Vision–Language Models for Fall Detection in Socially Assistive Robotics: Zero-Shot Prompting and Few-Shot Calibration

Mohamad Reza Shahabian Alashti[1], Khashayar Ghamati[1], Abolfazl Zaraki[1], Baobing Zhang[1], Patrick Holthaus[1], Shadiya Alingal Meethal[1], Vignesh Velmurugan[1], Gabriella Lakatos[1], Angela Dickinson[1], and Farshid Amirabdollahian[1]

Robotics Research Group, School of Physics, Engineering and Computer Science, University of Hertfordshire, Hatfield, AL10 9AB, United Kingdom
{m.r.shahabian,k.ghamati, a.zaraki, b.zhang6, s.alingal-meethal4, v.velmurugan, p.holthaus, g.lakatos, a.m.dickinson, f.amirabdollahian2}@herts.ac.uk

Abstract. Vision–language models (VLMs) offer a promising route to fall detection for socially assistive robots in home and care settings, where timely recognition can trigger assistance or further verification by carers or interactive robots. Most vision-based fall detectors are supervised and require task-specific labelled data and/or robust pose estimation, which can be brittle under occlusion and viewpoint changes and costly to adapt across deployments. This paper investigates whether pretrained VLMs can enable *data-free* fall detection via zero-shot prompting, and how much a lightweight *few-shot* calibration step improves performance without requiring backbone tuning. We present (i) a zero-shot detector based on a balanced contrastive prompt bank, and (ii) a few-shot variant that trains only a linear classifier on frozen VLM embeddings. We evaluate these alongside three skeleton-based supervised baselines (2D CNN, 3D CNN, ViT) and a rule-based heuristic on a balanced test set of 40 single-person videos (20 fall, 20 non-fall), with identical windowing (32 frames, 50% overlap) and video-level aggregation (majority vote). The few-shot VLM achieves **100%** accuracy, while the zero-shot VLM reaches **92.5%** accuracy without fall-specific training data (3 false positives on non-fall videos). Skeleton-based baselines achieve **97.5–100%** accuracy but require pose extraction, increasing pipeline complexity. These results suggest that pretrained VLMs can provide a practical perception trigger for robot-in-the-loop verification and escalation in assistive care, with zero-shot prompting achieving high recall at the cost of a small number of false alarms.

Keywords: Fall detection · Social robotics · Assistive technology · Human–robot interaction · Vision–language models · Zero-shot prompting · Few-shot calibration

Fig. 1. Conceptual social-robot fall response. Left: the robot approaches a person after detecting a potential fall to initiate a safety check. Right: a remote caregiver dashboard receives an incident log and status updates (detection, approach, check-in, escalation).

1 Introduction

Fall detection is a foundational capability for *socially assistive robots* and assistive technologies deployed in homes and care settings, where reliable recognition of hazardous events can trigger timely support (e.g., check-in interaction, alerting carers, escalation workflows) while preserving users' autonomy and dignity. Falls are a major public-health problem worldwide: the WHO reports approximately 684,000 fatal falls annually and around 37.3 million falls severe enough to require medical attention, with the highest fatality rates among adults over 60 [35]. In the UK, falls also represent a high-volume and high-cost burden for health services, motivating practical monitoring solutions in community and care environments [30]. Figure 1 illustrates the motivating social-robot scenario: a robot detects a potential fall, approaches to verify the event via dialogue, and escalates the incident to a remote caregiver interface.

Robots are increasingly being explored for care delivery, including socially assistive roles (companionship, facilitation, therapy) and safety-oriented functions [1,32]. However, deploying camera-based monitoring in private spaces raises acceptance and privacy concerns. Evidence syntheses show that acceptance of video-based AAL is highly conditional and strongly shaped by perceived benefits (especially emergency detection) relative to privacy costs, underscoring the need for deployable, transparent approaches that disclose failure modes [21].

Vision-based fall detection is attractive because it is non-contact and can leverage existing RGB cameras (fixed or robot-mounted), yet robust performance in real homes remains challenging. Recent reviews consistently highlight three recurring issues: (i) limited, staged, and often imbalanced labelled datasets; (ii) substantial domain shift across environments, viewpoints, and lighting; and (iii) ambiguity between falls and visually similar activities of daily living (e.g., sitting, lying down, kneeling) [14]. These challenges interact with real-time operational requirements: practical systems must maintain low latency and stable false-alarm rates, since excessive false alerts can reduce trust and willingness to use assistive systems over time [21]. While recent architectures (including transformer-based video models) improve in-domain accuracy, they typically remain supervision-heavy and require retraining or careful calibration to sustain performance under deployment shift [22,5].

A prominent strategy is to adopt skeleton-based representations to reduce sensitivity to background appearance and support more interpretable motion reasoning [12,26]. In these pipelines, pose estimation becomes a critical dependency: occlusion, unusual camera angles (common with mobile robots), and lighting can degrade keypoint quality, and per-frame pose inference adds non-trivial computational overhead compared to direct RGB processing [8,29,31]. Thus, even when skeleton models are efficient, end-to-end latency and robustness are bounded by the pose front-end.

Vision–language models (VLMs) offer a complementary route. CLIP-style pretraining aligns images and text in a shared embedding space, enabling *zero-shot* recognition via natural-language prompts without task-specific training data [23]. This capability is appealing for fall detection because collecting and annotating real fall events is expensive, privacy-sensitive, and difficult to scale. At the same time, applying CLIP to falls is non-trivial: falls are temporal events, and prompt-based inference can be sensitive to wording and calibration. Accordingly, this paper quantifies how far a frozen VLM can go for fall detection in a strictly *data-free* zero-shot setting, and how much can be gained through a minimal *few-shot* calibration step that avoids backbone fine-tuning.

We present a comparative study of (i) a zero-shot CLIP fall detector using a balanced contrastive prompt bank and sparse frame sampling, and (ii) a few-shot variant that trains only a lightweight linear classifier on frozen CLIP embeddings. We evaluate these alongside a rule-based heuristic and three skeleton-based supervised models (2D CNN/ResNet [17], 3D CNN, and ViT [10]) under a common windowing and aggregation protocol on a balanced single-person test set, reporting both detection performance and operational efficiency. These results inform how socially assistive robots can use fall detection as a practical trigger for check-in interaction and caregiver escalation, especially when collecting task-specific fall data is limited.

2 Related Work

We review prior work on fall detection for assistive care and socially assistive robotics, spanning video- and skeleton-based methods as well as recent vision–language models for zero/few-shot recognition, highlighting the need for approaches that reduce labelled-data requirements while remaining practical under deployment shift. Table 1 summarises the main families of fall-detection approaches discussed in the literature and highlights their typical trade-offs in training requirements, robustness, and deployment cost.

Table 1. High-level comparison of fall detection families and typical trade-offs.

