Uncertainty-Guided Counterfactual Sensor Selection and Safety-Gated Recognition for Autonomous Sensing Agents in IoT Systems

Shirong Guo, Hongming Zhou, Muhammad Attique Khan, Jinfeng Xu, Yonis Gulzar, Manar Mousa Altamimi, and Kunfu Wang

***Abstract*—Autonomous IoT systems rely on distributed sensing, but sensor faults and communication loss can leave multichannel evidence incomplete or unreliable. Their agents must therefore classify measurements, assess evidence sufficiency, select a sensor to reacquire, and hand off unsafe decisions. We formulate counterfactual sensing: a degraded window defines the factual state, each candidate restoration defines a controlled alternative, and a pre-request utility model estimates loss reduction. Physics-Guided Invariant-Kinematic Fusion (PGIK-Fusion) constructs class evidence, while calibrated risk and cost-sensitive utility select acceptance, sensor request, or handoff from factual information. On the Logistic Activity Recognition Challenge dataset, PGIK-Fusion achieves 0.768 accuracy and a macro-averaged F1 score (Macro-F1) of 0.696, exceeding extreme gradient boosting (XGBoost)–one-dimensional convolutional neural network (1D-CNN) fusion by 2.49 and 2.00 percentage points, respectively. Across seven predefined degradation conditions on held-out subjects, the policy reduces unsafe acceptance from 33.0% to 13.8%; relative to an action-rate-matched random policy, it raises automation accuracy by 18.5 percentage points and lowers expected cost by 24.7%. In the auxiliary within-event wind-turbine study, PGIK-Fusion reaches 0.975 Macro-F1, improves on XGBoost–1D-CNN by 1.17 percentage points, and reduces unsafe acceptance from 20.1% to 8.0%. The evaluation characterizes the controlled responses of a frozen recognition model to specified sensor restorations under predefined fault families and severities. Within this scope, the results demonstrate that calibrated recognition, targeted evidence acquisition, and conservative deferral can form a unified perception–action loop that reduces unsafe automation under sensor degradation.**

***Index Terms*—Autonomous sensing agents, Internet of Things, counterfactual sensor selection, physics-guided learning, safety-gated recognition, uncertainty quantification.**

I. Introduction

AN intelligent agent is a computational entity situated in an environment, capable of perceiving that environment and selecting actions with a degree of autonomy to pursue its objectives [1]–[7]. Sensor systems instantiate this perception–action abstraction when measurements influence what, where, or when to sense next. Recent uncertainty-aware autonomous sensing in the Internet of Things (IoT), for example, constructs a sensing agent whose policy adapts measurement placement to an application objective [8]. This progression moves sensing beyond passive telemetry, but leaves open how an agent should manage degraded observations in multichannel recognition.

Here, an *autonomous sensing agent in an IoT system* maps networked observations to three predefined actions: accept the recognition result, request one physical sensor, or hand the decision to an upper-level controller. Its *task-bounded* autonomy is constrained by a recognition objective, the current information state, and this finite action set, rather than by an application domain. *Safety-gated recognition* releases a factual label only after acceptance; a request delays release until one reacquisition and one model recomputation, after which the updated label is released directly, while handoff withholds the automated label. The state combines class evidence, sensor quality, predictive uncertainty, and action costs. This abstraction applies to logistics, industrial monitoring, connected transportation, and autonomous vehicles [9], [10]. Fig. 1 depicts the two representative applications evaluated here.

Missing channels, noise, packet loss, and delay can alter both the predicted class and the reliability of its evidence; predictive uncertainty may also deteriorate under dataset shift [11]. Clean-input accuracy therefore neither establishes confidence after degradation nor identifies which sensor could correct an error. Although uncertainty-aware IoT sensing can price additional measurements [12], recognition still requires candidate-specific rules for reacquisition, continued automation, and handoff.

We address this gap by separating factual perception from counterfactual evidence control. Physics-Guided Invariant-Kinematic Fusion (PGIK-Fusion) combines raw temporal encoding with physical descriptors adapted to each sensing system. Controlled missingness, noise, delay, and packet loss produce paired model responses: the degraded input defines the factual state, and restoration of one requestable sensor defines a candidate state. A utility model fitted on validation data predicts each candidate's reduction in classification loss from

Shirong Guo is with the Faculty of Engineering, Monash University, Clayton, Victoria 3800, Australia (e-mail: shirong.guo@monash.edu).

Muhammad Attique Khan is with the Center of AI, Prince Mohammad bin Fahd University, Alkhobar, Saudi Arabia (e-mail: attique.khan@ieee.org)

Jinfeng Xu is with the Department of Electrical and Computer Engineering, The University of Hong Kong, Hong Kong, China (e-mail: jinfeng@connect.hku.hk).

Yonis Gulzar is with the Department of Management Information Systems, College of Business Administration, King Faisal University, Al-Ahsa 31982, Saudi Arabia (e-mail: Ygulzar@kfu.edu.sa)

Manar Mousa Altamimi is with the Department of Information Systems, College of Computer and Information Sciences (CCIS), Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia (e-mail: mmaltamimi@pnu.edu.sa)

Hongming Zhou and Kunfu Wang are with the School of Advanced Manufacturing, Fuzhou University, Quanzhou 362251, China (e-mail: zhouhongming2025@126.com; okonfu23@gmail.com).

Hongming Zhou and Kunfu Wang are the corresponding authors.

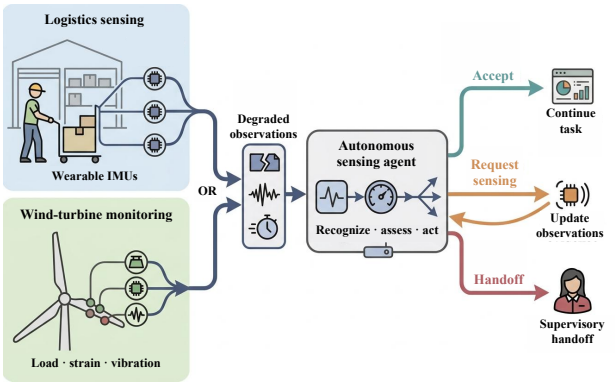


Fig. 1. Representative operating scenarios for evaluating the autonomous sensing agent in logistics activity recognition and wind-turbine monitoring.

factual evidence alone. Calibrated error risk, predicted benefit, and acquisition cost then determine acceptance, one sensor request, or handoff. The estimand is the frozen recognition model's response to a specified restoration.

Evaluation uses the Logistic Activity Recognition Challenge (LARa) version 03 inertial measurement unit (IMU) benchmark and an auxiliary within-event study using the Multimodal Dataset for Wind Turbine Blade Monitoring during Lightning Strikes (MDWTBM-LS) [13]–[15]. Models are retrained with domain-specific descriptors and requestable modalities in each regime.

The main contributions are:

1) A model-level counterfactual formulation that represents factual degradation and candidate sensor restorations as paired model responses with an explicit boundary to natural causal identification.
2) A physics-guided perception backbone that combines coordinate-robust and kinematic descriptors with raw temporal convolution.
3) A cost-sensitive policy that estimates candidate-specific recovery from pre-request evidence and selects acceptance, one sensor request, or handoff.
4) A task-bounded agent evaluation across human- and equipment-centered sensing instances, connecting recognition, uncertainty, controlled intervention, action allocation, unsafe acceptance, and expected cost.

II. Related Work

A. Autonomous Sensing Agents and Multichannel Perception

Agent-based sensing connects observations and internal state to objectives and actions [1], [8]. In air-quality and indoor-noise monitoring, objective-driven sensing increased critical-period detection by 78.5% and 357.3% while reducing energy use by 14.3% and 7.6%, respectively [8]; sensor-specific recovery, recognition risk, and handoff remain unresolved.

Deploying such agents near data sources introduces latency, bandwidth, privacy, and device-heterogeneity constraints [16]–[18]. Adaptive and hierarchical split federated learning address resource allocation during distributed training through model partitioning and device-specific adaptation [19]–[22]. Inference-time sensor recovery instead requires an explicit decision over physical observations.

Wearable activity recognition research has examined sensor placement, feature construction, subject variation, personalization, single-sensor efficiency, and modern sequence modeling [9], [23]–[29]. These methods improve recognition or efficiency but do not prescribe actions for unreliable sensors.

LARa provides synchronized inertial and semantic data, whereas MDWTBM-LS provides blade load, strain, and vibration [13]–[15]. Published LARa results vary across modalities, partitions, and metrics [13], [30]; they do not couple recognition to reacquisition and deferral.

B. Temporal and Physically Structured Representations

Multichannel time-series benchmarks span tree ensembles such as Random Forest, extreme gradient boosting (XGBoost), and the Light Gradient Boosting Machine (LightGBM) [31]–[33], together with raw-sequence encoders based on long short-term memory (LSTM), the gated recurrent unit (GRU), self-attention, and causal dilated convolution [34]–[37]. These families provide broad task coverage but do not specify which physical sensor should be restored after degradation.

Complementary representations encode temporal geometry through Gramian angular fields [38], while TimesNet, ModernTCN, and TimeMIL model periodicity, cross-variable interaction, and time-aware instance evidence [39]–[41]. In other recognition modalities, Interpretable Fixed-Length Representation for Fingerprint Matching via Vision Transformer (IFViT) combines global and local fixed-length representations, whereas Vision-Context Hierarchical Attention Autoregressive With Adaptive Permutation (HAAP) couples visual and contextual evidence. Graph and physics-guided methods additionally introduce sensor relations or domain structure [42], [43]. These approaches enrich factual representations but do not estimate the prospective value of restoring a degraded physical sensor.

C. Uncertainty and Intervention-Aware Sensing

Risk-aware automation requires confidence that tracks empirical correctness. Temperature scaling and shift-aware evaluation provide complementary evidence about calibration quality [11], [44]. Temperature scaling reduced expected calibration error from 12.67% to 0.96% for a 110-layer residual network on the Canadian Institute for Advanced Research 100 dataset [44], while distribution-shift experiments showed that clean-data calibration need not persist after input change [11]. Calibration does not identify which sensor would improve a decision.

Selective prediction controls risk and coverage through rejection or confidence ranking [45]–[47]; SelectiveNet reduced mean coverage violation from 11.98 to 3.63 percentage points [46]. Conformal methods provide finite-sample set coverage under exchangeability [48], [49]. Both assess current evidence, whereas sensor recovery requires prospective utility.

D. Counterfactual Reasoning and Cost-Aware Acquisition

Counterfactual foundations distinguish observed evidence from responses under specified alternatives through potential outcomes and identifiability conditions [50]–[52]. Algorithmic recourse further requires feasible interventions [53]. The present estimand is the fixed recognition model's controlled response to a specified restoration; identifying natural failure effects would require a separate causal design.

Active feature acquisition converts information value into cost-sensitive actions through reinforcement learning, variational representations, or generative surrogates [54]–[56]; IoT soft sensing also links uncertainty to operating cost [12]. The present task restores one degraded physical sensor, supervises candidate-specific loss reduction, and retains handoff when neither current nor restored evidence supports automation.

III. Methodology

The framework formulates task-bounded autonomous sensing as a finite evidence-management loop: PGIK-Fusion constructs class evidence, calibration characterizes reliability, controlled restorations supply candidate-utility targets, and a cost-sensitive gate selects acceptance, one request, or handoff.

Fig. 2 separates validation, autonomous agent operation, and domain-specific sensing while identifying the components fixed before deployment.

