

HSDC-Net: Privacy-Preserving and Trust-Aware Historical-State Decay Compensation for Secure Distributed Industrial Edge Monitoring

Jianxia Liu¹, Zicheng Zheng^{2*}

¹Guangxi Radio and Television General Studio Infrastructure Office, Nanning, 530022, Guangxi, China

²Department of Civil and Environmental Engineering, Faculty of Science and Technology, University of Macau, Macau, 519000, China

Abstract

INTRODUCTION: Distributed industrial edge monitoring requires long-sequence models to remain reliable under non-stationary operating conditions while preventing sensitive local data from being exposed through cross-node historical-state sharing.

OBJECTIVES: This work addresses secure reuse of historical representations under feature drift, privacy leakage, and poisoned-memory propagation.

METHODS: We propose HSDC-Net, which locally encodes heterogeneous process streams, releases clipped and differentially private latent anchors, estimates cross-node inconsistency and recursive source trust, aligns reliable historical states through low-rank context transport, and restricts correction to a learned nuisance-drift subspace.

RESULTS: Experiments on four industrial datasets show that HSDC-Net consistently improves predictive performance and drift robustness with moderate online latency. It also suppresses malicious-anchor acceptance while maintaining a finite privacy budget with only limited utility degradation relative to the non-private variant.

CONCLUSION: HSDC-Net provides a unified privacy-aware and poisoning-resilient mechanism for robust distributed industrial monitoring without centralizing raw manufacturing streams across heterogeneous monitoring and maintenance tasks.

Received on 10 August 2026; accepted on 30 September 2026; published on 15 October 2026

Keywords: Distributed industrial edge monitoring; historical-state decay; differential privacy; source trust; feature drift; poisoned-anchor suppression

Copyright © 2026 Jianxia Liu *et al.*, licensed to EAI. This is an open access article distributed under the terms of the [CC BY-NC-SA 4.0](#), which permits copying, redistributing, remixing, transformation, and building upon the material in any medium so long as the original work is properly cited.

doi:10.4108/eetsis.14417

1. Introduction

Distributed industrial monitoring is moving from centralized acquisition toward cloud-edge-device architectures in which sensors, cameras, controllers, and manufacturing machines continuously generate heterogeneous long-sequence data at multiple edge nodes. This deployment reduces raw-data transmission but changes the learning problem from conventional sequence prediction into secure historical-state reuse under distributed non-stationarity. Sensor

aging, material-batch variation, process-mode switching, acquisition noise, and equipment degradation gradually alter local feature distributions, while different edge nodes may operate under different contexts. Historical states are therefore useful only when their relevance to the current condition can be identified, because uniformly retaining old states causes stale-context interference whereas aggressively forgetting them discards reusable long-term process knowledge. Meanwhile, transmitting raw observations or unrestricted hidden states may expose proprietary process information, and an authenticated but compromised node may inject manipulated representations

*Corresponding author. Email: cc2022020224@163.com

into shared memory. Several algorithmic challenges follow. The difficulty is amplified because privacy perturbation changes the same latent space used for drift and inconsistency assessment. Rather than assuming that privacy noise can be explicitly separated from other latent deviations, it should be treated as an additional bounded uncertainty source whose influence must be controlled during anchor release and historical reuse. Benign drift, genuine task-relevant degradation, and malicious state manipulation can all appear as latent deviations, so they require complementary drift-, uncertainty-, and trust-related assessments rather than interpretation as uniquely identifiable causes from a single score. Historical states must be evaluated by age, context, uncertainty, and source credibility; privacy perturbation must preserve enough latent geometry for matching; and the whole mechanism must remain lightweight enough for real-time edge deployment.

Existing sequence models and online adaptation techniques address only parts of this problem. LSTM introduced gated recurrent memory for long-term dependency modeling, but its recurrent state does not explicitly judge whether historical information remains valid after distribution shift [20]. Informer improves long-sequence efficiency through sparse attention, yet efficient access to remote dependencies does not prevent obsolete or unsafe states from affecting current representations [21]. PatchTST strengthens long-horizon modeling through patch-based channel-independent representations, but its temporal statistics are designed for forecasting rather than trust-aware distributed state reuse [22]. ODEStream supports buffer-free streaming adaptation through an ODE-based adaptor, although parameter adaptation does not explicitly separate nuisance drift from task-relevant degradation or protect shared historical representations [23]. Black-box adaptation can respond to online distribution shifts without internal retraining, but the correction remains performance-driven rather than security-aware [1]. Retraining policies determine when model updates are justified, yet model-level retraining can be expensive for continuously distributed edge streams [2]. Bayesian weight enhancement improves test-time adaptation by controlling parameter uncertainty, but uncertainty-aware parameter adjustment does not establish whether a remote historical source is trustworthy [3]. Time-series attribution identifies when and where temporal shifts occur, while attribution alone cannot construct a protected historical reference or suppress poisoned memory [4]. Orthogonal-rotation continual learning improves streaming adaptation of time-series foundation models, but it mainly mitigates forgetting rather than privacy leakage or malicious cross-node state injection [5]. Drift-resilient residual boosting improves forecasting under changing distributions, yet it does not jointly model temporal

reliability, source trust, differential privacy, and task-preserving latent compensation [6]. The key research gap is therefore not merely how to adapt after drift, but how to securely reuse distributed historical information when that history is heterogeneous, privacy-sensitive, time-varying, and potentially contaminated. Existing approaches generally optimize sequence representation, online adaptation, or continual updating as separate objectives, while secure historical-state reuse additionally requires privacy-aware release, source-sensitive reliability assessment, temporal decay, context alignment, and selective compensation without assuming explicit causal identification of every latent deviation.

To address this gap, we propose HSDC-Net, a privacy-preserving and trust-aware historical-state decay compensation framework for secure distributed industrial edge monitoring across heterogeneous industrial monitoring tasks rather than a single manufacturing process. Each edge node uses a lightweight temporal encoder to transform multimodal observations into compact states while keeping raw measurements and images local. Stable states are summarized into bounded latent anchors, clipped to control sensitivity, and released through Gaussian perturbation with a per-release (ϵ_p, δ_p) differential-privacy guarantee; repeated releases consume additional privacy budget and are therefore controlled through bounded communication intervals and release counts for cumulative privacy accounting. At the gateway, HSDC-Net compares candidate anchors with context-compatible cross-node states, computes inconsistency scores, and recursively updates source trust, so anomalous or low-trust anchors are rejected or downweighted before reference construction. Accepted anchors are further weighted by temporal age, uncertainty, context discrepancy, source trust, and inconsistency, then transported toward the current operating context by a low-rank adapter. Rather than directly replacing the current state, HSDC-Net learns a low-dimensional nuisance-drift subspace and applies gated compensation only to the projected residual, preserving task-relevant degradation outside that subspace. Theoretical analysis establishes temporal-decay monotonicity, trust-dependent reliability, per-release differential-privacy protection, and an error bound including privacy noise and residual malicious influence. Experiments on four industrial datasets evaluate prediction, benign drift, efficiency, and poisoning robustness under distributed edge partitioning. The design evaluates benign feature drift and suspicious anchor inconsistency through distinct operational scores, allowing the drift estimator and source-trust mechanism to be examined for their different roles without claiming explicit causal separation between drift and malicious manipulation.

The contributions are: (1) privacy-protected historical-anchor sharing without centralizing raw streams; (2) source-trust and cross-node-consistency modeling for poisoned-anchor suppression; (3) context-aligned historical decay with nuisance-subspace compensation for suppressing nuisance drift while preserving task-relevant degradation; and (4) a unified theoretical and experimental analysis of prediction utility, privacy, robustness, and deployment efficiency.

2. Related Works

Industrial drift handling has increasingly shifted from static fault detection toward explicit modeling of non-stationary process behavior. Chapelin et al. developed a data-driven framework for detecting and diagnosing drift in heterogeneous production processes, demonstrating that distribution change itself can be monitored as an operational signal; however, drift diagnosis remains separated from the question of whether historical representations should be retained, discarded, or trusted across nodes [7]. Benatia et al. introduced continual learning for failure prediction under non-stationary condition-monitoring streams, which improves temporal adaptability but relies on model updating rather than constructing a bounded historical reference with explicit state-level reliability [8]. Process-mining-based IPDD further detects process drift from evolving execution behavior, yet its objective is to identify structural change rather than compensate latent representation drift while preserving task-relevant degradation [9]. The broader review of industrial process-monitoring approaches confirms the transition toward data-driven, adaptive, and intelligent monitoring, but also reveals that prediction, adaptation, privacy, and security are usually treated as separate system functions [10]. From the representation-sharing perspective, these studies largely focus on benign monitoring and adaptation and do not directly address privacy-protected cross-node state release or poisoning-aware admission of shared historical representations. From an algorithmic perspective, these studies motivate non-stationary monitoring but do not resolve the central problem considered here: a historical state can be statistically old, contextually mismatched, privacy-sensitive, or maliciously supplied, and these properties require different treatment rather than a single drift score or retraining trigger.

Recent manufacturing research further emphasizes multimodal representation learning and process-specific instability modeling. Deep-learning applications in metal additive manufacturing have demonstrated the value of visual, thermal, and sensor representations, while also highlighting variability, limited generalization, and data-management challenges across manufacturing conditions [11]. Oueslati et al.

explicitly coupled in-process monitoring with CTWD drift estimation for WAAM instability prediction, but the estimated drift is used mainly as a predictive indicator and does not determine the trustworthiness of shared historical latent states [12]. Physics-guided multi-source fusion improves LPBF melting-state recognition by integrating heterogeneous process knowledge and sensing information [13]. However, this line of work primarily focuses on predictive fusion and does not explicitly address privacy-preserving representation exchange in distributed monitoring environments. Heat-input control with deep-learning-based indirect stability measurement similarly strengthens process-state inference, but its representation is optimized for control observability rather than secure cross-node reuse [14]. Machine-learning monitoring criteria can provide process-sensitive evidence for WAAM instability detection [15], but this formulation does not consider adversarial manipulation of historical representations in distributed monitoring systems. Reproducible process-monitoring research also stresses consistent data processing, evaluation protocols, and transferable learning pipelines, but reproducibility alone does not solve privacy leakage or poisoned-memory propagation in distributed deployment [16]. These methods therefore clarify how informative industrial representations can be extracted and adapted, whereas privacy-preserving release, poisoning-aware validation, and trustworthy reuse of representations across nodes remain distinct representation-management requirements addressed by HSDC-Net.

Task-oriented industrial models further illustrate the distinction between predictive modeling and secure representation sharing. A locally spatiotemporal soft sensor for cement production captures neighboring temporal and process dependencies to improve key-indicator prediction, but its local correlation structure does not provide a mechanism for evaluating remote historical sources or protecting latent information exchanged between distributed nodes [17]. Mechanism-assisted learning for machining incorporates physical insight into nonequilibrium process dynamics, which helps preserve meaningful process evolution, yet it does not explicitly decompose shared latent deviations into nuisance drift and adversarial contamination [18]. Forecast-assisted remaining-useful-life prediction improves predictive-maintenance decisions by exploiting future degradation tendencies, but historical forecasting accuracy alone cannot prevent unreliable or privacy-sensitive states from entering a distributed reference bank [19]. HSDC-Net differs from these approaches at the representation-management level. It keeps raw manufacturing streams local, shares only clipped and privacy-perturbed anchors, evaluates cross-node inconsistency and recursive source trust before historical reuse, and then combines temporal

decay with low-rank context transport. Most importantly, the current-reference residual is projected onto a learned nuisance-drift subspace before compensation, so legitimate degradation outside that subspace is preserved. Accordingly, HSDC-Net addresses privacy protection, poisoning-aware reliability, and drift adaptation jointly within historical-reference construction rather than treating the absence of these mechanisms in task-specific industrial models as deficiencies of their original objectives.