Family	Inputs	Training Need	Strengths	Common Limitations
Wearables	IMU, pressure	Supervised / thresholds	Privacy, mobility-friendly	Compliance, placement, charging
Rule-based video	RGB / bbox	None	Simple, explainable	Viewpoint sensitivity, brittle thresholds
Supervised video DL	RGB / flow	High	Strong in-domain accuracy	Data-hungry, domain shift
Skeleton-based DL	2D/3D joints	Moderate–high	Background-robust, interpretable	Pose failures, preprocessing cost
Robot-in-the-loop	Multi-sensor + robot	Varies	Verification, escalation, telepresence	Integration complexity, sensing reliability
VLM zero-shot	RGB + prompts	None	No fall data needed, prompt interpretability	Prompt/domain sensitivity, temporal pooling
VLM few-shot	RGB + small labels	Low	Fast adaptation, minimal training	Requires calibration set

Fall detection is often treated as a sensing problem, but in *robot-assisted care* it also becomes an *interaction and escalation* problem: when an alarm triggers, a robot can approach, initiate a short safety dialogue, and route the event to caregivers to reduce unnecessary escalations while maintaining timely response. Robot-in-the-loop designs have been explored to *verify* potential falls and support tele-presence or remote decision-making, targeting the operational burden of false alarms [9]. More broadly, mobile/social robots have been proposed as part of Ambient Assisted Living stacks that combine monitoring with multi-modal user interaction and emergency handling [15,27]. These perspectives motivate fall detection pipelines that prioritise (i) robustness under viewpoint changes (including robot-mounted cameras), (ii) low latency for immediate check-in, and (iii) transparent failure modes integrated into downstream care workflows.

Vision-based fall detection has progressed from heuristic and silhouette-based methods to deep architectures that learn spatiotemporal patterns from RGB or intermediate representations [2,7]. Despite strong in-lab performance, surveys report deployment barriers: staged/acted falls (often by young participants), limited environmental diversity, and ambiguity with fall-like ADLs (e.g., sitting, lying, kneeling), contributing to domain shift and brittle thresholds [2]. This

has driven efforts toward (a) larger, more diverse datasets and (b) models that reduce reliance on curated training distributions.

Public datasets illustrate these trade-offs. UP-Fall provides a large multimodal resource (wearables, ambient, and vision) but uses simulated falls for safety and may differ from real-world falls [20]. Other datasets increase realism via variable illumination, occlusion and clutter (e.g., CAUCAFall) [16], multi-room/location protocols (e.g., URFD/UR Fall) [18], or low-cost/low-resolution recordings to stress robustness (e.g., GMDCSA-24) [3]. Evaluation resources also exist for surveillance-style settings with frame-level annotations and protocol variants [11]. Overall, *data availability and shift* remain key constraints, motivating approaches that operate with minimal task-specific data.

Supervised fall detectors use a sliding-window pipeline with 2D/3D CNNs, CNN–RNN hybrids, or transformer-style temporal encoders [6]. Recent transformer backbones operate directly on RGB clips (avoiding handcrafted features) and support streaming detection [22]. In parallel, edge-oriented work reduces compute (e.g., compression/pruning and lightweight temporal modules) to enable deployment on constrained devices [13]. However, these approaches remain supervision-heavy and often require retraining or recalibration when moving to new homes/cameras, motivating the use of frozen pretrained representations.

Skeleton-based fall detection inherits advances from skeleton action recognition, where spatiotemporal graph models (e.g., ST-GCN) have become a standard way to model joint dynamics [38]. For fall detection, skeleton representations can reduce sensitivity to background appearance and provide interpretability, but they introduce a dependency on pose quality [26,27]. In practice, pose estimation degrades under occlusion and at unusual viewpoints, and it incurs additional front-end cost. Common pose estimators include OpenPose-style part affinity fields and high-resolution networks [8,29], as well as efficient detectors used in applied pipelines [31]. Interpretable modular pipelines explicitly acknowledge this dependency by pairing privacy/selection modules with OpenPose-like extraction before classification [12]. These considerations are amplified in social-robot scenarios, where the camera viewpoint changes as the robot moves.

CLIP-style VLMs align images and text in a shared embedding space, enabling open-vocabulary recognition via prompting and similarity scoring [23,28]. Because falls are temporal events, a key challenge is extending image-level VLM representations to video. Prior work explores temporal pooling/alignment for retrieval and recognition (e.g., CLIP4Clip) [19], action-centric adaptations (e.g., ActionCLIP) [33], and open-vocabulary video CLIP variants that improve zero-shot recognition with lightweight temporal modelling (e.g., Open-VCLIP/Open-VCLIP++) [34,37]. A related line adapts VLMs for weakly supervised video anomaly detection (e.g., VadCLIP), supporting safety monitoring [36].

The most directly aligned fall-specific VLM work fuses object/action pipelines with CLIP, for example, a YOLO–CLIP fusion approach that uses detector outputs and CLIP similarity to improve fall recognition in engineered environments [4]. In contrast, our study isolates the question of *how far a frozen VLM can go* in a *data-free* setting using only prompt contrast on RGB, and quantifies the

Table 2. Compared methods in this study. "Preproc." indicates major preprocessing beyond resizing/normalisation.

Method	Input	Preproc.	Train. Data	Key property
Rule-based	RGB → pose	Pose+bbox	None	Simple posture heuristic
Zero-shot VLM	RGB frames	resize/norm.	None	Prompt contrast (99 vs 99)
Few-shot VLM	RGB frames	resize/norm.	Low	Frozen CLIP + linear probe
2D CNN (ResNet)	Skeleton windows	Pose+norm.	Moderate	CNN over time×joints
3D CNN (Simple)	Skeleton windows	Pose+norm.	Moderate	3D conv over (x,y,joint,time)
ViT (skeleton)	Skeleton windows	Pose+norm.	Moderate	Attention over patches

benefit of a minimal *few-shot linear probe* on frozen embeddings. This design is particularly relevant for social-robot and in-home deployment, where collecting fall-specific labels is difficult and fast calibration with limited data is desirable.

3 Methods

This section defines the fall-detection pipelines evaluated in this work and the common processing protocol used to ensure a fair comparison. We compare six fall-detection approaches spanning heuristics, pretrained vision–language models (VLMs), and skeleton-based supervised classifiers: (1) rule-based heuristic (pose + geometry), (2) zero-shot VLM (CLIP prompt contrast), (3) few-shot VLM (linear probe on frozen CLIP embeddings), (4) skeleton-based 2D CNN (ResNet-style), (5) skeleton-based 3D CNN, (6) skeleton-based ViT. All methods are evaluated under the same windowing and aggregation protocol (Sec. 3.1). Table 2 summarises the key differences.

