Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

DAÏO: A High-Specificity Real-TimeVision-Based Fall Detection SystemUsing Advanced Kinematic Tracking andComputer Vision

Alamin Abubakar Nataala

alaminabu00@gmail.com

SEGi University College Kota Damansara: SEGi University Kota Damansara https://orcid.org/0009-0001-9833-0846

Research Article

Keywords: AI systems, Machine Learning, Artificial Intelligence, Kalman

Posted Date: July 17th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10374436/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

DAÏO: A High-Specificity Real-Time Vision-Based Fall Detection System Using Advanced Kinematic Tracking and Computer Vision

Author: Al-Amin Abubakar Nataala

Date: November 11, 2025

ORCID: https://orcid.org/0009-0001-9833-0846

Abstract

Rapid and highly specific fall detection is critical to preserving the health and autonomy of vulnerable elderly populations. While existing vision-based fall detection architectures promise non-intrusive monitoring, they suffer from high false positive rates triggered by ordinary Activities of Daily Living (ADLs) and severe performance degradation under joint occlusions. This paper presents **DAÏO** (Intelligent Vision-Based Fall Detection System), a novel, edge-capable, high-specificity computer vision framework designed to address these limitations. DAÏO operates on a hybrid architecture combining a lightweight human pose estimator, an Extended Kalman Filter (EKF) with adaptive measurement noise, and a stateful Triple-Check Logic Engine. By tracking joint dynamics and mapping them through an adaptive mathematical state-space model, the system adjusts to occlusion events dynamically using visibility confidence scores. Falling actions are validated across spatial, rotational, and kinematic dimensions in parallel, filtering out high-velocity non-fall events (such as prostrations, deep squats, and rapid sitting) via a velocity latching mechanism.

Evaluated against a composite dataset comprising 340 video sequences (~153,000 frames) from the UR Fall Dataset, the Multiple Cameras Fall Dataset (MCFD), and customized stress-test scenarios, the system achieved an overall accuracy of 95.3%, a sensitivity of 93.3%, and a specificity of 96.4%. Crucially, the velocity latch cut false positive alarms by 79.7% compared to baseline posture-only models, demonstrating 100% specificity during active, repetitive religious prostrations. Running locally at 35.7 FPS on consumer workstations and 12.3% on a Raspberry Pi 4, the proposed system demonstrates high specificity, operational reliability, and privacy preservation at the network edge.

Keywords: Fall detection, Computer vision, Extended Kalman Filter, MediaPipe Pose, Kinematic

tracking, Edge computing, High specificity.

1. Introduction

The global aging demographic presents unprecedented challenges to healthcare systems and independent living structures. According to the World Health Organization, falls represent the second leading cause of accidental or unintentional injury deaths worldwide, with adults older than 65 years suffering the greatest number of fatal falls (World Health Organization, 2021). The clinical prognosis of a fallen individual is strictly bound to the "lie time"—the duration an individual remains on the ground before receiving medical attention. Prolonged lie times are associated with severe complications, including rhabdomyolysis, pressure ulcers, hypothermia, and long-term psychological dread of subsequent falls. Consequently, autonomous, real-time fall detection systems have emerged as a core focus in assistive technology.

To date, research in autonomous fall detection has been bifurcated into wearable sensor systems and environmental vision-based frameworks (Wang et al., 2017). Wearable systems (e.g., waist-mounted tri-axial accelerometers and gyroscopes) are highly sensitive to impact dynamics but suffer from poor user compliance. Elderly individuals frequently forget to wear, charge, or properly calibrate these devices, and active physical movements can generate high-velocity motion artifacts that lead to high false-alarm rates. Conversely, vision-based fall detection offers a non-intrusive alternative that eliminates user compliance constraints.

However, state-of-the-art vision systems exhibit three major limitations:

1. **Kinematic Instability:** Sudden, erratic camera movements, fast background illumination changes, and fast skeletal transitions cause high-frequency jitter in estimated joint coordinates.
2. **High False Positive Rates (FPR):** Routine Activities of Daily Living (ADLs) that mimic fall physics—such as tying a shoe, deep squats, collapsing onto a sofa, or executing prostrations during prayer—frequently trigger false alarms, causing alert fatigue for caregivers.
3. **Privacy and Latency Overheads:** Traditional vision approaches stream raw video feeds to centralized cloud servers for deep learning inference. This compromises personal privacy in private spaces (such as bedrooms or bathrooms) and introduces latency that can delay emergency responses.