3. Method

3.1. Problem Formulation and Framework Overview

Modern industrial monitoring increasingly relies on distributed industrial networks in which heterogeneous sensing, computation, and monitoring functions are deployed across production devices, industrial edge nodes, gateways, and cloud platforms [10]. Consider a distributed monitoring system consisting of M edge nodes,

$$\mathcal{E} = \{E_1, E_2, \dots, E_M\}, \quad (1)$$

where each node is connected to a local group of sensors, cameras, controllers, or manufacturing machines. Different edge nodes may correspond to different production lines, machines, material batches, operating stages, or geographically separated manufacturing sites.

At edge node E_i , the multimodal observation at time step t is defined as

$$\mathbf{o}_{i,t} = [\mathbf{x}_{i,t}; \mathbf{v}_{i,t}; \mathbf{c}_{i,t}] \in \mathbb{R}^{d_o}, \quad (2)$$

where $\mathbf{x}_{i,t}$ contains multivariate sensor measurements, $\mathbf{v}_{i,t}$ contains visual or thermal features, and $\mathbf{c}_{i,t}$ represents material formulation, batch identity, equipment configuration, operating stage, and other local process contexts, consistent with heterogeneous and multi-source industrial monitoring settings [11, 13]. When image data are unavailable, $\mathbf{v}_{i,t}$ is omitted without modifying the temporal framework.

The local production sequence observed by edge node E_i is

$$\mathcal{O}_{i,1:T_i} = \{\mathbf{o}_{i,1}, \dots, \mathbf{o}_{i,T_i}\}. \quad (3)$$

A local temporal encoder generates

$$\mathbf{h}_{i,t}^0 = f_\theta(\mathcal{O}_{i,1:t}), \quad \widehat{\mathbf{y}}_{i,t} = g_\omega(\mathbf{h}_{i,t}^0), \quad (4)$$

where $\mathbf{y}_{i,t}$ may denote a quality indicator, fault state, degradation score, or remaining useful life.

Under non-stationary manufacturing conditions, industrial monitoring representations may vary with operating-condition changes and evolving process distributions [7]. The uncorrected representation is modeled as

$$\mathbf{h}_{i,t}^0 = \mathbf{s}_{i,t} + \mathbf{d}_{i,t} + \boldsymbol{\epsilon}_{i,t}, \quad (5)$$

where $\mathbf{s}_{i,t}$ denotes the task-relevant physical state, $\mathbf{d}_{i,t}$ is nuisance drift caused by sensor aging, material variation, acquisition-condition changes, or process-mode switching, and $\boldsymbol{\epsilon}_{i,t}$ is encoding noise.

Unlike centralized monitoring, distributed industrial monitoring additionally introduces data-security and privacy risks. Raw sensor streams, manufacturing images, equipment trajectories, and detailed process contexts may contain proprietary production information and therefore should not be directly exposed to a coordinating gateway or other edge nodes. Meanwhile, authenticated edge nodes may become compromised and transmit manipulated historical states, causing poisoned representations to enter a shared historical memory and subsequently affect other nodes.

Accordingly, HSDC-Net considers two representation-level threats. First, an honest-but-curious coordinating gateway follows the prescribed protocol but may attempt to infer sensitive local information from transmitted representations. Second, a subset of authenticated edge nodes may be compromised and inject manipulated historical anchors. Authentication, communication-channel encryption, denial-of-service attacks, and identity forgery are assumed to be handled by the underlying network-security infrastructure and are outside the present scope.

To address non-stationary representation drift while reducing both privacy leakage and poisoned-memory propagation, HSDC-Net constructs

$$\mathbf{h}_{i,t}^c = \mathcal{C}_\Theta(\mathbf{h}_{i,t}^0, \mathcal{A}_t, \mathbf{c}_{i,t}, \mathbf{T}_t), \quad (6)$$

where $\mathcal{A}_t$ denotes a bounded privacy-protected historical anchor bank maintained by the coordinating gateway and periodically synchronized to registered edge nodes as a read-only snapshot, and

$$\mathbf{T}_t = [\mathbf{T}_{1,t}, \dots, \mathbf{T}_{M,t}] \quad (7)$$

contains the dynamic source-trust scores of all registered edge nodes. The synchronized snapshot contains only privacy-protected anchors and the reliability metadata required for local reference construction; raw observations and unrestricted local states are never broadcast.

Rather than directly sharing raw samples or unrestricted hidden states, each node constructs bounded latent prototypes from locally stable observations and applies privacy perturbation before transmission. The coordinating gateway evaluates the cross-node consistency of each received prototype and maintains source trust over time. After each gateway update, the resulting trusted anchor-bank snapshot and source-trust metadata are synchronized to the registered edge nodes for local drift estimation and historical-reference construction. Historical-anchor reliability is then jointly

determined by temporal age, contextual compatibility, latent similarity, predictive uncertainty, source trust, and cross-node consistency.

The complete inference and secure historical-sharing path is

$$\begin{aligned} \mathcal{O}_{i,1:t} &\rightarrow \mathbf{h}_{i,t}^0 \rightarrow \widehat{D}_{i,t} \rightarrow \mathcal{P}_{i,t} \rightarrow \mathcal{A}_t \\ &\rightarrow \{\rho_{k,i,t}\}_{k=1}^K \rightarrow \mathbf{r}_{i,t} \rightarrow \mathbf{h}_{i,t}^c \rightarrow \widehat{\mathbf{y}}_{i,t}. \end{aligned} \quad (8)$$

where $\widehat{D}_{i,t}$ denotes the current local drift intensity, $\mathcal{P}_{i,t}$ is a privacy-protected anchor packet, $\rho_{k,i,t}$ denotes the trust-aware reliability of historical anchor k for node E_i , and $\mathbf{r}_{i,t}$ is the trusted context-aligned historical reference.

Figure 1 presents the complete architecture of HSDC-Net. Heterogeneous manufacturing observations remain at their originating edge nodes and are locally encoded into long-sequence latent states. Only stable latent prototypes are clipped and privacy protected before transmission. At the coordinating gateway, cross-node consistency and historical source behavior are jointly evaluated to suppress potentially poisoned anchors. The resulting bounded trusted anchor bank supports adaptive historical-state decay, context transport, historical-reference construction, and subspace-selective drift compensation. Compensated representations are finally used for quality monitoring, fault diagnosis, and predictive maintenance.

3.2. Lightweight Long-Sequence Representation, Private Anchor Sharing, and Drift Characterization

To eliminate scale inconsistency and reduce outlier effects without introducing future information, each sensor channel at edge node E_i is causally normalized:

$$\widetilde{x}_{i,t,j} = \frac{x_{i,t,j} - \text{Med}(x_{i,t-w_n+1:t,j})}{\text{IQR}(x_{i,t-w_n+1:t,j}) + \varepsilon}. \quad (9)$$

Missing measurements are filled using the latest locally available valid value, and a binary missingness mask is appended to the input.

The fused local representation is

$$\mathbf{z}_{i,t} = \text{LN}(\mathbf{W}_x \widetilde{\mathbf{x}}_{i,t} + \mathbf{W}_v \mathbf{v}_{i,t} + \mathbf{W}_c \mathbf{c}_{i,t} + \mathbf{b}_z). \quad (10)$$

HSDC-Net combines a causal depthwise temporal convolution with a gated state-space recurrence. The local feature is

$$\mathbf{e}_{i,t} = \text{LN}[\mathbf{z}_{i,t} + \mathbf{W}_p \text{PWConv}(\text{DWConv}_{k,\delta}(\mathbf{z}_{i,t-k+1:t}))], \quad (11)$$

where k and δ denote the kernel size and dilation factor.

The long-term local state is updated as

$$\mathbf{a}_{i,t} = \sigma(\mathbf{W}_a[\mathbf{e}_{i,t}; \mathbf{c}_{i,t}] + \mathbf{b}_a), \quad (12)$$

$$\mathbf{b}_{i,t} = \sigma(\mathbf{W}_b[\mathbf{e}_{i,t}; \mathbf{c}_{i,t}] + \mathbf{b}_b), \quad (13)$$

$$\mathbf{s}_{i,t}^\ell = \mathbf{a}_{i,t} \odot \mathbf{s}_{i,t-1}^\ell + \mathbf{b}_{i,t} \odot \mathbf{W}_e \mathbf{e}_{i,t}, \quad (14)$$

$$\mathbf{h}_{i,t}^0 = \text{LN}(\mathbf{e}_{i,t} + \mathbf{W}_s \mathbf{s}_{i,t}^\ell). \quad (15)$$

This structure captures local process fluctuations and long-term equipment evolution with linear sequence complexity.

Each edge node maintains a temporary local prototype set

$$\mathcal{A}_{i,t}^L = \{\mathbf{m}_{i,k}, \mathbf{c}_{i,k}, \tau_{i,k}, u_{i,k}, n_{i,k}\}_{k=1}^{K_i}, \quad (16)$$

where $\mathbf{m}_{i,k}$ is a stable latent prototype, $\mathbf{c}_{i,k}$ is its local operating context, $\tau_{i,k}$ is its latest update time, $u_{i,k}$ is its uncertainty, and $n_{i,k}$ is the number of local states represented by this prototype.

To bound the contribution of an individual observation before historical-state sharing, each local compensated representation is clipped according to

$$\text{Clip}_C(\mathbf{h}) = \mathbf{h} \min\left(1, \frac{C}{\|\mathbf{h}\|_2 + \varepsilon}\right). \quad (17)$$

The local anchor is formed as

$$\mathbf{m}_{i,k} = \frac{1}{n_{i,k}} \sum_{j \in \mathcal{S}_{i,k}} \text{Clip}_C(\mathbf{h}_{i,j}^c), \quad (18)$$

where $\mathcal{S}_{i,k}$ denotes the local observation set assigned to prototype k .

For two neighboring local datasets differing in one observation, the sensitivity of the clipped mean is bounded by

$$\Delta_{i,k} \leq \frac{2C}{n_{i,k}}. \quad (19)$$

Before an anchor leaves the local edge node, Gaussian perturbation is applied:

$$\widetilde{\mathbf{m}}_{i,k} = \mathbf{m}_{i,k} + \mathbf{z}_{i,k}^p, \quad \mathbf{z}_{i,k}^p \sim \mathcal{N}(\mathbf{0}, \sigma_p^2 \Delta_{i,k}^2 \mathbf{I}). \quad (20)$$

Sensitive process contexts are not transmitted in their original form. A compact context embedding is first obtained as

$$\mathbf{e}_{i,k}^c = \mathbf{E}_c \mathbf{c}_{i,k}, \quad (21)$$

after which

$$\widetilde{\mathbf{e}}_{i,k}^c = \text{Clip}_{C_c}(\mathbf{e}_{i,k}^c) + \mathbf{z}_{i,k}^c, \quad \mathbf{z}_{i,k}^c \sim \mathcal{N}(\mathbf{0}, \sigma_c^2 \Delta_{i,k}^2 \mathbf{I}). \quad (22)$$

The transmitted historical-state packet is therefore

$$\mathcal{P}_{i,k} = \langle \widetilde{\mathbf{m}}_{i,k}, \widetilde{\mathbf{e}}_{i,k}^c, \tau_{i,k}, u_{i,k}, n_{i,k}, i \rangle. \quad (23)$$