3.1 Windowing and Video-Level Aggregation

All methods operate on fixed-length temporal windows of $T = 32$ frames with 50% overlap (stride 16). Each window produces either a fall probability $p_w \in [0, 1]$ or a binary label $\hat{y}_w \in \{0, 1\}$. For probabilistic methods, probabilities are converted to labels using a threshold τ (default $\tau = 0.5$):

$$\hat{y}_w = \mathbb{I}(p_w \geq \tau). \tag{1}$$

The video-level prediction is obtained by majority vote over windows:

$$\hat{y}_{\text{video}} = \mathbb{I}\left(\frac{1}{W}\sum_{w=1}^{W}\hat{y}_w \geq 0.5\right), \tag{2}$$

where W is the number of windows.

3.2 Rule-Based Heuristic (Pose Geometry)

We include a lightweight, explainable baseline that detects falls from simple geometric cues derived from pose estimation. For each frame, a single-person pose estimator returns 2D keypoints $\{(x_j, y_j)\}_{j=1}^{J}$ and an associated person extent (bounding box). We compute the tight pose bounding box $B_t = [x_{\min}, y_{\min}, x_{\max}, y_{\max}]$ from valid keypoints, with width $w_t = x_{\max} - x_{\min}$ and height $h_t = y_{\max} - y_{\min}$. A posture score is computed using the height-to-width ratio:

$$r_t = \frac{h_t}{w_t + \epsilon}, \tag{3}$$

where ϵ is a small constant to avoid division by zero. Upright postures tend to yield $r_t > 1$, while near-horizontal postures yield $r_t < 1$.

To reduce jitter, we apply a temporal median filter within each window:

$$\tilde{r}_t = \text{median}\{r_{t-\delta}, \ldots, r_{t+\delta}\}, \tag{4}$$

where δ is the half-window size of the filter (we use $\delta = 2$, i.e., a 5-frame median filter). A frame is flagged as "lying" if $\tilde{r}_t < \theta_r$, where θ_r is a fixed aspect-ratio threshold (we use $\theta_r = 0.8$). A window is classified as a fall if the proportion of "lying" frames exceeds θ_p:

$$\hat{y}_w = \mathbb{I}\left(\frac{1}{T}\sum_{t=1}^{T}\mathbb{I}(\tilde{r}_t < \theta_r) \geq \theta_p\right), \tag{5}$$

where θ_p controls how sustained the horizontal posture must be (we use $\theta_p = 0.5$). Video-level predictions use the same majority-vote aggregation (Sec. 3.1).

This heuristic approximates a common posture-based rule (sustained transition to a near-horizontal configuration) and provides a transparent reference point, while inheriting the limitations of pose quality and viewpoint sensitivity.

3.3 Zero-Shot VLM Fall Detection

We implement zero-shot fall detection using CLIP ViT-B/32 [23] with a balanced prompt-contrast scheme. For each temporal window, we sample k representative RGB frames $\{I_i\}_{i=1}^{k}$ and compute L2-normalised image embeddings $\mathbf{v}_i \in \mathbb{R}^{512}$ using the CLIP image encoder. We construct a prompt bank $\{t_j\}_{j=1}^{M}$ and precompute (cache) L2-normalised text embeddings $\mathbf{u}_j \in \mathbb{R}^{512}$ using the CLIP text encoder. Cosine similarity reduces to a dot product:

$$s_{ij} = \mathbf{v}_i^{\top}\mathbf{u}_j. \tag{6}$$

Table 3. Example prompts from the balanced bank (99 fall, 99 non-fall). The full list is omitted for space and can be provided on demand.

Fall prompts (examples)	**Non-fall prompts (examples)**
a person falling down to the ground	a person walking normally
a person losing balance and falling	a person standing upright
a person tripping and falling down	a person sitting on a chair
a person falling backwards	a person lying on a bed
a person falling and remaining on the floor	a person getting up safely

Balanced prompt bank and contrastive probability. We use $M = 198$ prompts with two equally sized sets: $M^+ = 99$ fall prompts and $M^- = 99$ non-fall prompts. Let $\mathcal{P}^+$ and $\mathcal{P}^-$ denote their indices. For a window, we average similarities across prompts within each set and across the k sampled frames:

$$\bar{s}^+ = \frac{1}{k}\sum_{i=1}^{k}\left(\frac{1}{M^+}\sum_{j\in\mathcal{P}^+} s_{ij}\right), \quad \bar{s}^- = \frac{1}{k}\sum_{i=1}^{k}\left(\frac{1}{M^-}\sum_{j\in\mathcal{P}^-} s_{ij}\right). \tag{7}$$

We use $k = 1$ frame per window (the middle frame) for the reported runtime measurements and form a contrastive logit $\Delta = \bar{s}^+ - \bar{s}^-$ and map it to a fall probability using a temperature-scaled sigmoid:

$$p_{\text{fall}} = \sigma\left(\frac{\Delta}{T_c}\right), \quad T_c = 0.1. \tag{8}$$

Intuitively, Eq. (8) measures whether a window is more similar to the fall prompt set than to the non-fall prompt set, reducing confusion with fall-like ADLs.

Prompt bank design (fall vs non-fall). A key component of our zero-shot formulation is a balanced prompt bank that represents both the target event and plausible confounders. Fall prompts cover direct fall descriptions, motion/causality variants (e.g., slipping/tripping/losing balance), and post-fall end states (e.g., lying on the floor, remaining still). Non-fall prompts represent common safe activities and postures (walking, standing, sitting) as well as benign lying (e.g., lying in bed), which is a frequent source of false alarms. The prompt bank can be adapted to the operating environment (e.g., bedroom vs living room, robot-mounted vs fixed camera) by adding or reweighting context-specific prompts, providing an interpretable mechanism to tune precision–recall behaviour without retraining the visual backbone.

3.4 Few-Shot VLM (Frozen CLIP + Linear Probe)

Few-shot adaptation trains only a lightweight classifier on top of frozen CLIP embeddings, avoiding backbone fine-tuning. For each window, we compute a

pooled embedding $\mathbf{z} \in \mathbb{R}^{512}$ by averaging the k sampled frame embeddings. We train a logistic regression (single linear layer) to predict fall probability:

$$p_{\text{fall}} = \sigma(\mathbf{w}^\top \mathbf{z} + b), \tag{9}$$

optimising binary cross-entropy while updating only $(\mathbf{w}, b)$. Window and video decisions follow Sec. 3.1.

3.5 Skeleton-Based Supervised Models

Skeleton baselines operate on 2D keypoints extracted per frame (COCO-17) using Ultralytics YOLOv11-pose [31]. Keypoints are normalised by centring and scaling to reduce translation and person-size variance. Each window yields a fixed-shape tensor that is classified by one of three supervised models: 2D CNN (ResNet-style), 3D CNN, or ViT.

Input representations. Let $J = 17$ joints and $T = 32$ frames. We use two layouts: (i) a "skeleton image" $[1, T, 2J] = [1, 32, 34]$ for the 2D CNN and ViT (time $\times$ concatenated (x, y) joints), and (ii) a 3D tensor $[2, J, T] = [2, 17, 32]$ for the 3D CNN (channels x, y).