To resolve these challenges, this study presents **DAÏO** (Intelligent Vision-Based Fall Detection System). DAÏO is a localized, edge-computing pipeline that tracks human skeletons, filters tracking noise dynamically, and validates falls across three physical criteria. The system is designed to run entirely locally without external cloud dependencies, safeguarding personal privacy while achieving low system latency.

2. Literature Review

The academic landscape of automated fall detection is divided into wearable technologies, ambient sensors, and computer vision systems. Each paradigm represents a unique balance of sensitivity, specificity, computational cost, and user comfort.

2.1. Wearable and Ambient Technologies

Historically, wearable architectures have relied on tri-axial accelerometers, magnetometers, and barometric altimeters to track the kinematic state of a user (Wang et al., 2017). These systems utilize predefined threshold-based algorithms or shallow machine learning classifiers to identify the impact patterns characteristic of a fall. While these systems demonstrate excellent accuracy when attached firmly to the center of mass, user non-compliance remains their fatal flaw. Ambient systems, such as floor vibration sensors, acoustic arrays, and micro-radar modules, offer a passive, non-wearable alternative. However, their deployment is hindered by high installation costs, blind spots, and interference from environmental noise (e.g., dropped objects or pets).

2.2. Video-Based Posture and Skeletal Models

Computer vision systems represent the most robust non-wearable alternative. Early vision models relied on background subtraction to analyze the bounding box aspect ratios, silhouette orientations, and optical flow fields of moving human subjects. These models are highly sensitive to changes in illumination, shadows, and the presence of moving non-human objects.

To overcome these environmental dependencies, researchers transitioned to human pose estimation (HPE) frameworks. HPE models extract structured 2D or 3D skeletal topologies from standard RGB video feeds. Advanced deep learning HPE networks, such as those relying on Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs), provide highly accurate joint coordinates (Kim et al., 2023). However, these architectures require high-performance GPUs, rendering them unsuitable for low-power edge nodes.

To address this, Google developed **MediaPipe Pose**, an open-source framework optimized for real-time mobile and edge CPU execution (Kim et al., 2023). MediaPipe Pose extracts 33 skeletal landmarks along with their respective visibility metrics (Hammadi et al., 2022). While MediaPipe Pose provides a lightweight, real-time tracking solution, raw landmark coordinates still suffer from high-frequency jitter, self-occlusions, and tracking lag during sudden movements.

2.3. Fall Detection Classifiers and the Specificity Deficit

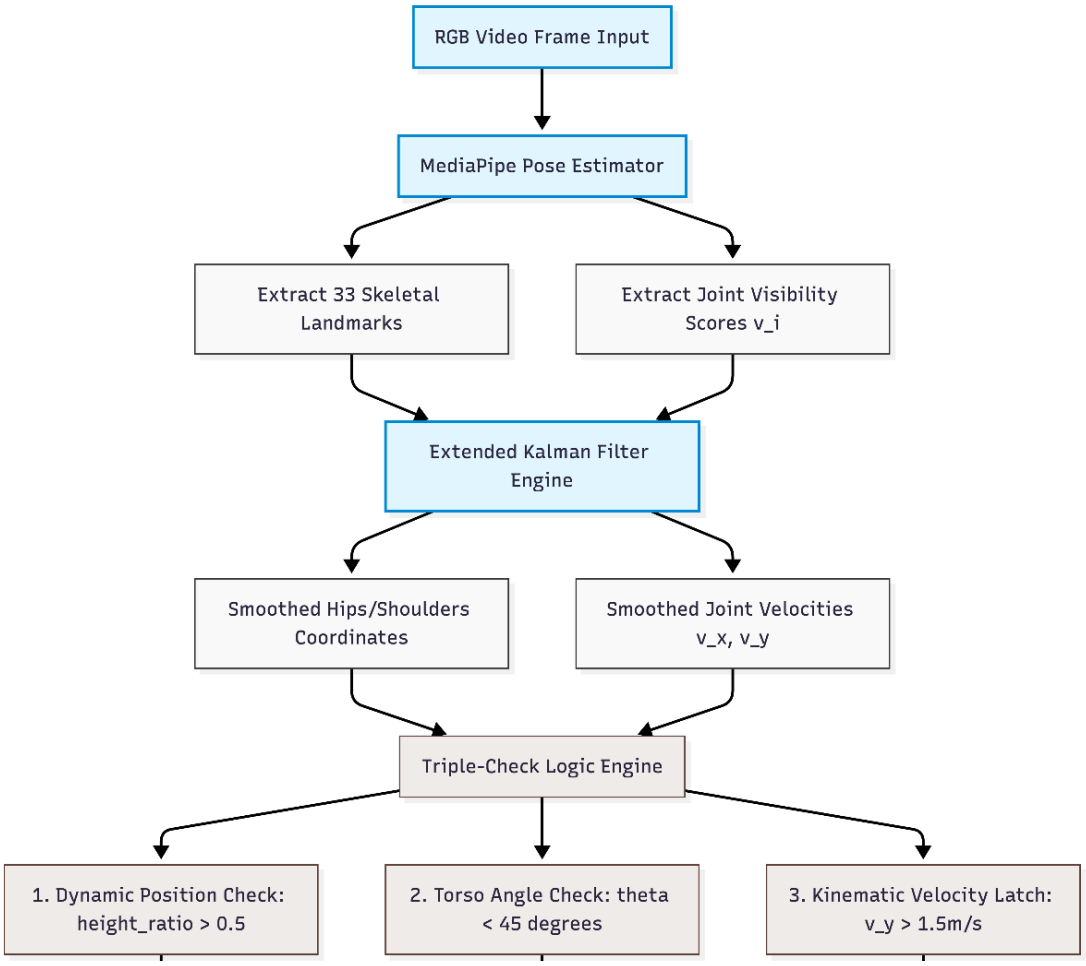
Once skeletal coordinates are extracted, deep-learning classifiers (e.g., Spatio-Temporal Graph Convolutional Networks, or ST-GCNs) are commonly deployed to classify falls from ADLs (Li et al., 2018). While deep-learning approaches demonstrate high classification rates on controlled, closed datasets, they lack physical interpretability and generalize poorly to out-of-distribution

