IBIMA Publishing
Communications of International Proceedings
https://ibimapublishing.com/p-articles/47ICT/2026/4726826/
Vol. 2026 (13), Article ID 4726826

Inactivity-Gated Fall-Like Detection Using Quaternion-Based Orientation Change on an ESP32–LSM9DS1 Wearable IMU Node*

Michal KIMBAR, Fryderyk GAJDZIK and Pawel KROWICKI

Wroclaw University of Science and Technology, Faculty of Mechanical Engineering, Wroclaw, Poland,

Correspondence should be addressed to: Fryderyk GAJDZIK, fryderykgajdzik@gmail.com

*Presented at the 47th IBIMA International Conference, 29-30 June 2026, Madrid, Spain

Abstract

False alarms remain a practical limitation of wearable inertial fall-detection systems, particularly when rapid non-fall movements produce acceleration or angular-velocity peaks comparable to fall-like events. This paper presents an inactivity-gated event logic implemented on a low-cost ESP32–LSM9DS1 wearable IMU node. The logic combines three sequential gates: dynamic-motion detection, quaternion-based orientation-change confirmation and post-event inactivity verification. Motion descriptors include acceleration magnitude, angular-velocity magnitude, jerk, stillness ratio and the angular distance between pre-event and current orientation quaternions. Five functional scenarios were tested at the gate level: static rest, slow orientation change, impulse without posture change, dynamic non-fall movement and controlled fall-like movement. The observed decisions followed the intended sequence: non-fall cases did not generate an alarm, whereas the controlled fall-like event generated an alarm. The study demonstrates a compact, interpretable and embedded-oriented decision structure for early-stage wearable IMU safety-monitoring prototypes.

Keywords: wearable IMU; fall-like event detection; post-event inactivity; quaternion orientation; ESP32; LSM9DS1; finite-state logic.

Introduction

Wearable inertial sensing is widely used in fall-detection research because it enables continuous motion monitoring without cameras, optical markers or fixed measurement infrastructure. A typical IMU node provides acceleration, angular velocity and, after sensor fusion, orientation estimates that describe both the dynamic phase of motion and the subsequent change in body or sensor posture.

A key limitation of threshold-based fall detection is that acceleration and angular-velocity peaks are not specific to falls. Similar peaks may occur during bending, rapid turning, sensor handling or intensive normal activity. A single threshold crossing is therefore insufficient for reliable alarm generation. A more interpretable approach is

____

Cite this Article as: Michal KIMBAR, Fryderyk GAJDZIK and Pawel KROWICKI Vol. 2026 (13) "Inactivity-Gated Fall-Like Detection Using Quaternion-Based Orientation Change on an ESP32–LSM9DS1 Wearable IMU Node " Communications of International Proceedings, Vol. 2026 (13), Article ID 4726826, https://doi.org/10.5171/2026.4726826

to analyse the temporal structure of the event: a dynamic peak should be followed by a relevant orientation change and by a post-event state consistent with inactivity.

Public datasets such as SisFall, MobiAct and KFall show that fall detection is usually evaluated against multiple activities of daily living and simulated fall scenarios (Sucerquia et al., 2017; Vavoulas et al., 2016; Yu et al., 2021). Recent reviews indicate that wearable fall-detection research increasingly combines inertial sensing, embedded systems, machine learning and IoT-oriented safety monitoring (Li et al., 2025; Ramachandran and Karuppiah, 2020). At the same time, practical deployment still depends on computational simplicity, interpretability and the ability to diagnose each decision step.

Sequential event logic, threshold-based inertial features and post-event verification have already been used in wearable, smartphone-based and IoT-oriented fall-detection systems (Noury et al., 2008; Bourke and Lyons, 2008; Abbate et al., 2012; Lin et al., 2023; Tseng et al., 2025). Low-power and edge-oriented wearable implementations have also been explored (Tian et al., 2024). The present work is not positioned as a new general fall-detection algorithm. Its contribution is an implementation-oriented formulation of the following decision sequence:

$$\text{dynamic event} \rightarrow \text{orientation change} \rightarrow \text{post} - \text{event inactivity}. \tag{1}$$

The sequence combines scalar inertial features, a quaternion-based posture-change metric and a finite-state-machine structure executable directly on a low-cost ESP32–LSM9DS1 IMU node.