At validation, frozen responses to factual and single-sensor-restored windows define targets for calibration, risk, and benefit models. At deployment, pre-request evidence determines factual risk, candidate benefit, and action cost. Accordingly, the deployed selector uses neither restored observations nor labels, and all fitted components remain fixed during test evaluation. A request reacquires or retransmits the selected source and recomputes affected features; acceptance releases the prediction, whereas handoff invokes the fallback. The fast Fourier transform (FFT) icon denotes the spectral component in Section III-B.

A. Task Formulation and Requestable Sensors

The decision problem is defined at the level of one synchronized window so that every action is tied to an explicit information state. Let $x_t^{(0)} \in \mathbb{R}^{L\times D}$ contain L samples from D channels in the observed, possibly degraded state, and let $y_t \in \{1,\dots,K\}$ be its label. Superscript (0) denotes this factual model input. The frozen perception model produces the uncalibrated class vector $\boldsymbol{\pi}_t^{(0)}$ and its associated prediction as

$$\begin{cases} \boldsymbol{\pi}_t^{(0)} = p_f(\cdot \mid x_t^{(0)}) \in [0,1]^K, \\ \hat{y}_t^{(0)} = \arg\max\limits_{1\le k\le K} \pi_{tk}^{(0)}. \end{cases} \tag{1}$$

Here p_f is the frozen fusion classifier and $\pi_{tk}^{(0)}$ is the uncalibrated probability of class k. Let $\mathcal{V} = \{v_j\}_{j=1}^{J}$ contain the J requestable sensors. The feasible action set is $\mathcal{A} = \{\text{accept}, \text{handoff}\} \cup \{\text{request}(v_j) \mid v_j \in \mathcal{V}\}$. Acceptance immediately releases $\hat{y}_t^{(0)}$; a request postpones release until candidate v_j is reacquired and the updated window is evaluated once, after which the resulting label is released directly without a second gate; handoff suppresses automated release and invokes a conservative fallback. This release rule defines safety-gated recognition independently of the perception architecture.

The pre-request state must distinguish predictive ambiguity from deficient sensor availability. It therefore concatenates the calibrated class vector $\mathbf{p}_t^{(0)}$ defined in Section III-D with three complementary uncertainty summaries and the sensor-quality vector:

$$\mathbf{z}_t = \left[(\mathbf{p}_t^{(0)})^\top, c_t, H_t, m_t, \mathbf{q}_t^\top\right]^\top \in \mathbb{R}^{K+3+J}, \tag{2}$$

where the scalar summaries are defined separately as

$$c_t = \max_k p_{tk}^{(0)}, \tag{3}$$

$$H_t = -\sum_{k=1}^{K} p_{tk}^{(0)} \log p_{tk}^{(0)}, \tag{4}$$

$$m_t = p_{t(1)}^{(0)} - p_{t(2)}^{(0)}. \tag{5}$$

These summaries encode distinct aspects of decision reliability. Equations (3)–(5) quantify peak confidence, distributional ambiguity, and separation between the two leading classes, respectively, with the convention $0\log 0 = 0$. Here $p_{t(1)}^{(0)}$ and $p_{t(2)}^{(0)}$ are the largest and second-largest calibrated probabilities, while $\mathbf{q}_t \in [0,1]^J$ records the quality of the J requestable sensors. Retaining the full probability vector distinguishes windows with equal confidence but different class evidence.

The sensing configuration defines the request set. For LARa, $\mathcal{V} = \{v_l, v_t, v_r\}$ denotes the left, trunk, and right IMUs; for MDWTBM-LS, the candidates are load, strain, and vibration. Each candidate is restored independently, and the decision layer contains no sensor graph.

B. Physics-Guided Invariant-Kinematic Fusion

PGIK-Fusion combines temporal representation learning with explicitly structured evidence through two independently trained probability estimators. The temporal branch maps the standardized multichannel window to class probabilities through a one-dimensional convolutional neural network (1D-CNN). The physics-guided branch concatenates statistical–spectral descriptors with sensing-system-specific physical descriptors before XGBoost classification. The branches do not exchange latent features; their probability vectors are combined only by the convex fusion rule defined below. This separation preserves the interpretation of the structured descriptors while allowing the temporal branch to learn local waveform patterns.

The structured branch first establishes a domain-neutral statistical–spectral basis so that raw channels and physically derived sequences can be summarized on the same scale. For a scalar sequence $s = (s_1,\dots,s_L)$, define

$$\begin{cases} \boldsymbol{\psi}(s) = \left[(\boldsymbol{\psi}^{\text{stat}}(s))^\top, (\boldsymbol{\psi}^{\text{spec}}(s))^\top\right]^\top, \\ \boldsymbol{\psi}^{\text{stat}}(s) = \Big[\mu_s, \sigma_s, \min\limits_{1\le\tau\le L} s_\tau, \max\limits_{1\le\tau\le L} s_\tau, \\ \qquad\qquad Q_{0.1}(s), Q_{0.5}(s), Q_{0.9}(s), \text{RMS}(s)\Big]^\top, \\ \boldsymbol{\psi}^{\text{spec}}(s) = \left[\bar{A}_1(s), \bar{A}_2(s), \bar{A}_3(s), \bar{A}_4(s)\right]^\top. \end{cases} \tag{6}$$

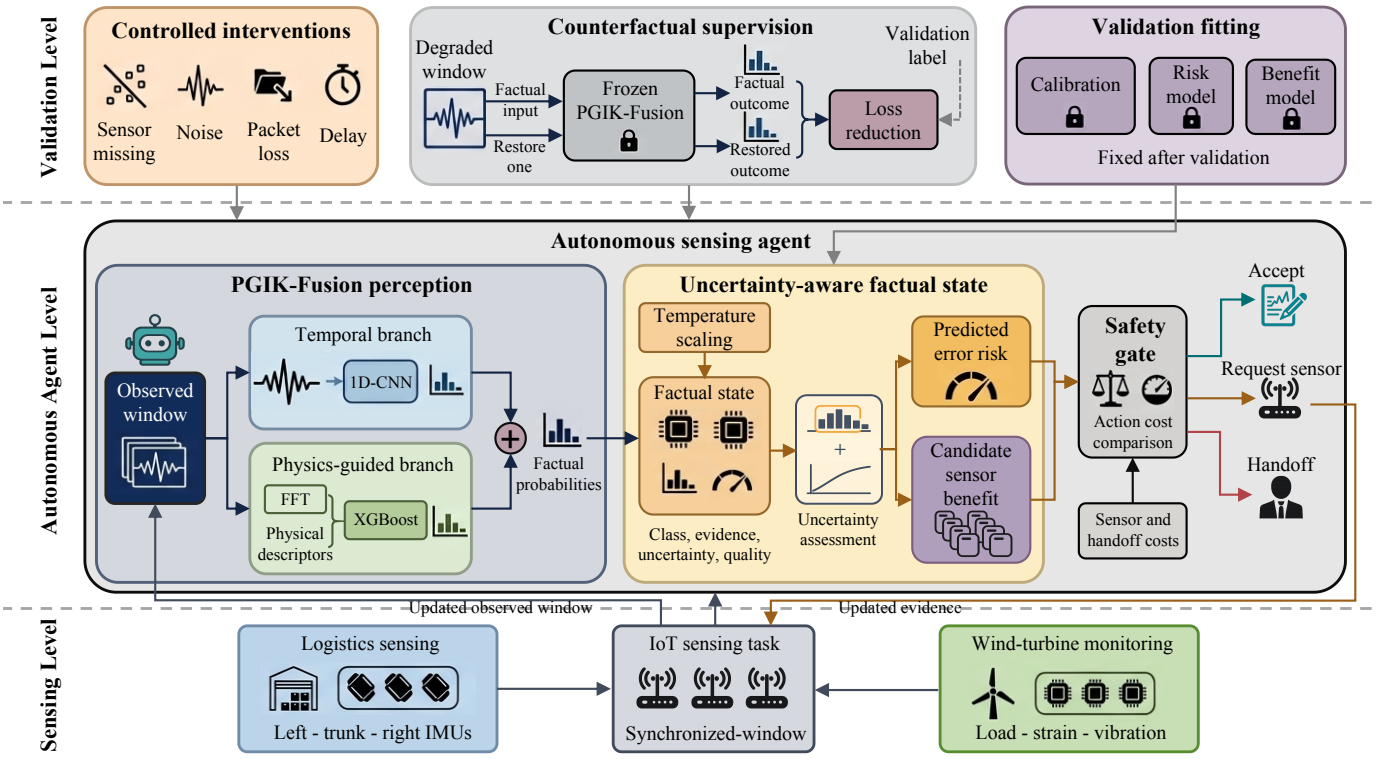


Fig. 2. End-to-end framework for uncertainty-guided counterfactual sensor selection and safety-gated recognition. The upper, middle, and lower layers denote validation, autonomous sensing agent operation, and representative evaluation instantiations, respectively.

Here μ_s, σ_s, $Q_q(s)$, and $\mathrm{RMS}(s)$ denote the sample mean, standard deviation, empirical q-quantile, and root-mean-square value. The term $\bar{A}_b(s)$ is the mean discrete-Fourier-transform magnitude in the bth of four equal nonnegative-frequency index bands, where $b \in \{1,2,3,4\}$. The operator therefore retains location, dispersion, range, distribution shape, signal energy, and coarse frequency allocation without requiring a domain-specific frequency boundary. Applied to the 27 LARa channels, (6) produces 324 base descriptors.

The physical augmentation then introduces invariances that are not guaranteed by channel-wise summaries. This construction follows theory-guided learning by exposing domain-structured evidence without imposing a complete governing equation [57]. Let $\mathbf{u}_\tau^{(s)} \in \mathbb{R}^3$ denote the triaxial signal at sample τ for modality $s \in \{a,\omega,m\}$, corresponding to acceleration, angular velocity, and magnetic field. Orientation-insensitive magnitude and first-order kinematic change are defined as

$$\rho_\tau^{(s)} = \left\|\mathbf{u}_\tau^{(s)}\right\|_2, \qquad s \in \{a,\omega,m\}, \tag{7}$$

$$\delta_\tau^{(s)} = \left\|\mathbf{u}_\tau^{(s)} - \mathbf{u}_{\tau-1}^{(s)}\right\|_2, \qquad s \in \{a,\omega\}. \tag{8}$$

Directional consistency complements the two norm-based operators by retaining relative orientation:

$$\kappa_\tau = \frac{\mathbf{a}_\tau^\top \mathbf{m}_\tau}{\|\mathbf{a}_\tau\|_2 \|\mathbf{m}_\tau\|_2 + \epsilon}. \tag{9}$$

Equation (7) applies for $\tau = 1,\ldots,L$, and (8) for $\tau = 2,\ldots,L$. Here $\mathbf{a}_\tau = \mathbf{u}_\tau^{(a)}$, $\mathbf{m}_\tau = \mathbf{u}_\tau^{(m)}$, and $\epsilon = 10^{-6}$ prevents instability when either vector norm is near zero. The operators $\rho_\tau^{(s)}$, $\delta_\tau^{(s)}$, and κ_τ represent magnitude, first-order change, and directional consistency, respectively. At each IMU, the six derived sequences are summarized by (6), adding 216 descriptors that encode axis-robust magnitude, transition strength, and relative direction.