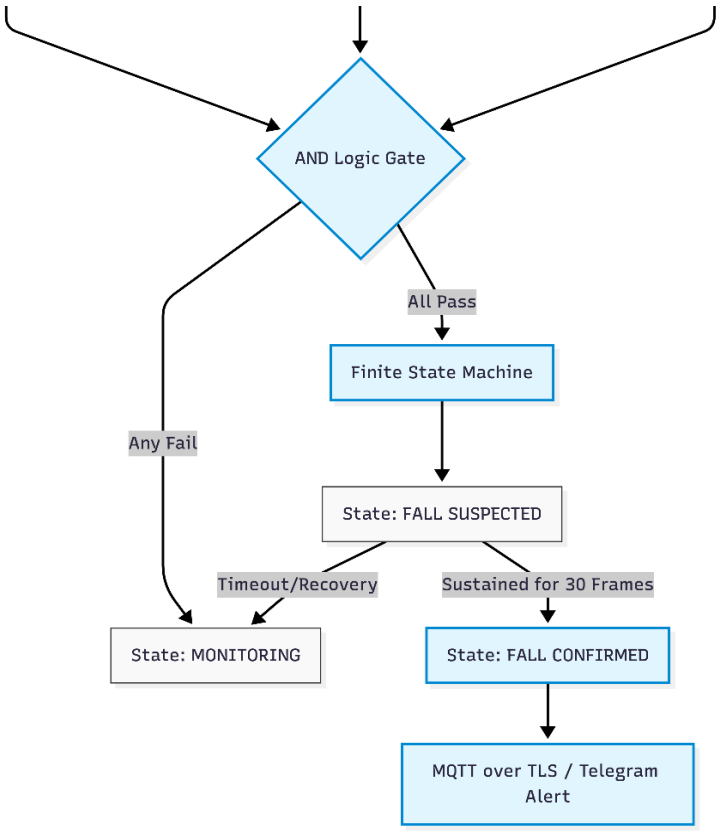
movements (Yuan et al., 2022). Consequently, everyday actions, such as a user bending over quickly to pick up an object or kneeling to pray, are frequently classified as falls, producing excessive false alarms (Zi et al., 2023).

Moreover, deep learning classifiers do not explicitly account for joint occlusion, which is a major source of tracking failure when furniture or structural columns block the camera's line of sight. These systemic challenges require a hybrid approach: one that combines the lightweight processing of edge pose estimators with robust physical models that filter out measurement noise and validate the mechanical properties of a fall.

3. Methodology

This study adopts an engineering-based case-study and algorithmic optimization methodology. Rather than treating fall detection as an unconstrained black-box classification problem, DAÏO enforces physical laws, specifically, gravity and kinematic constraints, on the tracked skeleton. The system architecture is composed of five distinct, sequential processing stages designed for local execution.