Quaternion-based posture-change representation

Euler angles provide an intuitive orientation description, but they depend on rotation order and may suffer from representation singularities. Quaternions provide a compact representation on SO(3) and allow the relative rotation between two postures to be computed directly. In the proposed logic, the post-event posture transition is represented by the angular distance between a pre-event quaternion and the current quaternion. Quaternion-based orientation estimation is well established in inertial and magnetic sensing (Sabatini, 2006; Mahony et al., 2008; Madgwick, 2010), and quaternion-based fall-detection logic has also been explored in wearable systems (Wu et al., 2015). The full sensor-fusion layer is not analysed here; the estimated orientation is treated as a firmware-level intermediate signal used by the decision logic.

Materials and implementation

The implementation was tested on a wearable IMU node based on an ESP32 microcontroller and the LSM9DS1 inertial module. The LSM9DS1 integrates a three-axis accelerometer, a three-axis gyroscope and a three-axis magnetometer (STMicroelectronics, 2016). The firmware provided calibrated IMU acquisition, orientation estimation, diagnostic telemetry and finite-state event logic. The preliminary tests reported in this paper used diagnostic telemetry; BLE transmission reliability and long-term communication stability were not evaluated.

This paper focuses only on the inactivity-gated decision layer. A broader system-level description of firmware architecture, communication backend, calibration persistence and sensor-fusion tuning is outside the scope of this short contribution.

For each sample k, the firmware provides:

$$\boldsymbol{a}_k = [a_x(k) a_y(k) a_z(k)]^T, \tag{2}$$

$$\boldsymbol{\omega}_k = [\omega_x(k) \omega_y(k) \omega_z(k)]^T, \tag{3}$$

$$\boldsymbol{q}_k = [q_w(k) q_x(k) q_y(k) q_z(k)]^T. \tag{4}$$

The decision logic uses the feature vector:

$$\boldsymbol{x}_k = [A_{vm}(k) G_{vm}(k) |J(k)| \Delta\theta_{\deg}(k) R_{still}(k)]^T. \tag{5}$$

This notation is used only to compactly represent threshold features; no vector-based machine-learning classifier is implied.

Method

Dynamic features

The acceleration magnitude normalized to gravitational acceleration is:

$$A_{vm}(k) = \frac{1}{g}\sqrt{a_x^2(k) + a_y^2(k) + a_z^2(k)}. \tag{6}$$

The angular-velocity magnitude is:

$$G_{vm}(k) = \sqrt{\omega_x^2(k) + \omega_y^2(k) + \omega_z^2(k)}. \tag{7}$$

The jerk feature is computed from consecutive acceleration-magnitude samples:

$$J(k) = \frac{A_{vm}(k) - A_{vm}(k-1)}{\Delta t}. \tag{8}$$

A dynamic candidate is detected when at least one dynamic threshold is exceeded:

$$C_{dyn}(k) = [A_{vm}(k) > T_A] \vee [G_{vm}(k) > T_G] \vee \left[|J(k)| > T_J\right], \tag{9}$$

where T_A, T_G and T_J are the acceleration-magnitude, angular-velocity-magnitude and jerk thresholds.

Quaternion-based orientation change

Let q_{pre} denote the orientation quaternion stored before the dynamic event and q_{cur} denote the current orientation quaternion. The relative rotation is:

$$\boldsymbol{q}_\Delta = \boldsymbol{q}_{pre}^{-1} \otimes \boldsymbol{q}_{cur}, \tag{10}$$

where $\otimes$ denotes quaternion multiplication.

