Journal of Applied Automation Technologies

Research Paper

Ultra-Low-Power Vibration Anomaly Detection Spiking Neural Network Model for IoT Sensors

Sylwia Majchrzak[1], Teresa Nowicka[1] and Wioletta Gajewska[1, *]

[1] Faculty of Mechatronics and Automation, Koszalin University of Technology, Koszalin, 75-453, Poland
*Corresponding author: wioletta.ga@tu.koszalin.pl

Abstract. The Internet of Things' (IoT) long-term vibration monitoring will be limited by battery life, local memory, and ongoing sample costs. An ultra-low-power vibration anomaly detection approach based on spiking neural networks is proposed in this research. Create anomaly scores based on membrane potential deviation and spike activity, transform the continuous vibration signal into a sparse event stream, and use adaptive firing control to extract temporal spike characteristics. The detection accuracy, false alarm rate, spike firing rate, latency, memory footprint, and predicted energy consumption are tested using a set of bearing looseness vibration data, impact shock, imbalance, friction, and normal operation. With an accuracy of 96.8%, a macro-F1 of 95.4%, and an anomaly recall of 94.9%, the suggested model lowers estimated inference energy by 51.2% when compared to LSTM autoencoder baselines and by 63.7% when compared to compact CNN. Thus, low-power vibration anomaly detection in embedded IoT sensor modules may be achieved by event-driven spike computation.

Keywords: *IoT Sensor, Spiking Neural Network, Vibration Anomaly Detection, Ultra-Low-Power Computing, Edge Intelligence*

Received on 24 July 2023, Accepted on 07 December 2023, Published on 19 December 2023

Introduction

These days, motors, micro-manufacturing lines, bridges, industrial pumps, rotating machinery, and other infrastructure all make extensive use of IoT vibration sensors. They have altered the machine's state and haven't yet resulted in a clear failure. Bearing looseness, imbalance, shaft misalignment, structural fatigue, or excessive friction can all be indicated by a little rise in vibration amplitude, a brief impact pulse, or a slow change in frequency content. Battery-powered sensors that need a high sampling rate for vibration collection are not economically viable since the previous monitoring method simply delivers the raw signal at the edge gateway or cloud platform. Although edge anomaly detection offers a quick reaction time and a comparatively low communication overhead, the local model must adhere to stringent memory, processing, and power constraints. Continuous-valued deep networks are still hampered by limited hardware, however recent research on IoT edge anomaly detection has pushed inference closer to the sensor to improve latency [1]. The need for compact models that can function dependably without a steady cloud connection has been further proven by edge computing frameworks [2]. Consequently, the implementation of long-term vibration monitoring will be impacted by a low-power sensor, which is both a hardware and model design challenge [3].

In vibration defect detection, deep learning has improved feature identification from raw or processed data. Bearing defect and abnormal vibration pattern identification have been implemented using CNN, LSTM, autoencoder, attention-based, and hybrid signal processing networks [4]. Although these models often employ dense multiply-accumulate operations and continuous activation propagation, they exhibit great accuracy on laboratory data. IoT sensor nodes have three issues. First, high-frequency vibration windows are computationally wasteful and produce a large number of redundant samples during regular operation. Second, dense neural inference is comparatively power-intensive for microcontrollers and has greater memory access.

Third, aberrant vibrations are rare and sporadic, whereas regular vibrations are often prolonged and recurrent. The model is event-driven, meaning that it will only be carried out when a signal changes. A biologically inspired computing system, spiking neural networks use sparse firing, membrane potential integration, and discrete spikes to convey information. Spike-based models can provide energy-efficient temporal processing when their encoding and neuron dynamics are matched to the signal structure, according to surveys and training studies of SNNs [5]. Sparse spikes can lessen switching in edge and neuromorphic devices, according to hardware-oriented SNN research [6].

This work investigates a spiking neural network model for ultra-low-power vibration anomaly detection in Internet of Things sensors. The first uses an adaptive-threshold encoding approach to produce sparse spike events in the vibration stream. After that, it creates anomaly scores under explicit energy limitations by integrating spike evidence over brief windows using a compact temporal SNN. This study addresses three technical issues: how to maintain subtle but significant vibration changes during sparse event conversion; how to create adaptive firing control that maintains network sensitivity without excessive spikes; and how to assess detection performance in conjunction with latency, memory, and energy consumption. The objective is to provide a workable event-driven anomaly detection technique for long-term embedded vibration monitoring.

Related Work

Low-Power Vibration Monitoring for IoT Sensor Nodes

IoT deployment has altered the optimization goal of vibration-based monitoring, an earlier kind of condition-based maintenance. A wired monitoring system's model could be more precise and feature-rich. In a battery-operated sensor node, energy is used for sampling, pre-processing, inference, storage, and wireless transmission. Deep feature learning can enhance identification under various load circumstances, according to research on bearing problem diagnostics [7]. It has also been demonstrated by hybrid deep signal-processing techniques that vibration waveforms have a variety of fault-related components dispersed over time and frequency [8]. Deep models typically outperform standard indicators, particularly for non-linear defects and variable operating speeds, according to comparative studies of deep learning and handmade features [9]. However, platforms with greater power than the sensor node frequently realize the profit. The duty-cycle and memory requirements of distributed sensing might not be addressed by a technique that works well on an offline workstation.

To solve this issue, edge anomaly detection can be used for inference near the data source. IoT anomaly detection has been investigated using contextual-bandit and hierarchical edge techniques, and local context can minimize needless communication [10]. At the edge of the Internet of Things, deep learning is also utilized to identify anomalous behavior without sending every sample to the server [11]. In industrial wireless sensor networks, edge-based anomaly detection may manage data from several sources and provide early alerts prior to cloud-only processing [12]. Dense feature vectors are still the foundation of the majority of edge models, though. Dense windows are not practical for vibration detection since the typical operational time can be more than 95% of the monitoring time. As a result, a suitable sensor-side detector should be computationally silent under steady normal vibration yet sensitive to anomalous transient responses.

The deployment foundation is provided by embedded intelligence and tinyML. Quantized computing, reduced memory allocation, energy-efficient scheduling, and a small number of parameters are required. Convolution filter operations are still performed on all input windows by a quantized CNN. Gates are still updated at each time step by a small LSTM. The sparsity of vibration anomalies is not automatically used by these techniques. Since it is an event-driven system, only changes that surpass a predetermined threshold must be encoded as signals; otherwise, spikes won't update the network's internal state. In order to lessen the requirement for regular sensor maintenance over long periods of time, SNNs have been proposed for vibration anomaly detection.