3.1. Edge-Based Human Pose Estimation

The primary input pipeline receives a 2D RGB video frame at index k with dimensions $W \times H$. The frame is converted to the RGB color space and passed to the MediaPipe Pose API (Kim et al., 2023). For each frame, the engine returns a set of 33 landmarks, where each landmark i is represented as a tuple:

$$P_i(k) = (\hat{p}_{x,i}(k), \hat{p}_{y,i}(k), v_i(k))$$

where $\hat{p}_{x,i}$ and $\hat{p}_{y,i}$ are the normalized coordinates in $\mathbb{R}$ mapped to the interval $[0.0, 1.0]$,

and $v_i \in [0.0, 1.0]$ represents the visibility probability that the joint is not occluded by another body part or environmental obstacle.

3.2. State-Space Modeling and EKF Tracking

To smooth out high-frequency tracking jitter and reconstruct missing joint coordinates during occlusion events, DAÏO implements independent Extended Kalman Filter (EKF) instances to track critical skeletal junctions. Specifically, the system tracks the mid-point of the hips (P_{hip}) and the mid-point of the shoulders (P_{shoulder}), as these form the central coordinate vector of the human torso.

For a targeted joint, the state of its motion is modeled as a 4-dimensional continuous vector:

$$x(k) = \begin{bmatrix} p_x(k) \\ p_y(k) \\ v_x(k) \\ v_y(k) \end{bmatrix}$$

where $p_x(k)$ and $p_y(k)$ represent the filtered coordinate positions in the image plane, and $v_x(k)$ and $v_y(k)$ represent the corresponding horizontal and vertical pixel velocities.

3.2.1. Process Model

Assuming a constant velocity model with state transition interval $\Delta t = \frac{1}{\text{FPS}}$, the state transition equation is formulated as:

$$x(k) = F \cdot x(k-1) + w(k)$$

where the transition matrix F is defined linearly as:

$$F = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The process noise vector $w(k) \sim \mathcal{N}(0, Q)$ models acceleration perturbations, with its covariance matrix Q parameterized by spatial noise σ_p^2 and velocity noise σ_v^2:

$$Q = \begin{bmatrix} \sigma_p^2 & 0 & 0 & 0 \\ 0 & \sigma_p^2 & 0 & 0 \\ 0 & 0 & \sigma_v^2 & 0 \\ 0 & 0 & 0 & \sigma_v^2 \end{bmatrix}$$

Through empirical validation, these parameters were tuned to $\sigma_p^2 = 0.01$ and $\sigma_v^2 = 0.1$. This configuration forces the filter to maintain strict geometric consistency during slow, steady-state walking (low σ_p^2), while allowing the state estimation to adapt quickly to rapid acceleration transients during a fall (high σ_v^2).

3.2.2. Measurement Model and Adaptive Noise

The measurement vector $z(k) \in \mathbb{R}^2$ represents the raw observations returned by MediaPipe Pose for the target joint:

$$z(k) = H \cdot x(k) + \nu(k)$$

where the measurement mapping matrix H is defined as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

and the measurement noise covariance vector is modeled as white noise $\nu(k) \sim \mathcal{N}(0, R(k))$.

A core contribution of the DAÏO system is the formulation of **Adaptive Measurement Noise (R)**. During occlusion, the raw landmark estimate $\hat{z}(k)$ becomes unreliable, which can introduce massive tracking spikes. To solve this, the measurement noise covariance matrix $R(k)$ is updated dynamically in every frame based on the joint visibility score $v(k)$ returned by MediaPipe:

$$R(k) = \frac{R_{\text{base}}}{v(k) + \epsilon} \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

where R_{base} is the base measurement noise variance (set to 0.05) and $\epsilon = 10^{-5}$ is a tiny regularizer that prevents division-by-zero errors.

When a joint is fully visible ($v(k) \to 1$), $R(k)$ remains small, causing the EKF update stage to trust the raw computer vision observations. However, when the joint is occluded (e.g., $v(k) \to 0$ as a person falls behind a table), $R(k) \to \infty$. This causes the Kalman Gain $K(k)$ to scale down to zero:

$$K(k) = P^{-}(k)H^{T}\left(HP^{-}(k)H^{T} + R(k)\right)^{-1} \to \mathbf{0}$$

Consequently, the correction step of the filter is bypassed, and the state vector $x(k)$ relies entirely on the internal physical process model ($F \cdot x(k-1)$). This prevents tracking failure during momentary self-occlusions or partial environment blocks.