Model families. The 2D CNN uses residual blocks [17] to model temporal patterns in the skeleton image. The 3D CNN applies 3D convolutions across (x/y), joints, and time to learn short-range spatiotemporal dynamics. The ViT uses patch embeddings and multi-head self-attention [10] on the skeleton image to capture longer-range dependencies. All skeleton methods output per-window probabilities and use the same video-level aggregation.

3.6 System Overview

Figures 2–4 summarise the end-to-end pipelines evaluated in this study. All approaches share the same temporal segmentation and decision logic (Sec. 3.1): videos are processed in overlapping 32-frame windows and aggregated to a video-level decision by majority vote. The pipelines differ in the *representation* and *where learning occurs*. The VLM pipelines operate directly on RGB frames: the zero-shot variant scores windows via prompt contrast (Eq. (8)), whereas the few-shot variant replaces prompt contrast with a lightweight linear classifier trained on frozen CLIP embeddings. In contrast, skeleton-based methods first extract COCO-17 keypoints using YOLOv11-pose and then classify normalised skeleton windows with supervised temporal models. To make these differences visually explicit, the diagrams use colour-coding: RGB/video processing, VLM/CLIP blocks, pose/skeleton processing, and decision/aggregation.

3.7 Algorithms

Algorithms 1 and 2 summarise the zero-shot prompt-contrast scoring and the few-shot linear-probe training procedure.

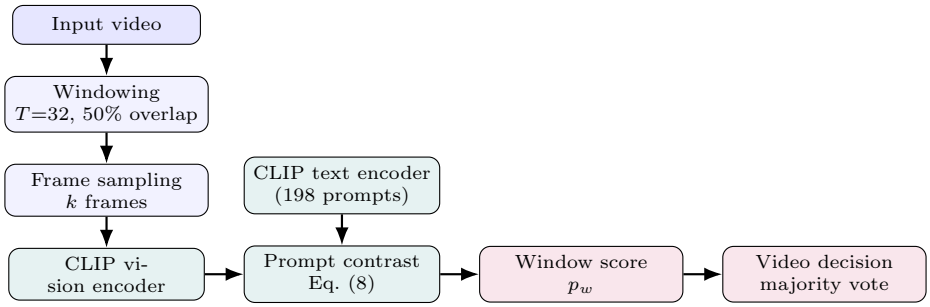

Fig. 2. Zero-shot VLM pipeline: windows are extracted and sampled from the input video, scored via CLIP prompt contrast and aggregated to a video-level decision.

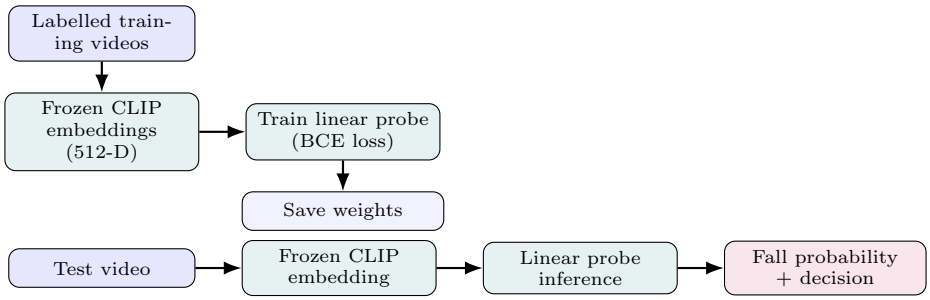

Fig. 3. Few-shot VLM adaptation: CLIP embeddings are extracted with a frozen backbone, and only a lightweight linear classifier is trained on these embeddings; predictions are then aggregated to a video-level decision under the shared windowing protocol.

4 Experimental Setup

This section specifies the data sources, sampling and split procedure, and evaluation protocol used to compare all methods under identical conditions. Training is performed only for supervised skeleton baselines and the few-shot linear probe; the zero-shot VLM and the rule-based heuristic require no task-specific training.

4.1 Datasets

We evaluate binary *fall vs. non-fall* recognition using two video sources (Table 4). Falls are taken from the FallVision benchmark video dataset [24]; we use single-person fall videos and exclude multi-person recordings to avoid confounding from crowd occlusion and multi-target tracking. Non-falls are sampled from the KTH human action dataset [25]; all six KTH actions are treated as the negative class.

4.2 Sampling, Splits, and Shared Test Set

All methods are evaluated on the same fixed test list to ensure a fair comparison. We apply an 80/20 stratified split *per class* (fall and non-fall separately), then

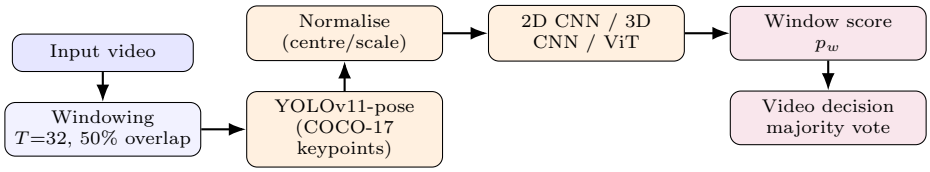

Fig. 4. Skeleton-based pipeline: COCO-17 keypoints are extracted per frame (YOLOv11-pose) and normalised before supervised window classification (2D CNN/3D CNN/ViT) and video-level aggregation.

Algorithm 1 Zero-shot VLM fall probability for one window

Require: Window frames $\mathcal{F}$, CLIP encoders $f_{\text{img}}, f_{\text{text}}$, prompt sets $\mathcal{P}^+, \mathcal{P}^-$, cached text embeddings $\{\mathbf{u}_j\}$, temperature T_c, number of sampled frames k

Sample k frames $\{I_i\}_{i=1}^{k}$ from $\mathcal{F}$ (uniform or middle-frame)

$\mathbf{v}_i \leftarrow \texttt{normalize}(f_{\text{img}}(I_i))$ for $i = 1..k$

$s_{ij} \leftarrow \mathbf{v}_i^\top \mathbf{u}_j$ ▷ $\mathbf{u}_j = \texttt{normalize}(f_{\text{text}}(t_j))$ cached once per run

$\bar{s}^+ \leftarrow \frac{1}{k}\sum_{i=1}^{k}\left(\frac{1}{|\mathcal{P}^+|}\sum_{j\in\mathcal{P}^+} s_{ij}\right)$

$\bar{s}^- \leftarrow \frac{1}{k}\sum_{i=1}^{k}\left(\frac{1}{|\mathcal{P}^-|}\sum_{j\in\mathcal{P}^-} s_{ij}\right)$

$\Delta \leftarrow \bar{s}^+ - \bar{s}^-$

return $p_{\text{fall}} \leftarrow \sigma(\Delta/T_c)$

combine the two test subsets to form a balanced test set. For the main results, we sample `num_samples`=100 videos per class with seed 42 and apply the 80/20 split, yielding a test set of 40 videos (20 fall, 20 non-fall).