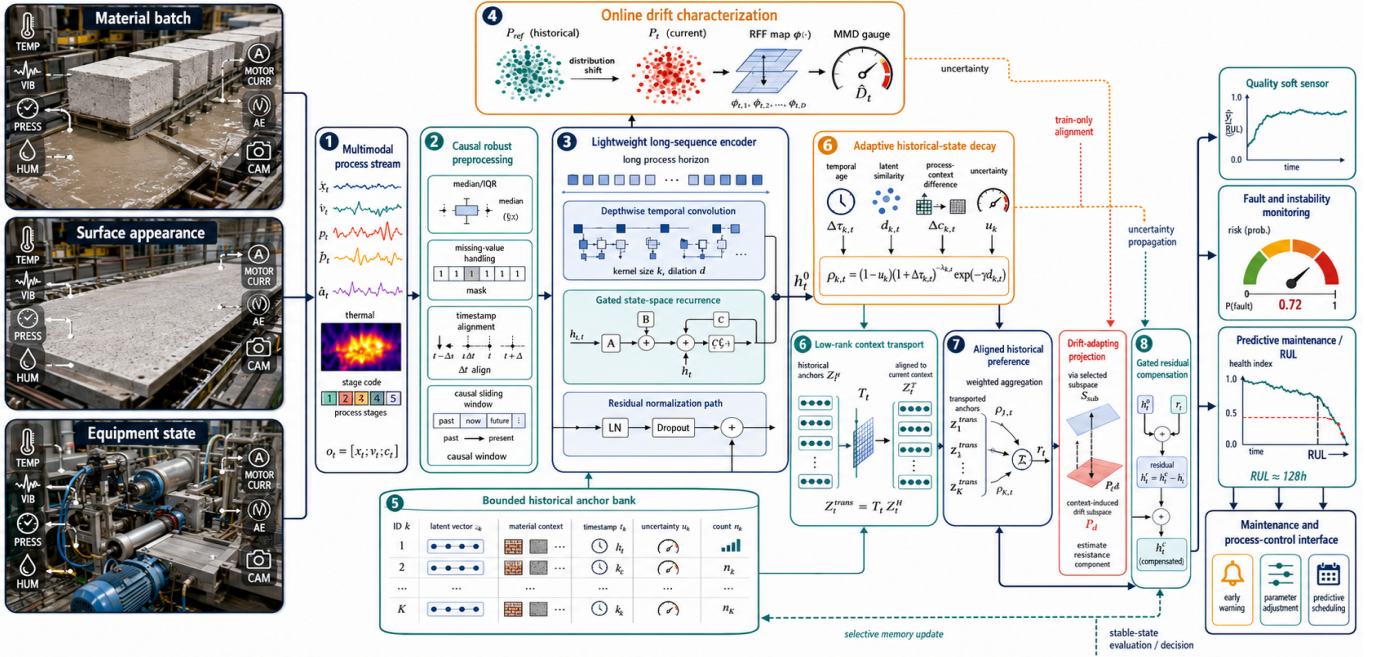


Figure 1. Overall architecture of HSDC-Net for privacy-preserving and trust-aware distributed industrial edge monitoring. Raw manufacturing observations remain at their originating edge nodes, while bounded latent anchors are privacy protected before transmission. Cross-node consistency and source-trust evaluation suppress unreliable or poisoned historical states, after which trusted anchors support adaptive historical-state decay, low-rank context transport, historical-reference construction, and gated compensation in the nuisance-drift subspace

Raw sensor vectors $\mathbf{x}_{i,t}$, visual observations $\mathbf{v}_{i,t}$, and complete local temporal trajectories are not included in $\mathcal{P}_{i,k}$.

The coordinating gateway maintains a bounded historical anchor bank

$$\mathcal{A}_t = \{\tilde{\mathbf{m}}_k, \tilde{\mathbf{e}}_k^c, \tau_k, u_k, n_k, s_k, A_k\}_{k=1}^K, \quad (24)$$

where s_k identifies the authenticated source node and A_k records the latest cross-node inconsistency score of anchor k . At each communication round, the gateway distributes a read-only synchronized snapshot of Equation (24), together with the corresponding source-trust values, to the registered edge nodes. Consequently, the following drift score is evaluated locally at E_i against this synchronized trusted snapshot rather than requiring the gateway to access the node's recent unperturbed latent sequence.

The recent local latent distribution is compared with the trusted anchor distribution through random Fourier features:

$$\widehat{D}_{i,t} = \left\| \frac{1}{w_d} \sum_{\ell=t-w_d+1}^t \phi(\mathbf{h}_{i,\ell}^0) - \sum_{k=1}^K \pi_{k,t} \phi(\tilde{\mathbf{m}}_k) \right\|_2^2, \quad (25)$$

where

$$\pi_{k,t} = \frac{n_k T_{s_k,t} \exp(-\gamma_A A_k)}{\sum_{j=1}^K n_j T_{s_j,t} \exp(-\gamma_A A_j) + \varepsilon}. \quad (26)$$

Thus, anchors contributed by historically unreliable nodes or exhibiting strong cross-node inconsistency have limited influence on drift characterization.

Current-state uncertainty is estimated by

$$u_{i,t} = \sigma(\mathbf{w}_u^\top \mathbf{h}_{i,t}^0 + b_u). \quad (27)$$

For a classification task, the predictive posterior and its maximum confidence are defined as

$$\mathbf{p}_{i,t} = \text{Softmax}(g_f(\mathbf{h}_{i,t}^0)), \quad p_{i,\max,t} = \max_c [\mathbf{p}_{i,t}]_c. \quad (28)$$

Thus, $p_{i,\max,t}$ measures the confidence of the current fault-state prediction rather than introducing an additional latent score. For regression-only monitoring tasks without a categorical posterior, the $p_{i,\max,t} > \tau_p$ condition below is omitted, while the uncertainty and drift conditions remain active.

A local state is eligible for historical-memory construction only when

$$u_{i,t} < \tau_u, \quad \widehat{D}_{i,t} < \tau_d, \quad p_{i,\max,t} > \tau_p, \quad T_{i,t} > \tau_T. \quad (29)$$

For classification tasks, the first three conditions reject unstable or uncertain local states, whereas the trust constraint prevents a persistently suspicious source from continuously contributing new historical information. For regression-only tasks, eligibility is determined by $u_{i,t} < \tau_u$, $\widehat{D}_{i,t} < \tau_d$, and $T_{i,t} > \tau_T$.

3.3. Trust-Aware Historical-State Decay and Latent Compensation

Privacy perturbation alone cannot prevent a compromised authenticated node from intentionally submitting manipulated latent anchors. HSDC-Net therefore evaluates each candidate anchor against context-compatible historical information contributed by multiple nodes before allowing it to significantly influence the shared reference.

For a candidate anchor $\widetilde{\mathbf{m}}_{i,k}$, let

$$\mathcal{N}_{i,k,t} = \left\{ (j, l) : \left\| \widetilde{\mathbf{e}}_{j,l}^c - \widetilde{\mathbf{e}}_{i,k}^c \right\|_2 < \tau_c \right\} \quad (30)$$

denote the set of context-compatible historical anchors.

A robust reference center is obtained through

$$\mathbf{m}_{i,k,t}^R = \text{GeoMed} \left(\left\{ \widetilde{\mathbf{m}}_{j,l} : (j, l) \in \mathcal{N}_{i,k,t} \right\} \right), \quad (31)$$

where $\text{GeoMed}(\cdot)$ denotes the geometric median.

The normalized cross-node inconsistency score of candidate anchor (i, k) is defined as

$$A_{i,k,t} = \frac{\left\| \widetilde{\mathbf{m}}_{i,k} - \mathbf{m}_{i,k,t}^R \right\|_2^2}{\text{Med} \left(\left\{ \left\| \widetilde{\mathbf{m}}_{j,l} - \mathbf{m}_{i,k,t}^R \right\|_2^2 : (j, l) \in \mathcal{N}_{i,k,t} \right\} \right) + \varepsilon}. \quad (32)$$

For all candidate anchors submitted by source E_i in the current communication round, the instantaneous consistency score is

$$R_{i,t} = \frac{1}{|\mathcal{B}_{i,t}|} \sum_{k \in \mathcal{B}_{i,t}} \exp(-A_{i,k,t}), \quad (33)$$

where $\mathcal{B}_{i,t}$ denotes the set of candidate anchors reported by node E_i .

A bounded source-trust score is recursively maintained as

$$T_{i,t} = \text{clip}_{[T_{\min}, 1]} \left[(1 - \kappa_T) T_{i,t-1} + \kappa_T R_{i,t} \right], \quad (34)$$

where $\kappa_T \in (0, 1)$ controls trust adaptation. Consequently, an isolated abnormal update does not immediately eliminate a node, while persistent inconsistent submissions progressively reduce its historical influence.

For global anchor k when serving current node E_i , the contextual discrepancy is

$$\Delta \mathbf{e}_{k,i,t}^c = \mathbf{E}_c \mathbf{c}_{i,t} - \widetilde{\mathbf{e}}_k^c. \quad (35)$$

HSDC-Net constructs the relation descriptor

$$\mathbf{q}_{k,i,t} = \left[1 - \frac{\widetilde{\mathbf{m}}_k^T \mathbf{h}_{i,t}^0}{\left\| \widetilde{\mathbf{m}}_k \right\|_2 \left\| \mathbf{h}_{i,t}^0 \right\|_2 + \varepsilon}; \left\| \Delta \mathbf{e}_{k,i,t}^c \right\|_2; \widehat{D}_{i,t}; u_k; u_{i,t}; 1 - T_{s_k,t}; A_k \right]. \quad (36)$$

Compared with a purely drift-oriented relation descriptor, the last two terms explicitly quantify source unreliability and cross-node inconsistency.

An adaptive positive decay rate is generated as

$$\lambda_{k,i,t} = \lambda_{\min} + \text{Softplus} \left[\mathbf{w}_\lambda^T \text{MLP}_\lambda(\mathbf{q}_{k,i,t}) + b_\lambda \right], \quad (37)$$

where $\lambda_{\min} > 0$.

The trust-aware historical reliability is then

$$\rho_{k,i,t} = (1 - u_k) T_{s_k,t} (1 + \Delta \tau_{k,t})^{-\lambda_{k,i,t}} \times \exp \left[-\gamma_c \left\| \Delta \mathbf{e}_{k,i,t}^c \right\|_2^2 - \gamma_A A_k \right]. \quad (38)$$

where

$$\Delta \tau_{k,t} = \max(0, t - \tau_k). \quad (39)$$

Equation (38) has two security-relevant properties. A low-trust source directly reduces the reliability of all historical states originating from that node, while an individually inconsistent anchor is additionally suppressed by $\exp(-\gamma_A A_k)$. Because source trust also enters the learned relation descriptor in Equation (36), the total dependence of reliability on trust additionally involves the trust sensitivity of the adaptive decay rate; this dependence is characterized explicitly in Proposition 2 below.

The normalized anchor weight is

$$\alpha_{k,i,t} = \frac{\exp \left[\log(\rho_{k,i,t} + \varepsilon) / \tau_\alpha \right]}{\sum_{j=1}^K \exp \left[\log(\rho_{j,i,t} + \varepsilon) / \tau_\alpha \right]}. \quad (40)$$

Because trusted historical anchors may originate from different production lines, material formulations, operating stages, or sensor environments, directly combining them may introduce systematic context bias. HSDC-Net therefore applies a low-rank context adapter:

$$\mathbf{g}_{k,i,t}^a = \sigma \left(\mathbf{W}_g^a \mathbf{q}_{k,i,t} + \mathbf{b}_g^a \right), \quad (41)$$

$$\overline{\mathbf{m}}_{k \rightarrow i,t} = \widetilde{\mathbf{m}}_k + \mathbf{U}_a \text{Diag}(\mathbf{g}_{k,i,t}^a) \mathbf{V}_a^T \left[\widetilde{\mathbf{m}}_k; \Delta \mathbf{e}_{k,i,t}^c \right], \quad (42)$$

where the adapter rank r_a is substantially smaller than the feature dimension d .