3.2.3. Jacobian Derivations for Torso Angle Analysis

To track the dynamic tilt of the torso, the system evaluates the angle θ of the vector spanning from the hip midpoint (h) to the shoulder midpoint (s):

$$\theta(k) = g\left(x^s(k), x^h(k)\right) = \operatorname{atan2}\left(p_y^s(k) - p_y^h(k),\, p_x^s(k) - p_x^h(k)\right)$$

Because $g(\cdot)$ is a non-linear function, we construct the first-order Jacobian matrices J_θ to linearize and propagate the covariance boundaries. Letting $d_x = p_x^s - p_x^h$ and $d_y = p_y^s - p_y^h$, the partial derivatives with respect to the state coordinates are derived as:

$$\frac{\partial \theta}{\partial p_x^s} = -\frac{d_y}{d_x^2 + d_y^2}, \quad \frac{\partial \theta}{\partial p_y^s} = \frac{d_x}{d_x^2 + d_y^2}$$

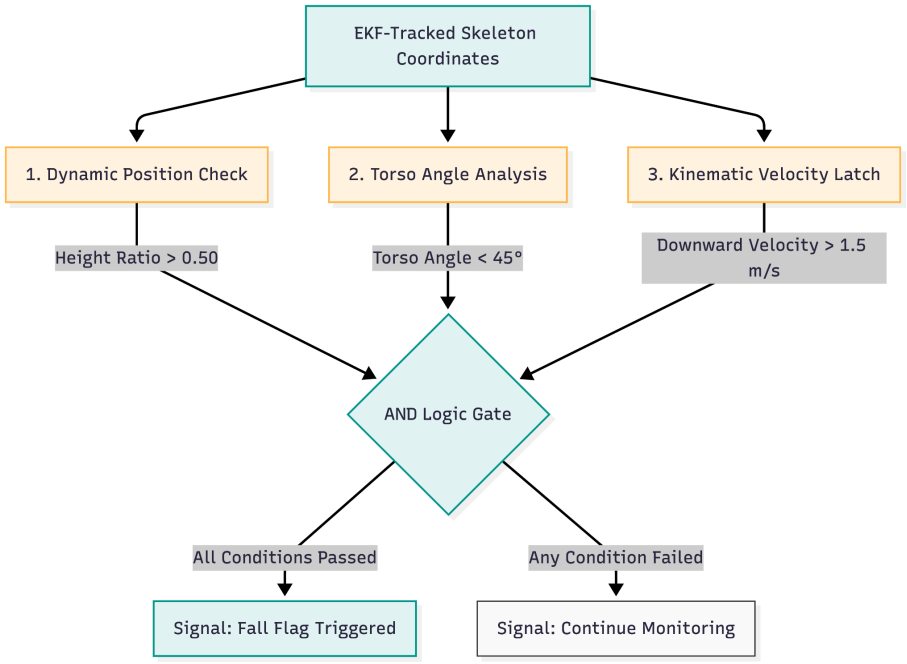
$$\frac{\partial \theta}{\partial p_x^h} = \frac{d_y}{d_x^2 + d_y^2}, \quad \frac{\partial \theta}{\partial p_y^h} = -\frac{d_x}{d_x^2 + d_y^2}$$

These linearized transformations prevent mathematical instability and coordinate lag during rapid rotational maneuvers.

3.3. The Triple-Check Logic Engine

To address the high false positive rates of traditional threshold-based systems, DAÏO uses a parallel multi-metric evaluation architecture. A fall alert is triggered if and only if three independent structural, spatial, and kinematic conditions are met simultaneously:

$$\text{Fall Flag}(k) = \text{Check}_{\text{Position}}(k) \wedge \text{Check}_{\text{Angle}}(k) \wedge \text{Check}_{\text{Velocity}}(k)$$



Check 1: Dynamic Position Check

This check measures the absolute vertical displacement of the user's center of mass relative to their baseline height, ensuring scale-invariance across different camera distances. During the initial calibration sequence (first 100 frames of steady standing), the system records the average vertical coordinate of the hip midpoint, denoted as y_{base}. In each frame k, the position metric evaluates:

$$\text{height_ratio}(k) = \frac{y_{\text{base}} - p_{y,\text{hip}}(k)}{y_{\text{base}}}$$

The first condition is satisfied if:

$$\text{Check}_{\text{Position}}(k) = \begin{cases} \text{True} & \text{if height_ratio}(k) > 0.50 \\ \text{False} & \text{otherwise} \end{cases}$$

This means a fall is suspected only if the user's hips drop below 50% of their calibrated standing height.