For two quaternions:

$$\boldsymbol{p} = [p_w \boldsymbol{p}_v^T]^T, \qquad \boldsymbol{q} = [q_w \boldsymbol{q}_v^T]^T, \tag{11}$$

their product is:

$$\boldsymbol{p} \otimes \boldsymbol{q} = \begin{bmatrix} p_w q_w - \boldsymbol{p}_v^T \boldsymbol{q}_v \\ p_w \boldsymbol{q}_v + q_w \boldsymbol{p}_v + \boldsymbol{p}_v \times \boldsymbol{q}_v \end{bmatrix}. \tag{12}$$

For a unit quaternion:

$$\boldsymbol{q}^{-1} = \boldsymbol{q}^* = [q_w - q_x - q_y - q_z]^T. \tag{13}$$

If:

$$\boldsymbol{q}_\Delta = [q_{\Delta,w} q_{\Delta,x} q_{\Delta,y} q_{\Delta,z}]^T, \tag{14}$$

then the angular orientation change is:

$$\Delta\theta_{\mathrm{rad}}(k) = 2\arccos\left(\left|q_{\Delta,w}(k)\right|\right), \tag{15}$$

$$\Delta\theta_{\deg}(k) = \frac{180}{\pi}\Delta\theta_{\mathrm{rad}}(k). \tag{16}$$

The absolute value accounts for the equivalent quaternion representations q and −q. Orientation change is confirmed when:

$$C_{ori}(k) = \left[\Delta\theta_{\deg}(k) > T_\theta\right], \tag{17}$$

where T_θ is the orientation-change threshold in degrees.

The same relative rotation can be expressed in matrix form:

$$R_\Delta = R(\boldsymbol{q}_{pre})^T R(\boldsymbol{q}_{cur}), \tag{18}$$

$$SO(3) = \{R \in \mathbb{R}^{3\times3}: R^T R = I,\ \det(R) = 1\}. \tag{19}$$

For $R_\Delta \in SO(3)$, the corresponding rotation angle is:

$$\Delta\theta_{\text{rad}} = \arccos\left(\frac{\operatorname{tr}(R_\Delta) - 1}{2}\right). \tag{20}$$

The firmware uses Eqs. (15)–(16), while Eq. (20) gives the geometric interpretation of the same feature as a relative rotation angle on SO(3).

Post-event inactivity gate

After orientation-change confirmation, a post-event time window is analysed. A sample is treated as static if:

$$C_{still}(k) = \left[|A_{vm}(k) - 1| < T_{A_s}\right] \wedge \left[G_{vm}(k) < T_{G_s}\right], \tag{21}$$

where T_{As} and T_{Gs} are stillness thresholds for acceleration and angular velocity.

For a window W containing N_W samples, the stillness ratio is:

$$R_{still} = \frac{1}{N_W} \sum_{k\in W} \mathbb{I}\left(C_{still}(k)\right), \tag{22}$$

where:

$$\mathbb{I}(P) = \begin{cases} 1, \text{if } P \text{ is true,} \\ 0, \text{otherwise.} \end{cases} \tag{23}$$

The post-event inactivity condition is:

$$C_{post}(k) = [R_{still}(k) > T_{still}]. \tag{24}$$

Finite-state event logic

The detector operates on the binary decision vector:

$$\boldsymbol{u}_k = [C_{dyn}(k) C_{ori}(k) C_{post}(k)]^T. \tag{25}$$

The finite-state machine is described by:

$$s_{k+1} = f(s_k, \boldsymbol{u}_k), \qquad s_k \in \{S_N, S_C, S_O, S_I, S_A, S_R\}. \tag{26}$$

The states denote normal operation, candidate event, orientation confirmation, inactivity analysis, alarm and rejection. S_R is used here as a logical rejection outcome rather than a persistent firmware state.