The invariant terminology follows from the behavior of these operators under a shared change of orthonormal sensor coordinates. For any $\mathbf{R} \in \mathbb{R}^{3\times3}$ satisfying $\mathbf{R}^\top\mathbf{R} = \mathbf{I}$, where $\mathbf{I}$ is the 3×3 identity matrix,

$$\begin{cases} \|\mathbf{R}\mathbf{u}_\tau\|_2 = \|\mathbf{u}_\tau\|_2, \\ \|\mathbf{R}(\mathbf{u}_\tau - \mathbf{u}_{\tau-1})\|_2 = \|\mathbf{u}_\tau - \mathbf{u}_{\tau-1}\|_2, \\ \kappa(\mathbf{R}\mathbf{a}_\tau, \mathbf{R}\mathbf{m}_\tau) = \kappa(\mathbf{a}_\tau, \mathbf{m}_\tau). \end{cases} \tag{10}$$

Here $\kappa(\cdot,\cdot)$ denotes the normalized inner-product operator in (9). Equation (10) establishes invariance to a common orthogonal rotation or reflection of the triaxial coordinate frame. It does not assume invariance to independent sensor displacement or time-varying misalignment, which remain represented through the observed sequences.

The complete LARa feature map makes the relation between these operators and the reported dimensionality explicit. Let $\mathcal{L} = \{l,t,r\}$ index the left, trunk, and right IMUs, and let $\mathcal{S}_{\mathrm{phys}}(x)$ collect the 18 derived sequences: $\rho_\ell^{(s)}$ for $s \in \{a,\omega,m\}$, $\delta_\ell^{(s)}$ for $s \in \{a,\omega\}$, and κ_ℓ, at every $\ell \in \mathcal{L}$. The physics-guided representation is

$$\boldsymbol{\phi}_{\mathrm{PGIK}}(x) = \mathrm{concat}\Big(\{\boldsymbol{\psi}(x_d)\}_{d=1}^{27}, \{\boldsymbol{\psi}(s)\}_{s\in\mathcal{S}_{\mathrm{phys}}(x)}\Big) \in \mathbb{R}^{540}. \tag{11}$$

In (11), x_d is the dth channel and concat denotes concatenation; the raw and derived blocks contain $27 \times 12 = 324$ and $18 \times 12 = 216$ descriptors. Their shared summary operator isolates the effect of physical structure without changing the classifier family.

The equipment-sensing instantiation preserves the representation principle while changing its physical primitives. For MDWTBM-LS, load, strain, and vibration at two positions form six modality–position vectors. Their magnitudes and first differences yield 12 sequences, while three cosine-similarity sequences compare positions within each modality. Summarization expands the 144-dimensional statistical–spectral representation to 324 dimensions. The shared construction encodes within-vector dynamics and cross-position agreement, while the wearable and structural sensor semantics remain domain specific.

Independent branch training preserves structured and temporal evidence without forcing a shared latent space. XGBoost [32] uses 250 trees, depth 6, learning rate 0.08, and row and feature subsampling of 0.85. Class k receives weight $\sqrt{N/n_k}$, where N and n_k are the total and class-k training counts, moderating rare-class weights. Following direct convolutional wearable encoding [58], the temporal branch uses 64, 128, and 192 channels with kernel widths 7, 5, and 5. The first two layers apply batch normalization, a Gaussian error linear unit, and max pooling; the third precedes global average pooling and a linear classifier. Inputs are standardized using training-subject statistics only.

Probability fusion provides an explicit balance between structured and temporal evidence. For input window x and class y, let $\pi_p(y \mid x)$ and $\pi_c(y \mid x)$ denote the uncalibrated physics-guided XGBoost and raw 1D-CNN probabilities, respectively. Their fused output is

$$\pi_f(y \mid x) = \lambda \pi_p(y \mid x) + (1 - \lambda)\pi_c(y \mid x). \tag{12}$$

The weight $\lambda \in [0, 1]$ controls structured evidence, and convexity preserves a valid probability vector. Validation Macro-F1 selects λ from $\{0, 0.05, \ldots, 1\}$, yielding 0.90 on LARa. Features, fitted branches, and λ are then fixed before calibration, interventions, and testing; p_f in (1) denotes this predictor.

C. Controlled Interventions and Sensor Utility

Controlled interventions pair a degraded factual input with one restored candidate. Complete IMU loss zeros one nine-channel group; Gaussian noise uses 0.10 or 0.25 times the training-channel standard deviation; packet loss zeros trunk samples 80–119 (20% of a 200-sample window); and delay shifts the right IMU by 10 samples with zero padding (0.10 s at 100 Hz). These conditions represent source removal, amplitude corruption, contiguous outage, and misalignment. Both branches and all derived features are recomputed after each intervention. The same predefined intervention families and severities are used for policy development and held-out-subject evaluation. The resulting evidence measures controlled robustness and action allocation within these conditions. Generalization to previously unseen fault mechanisms or severity levels requires separate evaluation.

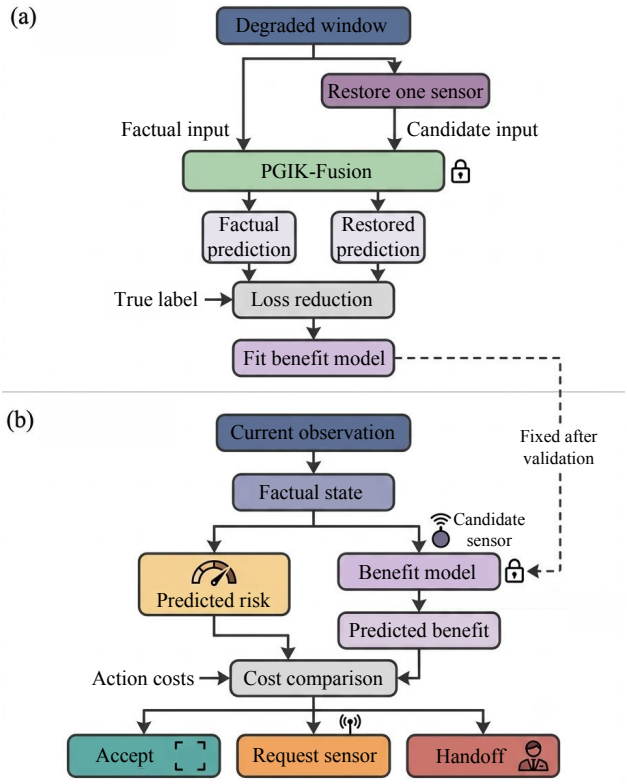


Fig. 3. Counterfactual sensor benefit and finite action selection: (a) validation-stage supervision from controlled restoration; (b) deployment using pre-request factual information.

Sensor-quality metadata describe degradation without revealing the unavailable clean signal. Clean and missing sensors receive 1 and 0; packet loss and delay use the retained-sample fractions 0.80 and 0.95; moderate and strong noise use one minus the normalized noise amplitude, giving 0.90 and 0.75. These indicators populate $\mathbf{q}_t$ as condition-supplied inputs rather than field fault-detector outputs.

Candidate restoration defines a supervised counterfactual target while preserving the deployment information boundary. Let $\mathcal{I}_j$ restore candidate v_j while leaving all other observed channels unchanged. The corresponding candidate state is

$$x_t^{(j)} = \mathcal{I}_j\left(x_t^{(0)}\right), \qquad j \in \{1, \ldots, J\}. \tag{13}$$

During target construction, the frozen classifier gives $\boldsymbol{\pi}_t^{(j)} = p_f(\cdot \mid x_t^{(j)})$ and $\hat{y}_t^{(j)} = \arg\max_k \pi_{tk}^{(j)}$. Equation (13) changes exactly one requestable source, thereby isolating its realized contribution while holding all nonrequested channels fixed. Fig. 3 separates validation supervision, where restorations and labels are available, from deployment selection, where the policy receives only $\mathbf{z}_t$ and candidate identity v_j.

The realized candidate benefit is defined on decision correctness rather than confidence change. For 0–1 classification loss $\ell(y, \hat{y}) = \mathbf{1}[y \neq \hat{y}]$, where $\mathbf{1}[\cdot]$ is the indicator function, the loss reduction under candidate j is

$$\Delta\ell_{tj} = \ell\left(y_t, \hat{y}_t^{(0)}\right) - \ell\left(y_t, \hat{y}_t^{(j)}\right). \tag{14}$$

The signed target equals 1 for error correction, 0 for unchanged correctness, and −1 for a newly introduced error. It is used only for target construction and retrospective analysis; neither y_t nor $x_t^{(j)}$ is available before a request.

The policy decomposes action value into factual error risk and candidate-specific recovery benefit because high risk alone does not identify which sensor should be requested. A standardized logistic model fitted on the S11–S12 validation subjects estimates

$$\hat{r}_t = \sigma\big(\beta_0 + \boldsymbol{\beta}^\top \widetilde{\mathbf{z}}_t\big) \approx \Pr\Big(\hat{y}_t^{(0)} \neq y_t \mid \mathbf{z}_t\Big), \tag{15}$$

where $\sigma(u) = (1+e^{-u})^{-1}$ is the logistic link, $\widetilde{\mathbf{z}}_t$ is standardized using S11–S12 statistics, β_0 is the intercept, and $\boldsymbol{\beta}$ contains the fitted coefficients. The logistic model uses an ℓ_2 penalty with inverse regularization strength 1 and a maximum of 1000 solver iterations; regularization stabilizes the estimate under correlated probability and uncertainty inputs. A standardized ridge model separately estimates the expected candidate benefit as

$$\hat{b}_{tj} = h_{\boldsymbol{\theta}}(\widetilde{\mathbf{z}}_t, v_j) \approx \mathbb{E}\big[\Delta\ell_{tj} \mid \mathbf{z}_t, v_j\big]. \tag{16}$$

Here $h_{\boldsymbol{\theta}}$ is a ridge predictor using standardized state $\widetilde{\mathbf{z}}_t$, one-hot candidate identity, and their interactions, allowing one factual window to assign different sensor utilities. Its penalty is fixed at 10 before test evaluation to shrink the interaction-expanded coefficient vector under the sparse nonzero target. Predictions are then clipped to the target support $[-1, 1]$ so that extrapolation cannot create an invalid loss reduction before cost evaluation.

Resource cost converts a predicted recovery into a deployable acquisition decision. For candidate v_j, the normalized cost is

$$C_j = w_E \widetilde{E}_j + w_D \widetilde{D}_j + w_B \widetilde{B}_j. \tag{17}$$

Here $\widetilde{E}_j$, $\widetilde{D}_j$, and $\widetilde{B}_j$ denote normalized energy, latency, and bandwidth burden; the nonnegative weights satisfy $w_E + w_D + w_B = 1$. Because the three LARa IMUs have equal channel widths and the dataset provides no platform telemetry, the primary experiment assigns $C_j = 0.10$ to every request and evaluates sensitivity over 0.05–0.25. For a deployed platform, $\widetilde{E}_j$ can be estimated from sensor wake-up, sampling, and transmission telemetry; $\widetilde{D}_j$ from reacquisition, queueing, and transmission delay relative to the decision deadline; and $\widetilde{B}_j$ from payload and retransmission volume relative to the available link budget. Normalizing these measurements by device-specific budgets and selecting w_E, w_D, and w_B from application priorities yields sensor- and context-dependent request costs without changing the decision rule. The MDWTBM-LS experiment retains the same normalized request and handoff costs for load, strain, and vibration; its validation-based modality selection and recovery gate are specified in Supplementary Section S1.3.