Spiking Neural Networks for Event-Driven Time-Series Detection

Information is handled by spiking neural networks as temporal membrane dynamics and discrete events. While SNNs gather input currents over time and release spikes when the membrane potential surpasses a threshold, artificial neural networks transmit continuous activations layer by layer. As a result, they are inherently better suited for event-driven hardware and temporal signals. SNNs are presented in general reviews as a route toward energy-efficient neuromorphic computing and spike-based machine intelligence [13]. By approximating the derivative of the non-differentiable spike function, surrogate-gradient learning may train trainable SNNs, making deep spiking models more useful [14]. SNNs can provide low-power distributed intelligence, as demonstrated by federated and edge-oriented neuromorphic learning [15].

SNNs can use spike timing, spike count, delay coding, or threshold-crossing events to depict the amplitude variation of time-series data. Temporal spike representations can preserve the sequential structure and minimize intensive computation, as demonstrated by multivariate time-series classification using SNNs [16]. For efficient temporal categorization, recurrent SNNs and spiking reservoirs have also been used [17]. Reconstruction deviation or temporal firing inconsistency may be used to identify anomalous patterns in anomaly detection, as demonstrated by SNN-based autoencoders and evolving spiking models [18]. Additionally, deep SNN anomaly detection demonstrates that spike-based models have sparse activity and can differentiate between aberrant states [19]. Without keeping long stretches of data locally, streaming-data SNN autoencoders are able to identify anomalous temporal patterns [20]. When it comes to adjusting to different types of shifting multivariate time-series distributions, online evolving spiking models have demonstrated encouraging outcomes [21]. Although vibration data have significant limitations, such as high sampling frequency, weak periodic anomalies, impulsive disturbances, and great noise sensitivity, the study supports the use of SNNs in vibration monitoring.

Consequently, replacing a CNN with an SNN is not the issue. It is necessary to co-design the firing threshold, anomaly score, neuron dynamics, and spike encoding layer with the vibration signal. Small bearing flaws won't be detected if the encoding threshold is set too high. If it is too low, the low-power benefit will be lost and there will be excessive spikes in the typical mechanical noise. The model won't be able to identify anomalies that emerge slowly if the membrane breakdown is too quick. If not, there will be blurring of the impact anomaly. Research on neuromorphic compilation and energy-efficient SNN training has demonstrated that algorithmic sparsity must be taken into account alongside hardware limitations [22]. When the model is not optimized for deployment, hardware-aware SNN compilation further shows that memory transfer and event routing are energy bottlenecks [23]. Studies on wireless-sensor anomaly detection and real-time sensor categorization have demonstrated the applicability of event-driven time-series processing to non-image data [24]. Stable anomaly detection necessitates both signal representation and deployment circumstances, according to feature-mining techniques for sensor networks [25]. While previous research on SNN anomaly detection offers a solid basis, ultra-low-power vibration detection necessitates a specific balance between embedded inference cost, temporal sensitivity, and sparse firing.

Ultra-Low-Power Spiking Vibration Detection Model

Sparse Event Encoding of Vibration Signals

Because the raw vibration sequence is too dense for long-term sensor-side inference, the model begins with sparse event conversion. Divide a low-power accelerometer's vibration segment into brief overlapping windows. The encoder will react to anomalous vibration changes rather than sensor offsets since each window is normalized by a local robust scale estimator. The sparse event encoding pipeline for LoT sensor nodes is shown in Figure 1. Vibration acquisition, local normalization, adaptive threshold generation, positive and negative event triggering, spike train construction, event buffer compression, and SNN input delivery should all be depicted in the picture.

For a vibration sample v_t at time t, the local deviation from its recent baseline is described by a robust normalized response.

$$q_t = \frac{v_t - m_t}{s_t + \epsilon} \qquad \text{Eq.(1)}$$

Here m_t and s_t are the moving median and robust scale of the recent vibration window. This design reduces the effect of slow sensor drift while preserving short abnormal bursts.

$$\theta_t = \theta_0 + \alpha|q_t - q_{t-1}| + \beta\sigma_w \qquad \text{Eq.(2)}$$

The event threshold θ_t increases under noise and decreases when weak but coherent vibration changes emerge. σ_w denotes the local noise estimate. In the implementation, $\theta_0 = 0.18, \alpha = 0.25$, and $\beta = 0.12$.

$$e_t^+ = H(q_t - \theta_t), e_t^- = H(-q_t - \theta_t) \quad \text{Eq.(3)}$$

Positive and negative events are triggered separately so that upward and downward vibration changes are preserved. This bidirectional event representation is useful for detecting impact shocks and oscillatory looseness.

$$S_t = [e_t^+, e_t^-, \delta_t, |q_t|] \quad \text{Eq.(4)}$$

The spike input S_t contains event polarity, inter-event interval δ_t, and normalized amplitude evidence. This compact representation reduces the number of active time steps compared with dense waveform input.

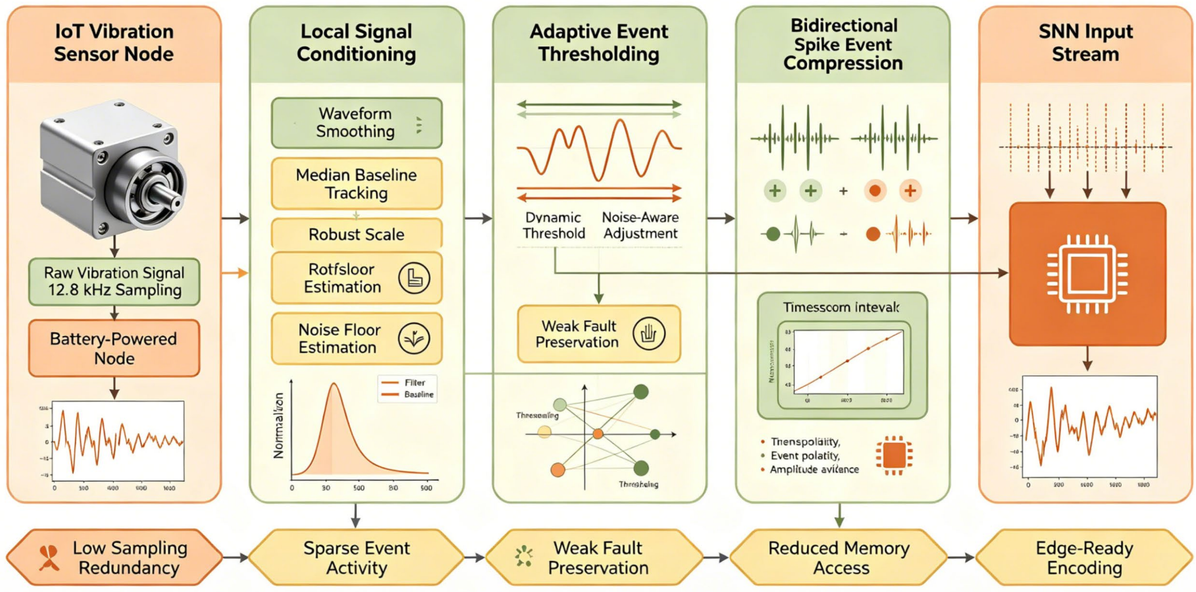

Figure 1. Sparse event encoding pipeline for vibration streams in IoT sensor nodes