The main transitions of the finite-state logic are summarized as:

$$\begin{gathered} S_N \overset{C_{\text{dyn}}}{\rightarrow} S_C, \\ S_C \overset{C_{\text{ori}}}{\rightarrow} S_O, \\ S_O \rightarrow S_I, \\ S_I \overset{C_{\text{post}}}{\rightarrow} S_A, \\ S_I \underset{t_W > T_W}{\overset{\neg C_{\text{post}}}{\rightarrow}} S_R. \end{gathered} \tag{27}$$

where t_W is the elapsed post-event analysis time and T_W is the maximum time allowed for inactivity confirmation.

A fall-like event is therefore defined as a temporally ordered sequence:

$$C_{alarm} = C_{dyn}(k_1) \wedge C_{ori}(k_2) \wedge C_{post}(k_3), \qquad k_1 < k_2 < k_3. \tag{28}$$

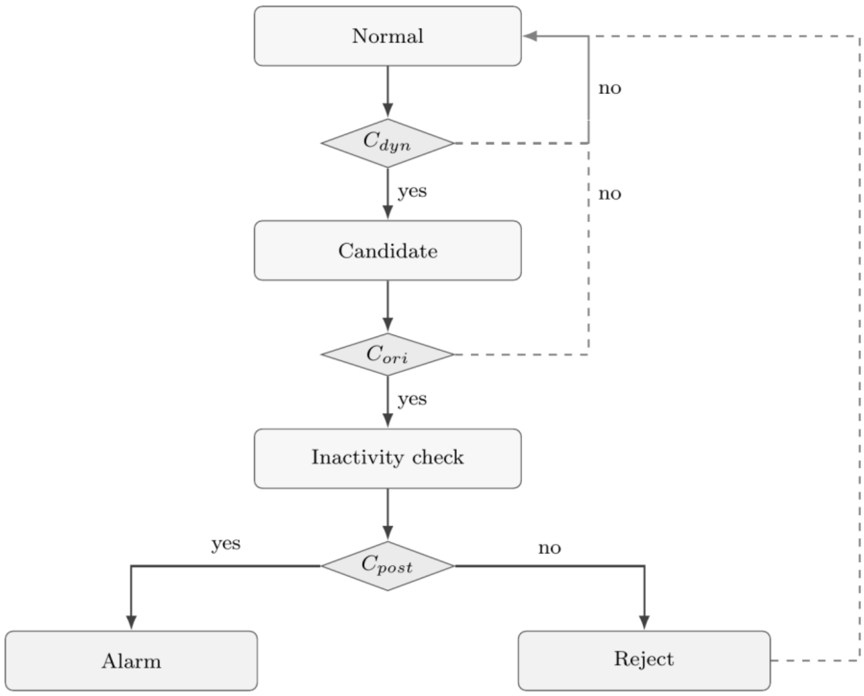

Fig. 1. Compact finite-state implementation of the inactivity-gated event logic

Preliminary functional tests

The functional tests used a prototype ESP32–LSM9DS1 node and the working thresholds listed in Tab 1. These values were selected to verify the logical behaviour of the decision sequence and were not optimized for a specific population, sensor placement or activity dataset. The tests should be interpreted as a functional proof-of-concept of the event logic, not as a statistical validation of fall-detection performance.

Tab. 1. Working thresholds used during functional tests

Parameter	Value	Meaning
T_A	2.50 g	acceleration-magnitude threshold
T_G	280 °/s	angular-velocity-magnitude threshold
T_J	18 g/s	jerk threshold
T_θ	45°	orientation-change threshold
T_{As}	0.25 g	stillness acceleration tolerance
T_{Gs}	30 °/s	stillness angular-velocity threshold
T_{still}	0.60	required stillness ratio
Orientation window	1000 ms	time allowed for orientation confirmation
Inactivity window	1500 ms	post-event stillness analysis window

Five functional scenarios were tested to verify whether each gate suppresses a different class of false alarm. Static rest was used to check baseline stability, slow orientation change tested whether posture change alone is insufficient, an impulse without posture change tested rejection before the inactivity stage, dynamic non-fall movement evaluated the inactivity gate, and a controlled fall-like event verified the full alarm path.