4.3 Evaluation Protocol, Training and Implementation Details

Windowing and video-level aggregation follow the common protocol defined in Sec. 3.1 (32-frame windows, 50% overlap, majority-vote aggregation).

Zero-shot VLM (no training). We use CLIP ViT-B/32 [23] with a *balanced* prompt bank of 198 prompts (99 fall, 99 non-fall) designed to cover both fall descriptions and realistic non-fall confounders, and a contrastive temperature $T_c = 0.1$ (Eq. (8)). We later analyse how prompt formulation influences zero-shot behaviour and error modes (Sec. 5.2).

Few-shot VLM (linear probe only). The CLIP backbone remains frozen. We train only a linear classifier (512→1) on CLIP window embeddings using binary cross-entropy. The linear probe is trained on 16 labelled videos (8 fall, 8 non-fall) randomly sampled from the full available pools (single-person falls from FallVision [24] and non-falls from KTH [25]), independent of the evaluation split. Training samples are generated at the window level using the shared 32-frame/50% overlap protocol (Sec. 3.1); we enforce class balance by sampling an equal number of fall and non-fall windows per epoch. Optimisation uses Adam (lr 0.001, wd 0.0001), batch size 8, for 50 epochs.

Algorithm 2 Few-shot VLM training (linear probe on frozen CLIP)

Require: Labelled videos $\{(V_n, y_n)\}$, frozen CLIP encoder f_{img}, epochs E, optimiser Opt
Initialise linear weights $(\mathbf{w}, b)$
for epoch $= 1..E$ **do**
  for mini-batch $\mathcal{B}$ of labelled windows **do**
    Sample k frame(s) per window and compute pooled embedding $\mathbf{z}_n \in \mathbb{R}^{512}$ (mean of normalised CLIP frame embeddings)
    $\hat{y}_n \leftarrow \sigma(\mathbf{w}^\top \mathbf{z}_n + b)$
    $\mathcal{L} \leftarrow \text{BCE}(\hat{y}_n, y_n)$
    Update $(\mathbf{w}, b)$ using Opt; keep f_{img} frozen
  end for
end for
return trained $(\mathbf{w}, b)$

Table 4. Datasets used in this study and their role in the binary classification task.

Dataset	Role	Content
FallVision [24]	Fall (positive)	Single-person fall videos (positive class).
KTH [25]	Non-fall (negative)	Boxing, handclapping, handwaving, jogging, running, walking (all mapped to non-fall).

Skeleton-based supervised baselines. For supervised baselines, we extract COCO-17 keypoints per frame using Ultralytics YOLOv11-pose [31]. We form 32-frame windows and normalise keypoints (centring and scaling) before classification. The 2D CNN (ResNet-style) and ViT operate on a reshaped skeleton "image" of size $[1, 32, 34]$ (time $\times$ 2D joints), while the 3D CNN operates on $[2, 17, 32]$. Skeleton baselines are trained on the training split using Adam (lr 0.001), batch size 8, for 50 epochs.

Rule-based heuristic (no training). The heuristic uses the pose-derived bounding box and flags a fall based on a fixed aspect-ratio criterion consistent with a transition to a near-horizontal posture. The same windowing and majority-vote aggregation are applied.

4.4 Metrics

We report video-level accuracy, precision, recall, and F1 on the shared balanced test set. For skeleton-based and rule-based methods, timing includes YOLOv11-pose extraction; for VLM methods, timing reflects direct RGB processing with cached text embeddings. Table 5 summarises the key evaluation parameters.

5 Results

This section reports quantitative and qualitative results on the shared balanced test set described in Sec. 4. We first compare video-level performance across all

Table 5. Key evaluation and training settings.

Setting	Value
Windowing / overlap	32 frames, 50% overlap (stride 16)
Test set (shared across all methods)	40 videos (20 fall, 20 non-fall);
Zero-shot VLM training	None (data-free prompting)
Zero-shot prompts / temperature	198 prompts (99 fall + 99 non-fall); $T_c = 0.1$
Zero-shot timing configuration	Cached text embeddings; $k = 1$ middle frame
Few-shot VLM training data	16 labelled videos (8 fall, 8 non-fall)
Few-shot optimiser / schedule	Adam lr 0.001, wd 0.0001, batch 8, 50 epochs
Skeleton baselines training data	Training split (supervised training; 2 classes)
Skeleton pose extractor	YOLOv11-pose (COCO-17 keypoints) [31]

Table 6. Balanced test set performance (40 videos: 20 fall, 20 non-fall).

Method	Accuracy	Precision	Recall	F1
Few-shot VLM	**100.0%**	1.00	1.00	1.00
Zero-shot VLM	92.5%	0.87	1.00	0.93
2D CNN (ResNet)	97.5%	1.00	0.95	0.97
3D CNN (Simple)	100.0%	1.00	1.00	1.00
ViT (skeleton)	100.0%	1.00	1.00	1.00
Rule-based heuristic	70.0%	1.00	0.40	0.57

methods, then analyse error profiles using confusion counts, and finally provide prompt-level visualisations to interpret zero-shot VLM decisions.

5.1 Balanced Test Set Results (40 videos)

Table 6 reports video-level metrics on the balanced test set (20 fall, 20 non-fall). Table 7 summarises the corresponding confusion counts. Zero-shot VLM shows a conservative profile on this test set (FN=0) but produces three false alarms on non-fall clips (FP=3), whereas the 2D CNN misses one fall (FN=1).

5.2 Qualitative Analysis of Zero-Shot Prompt Evidence

To complement aggregate metrics, we visualise the evidence produced by the zero-shot prompt-contrast mechanism. For each sampled frame, we log (i) the mean similarity to the fall and non-fall prompt sets together with the top-5 most similar prompts, and (ii) the full similarity matrices between frames and all prompts (99 fall prompts and 99 non-fall prompts). These diagnostics highlight cases where frame-level prompt competition is ambiguous and clarify why aggregation over windows/videos (Sec. 3.1) improves stability.

Figure 5 shows a non-fall (walking) clip. Some frames yield fall-leaning prompt matches despite the activity being benign, illustrating a mechanism for false alarms if decisions were made purely per frame. Figure 6 shows a fall clip, where

Table 7. Confusion matrices on the balanced test set (40 videos: 20 fall, 20 non-fall).

Method	TP	TN	FP	FN
Few-shot VLM	20	20	0	0
Zero-shot VLM	20	17	3	0
2D CNN (ResNet)	19	20	0	1
3D CNN (Simple)	20	20	0	0
ViT (skeleton)	20	20	0	0
Rule-based heuristic	8	20	0	12

fall-related prompts become consistently competitive as the subject transitions from upright to near-horizontal posture, supporting a stable fall decision.