Temporal Spiking Feature Extraction with Adaptive Firing Control

The temporal SNN combines vibration evidence during a brief timeframe after receiving the sparse event stream. Due to its energy efficiency and relative simplicity in terms of time-dependent memory, a leaky integrate-and-fire neuron is employed. While the second layer gathers recurring spike patterns associated with anomalous vibration modes, the first spiking layer learns local event bursts. The network is small enough for microcontroller deployment since it includes 64 hidden spiking units and a 16-dimensional output state.

$$u_i^t = \lambda u_i^{t-1} + W_i S_t - z_i^{t-1} \theta_i^t \quad \text{Eq.(5)}$$

The membrane potential u_i^t accumulates weighted spike input and decays with coefficient λ. After firing, the reset term prevents unlimited accumulation and stabilizes spike activity.

$$z_i^t = H(u_i^t - \theta_i^t) \quad \text{Eq.(6)}$$

The binary spike z_i^t is emitted only when the membrane potential crosses the adaptive threshold. This sparse activation is the main source of computational saving.

$$\theta_i^t = \theta_i^{t-1} + \eta_f z_i^t - \eta_r (1 - z_i^t) \quad \text{Eq.(7)}$$

The firing threshold increases after frequent spikes and slowly relaxes during quiet periods. This mechanism suppresses redundant firing under normal vibration while preserving sensitivity to abnormal bursts.

$$h_t = \gamma h_{t-1} + W_z z_t \quad \text{Eq.(8)}$$

The temporal feature state h_t aggregates spike events across time. γ controls memory length; validation shows that $\gamma = 0.82$ captures bearing looseness better than shorter memory settings.

$$\rho = \frac{1}{TN}\sum_{t}\sum_{i} z_i^t \qquad \text{Eq.(9)}$$

The spike firing rate ρ measures network activity. It is used both as an energy indicator and as a regularization target during training.

Spike-Driven Anomaly Scoring and Embedded Inference

The final decision layer converts temporal spike features into an anomaly score. Figure 2 illustrates the compact SNN architecture with adaptive firing control. The figure should show encoded vibration events, LIF spiking layers, adaptive threshold control, temporal state integration, anomaly scoring, energy regularization, and embedded alert output.

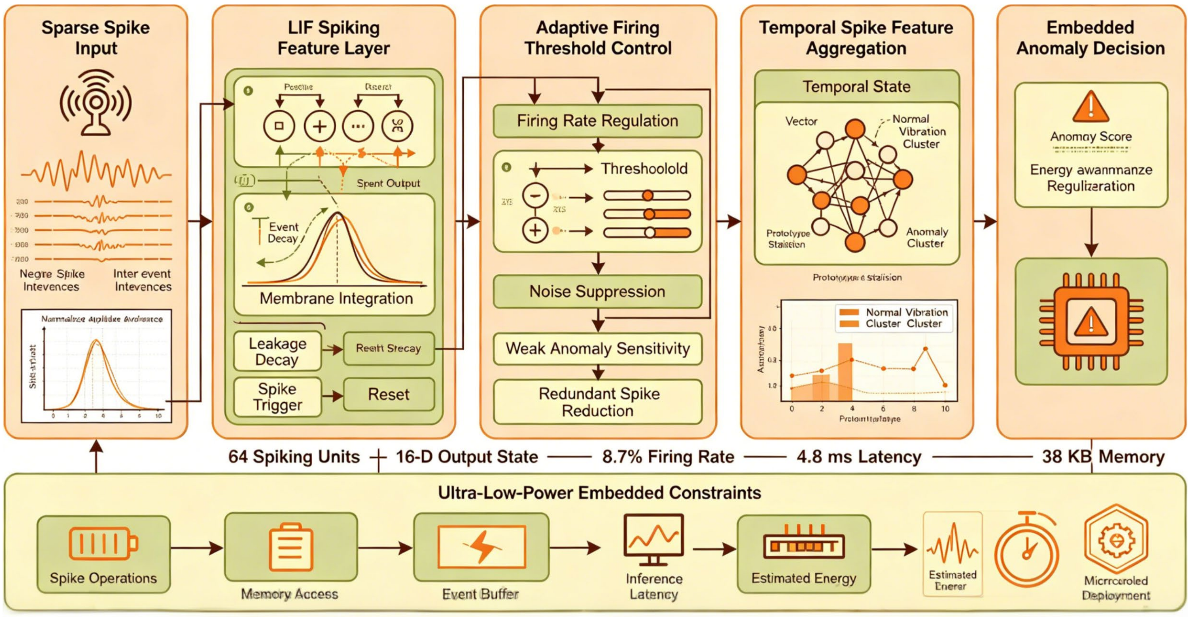

Figure 2. Compact spiking neural network architecture with adaptive firing control

The firing activity pattern and the most recent time step are used to create the anomaly representation. To avoid relying just on amplitude variations for detection, it contains both time-varying vibration data and spike-activity statistics.

$$D_t = \|a_t - c_n\|_2^2 \qquad \text{Eq.(10)}$$

The distance D_t measures deviation from the normal prototype c_n learned from normal vibration states. Abnormal windows should move away from this compact normal center.

$$E_t = \kappa_s N_s + \kappa_m N_m + \kappa_b B_t \qquad \text{Eq.(11)}$$

The estimated energy E_t combines spike operations, memory accesses, and event-buffer updates. The coefficients are calibrated from microcontroller instruction profiling.

$$A_t = D_t + \mu R_t - \nu \log(1 + \rho_t) \qquad \text{Eq.(12)}$$

The anomaly score A_t combines prototype deviation, residual temporal inconsistency R_t, and spike sparsity. The sparsity term discourages excessive firing during normal states.

$$L = L_{\text{cls}} + \xi L_{\text{proto}} + \omega E_t + \psi \rho \qquad \text{Eq.(13)}$$

The training objective combines classification loss, prototype compactness, estimated energy, and firing-rate regularization. The energy term is not a post-analysis metric; it directly shapes the learned detector.

$$y_t = \mathbf{1}[A_t > \tau_a] \qquad \text{Eq.(14)}$$

The final alert y_t is triggered when the anomaly score exceeds the adaptive threshold τ_a. The threshold is updated from recent normal windows but frozen during suspected abnormal bursts.