The trusted context-aligned historical reference is

$$\mathbf{r}_{i,t} = \sum_{k=1}^K \alpha_{k,i,t} \overline{\mathbf{m}}_{k \rightarrow i,t}. \quad (43)$$

Directly replacing the current representation with $\mathbf{r}_{i,t}$ is undesirable because legitimate equipment degradation or process evolution may differ from historical states. HSDC-Net therefore learns a low-rank nuisance-drift basis

$$\mathbf{U}_d \in \mathbb{R}^{d \times r_d}, \quad \mathbf{P}_d = \mathbf{U}_d \mathbf{U}_d^\top, \quad r_d \ll d. \quad (44)$$

The orthogonality condition

$$\mathbf{U}_d^\top \mathbf{U}_d \approx \mathbf{I}_{r_d} \quad (45)$$

encourages $\mathbf{P}_d$ to behave as a valid low-dimensional projection.

A channel-wise compensation gate is defined as

$$\begin{aligned} \mathbf{g}_{i,t}^c &= \sigma(\mathbf{W}_c[\mathbf{h}_{i,t}^0; \mathbf{r}_{i,t}; \widehat{\mathbf{D}}_{i,t}; u_{i,t}] + \mathbf{b}_c), \\ \mathbf{G}_{i,t} &= \text{Diag}(\mathbf{g}_{i,t}^c). \end{aligned} \quad (46)$$

The compensated state is

$$\mathbf{h}_{i,t}^c = \text{LN}[\mathbf{h}_{i,t}^0 - \mathbf{G}_{i,t} \mathbf{P}_d (\mathbf{h}_{i,t}^0 - \mathbf{r}_{i,t})]. \quad (47)$$

Figure 2 provides a geometric interpretation of the proposed mechanism. Panel A illustrates how benign manufacturing drift and maliciously manipulated representations may move observed states away from a stable latent manifold. Panel B shows that recent, context-compatible, low-uncertainty, low-anomaly, and trusted historical anchors receive larger reliability weights and are transported to the current operating context before aggregation. Panel C decomposes the current-reference residual into task-relevant evolution and nuisance components so that compensation is restricted to the learned drift subspace.

When drift is negligible, $\mathbf{g}_{i,t}^c$ approaches zero and the original representation is largely retained. When substantial nuisance drift occurs, the correction strength increases. When a historical state originates from an untrusted source or exhibits strong cross-node inconsistency, its reliability approaches zero before historical-reference construction, thereby limiting poisoned-memory propagation.

The final local outputs are

$$\widehat{\mathbf{y}}_{i,t}^q = g_q(\mathbf{h}_{i,t}^c), \quad \widehat{\mathbf{y}}_{i,t}^f = g_f(\mathbf{h}_{i,t}^c), \quad \widehat{y}_{i,t}^r = g_r(\mathbf{h}_{i,t}^c), \quad (48)$$

corresponding to quality prediction, fault monitoring, and remaining useful life estimation, respectively.

3.4. Learning Objective and Secure Online Inference

To explicitly identify nuisance-drift directions, moderate label-preserving perturbations are introduced during training:

$$x_{i,t,j}^a = a_j x_{i,t,j} + b_j + \kappa_j \frac{t}{T_i} + \epsilon_{i,t,j}, \quad (49)$$

where a_j , b_j , and κ_j simulate sensor-gain variation, offset bias, and slow baseline drift. Channel dropout, local time warping, random missingness, and visual brightness variation may also be applied within ranges that do not change the underlying task label.

Let $\mathbf{h}_{i,t}^{a,0}$ and $\mathbf{h}_{i,t}^{a,c}$ denote the uncorrected and compensated representations generated from the augmented sequence. The augmentation-induced difference should be captured primarily by the nuisance-drift subspace:

$$\mathcal{L}_{\text{sub}} = \frac{1}{N} \sum_{i,t} \|(\mathbf{I} - \mathbf{P}_d)(\mathbf{h}_{i,t}^{a,0} - \mathbf{h}_{i,t}^0)\|_2^2. \quad (50)$$

The compensated representation is required to remain invariant to these label-preserving perturbations:

$$\mathcal{L}_{\text{inv}} = \frac{1}{N} \sum_{i,t} \|\mathbf{h}_{i,t}^{a,c} - \mathbf{h}_{i,t}^c\|_2^2. \quad (51)$$

The multi-task prediction loss is

$$\mathcal{L}_{\text{task}} = \lambda_q \mathcal{L}_q + \lambda_f \mathcal{L}_f + \lambda_r \mathcal{L}_r, \quad (52)$$

where $\mathcal{L}_q$ may use the Huber loss for quality regression, $\mathcal{L}_f$ uses weighted cross-entropy for fault classification, and $\mathcal{L}_r$ uses mean-squared or heteroscedastic regression for remaining-useful-life prediction.

Reference consistency is imposed only within the nuisance-drift subspace:

$$\mathcal{L}_{\text{ref}} = \frac{1}{N} \sum_{i,t} \|\mathbf{P}_d(\mathbf{h}_{i,t}^c - \mathbf{r}_{i,t})\|_2^2. \quad (53)$$

Task-relevant information outside the estimated nuisance subspace is preserved by

$$\mathcal{L}_{\text{pre}} = \frac{1}{N} \sum_{i,t} \|(\mathbf{I} - \mathbf{P}_d)(\mathbf{h}_{i,t}^c - \text{sg}[\mathbf{h}_{i,t}^0])\|_2^2. \quad (54)$$

The drift basis and compensation gate are regularized by

$$\mathcal{L}_{\text{orth}} = \|\mathbf{U}_d^\top \mathbf{U}_d - \mathbf{I}_{r_d}\|_F^2, \quad \mathcal{L}_{\text{gate}} = \frac{1}{N} \sum_{i,t} \|\mathbf{g}_{i,t}^c\|_1. \quad (55)$$

To prevent the learned context transport from excessively amplifying privacy perturbation or abnormal historical deviations, the low-rank adaptation magnitude is additionally constrained through

$$\mathcal{L}_{\text{adapt}} = \frac{1}{N_a} \sum_{i,t,k} \|\overline{\mathbf{m}}_{k \rightarrow i,t} - \widetilde{\mathbf{m}}_k\|_2^2. \quad (56)$$

The complete training objective is

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_{\text{task}} + \beta_1 \mathcal{L}_{\text{sub}} + \beta_2 \mathcal{L}_{\text{inv}} + \beta_3 \mathcal{L}_{\text{ref}} \\ &\quad + \beta_4 \mathcal{L}_{\text{pre}} + \beta_5 \mathcal{L}_{\text{orth}} + \beta_6 \mathcal{L}_{\text{gate}} + \beta_7 \mathcal{L}_{\text{adapt}}. \end{aligned} \quad (57)$$

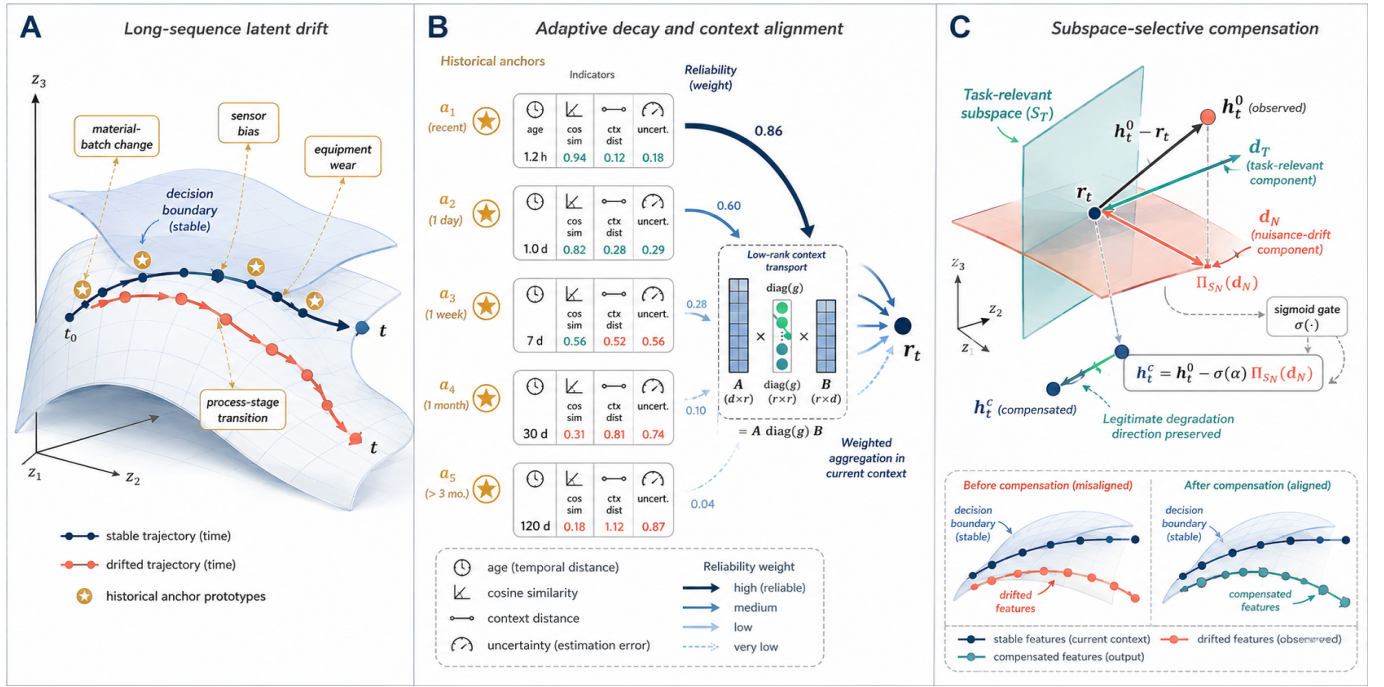


Figure 2. Mechanism of trust-aware historical-state decay and subspace-selective compensation. Panel A illustrates representation deviation caused by manufacturing drift and potentially manipulated historical states. Panel B shows source-trust evaluation, anomaly-aware reliability estimation, and low-rank context alignment of privacy-protected historical anchors. Panel C separates task-relevant evolution from nuisance drift and applies gated compensation only to the learned drift subspace

For an eligible local state, the closest context-compatible local anchor is selected by

$$k^* = \arg \min_k [\eta_h \|h_{i,t}^c - m_{i,k}\|_2^2 + \eta_c \|c_{i,t} - c_{i,k}\|_2^2]. \quad (58)$$

If the matching distance is below τ_m , the local prototype is updated through

$$m_{i,k^*} \leftarrow (1 - \zeta_{i,k^*}) m_{i,k^*} + \zeta_{i,k^*} h_{i,t}^c, \quad \zeta_{i,k} = \frac{1}{\min(n_{i,k}, n_{\max}) + 1}. \quad (59)$$

Its local context, uncertainty, timestamp, and sample count are updated similarly.

Before transmission, the updated local prototype is converted into the privacy-protected packet in Equation (23). At the coordinating gateway, a newly received anchor is admitted to the shared historical bank only when both its instantaneous cross-node inconsistency and its updated source-trust score satisfy

$$A_{i,k,t} < \tau_A, \quad T_{i,t} > \tau_T. \quad (60)$$

Anchors violating either condition are rejected and are not allowed to update the shared historical memory.

