S.—Min, W.: A Novel Real-Time Fall Detection Method Based on Head Segmentation and Convolutional Neural Network. Journal of Real-Time Image Processing, Vol. 17, 2020, No. 6, pp. 1939–1949, doi: 10.1007/s11554-020-00982-z.

[19] Nagaj, A.—Li, Z.—Papadopoulos, D. P.—Nasrollahi, K.: Visual Context-Aware Person Fall Detection. In: Czarnowski, I., Howlett, R. J., Jain, L. C. (Eds.): Intelligent Decision Technologies (KESIDT 2024). Springer, Singapore, Smart Innovation, Systems and Technologie, Vol. 411, 2024, pp. 215–226, doi: 10.1007/978-981-97-7419-7_19.

[20] Lian, J.—Yuan, X.—Li, M.—Tzeng, N. F.: Fall Detection via Inaudible Acoustic Sensing. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, Vol. 5, 2021, No. 3, Art. No. 114, doi: 10.1145/3478094.

[21] Fall Detection Dataset. `https://falldataset.com/`.

[22] Le2i Fall Dataset. `www.kaggle.com/datasets/tuyenldvn/falldataset-imvia`.

[23] Khanam, R.—Hussain, M.: YOLOv11: An Overview of the Key Architectural Enhancements. 2024, doi: 10.48550/arXiv.2410.17725 (arXiv Preprint arXiv:2410.17725).

[24] Zhao, R.—Qian, B.—Zhang, X.—Li, Y.—Wei, R.—Liu, Y.—Pan, Y.: Rethinking Dice Loss for Medical Image Segmentation. 2020 IEEE International Conference on Data Mining (ICDM), 2020, pp. 851–860, doi: 10.1109/ICDM50108.2020.00094.

[25] Lin, T. Y.—Goyal, P.—Girshick, R.—He, K.—Dollár, P.: Focal Loss for Dense Object Detection. 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2999–3007, doi: 10.1109/ICCV.2017.324.

[26] Rezatofighi, H.—Tsoi, N.—Gwak, J.—Sadeghian, A.—Reid, I.—Savarese, S.: Generalized Intersection over Union: A Metric and a Loss for Bounding Box Regression. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 658–666, doi: 10.1109/CVPR.2019.00075.

[27] Zheng, Z.—Wang, P.—Liu, W.—Li, J.—Ye, R.—Ren, D.: Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, No. 7, pp. 12993–13000, doi: 10.1609/aaai.v34i07.6999.

[28] De Miguel, K.—Brunete, A.—Hernando, M.—Gambao, E.: Home Camera-Based Fall Detection System for the Elderly. Sensors, Vol. 17, 2017, No. 12, Art. No. 2864, doi: 10.3390/s17122864.

[29] Zhou, Z.—Rahman Siddiquee, M. M.—Tajbakhsh, N.—Liang, J.: UNet++: A Nested U-Net Architecture for Medical Image Segmentation. In: Stoyanov, D., Taylor, Z., Carneiro, G. et al. (Eds.): Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support (DLMIA 2018, ML-CDS 2018). Springer, Cham, Lecture Notes in Computer Science, Vol. 11045, 2018, pp. 3–11, doi: 10.1007/978-3-030-00889-5_1.

[30] Asif, U.—Von Cavallar, S.—Tang, J.—Harrer, S.: SSHFD: Single Shot Human Fall Detection with Occluded Joints Resilience. In: De Giacomo, G. et al. (Eds.):

ECAI 2020. IOS Press, Frontiers in Artificial Intelligence and Applications, Vol. 325, 2020, pp. 2656–2663, doi: 10.3233/FAIA200403.

[31] Hu, J.—Cheng, F.—Liu, M.—Xu, X.—Li, X.: MicroFallNet: A Lightweight Model for Real-Time Fall Detection on Smart Wristbands. Pervasive and Mobile Computing, Vol. 109, 2025, Art. No. 102046, doi: 10.1016/j.pmcj.2025.102046.



Seif Eddine Mili is Assistant Professor at the Department of Computer Science, the Ecole Normale Supérieure Constantine, Algeria. He holds his Ph.D. in computer science from the Badji Mokhtar Annaba University, Algeria. His research interest includes artificial intelligence, Web of Things, IoT, multi agent systems, bio inspired system and model driven architecture.



Fouzi Lezzar is currently Assistant Professor at the Department of Software Technologies and Information Systems, the Faculty of New Technologies of Information and Communication, the University Constantine 2, Algeria. He holds his Ph.D. in computer science from the Batna 2 University (Batna, Algeria). His research interest includes artificial intelligence, Internet of Things, smart cities and e-health.

Nabila Azeri is a researcher in the field of computer science, affiliated with the Abbes Laghrour University, Algeria, where she earned her Ph.D. in computer science. Her research focuses on cyber-physical systems (CPS), artificial intelligence, and privacy-preserving technologies. She has made significant contributions to areas such as self-adaptive cyber-physical systems, federated learning, fault prediction, and enhancing resilience in dynamic systems.



Djamel Benmerzoug received his Ph.D. degree in computer science from the Pierre and Marie Curie University, Paris, France. He is currently Full Professor with the Department of TLSI, the Faculty of New Technologies of Information and Communication, the University Constantine 2, Algeria. He has published many papers in many international conferences and journals. He supervises many Master's and Ph.D. students. His current research interests include the Internet of Things, cloud computing, advanced enterprises systems, multiagent systems, service oriented computing, and business processes modeling and verification.