electronics

MDPI

Article

HCUNet: Hierarchical Consistency-Based Uncertainty Quantification Network for UAV Fault Diagnosis

Ruiqi Rao [1], Feng Jiang [1,*] and Xi Zha [2]

[1] School of Artificial Intelligence, Nanjing University of Information Science and Technology, Nanjing 210044, China; 202383460016@nuist.edu.cn
[2] School of Mathematics and Statistics, Nanjing University of Information Science and Technology, Nanjing 210044, China; 202413140031@nuist.edu.cn
* Correspondence: fjiang@nuist.edu.cn

Abstract

Unmanned aerial vehicle (UAV) fault diagnosis is critical for ensuring flight safety, yet existing methods face challenges from class-imbalanced datasets and the lack of reliable uncertainty estimates. We address these issues with HCUNet, a hierarchical consistency-based uncertainty quantification network that integrates multi-level feature aggregation with principled uncertainty estimation. The framework extracts complementary representations through three sequential stages. First, an adaptive data-driven multi-scale convolution module captures shallow multi-frequency patterns. Second, a hierarchical context bridging LSTM encodes middle-level temporal dynamics. Third, a rotary position embedding enhanced attention mechanism extracts deep semantic features. These hierarchical representations are fused by a latent-aware semantic extraction and aggregation module, which employs cross-scale dilated convolutions and random pooling sampling for robust feature integration. A single-pass mechanism disentangles epistemic and aleatoric uncertainties via prediction disagreement and feature divergence, eliminating the need for ensemble models. The hierarchical consistency loss jointly optimizes multi-level classification accuracy, inter-level agreement, and confidence calibration. Experiments on the ALFA dataset demonstrate that HCUNet achieves 99.55% accuracy, 99.12% F1-score, and 0.0024 expected calibration error, outperforming the strongest baseline by 0.66% in accuracy and 64.2% in calibration error reduction.

Keywords: UAV fault diagnosis; uncertainty quantification; hierarchical consistency; deep learning

Check for updates

Academic Editor: Ahmed Abu-Siada

Received: 8 June 2026
Revised: 11 July 2026
Accepted: 20 July 2026
Published: 22 July 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Unmanned aerial vehicles (UAVs) are now deployed in a wide range of missions—from surveillance, agriculture, and disaster response to package delivery. As these platforms move into safety-critical roles, reliable fault diagnosis becomes essential for flight safety and mission success. Yet UAV fault diagnosis presents several inherent challenges. Sensor measurements are routinely corrupted by environmental noise and measurement errors. Fault events are typically rare compared to normal operation, creating severely imbalanced datasets. Moreover, different fault types unfold across multiple time scales, from rapid transients on the order of milliseconds to gradual degradation trends that extend over minutes. Together, these factors make accurate and dependable fault diagnosis a demanding task.

Early fault diagnosis relied on model-based methods such as Kalman filters [1] and observers [2], which require accurate system models that are difficult to obtain under diverse UAV operating conditions. Signal processing techniques including wavelet transform [3] and empirical mode decomposition [4] combined with traditional classifiers [5,6] reduced modeling requirements but remained limited by hand-crafted features [7]. Deep learning fundamentally transformed this field by enabling automatic feature learning [8]: convolutional neural networks (CNNs) extract hierarchical features from raw signals [9,10], recurrent architectures model temporal dependencies [11,12], and attention mechanisms capture long-range relationships [13,14]. However, these methods produce deterministic predictions without quantifying diagnostic confidence—a critical limitation for safety-critical applications [15].

Bayesian approaches such as MC Dropout [16] and Deep Ensembles [17] have been used to model uncertainty, but their considerable computational cost makes them impractical for resource-constrained UAV platforms. Evidential deep learning [18] provides efficient single-pass uncertainty estimates; however, it still faces the fundamental challenge of separating epistemic uncertainty from aleatoric uncertainty [19]. This distinction is important because epistemic uncertainty reflects gaps in knowledge that can be reduced with more data, while aleatoric uncertainty captures irreducible noise. Moreover, existing methods often do not adequately handle the severe class imbalance in real-world UAV datasets [20,21], where standard classifiers are biased toward majority classes and neglect rare but safety-critical fault types.