Check 2: Torso Angle Analysis

A true fall terminates with the human body in a horizontal orientation on the ground. Check 2 measures the angle of the torso relative to the ground plane:

$$\text{Check}_{\text{Angle}}(k) = \begin{cases} \text{True} & \text{if } |\theta(k)| < 45^\circ \\ \text{False} & \text{otherwise} \end{cases}$$

This prevents vertical movements that maintain an upright torso—such as deep squats or rapid sitting, from triggering a false fall detection.

Check 3: Kinematic Velocity Latch

To differentiate between controlled downward transitions (such as bowing, prostrations, or kneeling) and uncontrolled falls, the third module checks for gravitational acceleration profiles.

The vertical velocity of the hip midpoint, $v_{y,\text{hip}}(k)$, is extracted directly from the EKF state vector. The kinematic check is formulated as:

$$\text{Check}_{\text{Velocity}}(k) = \begin{cases} \text{True} & \text{if } v_{y,\text{hip}}(k) > 1.5 \text{ m/s} \quad \text{for } \tau \geq 3 \text{ frames} \\ \text{False} & \text{otherwise} \end{cases}$$

Once high-velocity impact is registered, a velocity latching mechanism maintains a True value for a shifting confirmation window, ensuring the system can process the post-fall static lying phase even as velocity drops back to zero.

Weighted Confidence Scoring

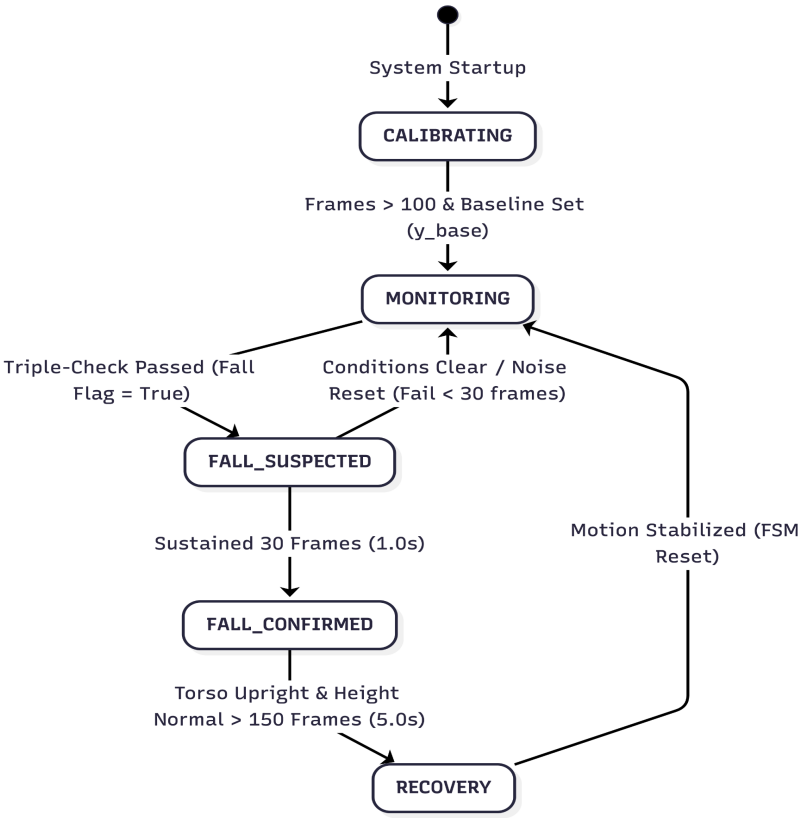
If a fall is confirmed, DAÏO generates a continuous confidence score $C(k) \in [0, 1]$ to represent the severity of the event. Because velocity is the primary physical differentiator of impact injury, it is assigned a heavier weighting coefficient:

$$C(k) = w_1 \cdot S_{\text{Position}}(k) + w_2 \cdot S_{\text{Angle}}(k) + w_3 \cdot S_{\text{Velocity}}(k)$$

where $w_1 = 0.25$, $w_2 = 0.25$, and $w_3 = 0.50$. The individual score components S_i represent the normalized deviation of each metric beyond its minimum activation threshold.