Tab. 2. Observed gate-level behaviour of the proposed event logic

Scenario	Dynamic gate	Orientation gate	Inactivity gate	Observed decision
Static rest	No	No	Stable only	No alarm
Slow orientation change	No	Satisfied or partially satisfied	Not evaluated	No alarm
Impulse without posture change	Yes	No	Not evaluated	Rejected before inactivity stage
Dynamic non-fall movement	Yes	Yes	No	Rejected due to lack of post-event inactivity
Controlled fall-like event	Yes	Yes	Yes	Alarm generated

The functional tests confirmed the intended role of the sequential gate structure. Static rest did not activate the dynamic gate. Slow reorientation did not produce an alarm because the dynamic gate was not satisfied. A short impulse without sufficient posture change was rejected before the inactivity stage. In the dynamic non-fall movement, both the dynamic and orientation-change gates were satisfied, but the event was rejected because post-event inactivity was absent. In the controlled fall-like event, all three gates were satisfied and an alarm was generated.

This result supports the intended interpretation of the detector as a sequential decision structure rather than a single-threshold alarm rule.

Discussion

The proposed logic should be interpreted as an embedded decision structure, not as a fully validated fall-detection classifier. Its main advantage is interpretability: every decision stage is explicit, threshold-based and available in event logs. This is useful during early-stage development of wearable safety-monitoring prototypes, where false positives should be attributable to a specific decision gate.

The functional test matrix separates the roles of individual gates. Static rest evaluates baseline stability, slow orientation change verifies that posture change alone is insufficient, an impulse without posture change tests rejection before the inactivity stage, dynamic non-fall movement evaluates the post-event inactivity gate, and the controlled fall-like event verifies the full alarm path. The observed decisions were consistent with these expected roles.

The key observation is that the post-event inactivity gate changed the final decision in a case where both the dynamic and orientation-change gates were already satisfied. This supports the practical value of evaluating the post-event state rather than relying solely on impact-like or rotation-like signatures. In wearable systems, abrupt sensor motion alone should not automatically trigger an alarm.

The quaternion-based orientation-change feature provides a compact posture-change descriptor. Instead of analysing roll, pitch and yaw separately, the firmware evaluates one relative rotation angle between the pre-event and current orientations. This avoids dependence on a selected Euler-angle sequence and remains consistent with common IMU orientation-estimation methods.

Only functional trials were performed, without multi-subject validation or a statistically meaningful number of repetitions. Sensitivity, specificity, precision and F1-score are therefore not reported. The influence of sensor placement, threshold tuning, BLE packet loss, magnetic disturbances and real-world industrial motion patterns remains to be evaluated.

Conclusions

This paper presented an ESP32–LSM9DS1 implementation of a lightweight inactivity-gated event logic for fall-like IMU screening. The logic combines dynamic inertial features, quaternion-based orientation-change confirmation and post-event inactivity verification. In functional tests covering static rest, slow reorientation, impulse-only motion, dynamic non-fall movement and a controlled fall-like event, the observed decisions followed the intended sequential structure. Non-fall cases did not generate an alarm, whereas the controlled fall-like event generated an alarm.

The results indicate that post-event inactivity can serve as an interpretable decision gate in compact wearable IMU systems. Future work should include repeated logged trials, comparison with a non-gated threshold variant, systematic threshold tuning and validation under conditions closer to the intended application environment.