If a received anchor matches an existing trusted anchor, the shared prototype is updated as

$$\tilde{m}_{k^*} \leftarrow (1 - \xi_{k^*}) \tilde{m}_{k^*} + \xi_{k^*} \tilde{m}_{i,k}, \quad (61)$$

where

$$\xi_k = \frac{T_{i,t} \exp(-\gamma_A A_{i,k,t})}{n_k + T_{i,t} \exp(-\gamma_A A_{i,k,t}) + \varepsilon}. \quad (62)$$

Thus, even among accepted states, updates from less reliable sources have a smaller effect on the stored anchor.

When the global bank is full, the anchor with the minimum utility

$$\mathcal{U}_k = \frac{n_k(1 - u_k) T_{s_k,t} \exp(-\gamma_A A_k)}{1 + t - \tau_k} \quad (63)$$

is replaced. The utility jointly penalizes obsolete, uncertain, anomalous, and low-trust historical states.

Figure 3 illustrates the online behavior of HSDC-Net in a distributed production trajectory. During stable operation, low-uncertainty local states generate privacy-protected historical anchors and maintain high source trust, while compensation remains weak. When benign raw-material variation or sensor drift emerges, incompatible historical anchors are gradually down-weighted and the compensation response increases. When a compromised edge node injects inconsistent anchor states, its anomaly score increases, its source trust decreases, and its anchors are either rejected or assigned negligible historical weight. In contrast, genuine equipment degradation is encouraged to remain outside the nuisance-drift subspace, while benign distribution variation and suspicious historical anchors are

Algorithm 1 Training and Secure Online Inference of HSDC-Net

Require: Distributed streams $\{\mathcal{O}_{i,1:T_i}\}_{i=1}^M$, targets $\{\mathcal{Y}_{i,1:T_i}\}_{i=1}^M$, anchor capacity K

Ensure: Predictions $\widehat{\mathcal{Y}}$, trusted anchor bank $\mathcal{A}_T$

- 1: Initialize HSDC-Net parameters, anchor banks, and $T_{i,0} = 1$
- 2: **for** each training epoch **do**
- 3: **for** each edge node E_i **do**
- 4: **for** each time step t **do**
- 5: Use the latest synchronized trusted-bank snapshot to encode $\mathbf{o}_{i,t}$ and locally estimate $\mathbf{h}_{i,t}^0$, $u_{i,t}$, and $\widehat{D}_{i,t}$
- 6: Compute trust-aware anchor weights and historical reference $\mathbf{r}_{i,t}$
- 7: Obtain compensated state $\mathbf{h}_{i,t}^c$ and task predictions
- 8: Update local anchors if Equation (29) holds
- 9: **end for**
- 10: **end for**
- 11: Optimize Equation (57) using AdamW
- 12: **end for**
- 13: **for** each online communication round **do**
- 14: **for** each edge node E_i **do**
- 15: Perform local inference and construct an eligible private anchor packet $\mathcal{P}_{i,k}$
- 16: **end for**
- 17: Evaluate anchor inconsistency and update source trust $T_{i,t}$
- 18: Reject anchors violating Equation (60)
- 19: Update the bounded trusted anchor bank
- 20: Synchronize the updated read-only trusted-bank snapshot and source-trust metadata to registered edge nodes
- 21: **end for**
- 22: **return** $\widehat{\mathcal{Y}}$ and $\mathcal{A}_T$

handled through the compensation and trust mechanisms without assuming explicit causal identification of their origins.

3.5. Theoretical Analysis and Computational Complexity

Proposition 1. For fixed context difference, uncertainty, source trust, cross-node inconsistency, and positive decay rate $\lambda_{k,i,t} > 0$, the historical reliability in Equation (38) decreases monotonically with temporal distance $\Delta\tau_{k,t}$.

Proof. For fixed quantities other than temporal distance, define

$$\rho(\Delta\tau) = C_\rho(1 + \Delta\tau)^{-\lambda}, \quad (64)$$

where

$$C_\rho = (1 - u_k)T_{s_k,t} \exp[-\gamma_c \|\Delta\mathbf{e}_{k,i,t}^c\|_2^2 - \gamma_A A_k] > 0. \quad (65)$$

Then

$$\frac{\partial \rho}{\partial \Delta\tau} = -C_\rho \lambda (1 + \Delta\tau)^{-\lambda-1} < 0. \quad (66)$$

Therefore, older states receive progressively smaller reliability even when their security and contextual attributes remain unchanged. $\square$

Proposition 2. For fixed temporal distance, context, uncertainty, and anomaly score, let the adaptive decay rate in Equation (37) be written as $\lambda(T)$ with $T = T_{s_k,t}$. The historical reliability in Equation (38) is monotonically increasing with source trust whenever the trust sensitivity of the learned decay rate satisfies

$$\frac{\partial \lambda(T)}{\partial T} \leq \frac{1}{T \ln(1 + \Delta\tau_{k,t})}, \quad \Delta\tau_{k,t} > 0. \quad (67)$$

For $\Delta\tau_{k,t} = 0$, reliability is strictly increasing in T independently of $\partial\lambda/\partial T$.

Proof. With all non-trust quantities fixed, define

$$\rho(T) = C_T T (1 + \Delta\tau_{k,t})^{-\lambda(T)}, \quad (68)$$

where

$$C_T = (1 - u_k) \exp[-\gamma_c \|\Delta\mathbf{e}_{k,i,t}^c\|_2^2 - \gamma_A A_k] > 0. \quad (69)$$

Taking the total derivative with respect to T gives

$$\frac{d\rho}{dT} = \rho(T) \left[\frac{1}{T} - \ln(1 + \Delta\tau_{k,t}) \frac{\partial \lambda(T)}{\partial T} \right]. \quad (70)$$

Because $\rho(T) > 0$, Equation (70) is nonnegative whenever Equation (67) holds. When $\Delta\tau_{k,t} = 0$, the logarithmic term vanishes and $d\rho/dT = \rho(T)/T > 0$. Thus, the direct multiplicative trust term always favors trusted sources, while global monotonicity in the presence of the learned trust-dependent decay rate requires the stated sensitivity condition rather than following automatically from the multiplicative factor alone. $\square$

Proposition 3. Suppose a clipped prototype has ℓ_2 sensitivity $\Delta_{i,k}$ as defined in Equation (19). For $0 < \epsilon_p < 1$ and $0 < \delta_p < 1$, if

$$\sigma_p \geq \frac{\sqrt{2 \ln(1.25/\delta_p)}}{\epsilon_p}, \quad (71)$$

then transmission of $\widetilde{\mathbf{m}}_{i,k}$ in Equation (20) satisfies (ϵ_p, δ_p) -differential privacy with respect to one observation contributing to the local anchor.

Proof. Equation (17) bounds every local representation by C , giving the mean prototype sensitivity

$$\Delta_{i,k} \leq \frac{2C}{n_{i,k}}. \quad (72)$$

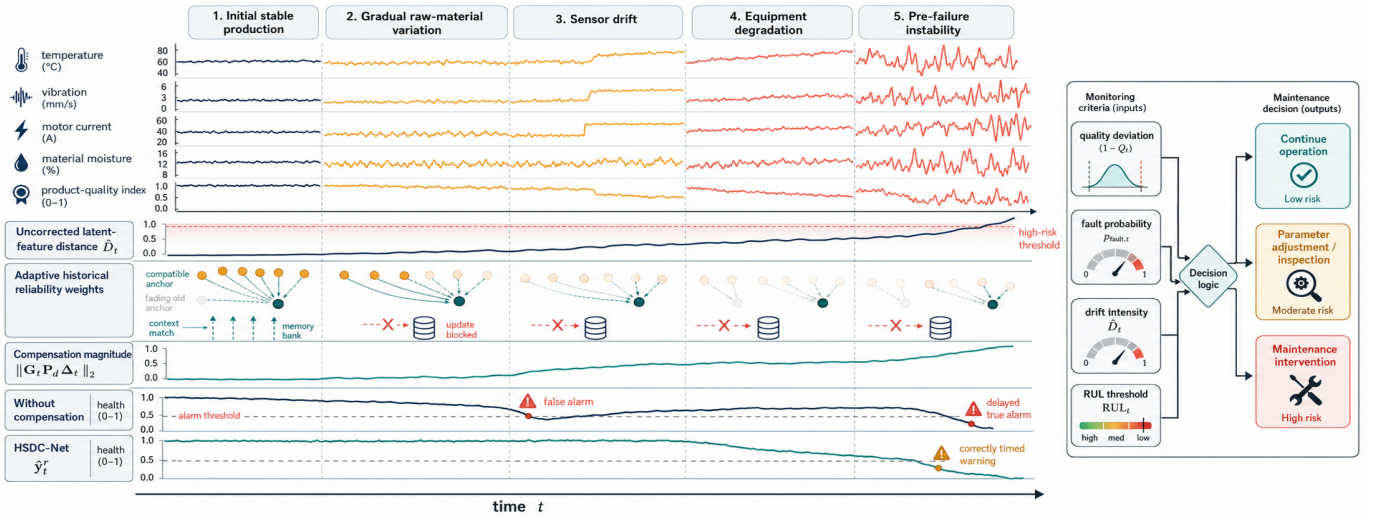


Figure 3. Secure online monitoring workflow of HSDC-Net over distributed manufacturing stages. Stable local states generate privacy-protected historical anchors, benign operating drift activates adaptive compensation, and inconsistent anchor injection reduces source trust and historical influence. The nuisance-drift subspace suppresses non-task-relevant variation while preserving genuine equipment degradation for quality monitoring, fault diagnosis, and predictive maintenance

Equation (20) adds zero-mean Gaussian noise with standard deviation $\sigma_p \Delta_{i,k}$. Under Equation (71), the standard Gaussian mechanism therefore provides (ϵ_p, δ_p) -differential privacy for each released prototype. $\square$

The privacy guarantee applies to the released prototype rather than raw observations. Repeated releases consume additional privacy budget; therefore, the communication interval and the number of anchor releases are explicitly bounded during deployment.

Theorem 1. Assume that the clean context-aligned historical reference satisfies

$$\mathbf{r}_{i,t}^* = \mathbf{s}_{i,t} + \mathbf{e}_{i,t}^r, \quad \|\mathbf{e}_{i,t}^r\|_2 \leq \epsilon_r. \quad (73)$$

Let privacy perturbation introduce aggregate reference error

$$\|\mathbf{e}_{i,t}^p\|_2 \leq \epsilon_p, \quad (74)$$

and suppose the total normalized weight assigned to residual malicious anchors after trust-aware weighting is at most ω_m . If every transported anchor satisfies

$$\|\bar{\mathbf{m}}_{k \rightarrow i,t}\|_2 \leq B_r, \quad (75)$$

then the trusted historical-reference error is bounded by

$$\|\mathbf{r}_{i,t} - \mathbf{s}_{i,t}\|_2 \leq \epsilon_r + \epsilon_p + 2B_r\omega_m. \quad (76)$$

Proof. Let $\mathcal{H}$ and $\mathcal{M}$ denote the accepted benign and malicious anchor sets, respectively, and define

$$\omega_m = \sum_{k \in \mathcal{M}} \alpha_{k,i,t}. \quad (77)$$

The actual reference may be decomposed into the clean reference, privacy perturbation, and residual malicious

contribution:

$$\mathbf{r}_{i,t} = \mathbf{r}_{i,t}^* + \mathbf{e}_{i,t}^p + \mathbf{e}_{i,t}^m. \quad (78)$$

Because both benign and transported malicious anchors are bounded by B_r , replacing a benign normalized contribution by a malicious one changes the aggregate reference by at most $2B_r$ per unit weight. Therefore,

$$\|\mathbf{e}_{i,t}^m\|_2 \leq 2B_r\omega_m. \quad (79)$$

Applying the triangle inequality to

$$\mathbf{r}_{i,t} - \mathbf{s}_{i,t} = \mathbf{e}_{i,t}^r + \mathbf{e}_{i,t}^p + \mathbf{e}_{i,t}^m \quad (80)$$

gives Equation (76). $\square$

Theorem 1 shows that representation privacy and poisoned-anchor robustness influence the historical reference through two explicit terms. Stronger privacy perturbation increases ϵ_p , whereas effective source-trust suppression decreases ω_m . This creates a measurable privacy-robustness-utility trade-off rather than assuming that privacy protection is cost free.