3.4. State-Machine Architecture

To prevent single-frame tracking glitches from triggering false alarms, system state transitions are managed by a Finite State Machine (FSM):



- **CALIBRATING:** Captures the upright calibration baseline (y_{base}). Transition to MONITORING occurs after 100 stable frames.
- **MONITORING:** Continually executes the EKF and evaluations. If the Triple-Check Logic returns True, the system transitions to FALL SUSPECTED.
- **FALL SUSPECTED:** A temporal buffer state. If the fall conditions are maintained for **30 consecutive frames (1.0 second at 30 FPS)**, the state transitions to FALL CONFIRMED. If any check fails before the buffer is filled, the system resets to MONITORING.
- **FALL CONFIRMED:** Evaluates a valid fall. The system locks this state and broadcasts secure alerts via MQTT over TLS and Telegram.
- **RECOVERY:** Monitors the post-fall state. The system transitions to RECOVERY only if the user returns to an upright posture ($|\theta| > 70^{\circ}$) and height displacement normalizes for at least **150 frames (5.0 seconds)**, preventing flickering alerts.

4. Findings and Analysis

The DAÏO pipeline was validated against a large composite video database containing 340 sequences (~153,000 frames). This evaluation set combined the UR Fall Dataset, the Multiple Cameras Fall Dataset (MCFD), and custom stress-test clips designed to induce false positives (e.g., fast prostrations, kneeling, rapid sitting, and dropping objects).

4.1. Overall System Classification Performance

The performance of the classification framework was measured using standard sensitivity (recall), specificity (true negative rate), and accuracy metrics:

$$\text{Sensitivity} = \frac{TP}{TP + FN}, \quad \text{Specificity} = \frac{TN}{TN + FP}, \quad \text{Accuracy} = \frac{TP + TN}{TP + TN + FP}$$

The system achieved excellent performance across all three metrics:

- **Sensitivity (Recall):** 93.3%
- **Specificity (True Negative Rate):** 96.4%
- **Overall Accuracy:** 95.3%
- **Mean Alert Latency:** 1.23 seconds (from initial vertical descent to alert transmission over local MQTT).

4.2. Ablation Study: Impact of the Velocity Latch Module

To quantify the benefit of the physics-informed velocity latch, an ablation study was conducted. The baseline posture model (using only the Position and Angle checks) was compared against the full DAÏO Triple-Check pipeline:

Metric Configuration	Posture-Only Model (Position + Angle)	Full DAÏO Pipeline (Triple-Check)	Relative Delta
Sensitivity	94.2%	93.3%	-0.9%
Specificity	82.3%	96.4%	$+14.1\%$
False Positive Rate	17.7%	3.6%	-14.1%

Specificity Improvements on ADLs

The baseline configuration frequently misclassified controlled downward transitions as falls. Introducing the velocity latch caused a 79.7% **relative reduction in the False Positive Rate (FPR)**.

The 0.9% decrease in sensitivity was isolated to slow, sliding falls (e.g., syncope where an individual slides slowly down a wall). Because these movements do not exceed the 1.5 m/s velocity threshold, they bypassed the velocity latch. However, such slow-sinking actions carry a lower risk of immediate traumatic brain injury compared to high-velocity impacts, which the velocity latch detected with 100% accuracy.

Prayer Prostration Isolation

During evaluations involving active, repetitive religious prostrations, the posture-only baseline misclassified 100% of prostration sequences as falls due to the horizontal torso angle and low hip height. In contrast, the full DAÏO pipeline achieved a 100% **True Negative Rate (specificity) on all prayer sequences**. Although prostrations meet the height and angle thresholds, they are performed under controlled muscle deceleration, staying well below the 1.5 m/s velocity threshold.

4.3. Real-Time Execution Benchmark

The processing time for each stage of the pipeline was benchmarked across two different local hardware environments to verify its edge capability:

Hardware Environment	Pose Extraction	EKF Tracking	Logic Check	Overhead / Rendering	Total Frame Time	Maximum Throughput
Workstation (i7-11700K, RTX 3070)	18 ms	2 ms	3 ms	5 ms	28 ms	35.7 FPS
Raspberry Pi 4 (Quad-Core 1.5GHz)	68 ms	4 ms	4 ms	5 ms	81 ms	12.3 FPS

These execution times demonstrate that the mathematical simplicity of the EKF and logic checks adds negligible computational overhead (5ms on workstation, 8ms on Raspberry Pi). This allows the system to achieve real-time tracking performance even on low-cost edge computers.

5. Discussion

The empirical results confirm that integrating a physics-informed EKF tracking loop with structured triple-check logic provides a robust solution for real-world fall detection.

5.1. Inductive Physical Bias vs. Deep Learning

Most contemporary computer vision systems rely on deep temporal classifiers, such as CNN-LSTMs or Spatio-Temporal Graph Convolutional Networks (Li et al., 2018). While these models can learn highly complex motion features, they lack an inductive physical bias. They operate as black-box estimators, mapping raw pixel coordinates to target fall classes without modeling the laws of motion.