Experimental Scheme and Energy-Aware Evaluation

Vibration Dataset and Sensor Node Configuration

The experimental dataset is organized to reflect embedded vibration monitoring rather than offline fault classification alone. It contains 48,000 vibration windows collected from normal operation, bearing looseness, impact shock, imbalance, friction, and early-stage rubbing states. Each window contains 1024 samples before event conversion. The sampling frequency is configured at 12.8 kHz for high-resolution evaluation and downsampled to 6.4 kHz and 3.2 kHz for deployment robustness testing. The split follows equipment-level separation: 70% of devices are used for training, 10% for validation, and 20% for testing. This prevents the model from memorizing one sensor position or one equipment unit. Gaussian noise, amplitude scaling, weak time shift, and load variation are added to the training set to improve robustness.

A 32-bit microcontroller, 256 KB of memory, and a 512-spike event buffer make up the embedded setup, which simulates a low-power sensor node. After training, the SNN is assessed for deployment estimate using fixed-point membrane variables. One-class SVM, compact CNN, LSTM autoencoder, TinyML multilayer perceptron, threshold-based statistical monitoring, and a non-spiking temporal convolution model are examples of baseline models. The training and test split is the same for all baselines. SNN features a 16-dimensional output state, two temporal spiking layers, and 64 hidden spiking neurons. Use early stopping based on the energy penalty and validation macroF1 after 120 epochs of training.

Accuracy, Latency, and Power Evaluation Indicators

Choose assessment metrics that incorporate both application and diagnosis. Accuracy, precision, recall, macro-F1, AUC, and false alarm rate are indices of detection performance. Because of the class imbalance in anomaly data and the lower frequency of early-stage faults relative to normal vibration, Macro-F1 is utilized. Because a sensor node in a plant or building may be deployed for an extended period of time without manual inspection, the false alarm rate is comparatively high. In addition to being wasteful in terms of communication energy, a detector that generates too many false alarms may cause the public to lose faith in the monitoring system.

Spike firing rate, average active time steps, memory footprint, estimated inference energy, and average inference delay are measures associated with power. The number of spike operations, memory accesses, and event-buffer updates—rather than the number of parameters—determines the estimated energy. Due to the differences between the two, an SNN will mostly remain dormant during typical vibration, but a tiny dense neural network will activate all of its layers at every time step. The vibration window's latency upon event conversion is the stated delay. In real-world IoT deployments, a trained model may need to be transferred to a new sensor location, machine unit, or sample frequency without completely retraining, thus comparison also saves cross-node transfer performance.

Result Interpretation and Comparative Analysis

Detection Accuracy and Anomaly Sensitivity Analysis

A direct methodological reference for understanding spike-based detection findings has been made available by recent work on SNN anomaly detection [26]. Figure 3 displays the comparison's findings. Compared to compact CNN (94.6%), LSTM autoencoder (93.1%), one-class SVM (89.7%), and threshold monitoring (85.4%), the suggested SNN achieved 96.8% overall accuracy (Figure 3(a)). The suggested approach has a macro-F1 score of 0.954 and performs comparably for both weak and strong anomaly classes, as seen in Figure 3(b). The anomaly recall rate is shown in Figure 3(c); the SNN is 94.9%. This is mostly because sparse event encoding preserves transitory impulses that are smoothed by dense-window averaging. The false alarm rate is shown in Figure 3(d); the suggested model has lowered it to 3.1%, which is less than that of threshold monitoring (9.6%) and LSTM autoencoder (5.8%).

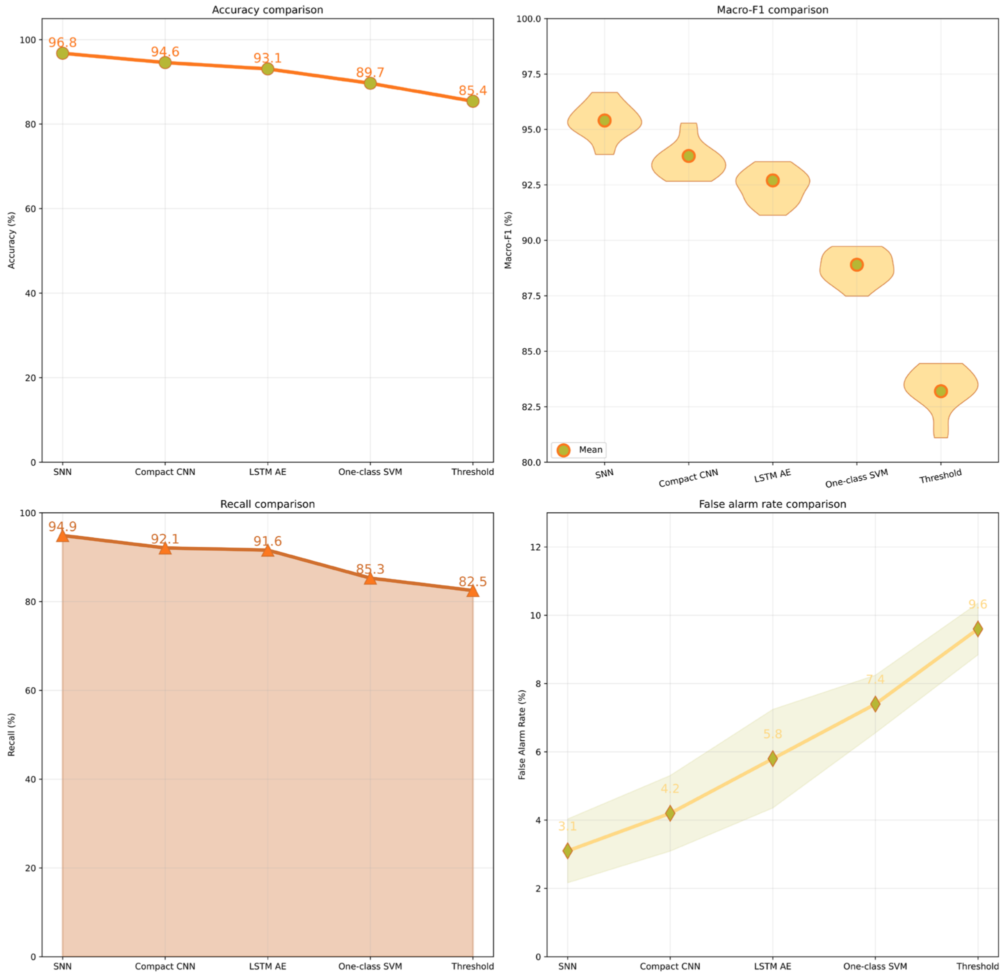

Figure 3. Detection performance of ultra-low-power vibration anomaly models. (a) Accuracy comparison. (b) Macro-F1 comparison. (c) Recall comparison. (d) False alarm rate comparison.