Theorem 2. Assume that

$$\|(\mathbf{I} - \mathbf{P}_d)\mathbf{d}_{i,t}\|_2 \leq \xi_d, \quad (81)$$

and let the minimum and maximum compensation activations be $g_{\min}$ and $g_{\max}$, respectively. Let $\mathcal{N}(\cdot)$ denote the LayerNorm mapping in Equation (47). Assume that, over the bounded latent domain encountered during inference,

$$\|\mathcal{N}(\mathbf{a}) - \mathcal{N}(\mathbf{b})\|_2 \leq L_{\text{LN}}\|\mathbf{a} - \mathbf{b}\|_2, \quad (82)$$

for a finite constant $L_{\text{LN}} > 0$, and that the task-relevant latent state is represented in the same normalized

coordinate system, i.e., $\mathcal{N}(\mathbf{s}_{i,t}) = \mathbf{s}_{i,t}$. Define

$$\chi_{i,t} = \|\mathbf{I} - \mathbf{G}_{i,t}\mathbf{P}_d\|_2 \leq 1 + g_{\max}. \quad (83)$$

Under the historical-reference bound in Equation (76), the compensated-state error satisfies

$$\begin{aligned} \|\mathbf{h}_{i,t}^c - \mathbf{s}_{i,t}\|_2 &\leq L_{\text{LN}} \left[(1 - g_{\min}) \|\mathbf{d}_{i,t}\|_2 + \xi_d + \chi_{i,t} \|\boldsymbol{\epsilon}_{i,t}\|_2 \right. \\ &\quad \left. + g_{\max} (\varepsilon_r + \varepsilon_p + 2B_r\omega_m) \right]. \end{aligned} \quad (84)$$

Proof. Define the pre-normalization compensated state as

$$\bar{\mathbf{h}}_{i,t}^c = \mathbf{h}_{i,t}^0 - \mathbf{G}_{i,t}\mathbf{P}_d(\mathbf{h}_{i,t}^0 - \mathbf{r}_{i,t}). \quad (85)$$

Then $\mathbf{h}_{i,t}^c = \mathcal{N}(\bar{\mathbf{h}}_{i,t}^c)$. Using Equation (82) and $\mathcal{N}(\mathbf{s}_{i,t}) = \mathbf{s}_{i,t}$ gives

$$\|\mathbf{h}_{i,t}^c - \mathbf{s}_{i,t}\|_2 \leq L_{\text{LN}} \|\bar{\mathbf{h}}_{i,t}^c - \mathbf{s}_{i,t}\|_2. \quad (86)$$

Substituting Equation (5) yields

$$\begin{aligned} \bar{\mathbf{h}}_{i,t}^c - \mathbf{s}_{i,t} &= (\mathbf{I} - \mathbf{G}_{i,t}\mathbf{P}_d)\mathbf{d}_{i,t} \\ &\quad + (\mathbf{I} - \mathbf{G}_{i,t}\mathbf{P}_d)\boldsymbol{\epsilon}_{i,t} \\ &\quad + \mathbf{G}_{i,t}\mathbf{P}_d(\mathbf{r}_{i,t} - \mathbf{s}_{i,t}). \end{aligned} \quad (87)$$

The drift term can be decomposed as

$$(\mathbf{I} - \mathbf{G}_{i,t})\mathbf{P}_d\mathbf{d}_{i,t} + (\mathbf{I} - \mathbf{P}_d)\mathbf{d}_{i,t}. \quad (88)$$

Since the gate entries lie in $[g_{\min}, g_{\max}]$,

$$\|(\mathbf{I} - \mathbf{G}_{i,t})\mathbf{P}_d\mathbf{d}_{i,t}\|_2 \leq (1 - g_{\min}) \|\mathbf{d}_{i,t}\|_2. \quad (89)$$

Moreover, Equation (83) gives

$$\|(\mathbf{I} - \mathbf{G}_{i,t}\mathbf{P}_d)\boldsymbol{\epsilon}_{i,t}\|_2 \leq \chi_{i,t} \|\boldsymbol{\epsilon}_{i,t}\|_2, \quad (90)$$

while $\|\mathbf{G}_{i,t}\mathbf{P}_d\|_2 \leq g_{\max}$. Combining these inequalities with the assumed projection error, Equation (76), and the triangle inequality yields Equation (84). $\square$

Compared with the uncorrected representation

$$\|\mathbf{h}_{i,t}^0 - \mathbf{s}_{i,t}\|_2 \leq \|\mathbf{d}_{i,t}\|_2 + \|\boldsymbol{\epsilon}_{i,t}\|_2, \quad (91)$$

HSDC-Net provides a smaller upper bound whenever

$$\begin{aligned} L_{\text{LN}} \left[(1 - g_{\min}) \|\mathbf{d}_{i,t}\|_2 + \xi_d + \chi_{i,t} \|\boldsymbol{\epsilon}_{i,t}\|_2 \right. \\ \left. + g_{\max} (\varepsilon_r + \varepsilon_p + 2B_r\omega_m) \right] < \|\mathbf{d}_{i,t}\|_2 + \|\boldsymbol{\epsilon}_{i,t}\|_2. \end{aligned} \quad (92)$$

Therefore, the benefit of compensation depends not only on the removable nuisance drift and reference quality but also on the local Lipschitz factor of LayerNorm, rather than assuming that the normalization operation is non-expansive.

Under the standard assumptions that $\mathcal{L}$ is lower bounded, has an L -Lipschitz gradient, and is optimized using unbiased stochastic gradients with bounded variance σ^2 , stochastic gradient descent with $\eta \leq 1/L$ satisfies

$$\frac{1}{N} \sum_{n=0}^{N-1} \mathbb{E} [\|\nabla \mathcal{L}(\Theta_n)\|_2^2] \leq \frac{2[\mathcal{L}(\Theta_0) - \mathcal{L}^*]}{\eta N} + L\eta\sigma^2. \quad (93)$$

Selecting

$$\eta = \mathcal{O}(N^{-1/2}) \quad (94)$$

gives an $\mathcal{O}(N^{-1/2})$ convergence rate to a first-order stationary point.

Let d denote the latent dimension, d_c the protected context dimension, K the global anchor capacity, r_a the context-adaptor rank, r_d the drift-subspace rank, d_ϕ the random-feature dimension, and w_d the drift-estimation window. The local online complexity per observation is

$$\mathcal{O}(d^2 + Kdr_a + Kd + Kd_c + dr_d + w_d d_\phi). \quad (95)$$

Let B_t denote the number of anchor candidates processed during one gateway update and I_g denote the number of iterations used by the geometric-median solver. Gateway-side trust evaluation requires

$$\mathcal{O}(I_g B_t d + B_t K d) \quad (96)$$

operations in the worst case, while this computation is executed only during bounded anchor-update rounds rather than for every raw sensor sample.

The memory complexity of an edge node is

$$\mathcal{O}(Kd + Kd_c + dr_a + dr_d), \quad (97)$$

and a transmitted anchor packet requires

$$\mathcal{O}(d + d_c) \quad (98)$$

floating-point values, independent of the length of the original production sequence.

Because K , r_a , r_d , d_c , and w_d are bounded, neither local inference memory nor communication size grows with the complete production-history length. Raw sensor sequences and manufacturing images remain local, while only compact privacy-protected latent prototypes are exchanged. HSDC-Net therefore retains the bounded-memory and linear-sequence characteristics of the original historical-state compensation framework while extending it to privacy-aware and poisoning-resilient distributed industrial edge monitoring.

4. Experiment

4.1. Experimental Settings

All experiments are conducted on a workstation with an Intel Core i9-13900K CPU, 64 GB RAM,

and an NVIDIA RTX 4090 GPU using Ubuntu 22.04, Python 3.10, PyTorch 2.2.2, and CUDA 12.1. Four industrial datasets are used: AMMT for melt-pool response prediction, TEP for fault recognition, NASA Milling Wear for tool-wear estimation, and Aluminum Lap-Joint Fatigue Crack Growth for crack-state prediction. For the prediction, drift-robustness, and efficiency experiments, in dependent builds, runs, cutters, or specimens are assigned to four logical edge nodes, with each trajectory stored locally and only privacy-protected historical anchors transmitted to the coordinating gateway. For the poisoning-robustness experiment, each of these four local partitions is further divided into ten non-overlapping authenticated logical edge clients, yielding $M_{\text{sec}} = 40$ clients; consequently, compromised ratios of 10%, 20%, and 40% correspond to 4, 8, and 16 compromised clients, respectively. Samples are chronologically divided into 60%, 20%, and 20% training, validation, and testing sets, and trajectories from the same physical instance do not cross partitions. Controlled benign shifts include sensor bias, gain variation, baseline drift, Gaussian noise, and random missingness at three severity levels. HSDC-Net is compared with LSTM [20], Informer [21], PatchTST [22], and ODEStream [23] using identical partitions, preprocessing, input windows, and prediction heads. HSDC-Net uses a latent dimension of 128, anchor capacity of 32, drift window of 64, context-adaptor rank of 8, drift-subspace rank of 16, and anchor-weight temperature of 0.5; the protected context dimension is $d_c = 16$, and the default per-release privacy setting is $\epsilon_p = 4$ and $\delta_p = 10^{-5}$. All models are optimized with AdamW using a batch size of 64, learning rate 3×10^{-4} , weight decay 10^{-4} , and early stopping. Regression results are reported using RMSE and R^2 , with MAE additionally reported for Milling Wear; classification uses Macro-F1 and AUROC. Drift robustness is evaluated using Performance Retention and feature-drift suppression rate (FDSR), while security robustness uses malicious-anchor acceptance rate (MAAR) and Trust Gap. Inference latency and peak GPU memory characterize online efficiency. All experiments are repeated with five random seeds and reported as mean $\pm$ standard deviation. No inferential p -value is reported from the five seeds.

For reproducibility, let Q^0 and Q^s denote the clean and shifted/attacked task metrics, respectively, and let D^0 and D^c denote the mean latent drift discrepancies before and after compensation under the same perturbation. The principal robustness metrics

are defined as

$$\begin{aligned} \text{PR} &= \begin{cases} 100 \frac{Q^s}{Q^0 + \varepsilon} \%, & \text{higher-is-better metric,} \\ 100 \frac{Q^0}{Q^s + \varepsilon} \%, & \text{lower-is-better metric,} \end{cases} \\ \text{FDSR} &= 1 - \frac{\overline{D}^c}{\overline{D}^0 + \varepsilon}, \quad \text{MAAR} = \frac{N_{\text{mal}}^{\text{acc}}}{N_{\text{mal}}^{\text{sub}} + \varepsilon}, \\ \text{TrustGap} &= \frac{1}{|\mathcal{H}|} \sum_{i \in \mathcal{H}} T_i - \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} T_i, \end{aligned} \quad (99)$$

where PR denotes Performance Retention, $N_{\text{mal}}^{\text{acc}}$ and $N_{\text{mal}}^{\text{sub}}$ are the numbers of admitted and submitted malicious anchors, and $\mathcal{H}$ and $\mathcal{M}$ are the benign and compromised source sets. For communication accounting, $\tilde{\mathbf{m}}_{i,k} \in \mathbb{R}^{128}$ and $\tilde{\mathbf{e}}_{i,k} \in \mathbb{R}^{16}$ are transmitted in FP32, requiring $(128 + 16) \times 4 = 576$ bytes. The timestamp $\tau_{i,k}$, uncertainty $u_{i,k}$, sample count $n_{i,k}$, and source identifier i each use one 32-bit field, adding 16 bytes. Thus, one anchor packet requires 592 bytes, i.e., 0.578 KiB, reported as 0.58 KB in Table 2.