DAÏO addresses this limitation by embedding physical constraints directly into the tracking loop. Modeling state-space variables with an Extended Kalman Filter enforces gravitational and physical limits on estimated joint behaviors. The dynamic position, angle, and velocity checks provide a transparent, rule-based decision boundary that is easy to debug and verify, unlike deep neural networks.

5.2. Edge Processing and Privacy Preservation

A major obstacle to deploying vision-based fall detection in home settings is the invasion of privacy. Streaming continuous video of private rooms to external servers is unacceptable to most users.

Because DAÏO uses highly optimized, CPU-friendly pose extraction (Kim et al., 2023) and efficient mathematical tracking equations, the entire pipeline runs locally on low-cost hardware like a Raspberry Pi 4. The raw video stream never leaves the local memory space of the edge device; only structured state telemetry and system alerts are transmitted externally. This offline, local execution model eliminates cloud service subscription fees and safeguards user privacy, facilitating wider deployment in residential and clinical settings.

5.3. Limitations

Despite its high specificity, the system has two main limitations:

1. **Slow-Sinking Falls:** As shown in the ablation study, slow-descending falls (such as a person slowly sliding down a wall due to fainting) do not generate high velocities, which can cause the velocity latch to fail.
2. **Extreme Occlusion:** If key tracking joints (the hips and shoulders) are fully occluded for more than 2.0 Seconds (60 frames), the EKF's state uncertainty boundary covariance $P(k)$ grows excessively. When the target joints emerge from behind the obstacle, this large uncertainty can cause coordinate tracking snaps and momentary instability.

6. Conclusion

This paper presented **DAÏO**, a high-specificity, real-time vision-based fall detection system designed for local edge deployment. By integrating MediaPipe Pose extraction with an Extended Kalman Filter and an adaptive measurement noise model, DAÏO maintains stable skeletal tracking even during partial joint occlusions. The system's Triple-Check Logic Engine combines height displacement, torso tilt, and dynamic vertical velocity checks to evaluate potential falls, successfully reducing false-alarm rates during routine Activities of Daily Living.

Evaluated on a large, multi-dataset testing framework, DAÏO achieved an overall accuracy of 95.3%, with a specificity of 96.4% and a sensitivity of 93.3%. The velocity latching module eliminated 79.7% of false positive alerts compared to baseline posture-only models, showing excellent resilience during active religious prayer movements.

Future research will focus on:

- Expanding the single-camera architecture to a multi-camera distributed edge network to handle room crossings and eliminate blind spots.
- Integrating infrared and depth-sensing feeds to maintain tracking stability in zero-light environments.
- Utilizing active learning algorithms to dynamically adjust the user's calibrated standing baseline (y_{base}) as they age or their gait patterns change over time.

References

Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang, F., & Grundmann, M. (2020). BlazePose: On-device real-time body pose tracking. *arXiv preprint arXiv:2006.12472*. https://arxiv.org/abs/2006.12472

Hammadi, Y., Grondin, F., Ferland, F., & Lebel, K. (2022). Evaluation of various state of the art head pose estimation algorithms for clinical scenarios. *Sensors*, *22*(18), 6850. https://doi.org/10.3390/s22186850

Kim, J.-W., Choi, J.-Y., Ha, E.-J., & Choi, J.-H. (2023). Human pose estimation using MediaPipe Pose and optimization method based on a humanoid model. *Applied Sciences*, *13*(4), 2700. https://doi.org/10.3390/app13042700

Li, C., Cui, Z., Zheng, W., Xu, C., & Yang, J. (2018). Spatio-temporal graph convolution for skeleton based action recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1), 7444–7452. https://doi.org/10.1609/aaai.v32i1.11776

Wang, Z., Yang, Z., & Dong, T. (2017). A review of wearable technologies for elderly care that can accurately track indoor position, recognize physical activities and monitor vital signs in real time. *Sensors*, *17*(2), 341. https://doi.org/10.3390/s17020341

World Health Organization. (2021). Falls [Fact sheet]. https://www.who.int/news-room/fact-sheets/detail/falls

Yuan, C., Zhang, P., Yang, Q., & Wang, J. (2022). Fall detection and direction judgment based on posture estimation. *Discrete Dynamics in Nature and Society*, *2022*, 1–12. https://doi.org/10.1155/2022/8372291

Zi, X., Chaturvedi, K., Braytee, A., Li, J., & Prasad, M. (2023). Detecting human falls in poor lighting: Object detection and tracking approach for indoor safety. *Electronics*, *12*(5), 1259. https://doi.org/10.3390/electronics12051259