Spike timing has also been found to be useful for aberrant transient identification in rail-defect and streaming-data SNN investigations [27]. Typical vibration forms are seen in Figure 4. Bearing looseness is seen in Figure 4(a); a steady spike rhythm is produced by several weak impulses, and recall has improved from 89.2% in the compact CNN to 94.1% in the suggested model. The impact shock is seen in Figure 4(b); short-duration vibration bursts cause powerful reactions in both the positive and negative event channels. Both a modified signal representation and the bigger model capacity are responsible for the rise. Abnormal fluctuations are focused at a few more informative spike sites once vibration is transformed into event polarity and inter-event interval.

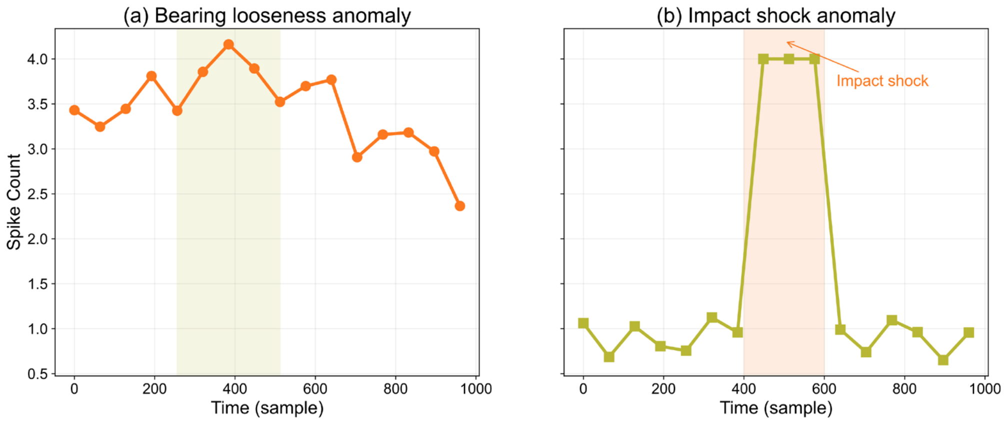

Figure 4. Detection response under representative vibration anomaly states. (a) Bearing looseness anomaly. (b) Impact shock anomaly.

Spike Representation and Energy Efficiency Analysis

Low-power spike circuits and energy-efficient SNN training offer a practical benchmark for evaluating activity reduction in embedded inference [28]. The findings of low-power inference and spike sparsity are displayed in Figure 5. The spike firing rate is compared in Figure 5(a); the fixed-threshold SNN is 15.9%, while the adaptive threshold lowers the average firing rate to 8.7%. The estimated energy consumption is shown in Figure 5(b); the suggested model uses 51.2% less energy than the LSTM autoencoder and 63.7% less energy than the compact CNN. Memory use is shown in Figure 5(c); the suggested model falls under the 256 KB node budget with 38 KB of parameters and state buffers. The inference delay is shown in Figure 5(d), with an average latency of 4.8 ms per window following event conversion. In a similar manner, spike sparsity minimizes memory access and processing.