6 Discussion

This study examines whether a large pretrained vision–language model can support a safety-critical perception task under minimal supervision. Unlike conventional supervised pipelines, CLIP-based inference is driven by natural-language prompts and similarity scoring, which introduces an element of prompt sensitivity and non-determinism in the sense that small changes in wording, calibration, or sampling can affect decisions. Nevertheless, the results indicate that pretrained VLMs provide a viable alternative to fully supervised fall detection pipelines when deployment constraints limit labelled data collection.

On the balanced 40-video test set, the zero-shot CLIP prompt-contrast approach achieves 92.5% accuracy while detecting all falls (recall 1.00) and making three false alarms on non-fall clips. This error profile is consistent with the qualitative prompt evidence: frame-level similarities to fall and non-fall prompt sets can be close for certain non-fall motions and postures, and small differences in prompt competition can flip individual window decisions. In assistive technology and human–robot interaction (HRI), this is a meaningful finding: collecting real fall data, particularly from older adults, is difficult and ethically constrained, yet prompt-based transfer provides a practical way to bootstrap a detector without fall-specific training data. In a robot-assisted care setting, false alarms can also be handled through a verification layer: a social robot can treat detection as a trigger to approach the user, perform a brief check-in, and escalate only if the response or subsequent observations indicate risk.

A small amount of task-specific calibration improves reliability. Training only a linear probe on frozen CLIP embeddings achieves 100% accuracy on the same test set. This supports the practical value of minimal-learning approaches: the representation remains general-purpose, while a lightweight classifier adapts decision boundaries to the deployment domain without backbone fine-tuning. It can also simplify operating-point selection because the classifier is trained directly for the fall/non-fall decision rather than relying on prompt-contrast scaling.

Compared with skeleton-based baselines, VLMs shift the cost profile rather than eliminating trade-offs. Skeleton models reach 97.5–100% accuracy but incur

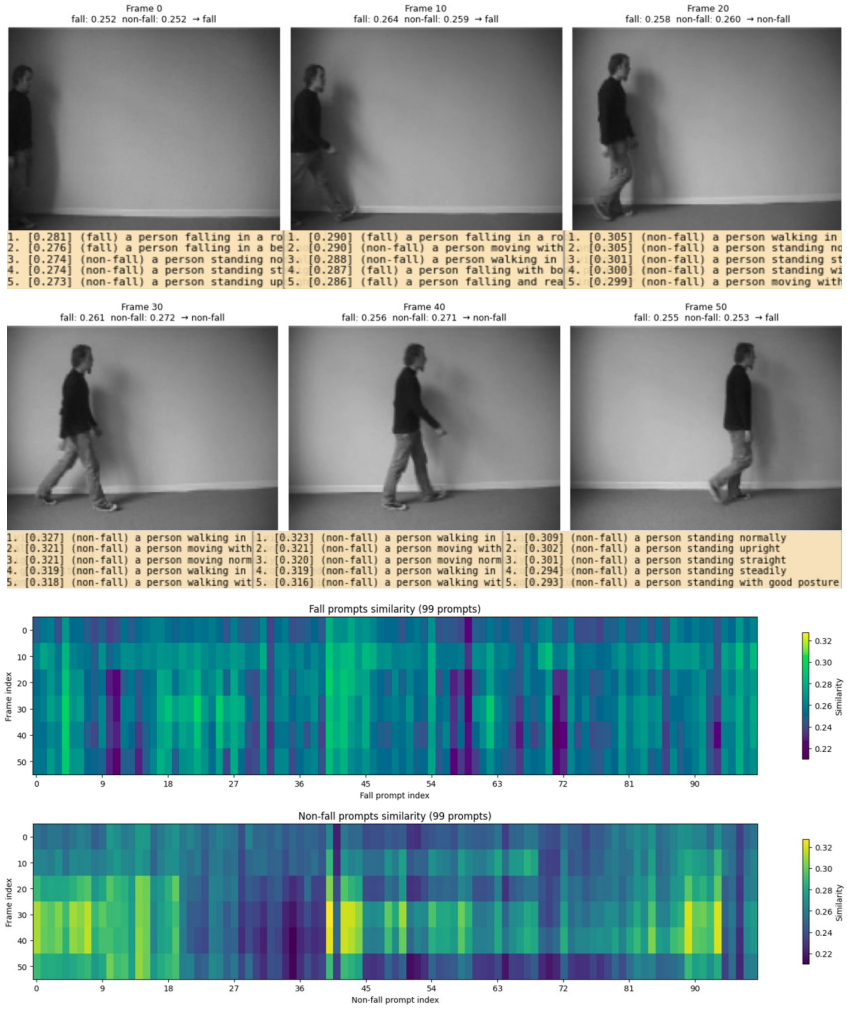

Fig. 5. Zero-shot prompt diagnostics for a non-fall example. Top: sampled frames with per-frame fall/non-fall similarity and top-5 prompts. Bottom: frame×prompt similarity matrices for fall prompts (top) and non-fall prompts (bottom).

pose-extraction overhead and are sensitive to failures under occlusion or non-standard viewpoints. VLMs avoid pose estimation and are attractive for mobile robots with variable camera geometry, but operate on RGB imagery and may raise privacy concerns unless combined with mitigation strategies (e.g., on-device processing, restricted retention, or privacy-preserving representations).

Relative to the closest CLIP-based fall-detection direction (YOLO–CLIP fusion) [4], our results isolate the contribution of prompt-based reasoning and minimal calibration under a shared evaluation protocol. The findings suggest that