References

- Abbate, S., Avvenuti, M., Bonatesta, F., Cola, G., Corsini, P. and Vecchio, A. (2012), 'A Smartphone-Based Fall Detection System', Pervasive and Mobile Computing, vol. 8, no. 6, pp. 883-899, doi: 10.1016/j.pmcj.2012.08.003.
- Bourke, A. K. and Lyons, G. M. (2008), 'A Threshold-Based Fall-Detection Algorithm Using a Bi-Axial Gyroscope Sensor', Medical Engineering & Physics, vol. 30, no. 1, pp. 84-90, doi: 10.1016/j.medengphy.2006.12.001.
- Li, Y., Liu, P., Fang, Y., Wu, X., Xie, Y., Xu, Z., Ren, H. and Jing, F. (2025), 'A Decade of Progress in Wearable Sensors for Fall Detection (2015-2024): A Network-Based Visualization Review', Sensors, vol. 25, no. 7, p. 2205, doi: 10.3390/s25072205.
- Lin, H. C., Chen, M. J., Lee, C. H., Kung, L. C. and Huang, J. T. (2023), 'Fall Recognition Based on an IMU Wearable Device and Fall Verification through a Smart Speaker and the IoT', Sensors, vol. 23, no. 12, p. 5472, doi: 10.3390/s23125472.
- Madgwick, S. O. H. (2010), 'An Efficient Orientation Filter for Inertial and Inertial/Magnetic Sensor Arrays', Internal Report, University of Bristol.
- Mahony, R., Hamel, T. and Pflimlin, J.-M. (2008), 'Nonlinear Complementary Filters on the Special Orthogonal Group', IEEE Transactions on Automatic Control, vol. 53, no. 5, pp. 1203-1218, doi: 10.1109/TAC.2008.923738.
- Noury, N., Rumeau, P., Bourke, A. K., ÓLaighin, G. and Lundy, J. E. (2008), 'A Proposal for the Classification and Evaluation of Fall Detectors', IRBM, vol. 29, no. 6, pp. 340-349, doi: 10.1016/j.irbm.2008.08.002.
- Ramachandran, A. and Karuppiah, A. (2020), 'A Survey on Recent Advances in Wearable Fall Detection Systems', BioMed Research International, vol. 2020, p. 2167160, doi: 10.1155/2020/2167160.
- Sabatini, A. M. (2006), 'Quaternion-Based Extended Kalman Filter for Determining Orientation by Inertial and Magnetic Sensing', IEEE Transactions on Biomedical Engineering, vol. 53, no. 7, pp. 1346-1356, doi: 10.1109/TBME.2006.875664.
- STMicroelectronics (2016), LSM9DS1: iNEMO Inertial Module: 3D Accelerometer, 3D Gyroscope, 3D Magnetometer. Datasheet, Rev. 4.
- Sucerquia, A., López, J. D. and Vargas-Bonilla, J. F. (2017), 'SisFall: A Fall and Movement Dataset', Sensors, vol. 17, no. 1, p. 198, doi: 10.3390/s17010198.
- Tian, J., Mercier, P. and Paolini, C. (2024), 'Ultra Low-Power, Wearable, Accelerated Shallow-Learning Fall Detection for Elderly At-Risk Persons', Smart Health, vol. 33, p. 100498, doi: 10.1016/j.smhl.2024.100498.

- Tseng, C.-K., Huang, S.-J. and Kau, L.-J. (2025), 'Wearable Fall Detection System with Real-Time Localization and Notification Capabilities', Sensors, vol. 25, no. 12, p. 3632, doi: 10.3390/s25123632.
- Vavoulas, G., Chatzaki, C., Malliotakis, T., Pediaditis, M. and Tsiknakis, M. (2016), 'The MobiAct Dataset: Recognition of Activities of Daily Living Using Smartphones', In: Proceedings of the 2nd International Conference on Information and Communication Technologies for Ageing Well and e-Health, pp. 143-151, doi: 10.5220/0005792401430151.
- Wu, F., Zhao, H., Zhao, Y. and Zhong, H. (2015), 'Development of a Wearable-Sensor-Based Fall Detection System', International Journal of Telemedicine and Applications, vol. 2015, p. 576364, doi: 10.1155/2015/576364.
- Yu, X., Jang, J.-H. and Xiong, S. (2021), 'A Large-Scale Open Motion Dataset (KFall) and Benchmark Algorithms for Detecting Pre-Impact Fall of the Elderly Using Wearable Inertial Sensors', Frontiers in Aging Neuroscience, vol. 13, p. 692865, doi: 10.3389/fnagi.2021.692865.