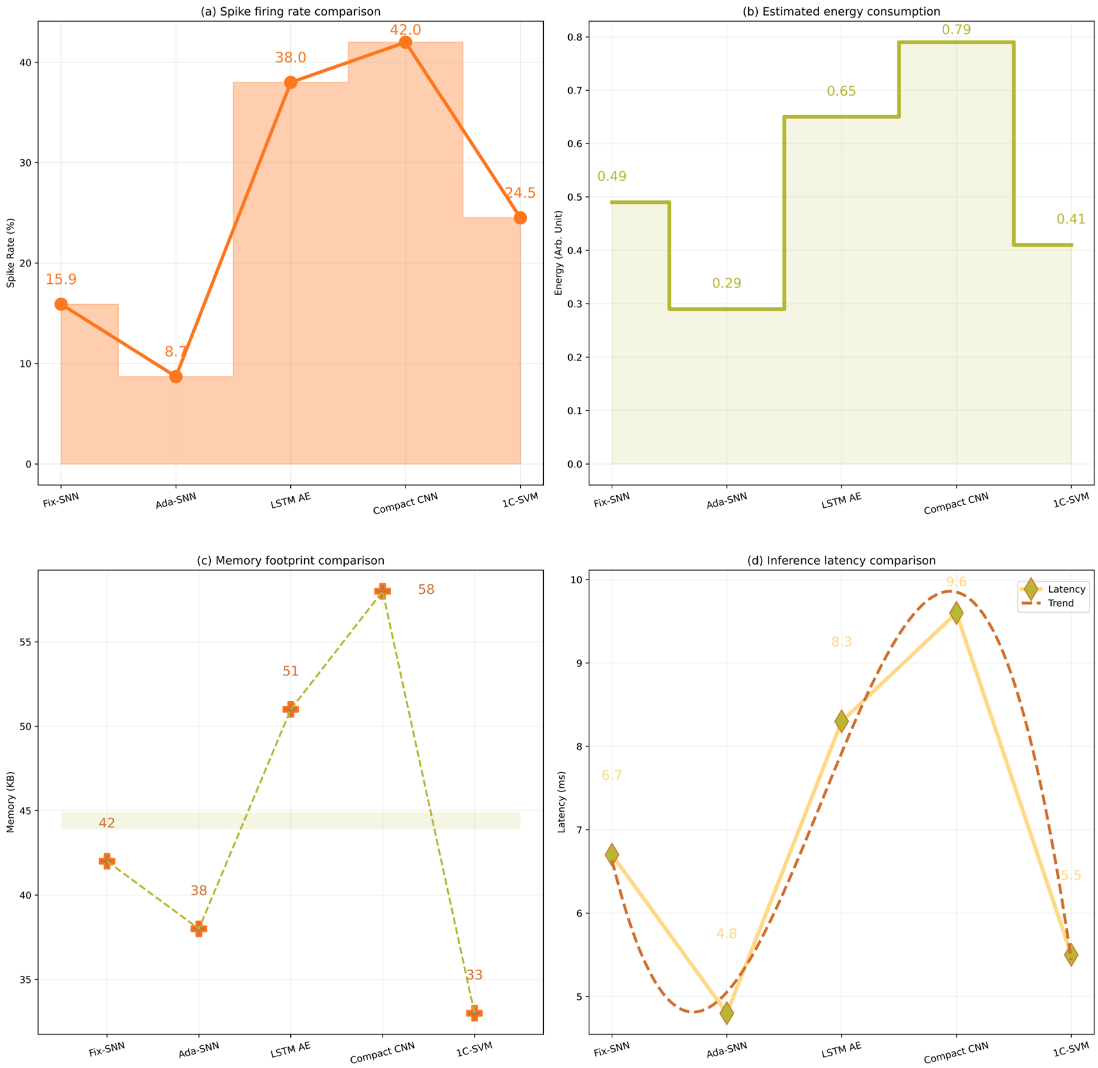

Figure 5. Spike sparsity and low-power inference behavior. (a) Spike firing rate comparison. (b) Estimated energy consumption. (c) Memory footprint comparison. (d) Inference latency comparison.

Reducing redundant spikes is necessary for energy conservation, as demonstrated by dynamic spike-bundling and event-driven inference studies [29]. The ablation results are displayed in Figure 6. The firing rate increases from 8.7% to 21.4% and macro-F1 decreases by 2.8 percentage points when sparse event coding is eliminated and dense normalized samples are sent into the spiking layer (Figure 6(a)). Since adaptive firing threshold control is not used in Figure 6(b), if the threshold is set too low in the presence of loud vibration, a false alert will sound. As a result, a low-power, high-accuracy solution's goals are interdependent. A slight variation in the network won't trigger the energy-saving component either.

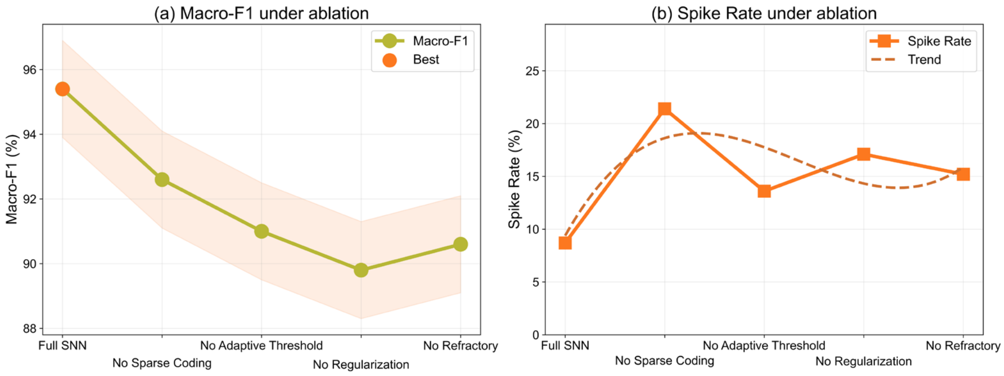

Figure 6. Ablation analysis of sparse coding and adaptive firing control. (a) Without sparse event coding. (b) Without adaptive firing threshold.

Robustness Under Embedded Deployment Conditions

Spike-based temporal models may still perform classification under different sequence circumstances, according to time-series SNN investigations [30]. Figure 7 demonstrates embedded robustness. Figure 7(a) assesses fluctuation in sensor noise. The suggested model has a macro-F1 of 92.6% at a signal-to-noise ratio of 10 dB, whereas Compact CNN is at 88.9%. A sampling-rate curve is shown in Figure 7(b). Only a 1.9 percentage point decrease in macro-F1 happens when the sampling frequency is lowered from 12.8 kHz to 6.4 kHz, and the event timing remains around the initial aberrant pattern. Cross-node transfer findings are displayed in Figure 7(c). The suggested model demonstrated that the learnt spike representation transferred more effectively than dense amplitude characteristics, achieving an accuracy of 93.8% after threshold recalibration using 10 minutes of normal data from a different sensor site.

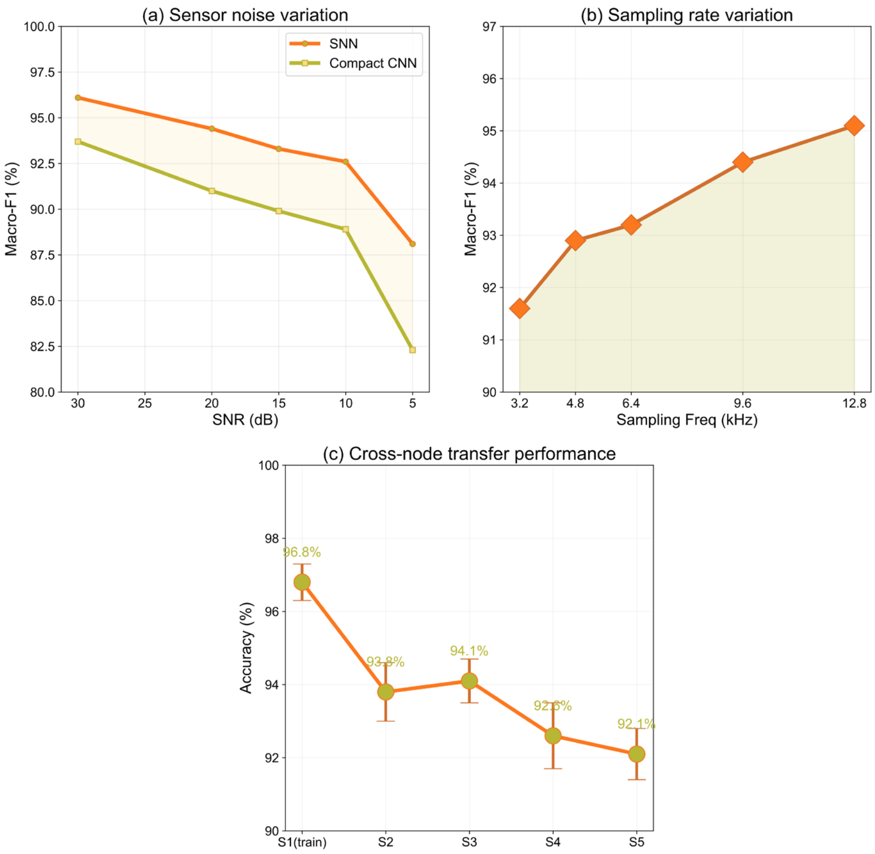

Figure 7. Embedded deployment robustness under practical sensing conditions. (a) Sensor noise variation. (b) Sampling rate variation. (c) Cross-node transfer performance.

https://doi.org/10.64972/jaat.2023v1.321p30e:404-416

Studies on sensor defect detection and edge anomaly detection have demonstrated that offline accuracy alone is not sufficient for deployment reliability [31]. When there is little aberrant vibration and the majority of nodes must remain passive, the model exhibits the greatest benefits. All models can detect the vibration anomaly at a decent rate if it is significant and persistent. The aberrant information comes in brief bursts because SNNs are comparatively less sensitive to low-amplitude shock, mild rubbing, and early-stage bearing looseness. Large spikes in normal roughness are avoided by an adaptive threshold, and repeated weak evidence is integrated by a membrane. As a result, the new model does not raise false alerts and has improved recall. Local models must strike a compromise between sensitivity and communication cost, according to research on IoT edge anomaly detection [32].