D. Uncertainty and Finite Safety Actions

Probability calibration is required because the risk estimator and action loss use class probabilities as decision evidence rather than only as a class ranking. Temperature scaling preserves the predicted class while adjusting probability sharpness [44]. For fused output $\boldsymbol{\pi}_t^{(0)}$, the calibrated probability of class k is

$$p_{tk}^{(0)}(T) = \frac{\left(\pi_{tk}^{(0)}\right)^{1/T}}{\sum_{h=1}^{K}\left(\pi_{th}^{(0)}\right)^{1/T}}, \qquad T > 0. \tag{18}$$

Here h indexes the classes. The transform is equivalent to applying softmax to $\log \boldsymbol{\pi}_t^{(0)}/T$ and covers sharpening ($T < 1$) and softening ($T > 1$). The calibration temperature is selected on the validation partition by

$$T^\star = \arg\min_{T\in\mathcal{T}} -\frac{1}{|\mathcal{D}_{\text{cal}}|}\sum_{i\in\mathcal{D}_{\text{cal}}} \log p_{iy_i}^{(0)}(T), \tag{19}$$

where $\mathcal{D}_{\text{cal}}$ denotes S11–S12, i indexes its windows, y_i is the associated label, and $\mathcal{T}$ contains 101 equally spaced values from 0.5 to 3.0. LARa selects $T^\star = 1.45$, and $\mathbf{p}_t^{(0)}$ henceforth denotes the calibrated vector used in (2)–(5). Calibration is assessed by 15-bin expected calibration error (ECE), negative log-likelihood (NLL), multiclass Brier score, and ten confidence-ranked coverage levels from 0.10 to 1.00. Fifteen bins balance probability resolution against validation occupancy, whereas the coverage grid provides a compact trajectory from selective to complete automation.

Set-valued uncertainty complements scalar calibration by retaining all labels compatible with a finite-sample threshold. For a scoring rule fixed independently of the calibration observations, the split-conformal rank construction targets marginal coverage under exchangeability without requiring a parametric error model [48], [49]. For n_{cal} calibration windows, define nonconformity scores $a_i = 1 - p_{iy_i}^{(0)}(T^\star)$ and their order statistics $a_{(1)} \le \cdots \le a_{(n_{\text{cal}})}$. The finite-sample threshold is

$$\hat{q} = a_{(r)}, \qquad r = \min\{n_{\text{cal}}, \lceil (n_{\text{cal}}+1)(1-\alpha)\rceil\}, \tag{20}$$

and the prediction set for factual window t is

$$\Gamma_t = \Big\{k \in \{1,\dots,K\} : 1 - p_{tk}^{(0)}(T^\star) \le \hat{q}\Big\}. \tag{21}$$

Here α is target miscoverage, r the corrected order-statistic index, and $\hat{q}$ the fixed threshold. We set $\alpha = 0.10$ to target 90% marginal coverage, yielding $\hat{q} = 0.933$ on LARa. The conformal set complements scalar calibration but does not enter the action rule. Since S11–S12 support both temperature selection and quantile estimation, the observed test coverage is reported empirically; a disjoint conformal-calibration partition would be required for the standard finite-sample guarantee.

The finite safety gate completes the evidence-management loop by placing acceptance, acquisition, and deferral on one cost scale. Safety gating here denotes cost-sensitive suppression of high-risk automated outputs through sensor request or handoff:

$$\mathcal{L}_t(a) = \begin{cases} \hat{r}_t, & a = \text{accept}, \\ \text{clip}(\hat{r}_t - \hat{b}_{tj}, 0, 1) + C_j, & a = \text{request}(v_j), \\ C_h, & a = \text{handoff}. \end{cases} \tag{22}$$

The branches in (22) share a unit because an emitted error has unit cost. Here $\text{clip}(u, 0, 1) = \min\{1, \max\{0, u\}\}$ bounds residual risk; acceptance incurs factual risk, a request adds

C_j after predicted recovery, and handoff pays C_h. An optimal action satisfies

$$a_t^\star \in \arg\min_{a\in\mathcal{A}} \mathcal{L}_t(a). \tag{23}$$

The action boundary becomes explicit by separating recoverable risk from acquisition burden. Define

$$\begin{cases} d_{tj} = \hat{r}_t - \mathrm{clip}(\hat{r}_t - \hat{b}_{tj}, 0, 1), \\ U_{tj} = d_{tj} - C_j, \\ \mathcal{L}_t(\mathrm{request}(v_j)) = \hat{r}_t - U_{tj}. \end{cases} \tag{24}$$

Here d_{tj} is the signed predicted change in factual error risk after restoring v_j; it is negative when restoration is predicted to increase risk. The quantity U_{tj} is the corresponding net utility after acquisition cost. Let $j_t^\dagger \in \arg\max_{1\le j\le J} U_{tj}$ denote any maximizing candidate. A strict preference for requesting this candidate over both acceptance and handoff is equivalent to

$$U_{tj_t^\dagger} > \max\{0, \hat{r}_t - C_h\}. \tag{25}$$

The first term in (25) requires predicted recovery to offset acquisition cost; the second requires the post-request loss to fall below the deferral cost. Thus, sensor identity is selected by candidate-specific net utility, while the release decision remains jointly controlled by factual risk and C_h. When action losses are equal, (23) represents the corresponding set of minimizing actions.

The cost ordering determines the remaining policy behavior without a hard confidence threshold. The handoff cost is $C_h = 0.30$, placing conservative deferral above a single request cost but below the unit cost assigned to an emitted error. With $C_j = 0.10 < C_h < 1$, low-risk observations can be accepted, recoverable high-risk observations can trigger acquisition, and high residual risk can trigger handoff. Let g and h denote the fitted risk and benefit models, respectively. Algorithm 1 summarizes the resulting deployment procedure.

Algorithm 1 Counterfactual Sensor Selection and Safety Gating

Require: Factual window $x_t^{(0)}$, quality $\mathbf{q}_t$, frozen PGIK-Fusion model p_f, risk model g, benefit model h, costs $\{C_j\}_{j=1}^J$ and C_h

Compute $\mathbf{p}_t^{(0)}$, $\hat{y}_t^{(0)}$, and pre-request state $\mathbf{z}_t$
Estimate $\hat{r}_t \leftarrow g(\mathbf{z}_t)$
for $j = 1$ to J **do**
  Estimate $\hat{b}_{tj} \leftarrow h(\mathbf{z}_t, v_j)$ without observing $x_t^{(j)}$
  Evaluate $\mathcal{L}_t(\mathrm{request}(v_j))$ using (22)
end for
Evaluate acceptance and handoff losses; select a minimizer $a_t^\star$ from (23)
if $a_t^\star = request(v_j)$ **then**
  Reacquire v_j, reconstruct all affected features, and rerun PGIK-Fusion
  Release the updated prediction directly without a second gate
end if
return Final label or handoff status, together with the action record

The evaluated policy is single-step: it permits at most one sensor request, and the post-request prediction is released without another accept/request/handoff decision.

Comparator policies isolate which part of the decision rule provides value. Confidence-only and quality-only gates use thresholds selected by validation expected cost, whereas the action-rate-matched random policy preserves aggregate accept, request, and handoff counts while randomizing their allocation and requested sensor. The Oracle policy accesses realized restoration outcomes only to define a retrospective lower bound. Unsafe acceptance is the error fraction among accepted windows. Risk–coverage follows selective classification [45], with calibrated confidence used as a post-hoc ranking score [47].

E. Datasets and Evaluation Protocols

The primary protocol evaluates person-independent logistics activity recognition. The MotionMiners IMU subset of LARa version 03 contains three IMUs, 27 channels, and 100-Hz sampling [14]. We use 200-sample (2-s) windows with 100-sample stride and majority labels. Removing 'None' and windows below 0.5 purity yields 21,952 windows in seven classes; *Handling centre* and *Synchronisation* constitute approximately 57% and 1.6%, respectively.

The MotionMiners IMU modality used in this study is available for S07–S14 but not for S01–S06 [14]. Modality availability therefore defines the eight-subject cohort. To prevent windows from the same person appearing across development and test partitions, S07–S10 (11,424 windows) are used for model fitting, S11–S12 (3,816 windows) for early stopping, fusion, calibration, and policy fitting, and S13–S14 (6,712 windows) for final evaluation. The resulting person-held-out estimates apply to the MotionMiners subset rather than to the full LARa subject population. Subject-level window and class counts are reported in Supplementary Table S3.

The auxiliary protocol evaluates the framework within the recorded event under equipment-sensing semantics. MDWTBM-LS contains synchronized load, strain, and vibration measurements around one recorded wind-turbine blade lightning event [15]. Each model uses a 30-min history to forecast the maximum Normal, Transition, or Anomaly state in the following 60 min. Models are retrained with seeds 2026, 2027, and 2028, and all seed-specific selection quantities are fitted on the corresponding validation partition. In this configuration, validation mean true-class NLL reduction selects the candidate modality for each degradation; validation-trained risk and recovery gates then allocate accept, request, and handoff actions. Label intervals, window construction, splitting, physical descriptors, gate configurations, and seven modality-degradation conditions are provided in Supplementary Section S1.

F. Baselines and Implementation Details

The comparison set includes feature-based trees [31]–[33]; recurrent, convolutional, and self-attention encoders [34]–[37]; Gramian angular field (GAF), recurrence plot (RP), and

short-time Fourier transform (STFT) models [38]; Sensor-GNN, TimesNet, ModernTCN, and TimeMIL [39]–[42]; and probability fusion. Transformed-domain and graph models are evaluated without XGBoost fusion. Supplementary Section S4.1 gives the configurations.

All LARa neural models, including the temporal branch of PGIK-Fusion, use class-weighted cross-entropy, AdamW with learning rate 2×10^{-3} and weight decay 10^{-4}, batch size 128, at most 50 epochs, and early stopping after 10 epochs without improvement in validation Macro-F1. The shared optimization budget controls training differences across neural baselines, while early stopping retains the checkpoint with the highest validation Macro-F1 for fusion, intervention, and efficiency analyses. Random-number generators in Python, NumPy, and PyTorch are initialized with the designated experimental seed. The PGIK-Fusion multi-seed experiment uses seeds 2026, 2027, and 2028.

Accuracy, balanced accuracy, and Macro-F1 are computed on held-out test windows. Structural scale counts trainable neural parameters and boosted-tree nodes. Experiments were run on a Mac mini with a 10-core Apple M4 processor and 16 GB unified memory under macOS 26.2. Neural-network training used the PyTorch Metal Performance Shaders (MPS) backend where supported; no Compute Unified Device Architecture (CUDA) device was used. All reported latency measurements placed models and tensors on the central processing unit (CPU) and did not invoke MPS. Single-window latency is measured with one CPU thread and batch size one after at least 20 warm-up calls, using 300 timed calls. PGIK-Fusion timing includes vectorized physical feature construction, XGBoost prediction, TorchScript 1D-CNN inference, and probability fusion. Its reported value is the median of seven trials, each preceded by 30 warm-up calls.

IV. Results and Discussion

A. Perception Comparison and Backbone Selection

Table I compares all models under the S07–S10/S11–S12/S13–S14 protocol. Selection uses validation Macro-F1, and every reported value is computed on held-out S13–S14.

PGIK-Fusion achieved the highest accuracy (0.768) and Macro-F1 (0.696), exceeding XGBoost–1D-CNN by 2.49 and 2.00 percentage points, respectively. Both models had a rounded balanced accuracy of 0.664, which motivates the class-level analysis below. The strongest modern standalone model, TimesNet, reached 0.596 Macro-F1. Overall, fusion models and boosted trees formed the leading group, and performance did not increase monotonically with architectural recency. Historical LARa results in Section II-A use different modalities, partitions, and metrics and are not numerically comparable with the present subject-held-out IMU protocol.