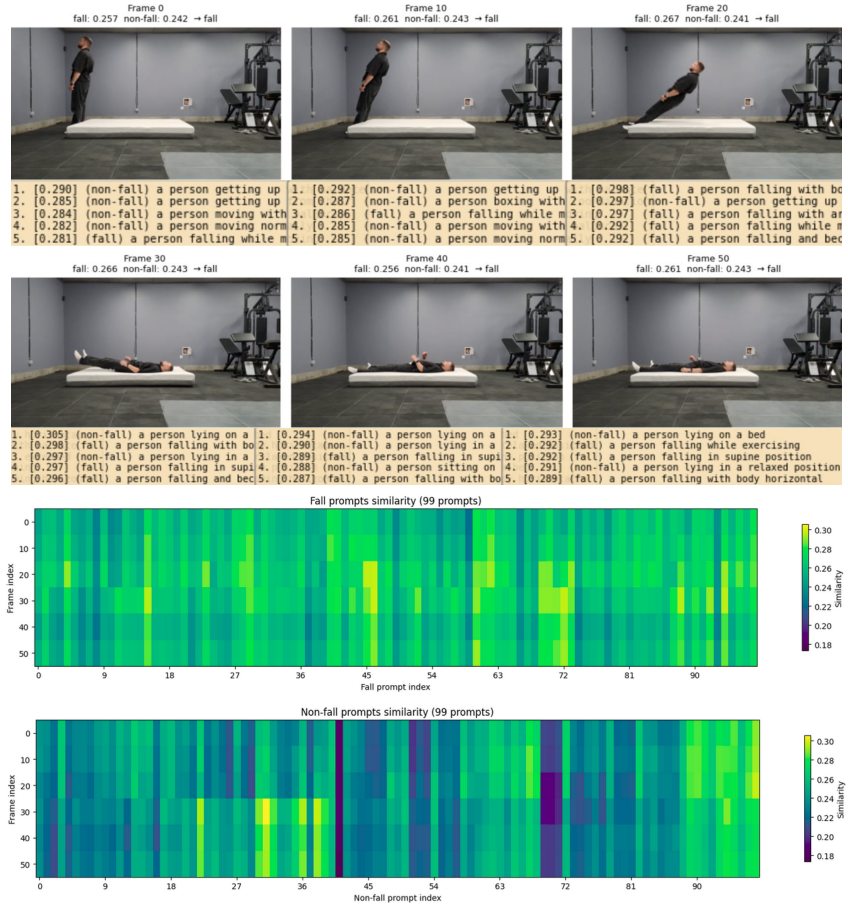

Fig. 6. Zero-shot prompt diagnostics for a fall example. Top: sampled frames with per-frame fall/non-fall similarity and top-5 prompts. Bottom: frame×prompt similarity matrices for fall prompts (top) and non-fall prompts (bottom).

prompt design is not merely an implementation detail: a balanced prompt bank that includes both fall descriptions and realistic non-fall confounders stabilises the contrastive signal, and the prompt set can be adjusted to the deployment context (e.g., bedroom vs living room) to better reflect expected activities.

Several limitations should be noted. First, the test set is small (40 videos), and multiple methods achieve perfect scores, which may not generalise to larger, heterogeneous collections. Second, non-falls are sourced from KTH, which differs in capture conditions from the fall dataset; cross-dataset separability may overestimate performance in real homes where falls and non-falls share similar environments and camera characteristics. As future work, we will benchmark additional public datasets (including matched fall/non-fall conditions) and evaluate real-time human–robot interactions in residential environments, including

scenarios that mimic realistic falls, to characterise robustness in practical deployments better.

A further consideration is that the reported zero-shot configuration uses k=1 representative frame per 32-frame window. Given the 50% overlap (stride 16), this still samples the video densely in time, and for RGB-based VLM scoring, the dominant cues are often scene context and gross posture rather than fine-grained motion, so adjacent windows may not differ substantially. Nevertheless, this setting does not explicitly model temporal dynamics within a fall event. An important direction for future work is to augment the VLM pipeline with lightweight temporal memory (e.g., aggregating evidence across multiple windows or maintaining a short-term state), enabling the system to capture temporal progression in addition to the scene-driven, potentially prompt-sensitive evidence from single frames.

Third, we restrict evaluation to single-person scenes; multi-person occlusion and ambiguity remain open. Finally, although the qualitative analyses clarify prompt competition effects, systematic prompt ablations, automated prompt selection, and multi-frame aggregation studies are needed to better characterise robustness across environments.

7 Conclusion

This work evaluates fall detection for social robotics and assistive technology, where a detector must be deployable in home-care settings and provide a reliable trigger for assistance. On a balanced 40-video benchmark, a zero-shot CLIP prompt-contrast detector achieves 92.5% accuracy without fall-specific training data, while a few-shot linear probe on frozen CLIP embeddings reaches 100% accuracy. These results show that a pretrained vision–language model can support a safety-critical classification task using language-driven similarity scoring, while indicating that prompt-based inference can be sensitive to prompt choice and sampling and benefits from careful calibration and reporting. The ability to operate in zero-shot or with minimal few-shot calibration is particularly relevant for older-adult care, where collecting real fall data is difficult, ethically constrained, and often impractical at scale. Skeleton-based baselines also achieve high accuracy but require pose extraction, adding pipeline complexity and a potential failure point under occlusion and viewpoint change.

While this paper does not implement or evaluate a full robot response pipeline, the proposed detector can serve as the perception component within a robot-in-the-loop process: detection can trigger the robot to approach and perform a brief check-in interaction, and unresolved cases can be escalated to a caregiver interface for follow-up. Future work should validate these findings on larger and more diverse datasets captured in real homes, extend evaluation to multi-person and cluttered scenarios, and study prompt robustness and streaming operation to better support robot-in-the-loop fall verification in the wild.

Acknowledgements

We kindly acknowledge the financial support of the Dinwoodie Charitable Company for the Hospital@Home project. We also acknowledge institutional support from the University of Hertfordshire.

References

1. Abdi, J., Al-Hindawi, A., Ng, T., Vizcaychipi, M.P.: Scoping review on the use of socially assistive robot technology in elderly care. BMJ open **8**(2), e018815 (2018). https://doi.org/10.1136/bmjopen-2017-018815
2. Alam, E., Sufian, A., Dutta, P., Leo, M.: Vision-based human fall detection systems using deep learning: A review. Computers in biology and medicine **146**, 105626 (2022). https://doi.org/10.1016/j.compbiomed.2022.105626
3. Alam, E., Sufian, A., Dutta, P., Leo, M., Hameed, I.A.: Gmdcsa-24: A dataset for human fall detection in videos. Data in Brief **57**, 110892 (2024). https://doi.org/10.1016/j.dib.2024.110892
4. An, J., Su, P., Liu, J., Li, G.: Implementation of YOLO-CLIP fusion algorithm for fall detection. Signal, Image and Video Processing **19**(10), 837 (2025), https://link.springer.com/article/10.1007/s11760-025-04397-w
5. Bamorovat Abadi, M., Shahabian Alashti, M.R., Holthaus, P., Menon, C., Amirabdollahian, F.: RHM: Robot house multi-view human activity recognition dataset. In: The Sixteenth International Conference on Advances in Computer-Human Interactions (ACHI 2023). pp. 181–187. IARIA (2023), https://uhra.herts.ac.uk/id/eprint/14434/
6. Bamorovat Abadi, M.H., Alashti, M.S., Holthaus, P., Menon, C., Amirabdollahian, F.: Robotic vision and multi-view synergy: Action and activity recognition in assisted living scenarios. In: 2024 10th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob). pp. 789–794. IEEE (2024). https://doi.org/10.1109/BioRob60516.2024.10719749
7. Benkaci, A., Sliman, L., Dellys, H.N.: Vision-based human fall detection systems: A review. Procedia Computer Science **241**, 203–211 (2024). https://doi.org/10.1016/j.procs.2024.08.028
8. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2D pose estimation using part affinity fields. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7291–7299 (2017), https://openaccess.thecvf.com/content_cvpr_2017/papers/Cao_Realtime_Multi-Person_2D_CVPR_2017_paper.pdf
9. Capra, M., Sapienza, S., Motto Ros, P., Serrani, A., Martina, M., Puiatti, A., Bonato, P., Demarchi, D.: Assessing the feasibility of augmenting fall detection systems by relying on uwb-based position tracking and a home robot. Sensors **20**(18), 5361 (2020). https://doi.org/10.3390/s20185361
10. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020), https://arxiv.org/abs/2010.11929
11. Dubois, J., Miteran, J.: Fall detection dataset (dataUBFC). dataUBFC repository record (2014). https://doi.org/10.25666/DATAUBFC-2024-04-09