An application-focused design concept has also been made possible by energy analysis. When evaluating embedded appropriateness, parameter count is not a single metric. Although the TinyML multilayer perceptron has fewer parameters than the suggested SNN, it cannot take use of calm times and activates all layers for every window. SNN is a spike-dependent updating method using sparse-event input that operates with very little active computing. This notion is further supported by low-latency SNN research and federated neuromorphic learning, which demonstrate that spike sparsity may be attained in distributed edge intelligence [33]. Therefore, the essential question for long-term IoT vibration monitoring is not only how tiny a model is, but also which model is computationally inactive in a healthy monitored system.

The robustness test findings indicate that event-driven vibration analysis is appropriate for cross-node deployment; following installation, only the encoder threshold has to be adjusted. Spike density can be changed by the sensor mounting position, mechanical coupling, and local noise floor. A single higher or lower limit might result in a false alarm or overlook little adjustments. A brief normal calibration time is still necessary since the new adaptive threshold is less sensitive. Edge-based industrial wireless sensor monitoring and anomaly detection at the IoT edge have recently been studied [34]. The fact that the energy usage metrics were acquired through instruction-level profiling rather than being monitored on an actual neuromorphic device is another flaw. The real enhanced battery life will be ascertained through future hardware measurements. Studies on neuromorphic in-sensor and FPGA-based SNN have demonstrated that translating spike computation to event-driven hardware can result in further energy savings [35].

Conclusion

An ultra-low-power spiking neural network model for vibration anomaly detection in Internet of Things sensor nodes is proposed in this research. The model transforms dense vibration waveforms into sparse spike events, using adaptive leaky integrate-and-fire dynamics to extract temporal characteristics, and evaluates anomalies using energy-aware scoring and spike-driven prototype deviation. In comparison to the compact CNN, LSTM autoencoder, and dense TinyML baselines, the findings demonstrate that the suggested model can retain high anomaly detection performance while lowering predicted inference energy and active computation.

The primary findings indicate that embedded vibration monitoring requires both adaptive firing control and sparse event conversion. An adjustable threshold regulator stops noise-induced spike bursts, while sparse encoding minimizes unnecessary computations in the normal state. To enhance early-stage anomaly identification, a membrane-potential temporal-integration module will gather data from many mild vibrations. When combined, they can simultaneously lower false alarms and boost the detector's recall, which is essential for IoT sensors to operate for longer periods of time.

The system can be expanded in the future to include online threshold adaptation, cross-device continuous learning, hardware-in-the-loop energy monitoring, and deployment on neuromorphic processors. Incorporate more practical field testing to address mounting looseness, temperature fluctuations, sensor drift, and communication duty cycle. These modifications would aid in transforming the spike-based vibration anomaly detection model into an effective design for a useful low-power industrial equipment and structure health monitoring system.

Author Contributions

Sylwia Majchrzak contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision, project administration, and funding acquisition. Teresa Nowicka contributes to validation, analysis, investigation, data collection, draft

preparation. Wioletta Gajewska contributes to methodology, software, analysis, data collection, draft preparation, manuscript editing. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Utomo, D. S., & Hsiung, P. A. (2019). Anomaly detection at the IoT edge using deep learning. 2019 IEEE International Conference on Consumer Electronics - Taiwan, 1-2. https://doi.org/10.1109/ICCE-TW46550.2019.8991929

[2] Anantha, P., Tobianto Daely, P., & Lee, J. M. (2020). Edge computing-based anomaly detection for multi-source monitoring in industrial wireless sensor networks. 2020 International Conference on Information and Communication Technology Convergence, 510-515. https://doi.org/10.1109/ICTC49870.2020.9289271

[3] Ngo, H. Q., Luo, X., & Chaouchi, H. (2020). Contextual-bandit anomaly detection for IoT data in distributed hierarchical edge computing. 2020 IEEE 40th International Conference on Distributed Computing Systems, 1229-1234. https://doi.org/10.1109/ICDCS47774.2020.00191

[4] He, M., & He, D. (2020). A new hybrid deep signal processing approach for bearing fault diagnosis using vibration signals. Neurocomputing, 396, 542-555. https://doi.org/10.1016/j.neucom.2018.12.088

[5] Tavanaei, A., Ghodrati, M., Kheradpisheh, S. R., Masquelier, T., & Maida, A. (2019). Deep learning in spiking neural networks. Neural Networks, 111, 47-63. https://doi.org/10.1016/j.neunet.2018.12.002

[6] Krithivasan, S., Sen, S., Venkataramani, S., & Raghunathan, A. (2019). Dynamic spike bundling for energy-efficient spiking neural networks. 2019 IEEE/ACM International Symposium on Low Power Electronics and Design, 1-6. https://doi.org/10.1109/ISLPED.2019.8824897

[7] Wang, X., Xie, L., Zhang, X., & Zhou, Z. (2019). Deep learning for bearing fault diagnosis under different working loads and non-fault location point. Journal of Low Frequency Noise, Vibration and Active Control, 39(3), 735-748. https://doi.org/10.1177/1461348419889511

[8] Alabsi, A. A., Liao, Y., & Nabulsi, A. A. (2020). Bearing fault diagnosis using deep learning techniques coupled with handcrafted feature extraction: A comparative study. Journal of Vibration and Control, 27(3-4), 404-417. https://doi.org/10.1177/1077546320929141

[9] Gu, J., & Huang, Z. (2020). Fault diagnosis method for bearing of high-speed train based on multitask deep learning. Shock and Vibration, 2020, 8873504. https://doi.org/10.1155/2020/8873504

[10] Sahu, S., & Mukherjee, I. (2020). Machine learning based anomaly detection for IoT network. 2020 4th International Conference on Trends in Electronics and Informatics, 787-794. https://doi.org/10.1109/ICOEI48184.2020.9142921