Deployment suitability depends jointly on predictive quality and execution cost. Fig. 4 therefore relates test Macro-F1 to single-window CPU latency for all 16 models; color denotes structural scale and the dashed line joins operating points for which neither metric can be improved without worsening the other.

PGIK-Fusion occupied the highest-performance endpoint of the nondominated frontier. Relative to XGBoost–1D-CNN, its 2.00-percentage-point Macro-F1 gain increased median latency from 0.278 to 0.520 ms per window; both remained below 1 ms. XGBoost was faster at 0.089 ms but 2.20 percentage points lower in Macro-F1. The comparison therefore supports PGIK-Fusion when the additional 0.242 ms is acceptable, while identifying XGBoost as the lower-latency operating point. Structural scale and latency were not monotonically related, as illustrated by the compact but slower ModernTCN.

Table II compares compact temporal backbones using validation-selected XGBoost weights of 0.50, 0.45, 0.90, and 0.65 for GRU, ModernTCN, Transformer, and 1D-CNN. Their combined structural scales are 212,491, 166,411, 272,875, and 297,995 counted units, respectively. TimesNet remains standalone because its fusion would contain 434,123 counted units. Latency is the seven-trial median for joint branch inference under the batch-one, single-thread CPU protocol in Section III-E.

XGBoost–1D-CNN provided the highest Macro-F1 (0.676) and balanced accuracy (0.664). XGBoost–Transformer was

TABLE I
TEST PERFORMANCE, STRUCTURAL SCALE, AND SINGLE-WINDOW CPU LATENCY UNDER THE FIXED SUBJECT PROTOCOL. ACC. AND BAL. ACC. DENOTE ACCURACY AND BALANCED ACCURACY; SCALE COUNTS TRAINABLE PARAMETERS, BOOSTED-TREE NODES, OR BOTH FOR FUSION MODELS.

Model	Acc.	Bal. Acc.	Macro-F1	Scale	Time (ms)
PGIK-Fusion	**0.768**	**0.664**	**0.696**	296,591	0.520
XGBoost+1D-CNN	0.743	0.664	0.676	297,995	0.278
XGBoost	0.753	0.630	0.674	119,940	0.089
LightGBM	0.746	0.618	0.660	102,196	0.460
STFT-CNN	0.739	0.615	0.630	169,543	0.195
TimesNet	0.627	0.654	0.596	314,183	1.632
TimeMIL	0.588	0.658	0.585	240,728	0.256
ModernTCN	0.659	0.599	0.568	46,471	2.017
Transformer	0.647	0.632	0.593	152,935	0.672
GRU	0.587	0.632	0.574	92,551	2.697
1D-CNN	0.546	0.646	0.546	178,055	0.198
Sensor-GNN	0.603	0.560	0.537	177,975	0.306
RP-CNN	0.431	0.476	0.425	100,679	0.511
tCNN-softmax	0.621	0.621	0.567	71,303	0.174
tCNN-semantic	0.580	0.645	0.566	72,538	0.179
ResNeXt-1D	0.583	0.633	0.564	646,027	0.875

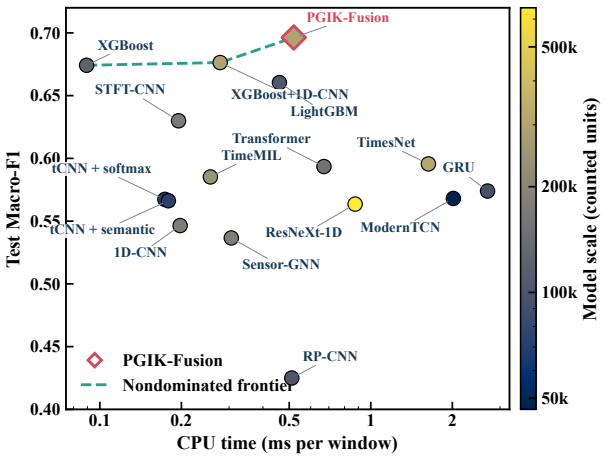


Fig. 4. Test Macro-F1 versus single-window CPU latency on S13–S14; marker color denotes structural scale, and the dashed line joins nondominated points.

TABLE II
TEST PERFORMANCE AND SINGLE-WINDOW CPU LATENCY OF COMPACT FUSION BACKBONES ON S13–S14.

Fusion backbone	Accuracy	Bal. Acc.	Macro-F1	Time (ms)
XGBoost+GRU	0.701	0.654	0.638	2.923
XGBoost+ModernTCN	0.744	0.647	0.660	2.232
XGBoost+Transformer	**0.751**	0.630	0.672	0.777
XGBoost+1D-CNN	0.743	**0.664**	**0.676**	**0.278**

TABLE III
TEST PERFORMANCE OF THE PHYSICAL-FEATURE ABLATION ON S13–S14.

Feature configuration	Accuracy	Bal. Acc.	Macro-F1
Raw fusion	0.743	0.664	0.676
+ Vector magnitude	0.755	0.678	0.690
+ Kinematic difference	0.743	**0.698**	0.685
+ Directional consistency	0.755	0.684	0.694
Full PGIK-Fusion	**0.768**	0.664	**0.696**

0.004 lower in Macro-F1 and required 2.8 times the latency, while XGBoost–ModernTCN reduced scale by 44% but lost 1.6 percentage points of Macro-F1 and required 8.0 times the latency. The 1D-CNN therefore provided the strongest performance–latency balance among the tested fusion backbones.

B. Physical Evidence, Class Effects, and Efficiency

Table III adds vector magnitude, kinematic difference, and directional consistency separately before evaluating their joint combination. This design distinguishes descriptor-specific physical information from a generic increase in feature dimension.

Vector magnitude, kinematic difference, and directional consistency increased Macro-F1 by 1.34, 0.82, and 1.73 percentage points over raw fusion, respectively. Directional consistency recovered approximately 86% of the full 2.00-percentage-point gain, while the joint combination added only 0.27 percentage points beyond that variant. Conversely, the kinematic-only variant attained the highest balanced accuracy (0.698), 3.43 percentage points above the full model. The descriptor effects were therefore neither additive nor uniform across metrics, motivating the classwise analysis below.

To examine how physical augmentation reorganized the structured feature space, Fig. 5 compares deterministic t-distributed stochastic neighbor embedding (t-SNE) projections of the same sampled test windows before and after adding the invariant–kinematic descriptors. Up to 300 windows per class are sampled with seed 2026; each representation is standardized, reduced to 30 principal components, and embedded with perplexity 30 over 1200 iterations.

The physics-guided projection formed locally tighter *Cart* and *Synchronisation* regions, whereas the handling activities remained substantially overlapped. This geometry is consistent with selective class separation rather than uniform disentanglement. Because neighborhood projections do not quantify classification error, the interpretation is treated as descriptive

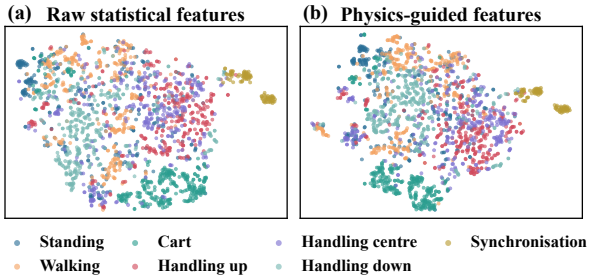


Fig. 5. Deterministic t-SNE projections of sampled test windows using (a) raw statistical features and (b) physics-guided structured features.

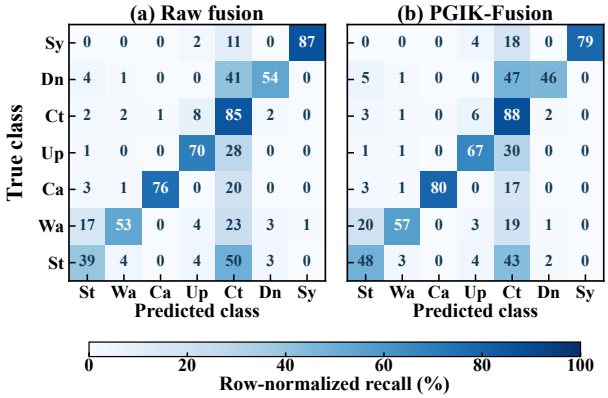


Fig. 6. Row-normalized test confusion matrices for (a) raw XGBoost–1D-CNN fusion and (b) full PGIK-Fusion. St, Wa, Ca, Up, Ct, Dn, and Sy denote Standing, Walking, Cart, Handling up, Handling centre, Handling down, and Synchronisation, respectively.

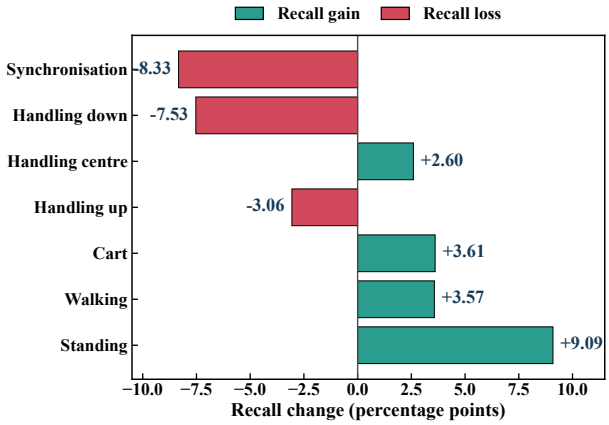


Fig. 7. Per-class test recall change from raw fusion to full PGIK-Fusion.

and is evaluated against the confusion and recall evidence below.

To determine whether the projected changes translated into classwise decisions, Fig. 6 reports the complete row-normalized confusion matrices for raw fusion and PGIK-Fusion.

The matrices show that physical augmentation redistributed rather than uniformly reduced errors. Fig. 7 summarizes the diagonal differences as signed recall changes.

PGIK-Fusion increased recall for *Standing*, *Walking*, *Cart*,

and *Handling centre* by 9.09, 3.57, 3.61, and 2.60 percentage points, respectively. Recall decreased by 3.06 percentage points for *Handling up*, 7.53 percentage points for *Handling down*, and 8.33 percentage points for *Synchronisation*. The 17.42-percentage-point span between the largest gain and loss shows substantial class-level redistribution beneath the 2.00-percentage-point aggregate gain. This redistribution motivates classwise reliability analysis alongside aggregate Macro-F1. The reported safety improvement is aggregate: the gate routed 83.9% of *Standing* and 84.85% of *Handling down* class-condition windows to request or handoff. Direct acceptance remained unreliable for these classes, with erroneous predictions in all accepted *Standing* instances and 77.84% of accepted *Handling down* instances. Relative to releasing every factual prediction, erroneous accepted outputs across all class-condition windows fell from 76.3% to 16.1% for *Standing* and from 65.23% to 11.79% for *Handling down*. For *Synchronisation*, routing 57.82% of instances to request or handoff reduced this exposure from 43.54% to 4.25%. Protection for *Standing* and *Handling down* therefore arose mainly from routing rather than reliable direct acceptance, and the classwise reliability imbalance persisted; Supplementary Table S6 reports the complete action and error profile.