4.2. Overall Distributed Process-Monitoring Performance

The first experiment evaluates the basic predictive performance of HSDC-Net when manufacturing trajectories are distributed across independent edge nodes and raw data remain local. All methods use identical distributed partitions and task-specific prediction settings.

As reported in Table 1, HSDC-Net achieves the best results across all four datasets. Compared with ODEStream, it reduces AMMT RMSE from 0.151 to 0.132, increases TEP Macro-F1 from 0.919 to 0.941, reduces Milling Wear MAE from 0.076 to 0.064, and reduces Aluminum Lap-Joint RMSE from 0.206 to 0.181. LSTM, Informer, and PatchTST model temporal dependencies but do not explicitly correct the validity of historical representations under distributed non-stationarity, while ODEStream performs online adaptation without separating nuisance drift from task-relevant degradation. HSDC-Net instead reuses privacy-protected historical anchors according to temporal, contextual, uncertainty, and source-reliability information and applies correction only within the learned nuisance-drift subspace. The results show that useful historical information can be shared across edge nodes without centralizing the original manufacturing streams.

4.3. Robustness under Progressive Feature Drift

The second experiment evaluates robustness to benign non-stationarity by introducing sensor bias, gain variation, baseline drift, Gaussian noise, and missingness at low, medium, and high severity levels. All edge nodes

Table 1. Overall process-monitoring performance of different methods on four industrial datasets. The best result is highlighted in bold and the second-best result is underlined. Values are reported as mean $\pm$ standard deviation over five random seeds; no inferential significance marker is used

Method	AMMT		TEP		Milling Wear		Aluminum Lap-Joint	
	RMSE $\downarrow$	R^2 $\uparrow$	Macro-F1 $\uparrow$	AUROC $\uparrow$	MAE $\downarrow$	R^2 $\uparrow$	RMSE $\downarrow$	R^2 $\uparrow$
LSTM	0.184 $\pm$ 0.006	0.842 $\pm$ 0.009	0.884 $\pm$ 0.008	0.931 $\pm$ 0.006	0.091 $\pm$ 0.004	0.817 $\pm$ 0.011	0.237 $\pm$ 0.009	0.793 $\pm$ 0.013
Informer	0.169 $\pm$ 0.005	0.861 $\pm$ 0.008	0.901 $\pm$ 0.007	0.944 $\pm$ 0.005	0.085 $\pm$ 0.004	0.836 $\pm$ 0.010	0.224 $\pm$ 0.008	0.811 $\pm$ 0.012
PatchTST	0.158 $\pm$ 0.005	0.874 $\pm$ 0.007	0.913 $\pm$ 0.006	0.951 $\pm$ 0.005	0.079 $\pm$ 0.003	0.851 $\pm$ 0.009	0.211 $\pm$ 0.007	0.829 $\pm$ 0.010
ODEStream	0.151 $\pm$ 0.004	0.881 $\pm$ 0.006	0.919 $\pm$ 0.005	0.956 $\pm$ 0.004	0.076 $\pm$ 0.003	0.858 $\pm$ 0.008	0.206 $\pm$ 0.006	0.837 $\pm$ 0.009
HSDC-Net	0.132 $\pm$ 0.003	0.909 $\pm$ 0.005	0.941 $\pm$ 0.004	0.972 $\pm$ 0.003	0.064 $\pm$ 0.002	0.889 $\pm$ 0.006	0.181 $\pm$ 0.005	0.872 $\pm$ 0.007

remain benign in this experiment so that the reported degradation reflects benign distribution shift rather than poisoned-anchor injection.

As shown in Fig. 4, HSDC-Net consistently maintains the highest performance retention as drift severity increases. Under high drift, it retains 92.1% performance on AMMT, 93.5% on TEP, 91.4% on Milling Wear, and 90.2% on Aluminum Lap-Joint. These results indicate that HSDC-Net can suppress benign distribution variation while preserving task-relevant process evolution across heterogeneous edge nodes. Unlike conventional sequence models that retain drifted historical statistics, HSDC-Net evaluates context-compatible historical anchors and restricts compensation to the learned nuisance-drift subspace, thereby handling ordinary non-stationarity without unnecessarily activating security-oriented source rejection.

4.4. Efficiency–Drift Suppression Trade-Off

The third experiment evaluates whether HSDC-Net achieves stronger drift suppression without excessive online inference overhead. Average latency and FDSR are jointly compared, with the upper-left region of Fig. 5 indicating a better efficiency–robustness trade-off.

As illustrated in Fig. 5, HSDC-Net consistently achieves high FDSR with moderate inference latency. It obtains FDSR/latency pairs of 0.724/4.86 ms on AMMT, 0.751/3.72 ms on TEP, 0.708/4.08 ms on Milling Wear, and 0.694/4.29 ms on Aluminum Lap-Joint, substantially outperforming ODEStream in both drift suppression and latency. Although LSTM has lower computational cost, its FDSR remains only 0.158–0.196. The efficiency of HSDC-Net results from its linear-complexity temporal encoder, bounded anchor bank, low-rank context transport, and low-dimensional drift compensation. Privacy perturbation and source-trust evaluation are executed only during anchor-update rounds, so the added security mechanisms do not substantially increase per-window inference cost.

4.5. Security Robustness and Privacy–Utility Analysis

The fourth experiment evaluates the robustness of HSDC-Net against poisoned historical-anchor sharing in distributed edge networks and examines the utility cost introduced by privacy protection. The $M_{\text{sec}} = 40$ authenticated logical edge clients defined in Section 4.1 are used here, and 4, 8, or 16 clients are randomly designated as compromised, corresponding to compromised-client ratios of 10%, 20%, and 40%, respectively. HSDC-Net is compared with variants without source-trust weighting and without differential privacy.

As shown in Table 2, source-trust evaluation is critical for resisting poisoned historical anchors. When the compromised-client ratio increases from 10% to 40%, HSDC-Net maintains performance retention from 98.1% to 90.4%, whereas HSDC-Net w/o Trust decreases from 94.2% to 74.8%. The complete model also reduces MAAR from 0.238 to 0.056 and produces a Trust Gap of 0.603, indicating that inconsistent malicious sources are effectively separated from benign nodes and prevented from dominating historical-reference construction. HSDC-Net w/o DP provides slightly higher robustness, reaching 91.2% retention at 40% compromised clients, but its $\epsilon_p = \infty$ indicates the absence of differential-privacy protection. In comparison, HSDC-Net uses a finite per-release privacy parameter of $\epsilon_p = 4$ with less than one percentage point of performance loss relative to the non-private variant. With $d = 128$, $d_c = 16$, FP32 representation fields, and four 32-bit metadata fields, each transmitted packet contains 592 bytes (0.578 KiB), which is reported as 0.58 KB. Thus, the proposed privacy and trust mechanisms provide a favorable security–privacy–utility trade-off without increasing communication size.

5. Conclusion

This paper proposed HSDC-Net, a privacy-preserving and trust-aware historical-state decay compensation framework for secure distributed industrial edge monitoring under non-stationary operating conditions. HSDC-Net keeps raw sensor and visual streams at

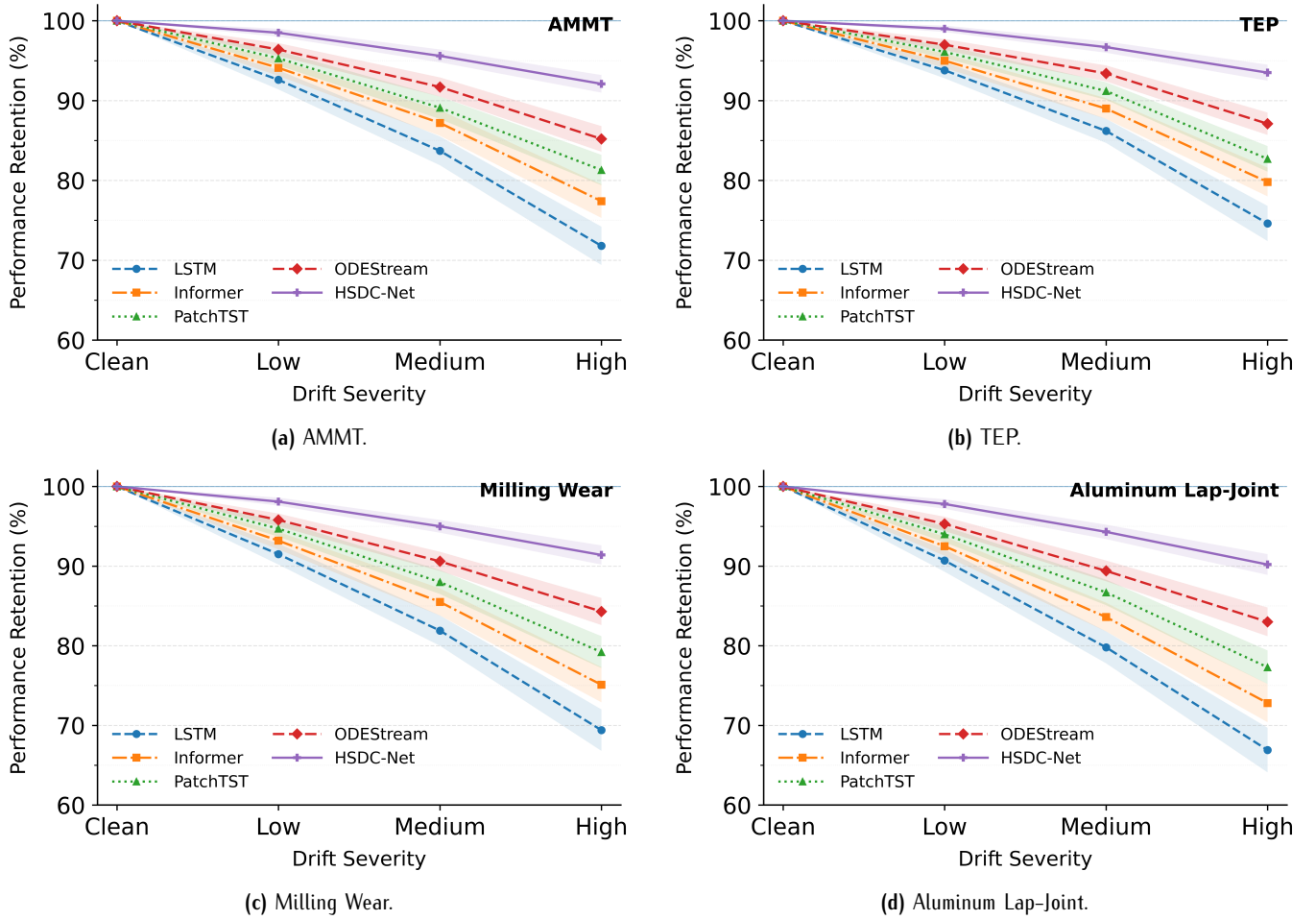


Figure 4. Performance retention of different methods under progressively intensified benign feature drift in the distributed edge setting. The curves represent the mean results over five independent runs, and the shaded regions denote the corresponding standard deviations

Table 2. Security robustness and privacy-utility performance of HSDC-Net under distributed historical-anchor sharing. The privacy parameter ϵ_p is specified per released anchor