[11] Yasaei, R., Hernandez, F., & Faruque, M. A. A. (2020). IoT-CAD: Context-aware adaptive anomaly detection in IoT systems. Proceedings of the 39th International Conference on Computer-Aided Design, 1-9. https://doi.org/10.1145/3400302.3415672

[12] R, S., P, S., & V, S. (2021). RANFO: An intelligent anomaly detection in IoT edge devices. 2021 Innovations in Power and Advanced Computing Technologies, 1-6. https://doi.org/10.1109/i-PACT52855.2021.9697006

[13] Roy, K., Jaiswal, A., & Panda, P. (2019). Towards spike-based machine intelligence with neuromorphic computing. Nature, 575, 607-617. https://doi.org/10.1038/s41586-019-1677-2

[14] Neftci, E. O., Mostafa, H., & Zenke, F. (2019). Surrogate gradient learning in spiking neural networks. IEEE Signal Processing Magazine, 36(6), 61-63. https://doi.org/10.1109/MSP.2019.2931595

[15] Skatchkovsky, N., Jang, H., & Simeone, O. (2020). Federated neuromorphic learning of spiking neural networks for low-power edge intelligence. ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing, 8524-8528. https://doi.org/10.1109/ICASSP40776.2020.9053861

[16] Fang, H., Shrestha, A., & Qiu, Q. (2020). Multivariate time series classification using spiking neural networks. 2020 International Joint Conference on Neural Networks, 1-7. https://doi.org/10.1109/IJCNN48605.2020.9206751
[17] Dey, S., Banerjee, A., & George, N. V. (2022). Efficient time series classification using spiking reservoir. 2022 International Joint Conference on Neural Networks, 1-8. https://doi.org/10.1109/IJCNN55064.2022.9892728
[18] Demertzis, K., Iliadis, L., & Bougoudis, I. (2019). Gryphon: A semi-supervised anomaly detection system based on one-class evolving spiking neural network. Neural Computing and Applications, 32, 4303-4314. https://doi.org/10.1007/s00521-019-04363-x
[19] Hu, Y., Liu, Y., & Qiu, Q. (2022). A deep spiking neural network anomaly detection method. Computational Intelligence and Neuroscience, 2022, 6391750. https://doi.org/10.1155/2022/6391750
[20] Stratton, B., Wabnitz, A., & Hamilton, T. J. (2020). A spiking neural network-based auto-encoder for anomaly detection in streaming data. 2020 IEEE Symposium Series on Computational Intelligence, 1997-2004. https://doi.org/10.1109/SSCI47803.2020.9308187
[21] Bäßler, T., Kortus, J., & Gühring, G. (2022). Unsupervised anomaly detection in multivariate time series with online evolving spiking neural networks. Machine Learning, 111, 4391-4414. https://doi.org/10.1007/s10994-022-06129-4
[22] Srinivasan, G., Lee, C., & Sengupta, A. (2020). Training deep spiking neural networks for energy-efficient neuromorphic computing. ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing, 8544-8548. https://doi.org/10.1109/ICASSP40776.2020.9053914
[23] Balaji, A., & Das, A. (2020). Compiling spiking neural networks to mitigate neuromorphic hardware constraints. 2020 11th International Green and Sustainable Computing Workshops, 1-7. https://doi.org/10.1109/IGSC51522.2020.9290830
[24] Vanarse, A., Osseiran, A., & Rassau, A. (2019). Real-time classification of multivariate olfaction data using spiking neural networks. Sensors, 19(8), 1841. https://doi.org/10.3390/s19081841
[25] Ding, L., & Feng, L. (2021). An anomaly detection method based on feature mining for wireless sensor networks. International Journal of Sensor Networks, 36(3), 136-147. https://doi.org/10.1504/IJSNET.2021.117233
[26] Phusakulkajorn, W., Hendriks, M., & Moraal, P. (2022). A multiple spiking neural network architecture based on fuzzy intervals for anomaly detection: A case study of rail defects. 2022 IEEE International Conference on Fuzzy Systems, 1-8. https://doi.org/10.1109/FUZZ-IEEE55066.2022.9882864
[27] Jaoudi, A., Yakopcic, C., & Taha, T. M. (2020). Conversion of an unsupervised anomaly detection system to spiking neural network for car hacking identification. 2020 11th International Green and Sustainable Computing Workshops, 1-7. https://doi.org/10.1109/IGSC51522.2020.9291232
[28] Asghar, M. S., Arslan, T., & Kim, H. (2020). Low power spiking neural network circuit with compact synapse and neuron cells. 2020 International SoC Design Conference, 152-153. https://doi.org/10.1109/ISOCC50952.2020.9333105
[29] Takuya, H., Zhang, Y., & Nakashima, Y. (2021). Training low-latency spiking neural network through knowledge distillation. 2021 IEEE Symposium in Low-Power and High-Speed Chips, 1-3. https://doi.org/10.1109/COOLCHIPS52128.2021.9410323
[30] Gautam, C., & Singh, A. (2019). CLR-based deep convolutional spiking neural network with validation based stopping for time series classification. Applied Intelligence, 50, 1421-1434. https://doi.org/10.1007/s10489-019-01552-y
[31] Hwang, J., Lee, J., & Yun, J. (2021). E-SFD: Explainable sensor fault detection in the ICS anomaly detection system. IEEE Access, 9, 140470-140486. https://doi.org/10.1109/ACCESS.2021.3119573
[32] Utomo, D. S., & Hsiung, P. A. (2020). A multitiered solution for anomaly detection in edge computing for smart meters. Sensors, 20(18), 5159. https://doi.org/10.3390/s20185159
[33] Boone, C., Zhang, J., & Li, P. (2021). Efficient biologically-plausible training of spiking neural networks with precise timing. International Conference on Neuromorphic Systems 2021, 1-8. https://doi.org/10.1145/3477145.3477147
[34] Kong, L., Liu, X., & Zhang, Y. (2020). Sensor anomaly detection in the industrial internet of things based on edge computing. Turkish Journal of Electrical Engineering & Computer Sciences, 28(5), 2716-2728. https://doi.org/10.3906/elk-1906-55

https://doi.org/10.64972/jaat.2023v1.321p30e:404-416

[35] Sankaran, A., Detterer, P., & Kannan, S. (2022). An event-driven recurrent spiking neural network architecture for efficient inference on FPGA. Proceedings of the International Conference on Neuromorphic Systems 2022, 1-8. https://doi.org/10.1145/3546790.3546802