Across seeds 2026, 2027, and 2028, PGIK-Fusion yielded a test Macro-F1 of 0.694 ± 0.006, and all three runs remained above the raw XGBoost–1D-CNN fusion. Per-seed metrics are provided in Supplementary Fig. S1. Table I shows that PGIK-Fusion contains 118,536 boosted-tree nodes and 178,055 trainable 1D-CNN parameters. Under the stated CPU protocol, vectorized feature construction, XGBoost prediction, TorchScript 1D-CNN inference, and probability fusion together required a median of 0.520 ms per window. The post-perception gate comprises 70 fitted coefficients across the logistic risk model and ridge utility model. Under the same batch-one, single-thread protocol, including construction and evaluation of all three candidate utility inputs and the final cost comparison, its median latency was 0.179 ms per window, or 34.4% of the 0.520-ms perception latency; the measured perception-plus-policy path was therefore 0.698 ms. Fig. 4 places the perception measurement in the performance–latency distribution. Structural scale and latency characterize complementary aspects of deployment: the former combines heterogeneous neural-parameter and tree-node counts and is not a proxy for memory consumption or floating-point operations. Supplementary Table S5 reports the timing distribution.

The directional-consistency variant recovered most of the aggregate gain, but class effects remained nonuniform. The submillisecond perception-plus-policy path supports a separate safety layer for unreliable classes and sensor states.

C. Calibrated Evidence and Selective Risk

All calibration and selection parameters were fitted on S11–S12. Fig. 8 ranks test windows by calibrated confidence and reports selective risk as coverage increases.

Temperature scaling selected $T = 1.45$. Selective risk increased from 0.028 at 10% coverage to 0.083 at 50% and 0.232 at full coverage, corresponding to accepted accuracies of

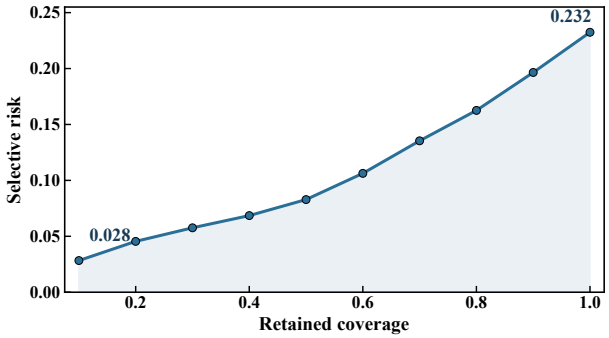


Fig. 8. Selective test risk as retained coverage increases for temperature-scaled PGIK-Fusion probabilities.

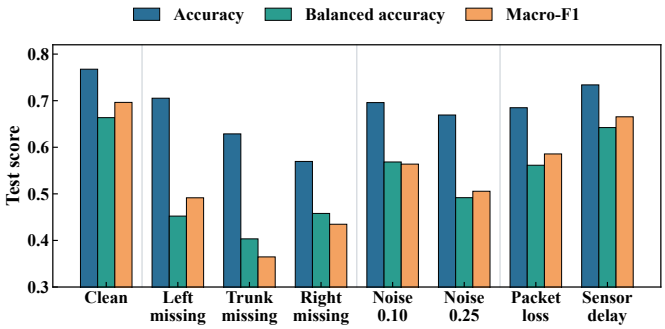


Fig. 9. Test recognition performance under controlled missingness, noise, packet-loss, and delay interventions.

97.2%, 91.7%, and 76.8%. Increasing coverage from 10% to 50% therefore tripled risk while adding 40 percentage points of automation; full coverage added a further 14.9 percentage points of risk. At $\alpha = 0.10$, split conformal prediction achieved 0.929 empirical coverage, 2.88 percentage points above the nominal target, with a mean set size of 1.80. The risk–coverage curve and conformal sets characterize confidence-ranked and set-valued uncertainty, respectively, for the evaluated split.

D. Counterfactual Sensor Recovery and Safety Actions

All structured and 1D-CNN inputs were recomputed after each controlled intervention. Fig. 9 compares complete IMU loss, distributed noise, packet loss, and temporal delay.

Missing left, trunk, and right IMUs reduced Macro-F1 from 0.696 to 0.492, 0.365, and 0.435, corresponding to losses of 20.48, 33.17, and 26.16 percentage points. Trunk loss removed 47.6% of the clean Macro-F1 and was the most damaging intervention. Moderate noise, strong noise, packet loss, and right-IMU delay retained 0.564, 0.506, 0.586, and 0.665, respectively. The 3.09-percentage-point delay loss was less than one sixth of the smallest complete-sensor loss, confirming that recognition severity differed markedly by fault type.

Recognition loss alone does not establish whether an additional sensor should be requested. Fig. 10 therefore evaluates candidate-specific net utility before the final action comparison.

Restoring the affected sensor produced positive mean net utility for trunk and right-IMU loss (0.021 and 0.062), whereas

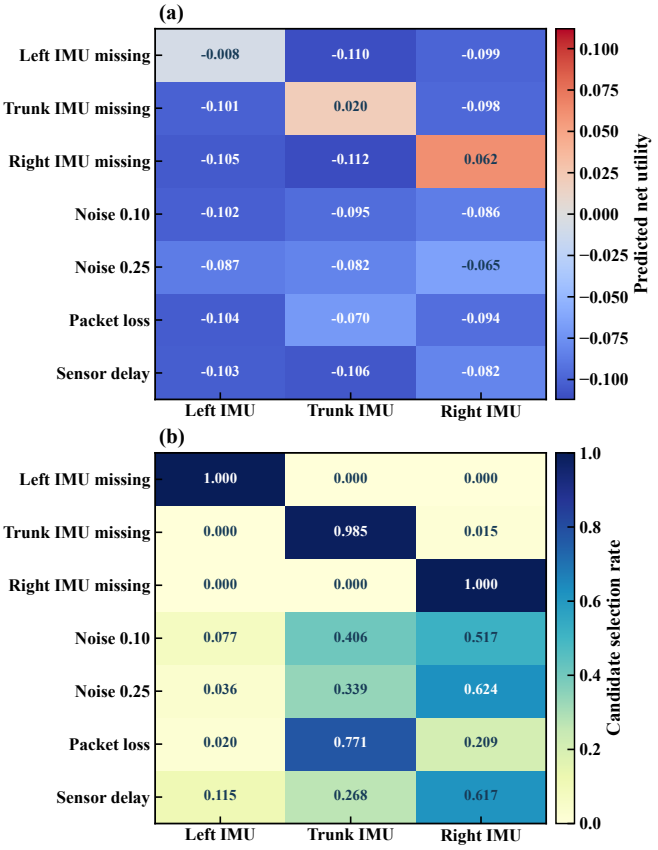


Fig. 10. Window-specific candidate sensor evaluation under seven LARa degradations: (a) mean predicted benefit minus request cost; (b) fraction assigned to each candidate before the final action comparison.

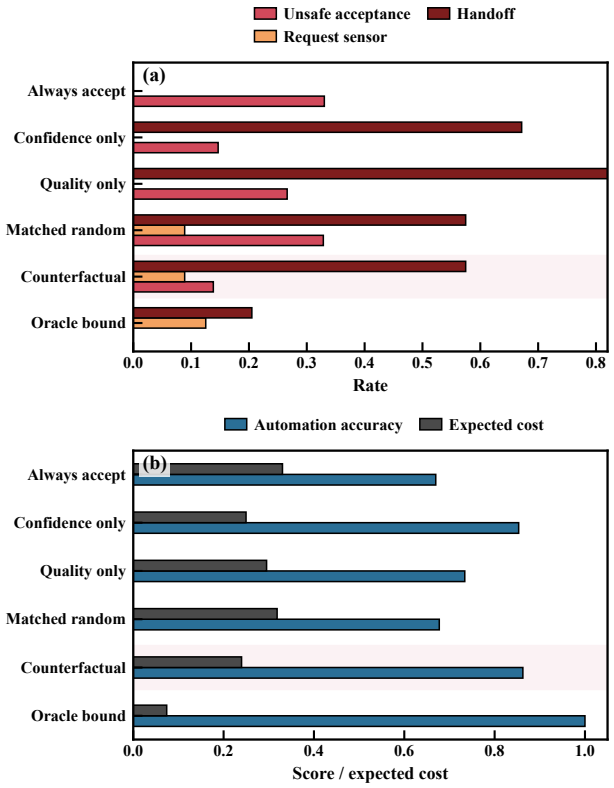


Fig. 11. Safety-policy comparison across seven degraded test conditions: (a) unsafe acceptance and action rates; (b) automated accuracy and expected cost.

restoring unaffected sensors produced negative values. The selector chose the matching sensor for 98.5% of trunk-loss windows and all left- and right-loss windows. Under distributed noise, packet loss, and delay, all mean utilities were negative after request cost. The highest-ranked candidate was therefore not necessarily preferable to acceptance or handoff. Degradation severity and recoverability were distinct: the most corrupted signal was not always cost-effective to reacquire.

Held-out restoration outcomes evaluate the utility ranking against an imbalanced signed target. Among 140,952 window–candidate pairs, 6,348 (4.50%) were beneficial, 132,453 (93.97%) were neutral, and 2,151 (1.53%) were harmful. The three candidates for each of 46,984 degraded window–condition instances account for these pairs. Of the 46,984 instances, 5,888 (12.53%) contained at least one corrective restoration; the selected candidate corrected 5,018, yielding an opportunity-capture rate of 85.22%. Predicted benefit had a mean absolute error of 0.087 and a Pearson correlation of 0.219 with realized 0–1 loss reduction. The predominance of neutral targets contributes to the low mean absolute error, while the modest correlation indicates limited global linear agreement. The opportunity-capture rate quantifies candidate ranking within the corrective-opportunity subset. The estimand remains the controlled frozen-model response defined in Section III-C. Mean regret relative to the retrospectively best candidate was 0.048 loss units per window. Supplementary Table S4 reports the target distribution. Together, Figs. 9 and 10 separate fault severity from recoverability and connect both quantities to the finite action space.

E. Counterfactual Policy Comparison and Agent Decisions

To test whether candidate-specific utility improves the allocation of a fixed action budget, Fig. 11 compares the counterfactual policy with always accepting, confidence-only and quality-only gating, action-rate-matched random allocation, and the Oracle lower bound. The random policy preserves the counterfactual policy's aggregate accept, request, and handoff rates while randomizing their allocation and requested sensor. Differences from this comparator therefore reflect where the policy acts rather than how often it acts. Supplementary Table S7 provides the numerical values.

Across seven degraded conditions, always accepting produced 33.03% unsafe acceptance. The counterfactual policy reduced this rate to 13.84%, an absolute decrease of 19.19 percentage points and a relative decrease of 58.1%. It accepted 33.64%, requested one sensor for 8.89%, and handed off 57.47% of windows. At the same action rates, the action-rate-matched random policy produced 32.86% unsafe acceptance, 0.678 automation accuracy, and 0.319 expected cost. Candidate-specific allocation reduced unsafe acceptance by 19.02 percentage points, increased automation accuracy by 18.51 percentage points, and lowered cost by 24.7% to 0.240. The gain therefore arose from action allocation rather than a higher handoff rate.