To address these challenges, we propose HCUNet, a hierarchical consistency-based uncertainty quantification network for UAV fault diagnosis. Its core idea is to exploit the complementary information captured at different network depths through a three-level feature aggregation architecture. Shallow features extracted after multi-scale convolution capture local waveform patterns, middle features from the hierarchical context bridging LSTM (HCB-LSTM) encode temporal dynamics and sequential dependencies, and deep features produced by attention mechanisms represent high-level semantic categories. Instead of processing these hierarchical representations independently, HCUNet employs a consistency-aware fusion mechanism that measures prediction agreement across the three depths. The central idea is that disagreement among the level predictions signals a lack of model knowledge, while divergence between their feature representations reflects inherent signal noise. To the best of our knowledge, this represents the first approach in UAV fault diagnosis to exploit hierarchical prediction consistency for epistemic uncertainty estimation, enabling principled uncertainty decomposition through the network's intrinsic structure. From a theoretical perspective, the three hierarchical prediction heads can be viewed as an implicit ensemble that share a common feature extraction backbone. Whereas Deep Ensemble trains multiple independent networks and aggregates their predictions to estimate epistemic uncertainty, HCUNet obtains analogous prediction diversity from a single network through its depthwise hierarchical heads. The disagreement among these heads therefore serves as a computationally efficient surrogate for ensemble variance, capturing epistemic uncertainty in a single forward pass and avoiding the substantial training and inference overhead of maintaining multiple independent models. Several technical innovations are introduced to realize this hierarchical consistency paradigm. Specifically, an adaptive data-driven multi-scale convolution module dynamically routes signals to scale-specific branches based on spectral characteristics. Apart from that, the model uses rotary position embedding enhanced attention to model long-range temporal dependencies. The latent-aware semantic extraction and aggregation module is reformulated from two-dimensional visual features to one-dimensional temporal sequences, performing cross-scale fusion with random pooling as implicit regularization. A hierarchical consistency loss

jointly optimizes classification accuracy, inter-level agreement, and confidence calibration. The main contributions of this work are as follows:

1. A hierarchical three-level feature aggregation architecture is proposed that systematically integrates shallow convolutional features, middle-level sequential features, and deep semantic features for comprehensive fault pattern characterization.
2. An ADMS-CSI module is developed to dynamically route input signals to scale-specific branches based on spectral characteristics, incorporating bidirectional cross-scale feature pyramids and temporal sparse gating for enhanced multi-frequency pattern extraction.
3. A consistency-based mechanism disentangles epistemic and aleatoric uncertainty via hierarchical prediction disagreement in a single efficient forward pass.
4. A LaSEA module is reformulated and adapted for one-dimensional temporal sequences, employing cross-scale dilated convolutions combined with random pooling sampling for implicit regularization. This module mitigates feature degradation for robust hierarchical fusion.

2. Related Work

We briefly review fault diagnosis, uncertainty quantification, and multi-scale temporal feature learning, focusing on aspects most relevant to HCUNet.

2.1. Fault Diagnosis Methods

Early work on UAV fault diagnosis employed model-based methods, such as Kalman filter-based techniques [1] and observer-based approaches [2]. These methods detect faults by comparing residual signals against predefined thresholds. However, building accurate dynamic models for UAVs operating under diverse conditions is challenging. To alleviate the modeling burden, signal processing techniques were introduced to extract fault-related features from time–frequency representations, including wavelet transform [3], empirical mode decomposition [4], and variational mode decomposition [22]. Combined with traditional classifiers such as support vector machines [5], random forests [6], and XGBoost [23], these approaches achieved reasonable accuracy. Although they alleviated some modeling difficulties and provided interpretable feature representations, their designs continued to rely heavily on hand-crafted features, which limited adaptability to complex, varying fault patterns.