Method	Performance Retention under Compromised Edge Clients			MAAR ↓	Trust Gap ↑	ϵ_p /release	Comm. / Anchor ↓
	10%	20%	40%				
HSDC-Net w/o Trust	94.2%	87.5%	74.8%	0.238	0.000	4	0.58 KB
HSDC-Net w/o DP	98.8%	96.3%	91.2%	0.047	0.624	∞	0.58 KB
HSDC-Net	98.1%	95.6%	90.4%	0.056	0.603	4	0.58 KB

local edge nodes and exchanges only bounded privacy-perturbed historical anchors, while cross-node consistency and recursively updated source trust prevent unreliable or manipulated anchors from dominating the shared memory. Adaptive historical-state decay further combines temporal age, context discrepancy, uncertainty, and source reliability, and a low-rank context adapter transports reliable historical information toward the current operating condition. Instead of

directly forcing current representations toward historical references, the proposed nuisance-drift projection restricts compensation to task-irrelevant variation and preserves genuine equipment degradation and process evolution. Experiments on four industrial datasets showed consistent improvements in prediction accuracy and robustness to progressive benign drift while maintaining moderate online latency. Security experiments further demonstrated that source-trust evaluation substantially suppresses malicious-anchor acceptance and preserves monitoring performance as the compromised-node ratio increases, while the finite

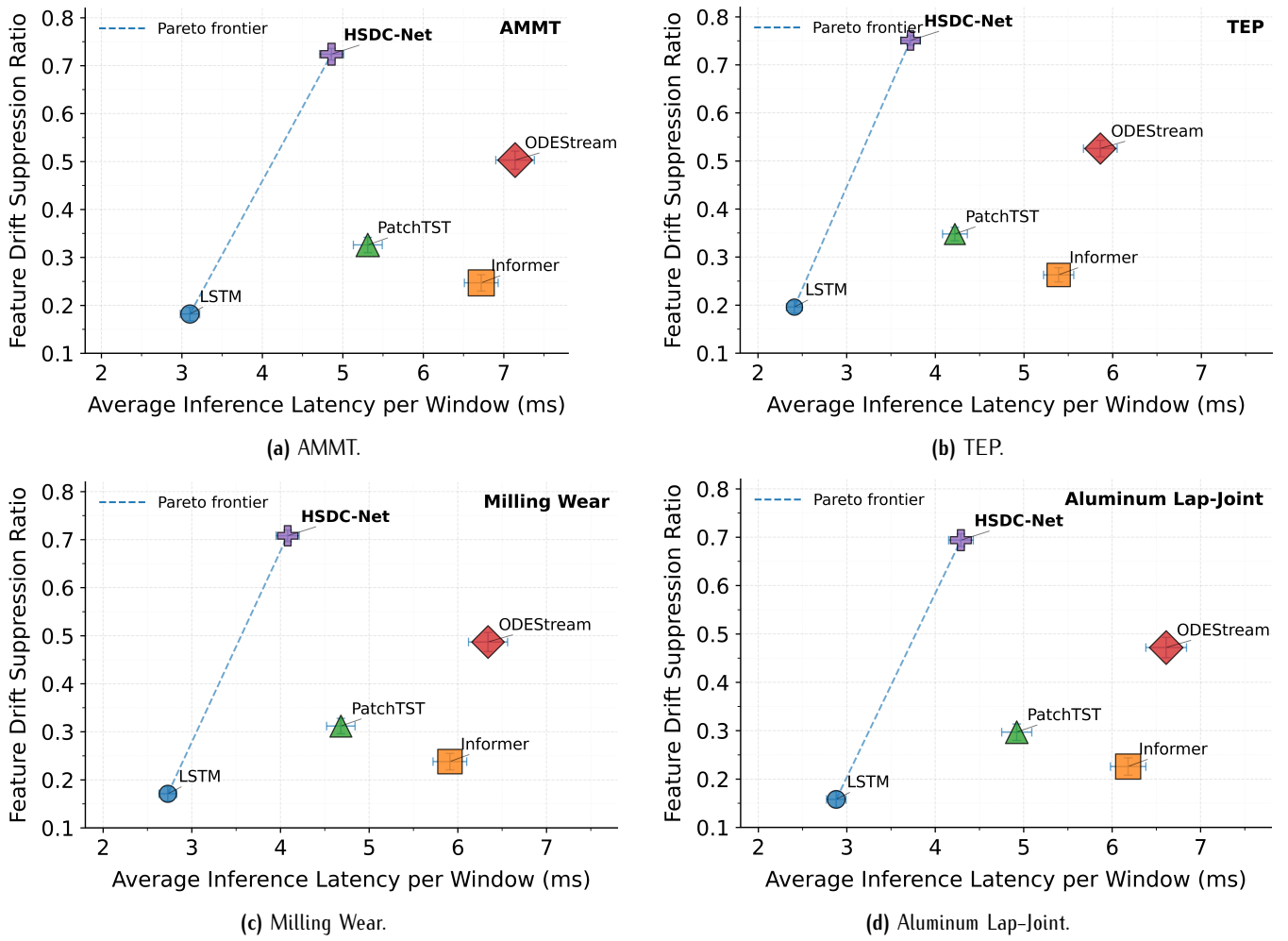


Figure 5. Efficiency-drift suppression trade-off of different methods on the four datasets under distributed edge monitoring. The horizontal and vertical error bars denote the standard deviations of inference latency and FDSR over five independent runs, respectively. Marker size represents peak GPU memory consumption, and the dashed line indicates the Pareto frontier considering lower latency and higher drift suppression

privacy setting introduces only limited degradation relative to the non-private variant without increasing transmitted-anchor size. The theoretical analysis additionally characterizes temporal reliability, trust dependence, prototype privacy, and the influence of privacy noise and residual malicious anchors on compensation error. Nevertheless, the current study evaluates distributed operation through logically partitioned industrial datasets and assumes authenticated communication and a coordinating gateway following the prescribed protocol; repeated private releases also require explicit privacy-budget management, and trust estimation may become more difficult when benign nodes exhibit strongly heterogeneous contexts or multiple malicious sources collude. Future work will therefore investigate adaptive privacy-budget allocation, stronger Byzantine-robust anchor aggregation, collusion-aware trust modeling, secure aggregation, and deployment

across physically distributed industrial edge devices and production sites.

References

- [1] D. Baby, B. Han, S. Zhang, C. Hu, B. Wang, Y.-X. Wang, Adapting to online distribution shifts in deep learning: A black-box approach, in: International Conference on Artificial Intelligence and Statistics, PMLR, 2025, pp. 3853–3861.
- [2] F. Regol, L. Schwinn, K. Sprague, M. Coates, T. Markovich, When to retrain a machine learning model, in: International Conference on Machine Learning, PMLR, 2025, pp. 51369–51404.
- [3] J.-H. Lee, Bayesian weight enhancement with steady-state adaptation for test-time adaptation in dynamic environments, in: International Conference on Machine Learning, PMLR, 2025, pp. 32889–32906.
- [4] J. Liu, N. Seedat, P. Cui, M. van der Schaar, [Going beyond static: Understanding shifts with time-series attribution](#),

- in: The Thirteenth International Conference on Learning Representations, 2025.
URL <https://openreview.net/forum?id=XQ1ccqJpCC>
- [5] L. Lin, X. Zhang, Q. Zhang, S. Wang, K. Xia, Ortcl: Towards continual learning of time series foundation models on streaming data via orthogonal rotation, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 40, 2026, pp. 23541–23549.
 - [6] D. Liang, J. Chen, X. Wang, Y. Wang, S. Li, Deepboots: Dual-stream residual boosting for drift-resilient time-series forecasting, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 40, 2026, pp. 23391–23400.
 - [7] J. Chapelin, A. Voisin, B. Rose, B. Iung, L. Steck, L. Chaves, M. Lauer, O. Jotz, Data-driven drift detection and diagnosis framework for predictive maintenance of heterogeneous production processes: Application to a multiple tapping process, Engineering Applications of Artificial Intelligence 139 (2025) 109552.
 - [8] M. A. Benatia, M. Hafsi, S. B. Ayed, A continual learning approach for failure prediction under non-stationary conditions: Application to condition monitoring data streams, Computers & Industrial Engineering 204 (2025) 111049.
 - [9] D. M. V. Sato, E. Ruschel, E. E. Scalabrin, E. d. F. R. Loures, E. A. P. Santos, Interactive process drift detection (ipdd) for condition-based maintenance using process mining, Advanced Engineering Informatics 66 (2025) 103397.
 - [10] S. Li, B. Zhou, J. Shang, X. Chen, J. Yu, Review of recent applications and future perspectives on process monitoring approaches in industrial processes, Journal of Manufacturing Systems 82 (2025) 509–530.
 - [11] T. Özel, Deep learning-based applications in metal additive manufacturing processes: Challenges and opportunities—a review, International Journal of Lightweight Materials and Manufacture 8 (4) (2025) 453–468.
 - [12] S. Oueslati, M. Ritou, E. Paquet, F. Belkadi, P. Le Bot, Prediction of process instability by waam in-process monitoring and ctwd drift estimation, Additive Manufacturing (2026) 105127.
 - [13] Y. Mao, J. Yan, X. Lin, C. K. Chua, K. Zhu, A physics-guided multi-source data fusion model for lpb process monitoring: From data generation to melting state recognition, Additive Manufacturing (2026) 105200.
 - [14] A. Caggiano, G. Mattera, Y. Zhang, R. Teti, Heat input control and deep learning-based indirect measure of process and deposition stability in wire arc additive manufacturing, CIRP Annals (2026).
 - [15] S. Oueslati, E. Paquet, F. Belkadi, M. Ritou, P. Le Bot, New monitoring criteria for instability detection by machine learning in additive manufacturing: Application to wire arc additive manufacturing process, IFAC-PapersOnLine 59 (10) (2025) 2250–2255.
 - [16] J. Xie, M. Safdar, A. M. Romascanu, Y. Lu, H. Ko, Z. Yang, Y. F. Zhao, Towards reproducible machine learning-based process monitoring and quality prediction research for additive manufacturing, in: International design engineering technical conferences and computers and information in engineering conference, Vol. 88346, American Society of Mechanical Engineers, 2024, p. V02AT02A033.
 - [17] Q. Liu, R. Yin, X. Guo, W. Wang, Z. Gao, M. Jia, Y. Liu, Locally spatiotemporal soft sensor for key indicator prediction in cement production process, Chemical Engineering Science 307 (2025) 121386.
 - [18] J. Li, X. Lin, G. S. Hong, K. Zhu, Uncovering nonequilibrium dynamics in machining via mechanism-assisted machine learning, npj Advanced Manufacturing 2 (1) (2025) 33.
 - [19] G. Mattera, R. Mattera, L. Nele, A forecast-assisted approach to remaining useful life prediction: A predictive maintenance case study in hybrid al/cfrp stack drilling, International Journal of Production Research 64 (9) (2026) 3400–3422.
 - [20] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Computation 9 (8) (1997) 1735–1780. doi:10.1162/neco.1997.9.8.1735.
 - [21] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, W. Zhang, Informer: Beyond efficient transformer for long sequence time-series forecasting, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 11106–11115. doi:10.1609/aaai.v35i12.17325.
 - [22] Y. Nie, N. H. Nguyen, P. Sinthong, J. Kalagnanam, A time series is worth 64 words: Long-term forecasting with transformers, in: The Eleventh International Conference on Learning Representations, 2023.
URL <https://openreview.net/forum?id=Jbdc0vT0co1>
 - [23] F. M. Abushaqra, H. Xue, Y. Ren, F. D. Salim, ODEStream: A buffer-free online learning framework with ODE-based adaptor for streaming time series forecasting, Transactions on Machine Learning Research (2025).
URL <https://openreview.net/forum?id=TW0TKhwU5n>