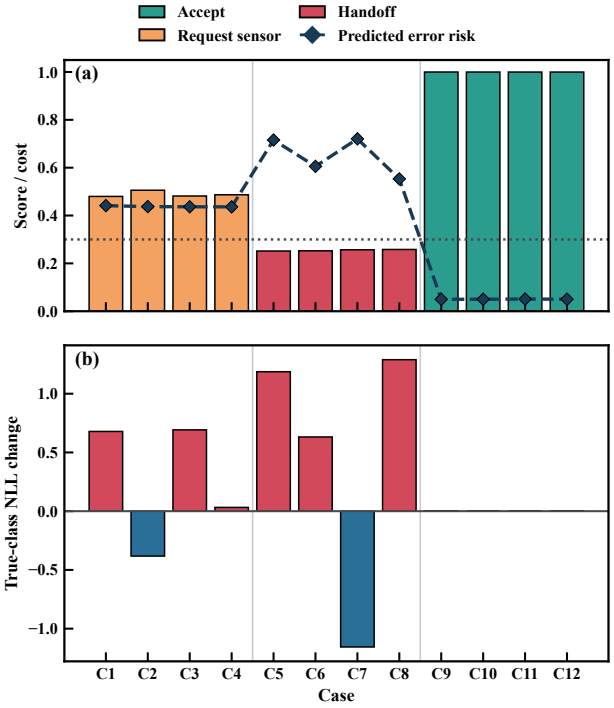


Fig. 12. Illustrative one-step decisions under IoT sensor degradation: (a) confidence, predicted error risk, and selected action; (b) retrospective change in true-class NLL.

Validation cost minimization selected no request actions for either scalar comparator. The confidence-only policy obtained 14.68% unsafe acceptance, 0.853 automation accuracy, and 0.250 expected cost. Quality-only gating accepted 14.3% of windows and obtained 26.6% unsafe acceptance, 0.734 automation accuracy, and 0.295 expected cost, showing that degradation metadata alone did not resolve class uncertainty. Relative to confidence-only gating, the counterfactual policy reduced unsafe acceptance by 0.83 percentage points, increased automation accuracy by 0.93 percentage points, and reduced cost by 4.0%, while retaining targeted reacquisition. Across request costs of 0.05–0.25, the request rate fell from 22.37% to zero at costs of at least 0.20; expected cost ranged from 0.236 to 0.242.

The counterfactual policy remained 0.166 above the Oracle cost of 0.074. Observed sources of this gap include candidate-selection regret and errors in the estimated risk and benefit. Its performance therefore remained between action-rate-matched random allocation and the retrospective Oracle lower bound.

Aggregate policy metrics do not show how an individual sensing decision is formed. Fig. 12 therefore traces representative decisions from observed risk to action. Retrospective true-class NLL is used only to evaluate the selected action.

Cases C1–C4 are right-IMU-loss windows for which the predicted risk reduction exceeded request cost. Cases C5–C8 were handed off because their predicted residual risk remained high. Cases C9–C12 combined low predicted error risk with no cost-effective request and were accepted. The cases illustrate the finite, single-step decisions defined in Section III-A without supplying post-request outcomes to the policy. The policy comparison and cases jointly support the finite sensing-agent formulation by linking pre-request evidence to actions over a defined set of independently restorable sensors.

TABLE IV
THREE-SEED TEST PERFORMANCE FOR 60-MIN-AHEAD WIND-TURBINE BLADE STATE FORECASTING ON MDWTBM-LS. ENTRIES ARE MEAN ± STANDARD DEVIATION OVER SEEDS 2026, 2027, AND 2028; ALL PARTITIONS ARE DERIVED FROM THE SAME RECORDED LIGHTNING EVENT.

Model	Acc.	Bal. Acc.	Macro-F1
XGBoost	0.935 ± 0.050	0.904 ± 0.063	0.924 ± 0.052
XGBoost+1D-CNN	0.971 ± 0.023	0.946 ± 0.026	0.963 ± 0.020
PGIK-Fusion	**0.978 ± 0.022**	**0.964 ± 0.031**	**0.975 ± 0.022**

F. Auxiliary Study in Wind-Turbine Blade Monitoring

To examine cross-domain re-instantiation within the recorded event, the models were retrained under the MDWTBM-LS block-partition protocol with seeds 2026, 2027, and 2028. The auxiliary task forecasts the maximum Normal, Transition, or Anomaly state in the next 60 min from wind-turbine blade load, strain, and vibration histories. Table IV compares the retrained perception models.

PGIK-Fusion reached 0.975 ± 0.022 Macro-F1, compared with 0.963 ± 0.020 for XGBoost–1D-CNN and 0.924 ± 0.052 for XGBoost. The paired changes over raw fusion were +1.77, +1.73, and +0.00 percentage points, yielding a mean increase of 1.17 percentage points. Two seeds improved and one tied; the gain was therefore positive on average but not uniform across seeds.

Calibration effects differed by metric: temperature scaling reduced mean ECE and Brier score but increased test NLL. At $\alpha = 0.10$, conformal coverage was 0.909 ± 0.110. Supplementary Fig. S3(a) reports the complete calibration and risk–coverage evidence. Removing load, strain, and vibration reduced Macro-F1 by 32.25, 38.50, and 37.70 percentage points, respectively. High clean performance therefore coexisted with substantial modality dependence; complete condition-level results are provided in Supplementary Fig. S2.

To compare decision policies under these degraded conditions, Fig. 13 presents the mean unsafe acceptance, request and handoff rates, automation accuracy, and expected cost across the three runs. Acceptance is the complement of request and handoff and is therefore omitted as a redundant series. The action-rate-matched random policy uses the same aggregate action rates as the counterfactual policy, and the Oracle is retained as a retrospective lower bound. Supplementary Table S8 lists the corresponding means and standard deviations, including the complementary acceptance rate.

The counterfactual policy reduced unsafe acceptance from 0.201±0.030 under always accepting to 0.080±0.057, a 60.2% relative reduction. Against the action-rate-matched random policy, it reduced unsafe acceptance by 11.28 percentage points, increased automation accuracy by 9.57 percentage points, and lowered expected cost by 44.5%, from 0.194 ± 0.024 to 0.108 ± 0.033. Relative to confidence-only gating, unsafe acceptance fell by 3.76 percentage points and cost by

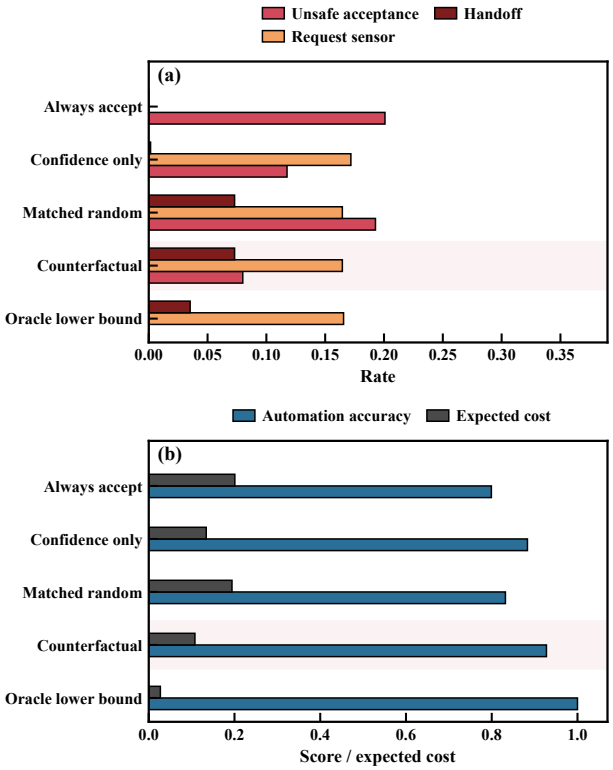


Fig. 13. Finite-action policy comparison across seven degraded MDWTBM-LS sensing conditions: (a) unsafe acceptance, sensor request, and handoff rates; (b) automation accuracy and expected cost.

19.7%, while request rates were similar (0.165 versus 0.172). The counterfactual policy also handed off 7.3% of windows.

Condition-specific results in Supplementary Fig. S2 and uncertainty and case analyses in Supplementary Fig. S3 complement the aggregate outcomes in Table IV and Fig. 13. Because all training, validation, and test blocks originate from one recorded lightning event, variation across seeds measures partition and model-fitting sensitivity within that event. Four-window grouping reduces direct adjacency across partitions, but the overlapping histories and forecast horizons leave residual temporal dependence between nearby blocks. The Macro-F1 standard deviations of 0.020–0.052 in Table IV, together with the wider condition-level intervals in Supplementary Fig. S2, quantify this within-event sensitivity. The 1.17-percentage-point mean perception gain and policy improvements show that the finite control abstraction can be configured with distinct sensor semantics within the recorded event. Cross-event generalization requires evaluation on independent lightning events.

V. Conclusion

This study defines a task-bounded counterfactual evidence manager for autonomous IoT sensing. PGIK-Fusion combines physical descriptors with temporal evidence, while pre-request risk and utility connect factual and candidate-restoration states to acceptance, one sensor request, or handoff. The formulation distinguishes recognition quality, recoverability, and evidence sufficiency.

The LARa experiments provide the primary evidence. PGIK-Fusion reached 0.696 Macro-F1, 2.00 percentage points above raw XGBoost–1D-CNN fusion. Pre-request utility captured 85.22% of windows with a corrective restoration. At matched action rates, the counterfactual policy improved automation accuracy by 18.51 percentage points and reduced expected cost by 24.7% relative to the action-rate-matched random policy. In the auxiliary within-event study, PGIK-Fusion attained 0.975 Macro-F1 on MDWTBM-LS, and the counterfactual policy reduced unsafe acceptance from 20.1% under always accepting to 8.0%. These auxiliary estimates quantify partition and model-fitting sensitivity within one recorded event; cross-event generalization requires independent events.

The testbeds provide complementary wearable and structural sensing evidence. Deployment requires sensor-quality metadata, reacquisition or retransmission, and action costs derived from energy, latency, bandwidth, battery state, and supervisory response. The empirical evidence covers subject-held-out logistics data, one recorded wind-turbine lightning event, predefined simulated fault families and severities, and single-step actions. Unseen fault families and cross-event behavior require separate validation. Long-horizon planning, continual adaptation, and autonomous physical control define extensions beyond the present single-step formulation. Within the evaluated setting, physically structured recognition, targeted evidence acquisition, and conservative deferral form an experimentally grounded perception–action loop across distinct IoT sensing configurations.

Data Availability

The implementation, experiment configurations, pretrained weights, and reference results are available at https://github.com/shannongsr/IoTJ_counterfactual_sensor_selection.

References

[1] M. Wooldridge and N. R. Jennings, "Intelligent agents: theory and practice," *The Knowledge Engineering Review*, vol. 10, no. 2, pp. 115–152, 1995.

[2] T. Duan, Z. Zhang, Z. Lin, S. Guo, X. Guan, G. Wu, Z. Fang, H. Meng, X. Du, J.-Z. Zhou *et al.*, "LLM-Driven Stationarity-Aware Expert Demonstrations for Multi-Agent Reinforcement Learning in Mobile Systems," *arXiv preprint arXiv:2511.19368*, 2025.

[3] D. Huang, M. Du, J. M. Zhang, Z. Lin, M. Luo, Q. Zhang, and S.-K. Ng, "Nexus: Execution-grounded multi-agent test oracle synthesis," *arXiv preprint arXiv:2510.26423*, 2025.

[4] J. Xu, Z. Chen, S. Yang, J. Li, P. Wu, Z. Liu, Z. Lin, J. Tang, J. Yang, W. Wang *et al.*, "Agents as knowledge integrator and utilizer in multimodal recommendation," *arXiv preprint arXiv:2608.29410*, 2026.