This limitation motivated the adoption of deep learning, which enables end-to-end feature learning and has fundamentally transformed UAV fault diagnosis. CNNs demonstrated the ability to extract hierarchical features directly from raw waveforms [9,10]. The sequential nature of sensor data then drove the use of Long Short-Term Memory (LSTM) [11] and Gated Recurrent Unit (GRU) [12] networks to model long-term temporal dependencies, while bidirectional variants [24] captured both forward and backward context. Hybrid architectures combining CNNs with LSTMs [25] further integrated local feature extraction with temporal aggregation, leveraging their complementary strengths. More recently, attention mechanisms [13] and efficient variants such as Informer [14] have enabled direct modeling of relationships between arbitrary time steps. Recent studies have further explored physics-informed and interpretable fault diagnosis approaches, including heterogeneous multimodal fusion for rotating machinery [26], physical-information constrained large language models for imbalanced reactor data [27], quadratic neural networks under zero-fault-sample conditions [28], digital twin-based virtual-real sound-vibration interaction [29], and sound-vibration physical-information fusion constraints [30]. Despite these advances, the majority of existing methods still produce deterministic point estimates and do not quantify the uncertainty inherent in real-world fault diagnosis. Recent

work has explored uncertainty quantification using machine learning and Bayesian model averaging [31], while uncertainty propagation in UAV-mounted systems has also been studied [32]. However, systematic uncertainty decomposition within temporal sensor sequences for UAV fault diagnosis remains underexplored.

2.2. Uncertainty Quantification in Deep Learning

The absence of reliable uncertainty estimates represents a significant limitation of conventional deep learning methods, particularly for UAV fault diagnosis where overconfident misclassifications could lead to catastrophic failures. A nuanced understanding of uncertainty requires distinguishing between its two fundamental sources. Epistemic uncertainty, also known as model uncertainty, reflects limitations in our knowledge about the optimal model parameters, typically arising from insufficient or non-representative training data. Aleatoric uncertainty, or data uncertainty, captures inherent randomness in the observations that cannot be reduced even with infinite data [33]. Effective uncertainty quantification must account for both sources while ideally providing the means to separate their contributions.

The Bayesian framework [34] provides a principled approach to uncertainty quantification by treating network weights as random variables. Since exact posterior inference is computationally prohibitive, practical implementations rely on approximations. Monte Carlo Dropout [16] reinterprets dropout regularization as approximate variational inference, obtaining uncertainty estimates through multiple stochastic forward passes. Deep Ensembles [17] train multiple networks with different initializations and aggregate their predictions. Evidential deep learning [18] offers an efficient alternative that predicts Dirichlet distributions over class probabilities in a single forward pass, naturally distinguishing between epistemic uncertainty from limited knowledge and aleatoric uncertainty from inherent data noise. However, effectively disentangling these two uncertainty types in temporal fault patterns remains challenging.

Recent model-based approaches have also addressed UAV fault diagnosis through parametric methods. For instance, Tutsoy et al. [35] proposed a robust reduced-order Thau observer integrated with an adaptive fault estimator for quadrotor UAVs, which provides explicit fault magnitude quantification. Such parametric methods offer interpretable fault estimates grounded in physical dynamics but require accurate system models that may be difficult to obtain under diverse operating conditions. In contrast, non-parametric approaches like deep learning obviate the need for explicit dynamical models and can learn complex fault patterns directly from data. However, they typically lack the physical interpretability of model-based methods. HCUNet bridges this gap by incorporating hierarchical consistency constraints that implicitly encode structural priors, combining the flexibility of data-driven learning with principled uncertainty quantification.

2.3. Enhanced Temporal Representation Learning

Fault signatures manifest across multiple temporal scales, from rapid transients to gradual degradation trends. Dilated convolutions [36] provide an elegant mechanism to expand receptive fields exponentially without proportionally increasing parameters, enabling efficient multi-scale analysis. Parallel multi-scale architectures [37] process signals through branches with different temporal resolutions simultaneously.

The concept of hierarchical feature fusion, popularized by feature pyramid networks [38] in computer vision, has inspired analogous developments for sequential data. A particularly relevant advancement is the LaSEA module [39], which employs random pooling sampling to enhance discriminative features while attenuating noise—an approach that translates naturally to the temporal domain. Knowledge distillation [40] offers another

perspective on hierarchical learning, transferring rich representations from deeper layers to shallower ones. The potential of hierarchical consistency for uncertainty estimation, however, remains largely untapped.