12. Dutt, M., Gupta, A., Goodwin, M., Omlin, C.W.: An interpretable modular deep learning framework for video-based fall detection. Applied Sciences **14**(11), 4722 (2024). https://doi.org/10.3390/app14114722
13. Garmendia-Orbegozo, A., Anton, M.A., Nuñez-Gonzalez, J.D.: Reduction of vision-based models for fall detection. Sensors **24**(22), 7256 (2024). https://doi.org/10.3390/s24227256
14. Gaya-Morey, F.X., Manresa-Yee, C., Buades-Rubio, J.M.: Deep learning for computer vision based activity recognition and fall detection of the elderly: a systematic review: Gm f. xavier et al. Applied Intelligence **54**(19), 8982–9007 (2024). https://doi.org/10.1007/s10489-024-05645-1
15. Gomez-Donoso, F., Escalona, F., Rivas, F.M., Cañas, J.M., Cazorla, M.: Enhancing the ambient assisted living capabilities with a mobile robot. Computational intelligence and neuroscience **2019**(1), 9412384 (2019). https://doi.org/10.1155/2019/9412384
16. Guerrero, J.C.E., España, E.M., Añasco, M.M., Lopera, J.E.P.: Dataset for human fall recognition in an uncontrolled environment. Data in brief **45**, 108610 (2022). https://doi.org/10.1016/j.dib.2022.108610
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016), https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html
18. Kwolek, B., Kepski, M.: Improving fall detection by the use of depth sensor and accelerometer. Neurocomputing **168**, 637–645 (2015). https://doi.org/10.1016/j.neucom.2015.05.061
19. Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., Li, T.: Clip4clip: An empirical study of clip for end to end video clip retrieval. arXiv preprint arXiv:2104.08860 (2021). https://doi.org/10.48550/arXiv.2104.08860
20. Martínez-Villaseñor, L., Ponce, H., Brieva, J., Moya-Albor, E., Núñez-Martínez, J., Peñafort-Asturiano, C.: Up-fall detection dataset: A multimodal approach. Sensors **19**(9), 1988 (2019). https://doi.org/10.3390/s19091988
21. Mujirishvili, T., Maidhof, C., Florez-Revuelta, F., Ziefle, M., Richart-Martinez, M., Cabrero-García, J.: Acceptance and privacy perceptions toward video-based active and assisted living technologies: scoping review. Journal of Medical Internet Research **25**, e45297 (2023). https://doi.org/10.2196/45297
22. Núñez-Marcos, A., Arganda-Carreras, I.: Transformer-based fall detection in videos. Engineering Applications of Artificial Intelligence **132**, 107937 (2024), https://doi.org/10.1016/j.engappai.2024.107937
23. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021), https://proceedings.mlr.press/v139/radford21a.html
24. Rahman, N.N., Mahi, A.B.S., Mistry, D., Al Masud, S.M.R., Saha, A.K., Rahman, R., Islam, M.R.: FallVision: A benchmark video dataset for fall detection. Data in Brief **59**, 111440 (2025). https://doi.org/10.1016/j.dib.2025.111440
25. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local SVM approach. In: Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. vol. 3, pp. 32–36. IEEE (2004). https://doi.org/10.1109/ICPR.2004.1334462

26. Shahabian Alashti, M.R., Bamorovat Abadi, M., Holthaus, P., Menon, C., Amirabdollahian, F.: RH-HAR-SK: a multi-view dataset with skeleton data for ambient assisted living research. In: The Sixteenth International Conference on Advances in Computer-Human Interactions (ACHI 2023). pp. 181–187. IARIA (2023), https://uhra.herts.ac.uk/id/eprint/14432/
27. Shahabian Alashti, M.R., Bamorovat Abadi, M.H., Holthaus, P., Menon, C., Amirabdollahian, F.: Efficient skeleton-based human activity recognition in ambient assisted living scenarios with multi-view CNN. In: 2024 10th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob). pp. 979–984. IEEE (2024). https://doi.org/10.1109/BioRob60516.2024.10719939
28. Shahabian Alashti, M., Ghamati, K., Samani, H., Zaraki, A.: Towards memory-driven Agentic AI for human activity recognition. In: International Conference on Social Robotics. pp. 356–369. Springer (2025), https://link.springer.com/chapter/10.1007/978-981-95-2398-6_25
29. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5693–5703 (2019)
30. UK Government: Falls: applying all our health. https://www.gov.uk/government/publications/falls-applying-all-our-health/falls-applying-all-our-health, accessed: 2026-03-04
31. Ultralytics: YOLO11 pose estimation documentation. https://docs.ultralytics.com/tasks/pose/ (2025), accessed: 2026-01-05
32. Vandemeulebroucke, T., Dzi, K., Gastmans, C.: Older adults' experiences with and perceptions of the use of socially assistive robots in aged care: A systematic review of quantitative evidence. Archives of Gerontology and Geriatrics **95**, 104399 (2021). https://doi.org/10.1016/j.archger.2021.104399
33. Wang, M., Xing, J., Liu, Y.: Actionclip: A new paradigm for video action recognition. arXiv preprint arXiv:2109.08472 (2021). https://doi.org/10.48550/arXiv.2109.08472
34. Weng, Z., Yang, X., Li, A., Wu, Z., Jiang, Y.G.: Open-vclip: Transforming clip to an open-vocabulary video model via interpolated weight optimization. In: International conference on machine learning. pp. 36978–36989. PMLR (2023). https://doi.org/10.48550/arXiv.2302.00624
35. World Health Organization: Falls. https://www.who.int/news-room/fact-sheets/detail/falls (2021), accessed: 2025-12-31
36. Wu, P., Zhou, X., Pang, G., Zhou, L., Yan, Q., Wang, P., Zhang, Y.: VadCLIP: Adapting vision-language models for weakly supervised video anomaly detection. In: Proceedings of the AAAI conference on artificial intelligence. vol. 38, pp. 6074–6082 (2024). https://doi.org/10.1609/aaai.v38i6.28423
37. Wu, Z., Weng, Z., Peng, W., Yang, X., Li, A., Davis, L., Jiang, Y.: Building an Open-Vocabulary video CLIP model with better architectures. Optimization and Data (2024). https://doi.org/10.48550/arXiv.2310.05010
38. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018). https://doi.org/10.1609/aaai.v32i1.12328