[5] Y. Zhang, M. Hu, Z. Lin, X. Fan, F. Xie, Z. Fang, J. Yang, W. Zhu, Z. Chen, C. Lv *et al.*, "Hera: Learning long-horizon coordination for device-cloud collaborative llm agents," *arXiv preprint arXiv:2605.24598*, 2026.

[6] H. Zhang, X. Mao, G. Dong, Z. Li, X. Su, K. Chen, J. Yang, and Z. Lin, "Memmark: State-evolution attribution watermarking for agent long-term memory systems," *arXiv preprint arXiv:2605.25002*, 2026.

[7] T. Duan, Z. Zhang, S. Guo, D. Huang, Y. Zhao, Z. Lin, Z. Fang, D. Luan, H. Cui, and Y. Cui, "Leed: A highly efficient and scalable llm-empowered expert demonstrations framework for multi-agent reinforcement learning," *arXiv preprint arXiv:2509.14680*, 2025.

[8] A. Murad, F. A. Kraemer, K. Bach, and G. Taylor, "Uncertainty-aware autonomous sensing with deep reinforcement learning," *Future Generation Computer Systems*, vol. 156, pp. 242–253, 2024.

[9] V. Bianchi, M. Bassoli, G. Lombardo, P. Fornacciari, M. Mordonini, and I. De Munari, "IoT Wearable Sensor and Deep Learning: An Integrated Approach for Personalized Human Activity Recognition in a Smart Home Environment," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8553–8562, 2019.

[10] T. Yang and C. Lv, "A Secure Sensor Fusion Framework for Connected and Automated Vehicles Under Sensor Attacks," *IEEE Internet of Things Journal*, vol. 9, no. 22, pp. 22 357–22 365, 2022.

[11] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. V. Dillon, B. Lakshminarayanan, and J. Snoek, "Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift," in *Advances in Neural Information Processing Systems*, vol. 32. Curran Associates, Inc., 2019, pp. 13 969–13 980.

[12] H.-S. Lee, S.-H. Park, and S.-Y. Kim, "Cost-Efficient Sensing With Self-Improving Soft Sensor Based on Predictive Uncertainty Estimation in IoT Environments," *IEEE Internet of Things Journal*, vol. 13, no. 2, pp. 3055–3068, 2026.

[13] F. Niemann, C. Reining, F. Moya Rueda, N. R. Nair, J. A. Steffens, G. A. Fink, and M. ten Hompel, "LARa: Creating a Dataset for Human Activity Recognition in Logistics Using Semantic Attributes," *Sensors*, vol. 20, no. 15, p. 4083, 2020.

[14] F. Niemann, C. Reining, F. Moya Rueda, N. R. Nair, P. Oberdiek, H. Bas, R. Spiekermann, E. Altermann, J. A. Steffens, G. A. Fink, and M. ten Hompel, "Logistic Activity Recognition Challenge (LARa Version 03) – A Motion Capture and Inertial Measurement Dataset," 2023.

[15] T. Li, C. Li, Y. Qin, L. Tan, B. Jiang, and H. Deng, "Multimodal dataset for wind turbine blade monitoring during lightning strikes," *Scientific Data*, vol. 12, no. 1, p. 1524, 2025.

[16] Z. Lin, G. Qu, Q. Chen, X. Chen, Z. Chen, and K. Huang, "Pushing Large Language Models to the 6G Edge: Vision, Challenges, and Opportunities," *IEEE Communication Magazine*, 2025.

[17] Z. Sun, X. Guan, Z. Lin, Y. Qing, H. Song, Z. Fang, Z. Chen, F. Liu, H. Cui, W. Ni *et al.*, "Rrto: A high-performance transparent offloading system for model inference in mobile edge computing," *arXiv preprint arXiv:2507.21739*, 2025.

[18] J. Zhan, H. Shen, Z. Lin, and T. He, "Prism: Privacy-aware routing for adaptive cloud-edge llm inference via semantic sketch collaboration," *arXiv preprint arXiv:2511.22788*, 2025.

[19] W. Wei, Z. Lin, T. Li, X. Li, and X. Chen, "Pipelining split learning in multi-hop edge networks," *arXiv preprint arXiv:2505.04368*, 2025.

[20] Z. Lin, Y. Zhang, Z. Chen, Z. Fang, X. Chen, P. Vepakomma, W. Ni, J. Luo, and Y. Gao, "HSplitLoRA: A Heterogeneous Split Parameter-Efficient Fine-Tuning Framework for Large Language Models," *arXiv preprint arXiv:2505.02795*, 2025.

[21] S. Lyu, Z. Lin, G. Qu, X. Chen, X. Huang, and P. Li, "Optimal resource allocation for u-shaped parallel split learning," in *2023 IEEE Globecom Workshops (GC Wkshps)*, 2023, pp. 197–202.

[22] Z. Lin, Z. Chen, X. Chen, W. Ni, and Y. Gao, "HASFL: Heterogeneity-aware Split Federated Learning over Edge Computing Systems," *IEEE Trans. Mobile Comput.*, 2026.

[23] N. Y. Hammerla, S. Halloran, and T. Plötz, "Deep, Convolutional, and Recurrent Models for Human Activity Recognition Using Wearables," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*. AAAI Press, 2016, pp. 1533–1540.

[24] J. Lu, X. Zheng, M. Sheng, J. Jin, and S. Yu, "Efficient Human Activity Recognition Using a Single Wearable Sensor," *IEEE Internet of Things Journal*, vol. 7, no. 11, pp. 11 137–11 146, 2020.

[25] S. Li, T. Zhu, M. Nie, H. Ning, Z. Liu, and L. Chen, "P2LHAP: Wearable-Sensor-Based Human Activity Recognition, Segmentation, and Forecast Through Patch-to-Label Seq2Seq Transformer," *IEEE Internet of Things Journal*, vol. 12, no. 6, pp. 6818–6830, 2025.

[26] S. Li, T. Zhu, F. Duan, L. Chen, H. Ning, C. Nugent, and Y. Wan, "HARMamba: Efficient and Lightweight Wearable Sensor Human Activity Recognition Based on Bidirectional Mamba," *IEEE Internet of Things Journal*, vol. 12, no. 3, pp. 2373–2384, 2025.

[27] H. Yuan, Z. Chen, Z. Lin, J. Peng, Y. Zhong, X. Hu, S. Xue, W. Li, and Y. Gao, "Constructing 4D Radio Map in LEO Satellite Networks with Limited Samples," *IEEE INFOCOM*, 2025.

[28] J. Peng, Z. Chen, Z. Lin, H. Yuan, Z. Fang, L. Bao, Z. Song, Y. Li, J. Ren, and Y. Gao, "SUMS: Sniffing Unknown Multiband Signals under Low Sampling Rates," *IEEE Trans. Mobile Comput.*, 2024.

[29] H. Yuan, Z. Chen, Z. Lin, J. Peng, Z. Fang, Y. Zhong, Z. Song, and Y. Gao, "SatSense: Multi-Satellite Collaborative Framework for Spectrum Sensing," *IEEE Trans. Cogn. Commun. Netw.*, 2025.

[30] S. Mekruksavanich, D. Tancharoen, and A. Jitpattanakul, "Human Activity Recognition in Logistics Using Wearable Sensors and Deep Residual Network," in *TENCON 2023 – 2023 IEEE Region 10 Conference (TENCON)*. IEEE, 2023, pp. 194–198.

[31] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[32] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 785–794.

[33] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017, pp. 3146–3154.

[34] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[35] K. Cho, B. van Merriënboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2014, pp. 1724–1734.

[36] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is All you Need," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017, pp. 5998–6008.

[37] S. Bai, J. Z. Kolter, and V. Koltun, "An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[38] Z. Wang and T. Oates, "Imaging Time-Series to Improve Classification and Imputation," in *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*. AAAI Press, 2015, pp. 3939–3945.

[39] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis," in *International Conference on Learning Representations*, 2023.

[40] D. Luo and X. Wang, "ModernTCN: A Modern Pure Convolution Structure for General Time Series Analysis," in *International Conference on Learning Representations*, 2024.

[41] X. Chen, P. Qiu, W. Zhu, H. Li, H. Wang, A. Sotiras, Y. Wang, and A. Razi, "TimeMIL: Advancing Multivariate Time Series Classification via a Time-aware Multiple Instance Learning," in *Proceedings of the 41st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 235. PMLR, 2024, pp. 7190–7206.

[42] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," in *International Conference on Learning Representations*, 2017.

[43] J. Willard, X. Jia, S. Xu, M. Steinbach, and V. Kumar, "Integrating Scientific Knowledge with Machine Learning for Engineering and Environmental Systems," *ACM Computing Surveys*, vol. 55, no. 4, pp. 1–37, 2022.

[44] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On Calibration of Modern Neural Networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 70. PMLR, 2017, pp. 1321–1330.

[45] R. El-Yaniv and Y. Wiener, "On the Foundations of Noise-free Selective Classification," *Journal of Machine Learning Research*, vol. 11, pp. 1605–1641, 2010.

[46] Y. Geifman and R. El-Yaniv, "SelectiveNet: A Deep Neural Network with an Integrated Reject Option," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 97. PMLR, 2019, pp. 2151–2159.

[47] L. F. Cattelan and D. Silva, "How to Fix a Broken Confidence Estimator: Evaluating Post-hoc Methods for Selective Classification with Deep Neural Networks," in *Proceedings of the Fortieth Conference on Uncertainty in Artificial Intelligence*, ser. Proceedings of Machine Learning Research, vol. 244. PMLR, 2024, pp. 547–584.

[48] G. Shafer and V. Vovk, "A Tutorial on Conformal Prediction," *Journal of Machine Learning Research*, vol. 9, pp. 371–421, 2008.

[49] A. N. Angelopoulos and S. Bates, "Conformal Prediction: A Gentle Introduction," *Foundations and Trends in Machine Learning*, vol. 16, no. 4, pp. 494–591, 2023.

[50] D. B. Rubin, "Estimating causal effects of treatments in randomized and nonrandomized studies," *Journal of Educational Psychology*, vol. 66, no. 5, pp. 688–701, 1974.

[51] J. Pearl, "The Seven Tools of Causal Inference, with Reflections on Machine Learning," *Communications of the ACM*, vol. 62, no. 3, pp. 54–60, 2019.

[52] B. Schölkopf, F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio, "Toward Causal Representation Learning," *Proceedings of the IEEE*, vol. 109, no. 5, pp. 612–634, 2021.

[53] A.-H. Karimi, B. Schölkopf, and I. Valera, "Algorithmic Recourse: From Counterfactual Explanations to Interventions," in *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. ACM, 2021, pp. 353–362.

[54] J. Janisch, T. Pevný, and V. Lisý, "Classification with Costly Features Using Deep Reinforcement Learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, pp. 3959–3966, 2019.

[55] C. Ma, S. Tschiatschek, K. Palla, J. M. Hernández-Lobato, S. Nowozin, and C. Zhang, "EDDI: Efficient Dynamic Discovery of High-Value Information with Partial VAE," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 97. PMLR, 2019, pp. 4234–4243.

[56] Y. Li and J. Oliva, "Active Feature Acquisition with Generative Surrogate Models," in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 139. PMLR, 2021, pp. 6450–6459.

[57] A. Karpatne, G. Atluri, J. H. Faghmous, M. Steinbach, A. Banerjee, A. Ganguly, S. Shekhar, N. Samatova, and V. Kumar, "Theory-Guided Data Science: A New Paradigm for Scientific Discovery from Data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 10, pp. 2318–2331, 2017.

[58] F. J. Ordóñez and D. Roggen, "Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.