Transformers lack inherent sensitivity to sequence order, necessitating explicit position encoding. Learned absolute embeddings [41] and sinusoidal encodings [13] represent early solutions, though both struggle with length generalization. Rotary position embedding (RoPE) [42] offers a more sophisticated approach by encoding relative positions through rotation operations applied to attention queries and keys. Its adoption in models like LLaMA [43] and Mistral [44] demonstrates superior sequence extrapolation. Integrating RoPE into fault diagnosis architectures presents an unexplored opportunity to enhance temporal modeling.

3. Methodology

3.1. Uncertainty Sources and Mathematical Formulation

In UAV fault diagnosis, uncertainty arises from two distinct sources that require separate treatment. Let $\mathbf{x}_t \in \mathbb{R}^D$ denote the sensor measurement at time t and $y \in \{1, \ldots, K\}$ the fault category.

Aleatoric uncertainty captures irreducible noise inherent in sensor measurements and fault manifestations. We model the observed signal as:

$$\mathbf{x}_t = \mathbf{x}_t^* + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\text{sensor}}) \tag{1}$$

where $\mathbf{x}_t^*$ represents the true system state and $\boldsymbol{\epsilon}_t$ denotes measurement noise with covariance $\boldsymbol{\Sigma}_{\text{sensor}}$. Additionally, fault boundaries exhibit intrinsic ambiguity when multiple fault modes produce overlapping signatures.

Epistemic uncertainty reflects limitations in model knowledge due to finite training data. For a neural network with parameters $\boldsymbol{\theta}$, the predictive distribution integrates over the posterior:

$$p(y|\mathbf{x}, \mathcal{D}) = \int p(y|\mathbf{x}, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) d\boldsymbol{\theta} \tag{2}$$

where $\mathcal{D}$ denotes the training set. Since exact integration is intractable, HCUNet approximates this through hierarchical prediction disagreement, where multiple network depths provide implicit samples from the functional posterior.

These uncertainties propagate through the diagnostic pipeline: sensor noise contributes to aleatoric uncertainty captured by feature divergence, while data scarcity for rare fault types manifests as epistemic uncertainty reflected in hierarchical prediction disagreement.

3.2. Overall Architecture

UAV systems are governed by coupled, nonlinear, time-varying dynamics subject to environmental disturbances and parametric uncertainties. Rather than relying on an explicit analytical dynamic model, HCUNet treats fault diagnosis as a data-driven sequence classification problem. Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T] \in \mathbb{R}^{T \times D}$ denote a window of multivariate sensor measurements, where each $\mathbf{x}_t \in \mathbb{R}^D$ jointly encodes the translational and rotational states of the six-degree-of-freedom rigid-body motion. Under a fault condition c, the measurement distribution shifts from the nominal one, i.e., $p(\mathbf{X} \mid c) \neq p(\mathbf{X} \mid \text{normal})$, producing anomalous spatio-temporal patterns across the coupled state channels. HCUNet learns a mapping $f_{\boldsymbol{\theta}} : \mathbf{X} \mapsto (\hat{y}, u_{\text{epi}}, u_{\text{alea}})$ that directly infers the fault category together with its associated uncertainties, while the hierarchical architecture implicitly captures the multi-scale temporal patterns corresponding to different coupled dynamic modes without requiring an explicit system identification step.

The complete framework of HCUNet is illustrated in Figure 1. The network adopts a hierarchical feature extraction paradigm that progressively captures multi-level temporal representations from flight sensor data. HCUNet employs a unified architecture integrating adaptive data-driven multi-scale convolution, hierarchical consistency learning, and uncertainty estimation based on hierarchical prediction consistency and feature divergence. With LaSEA-inspired cross-scale feature fusion and RoPE-based temporal attention, it effectively separates epistemic and aleatoric uncertainty while maintaining competitive diagnostic accuracy. Hierarchical prediction heads enable intermediate supervision and disagreement-based uncertainty estimation. Here, the hierarchical context bridging LSTM (HCB-LSTM) denotes a two-layer bidirectional LSTM that models forward and backward temporal dependencies over the multi-scale features produced by ADMS-CSI. Its output is projected back to the hidden dimension and residually bridged to the preceding shallow convolutional representation, so that local waveform cues are preserved while sequential context is injected—hence "context bridging." This bridged middle-level representation is then passed to the RoPE-based temporal attention for deep semantic encoding.

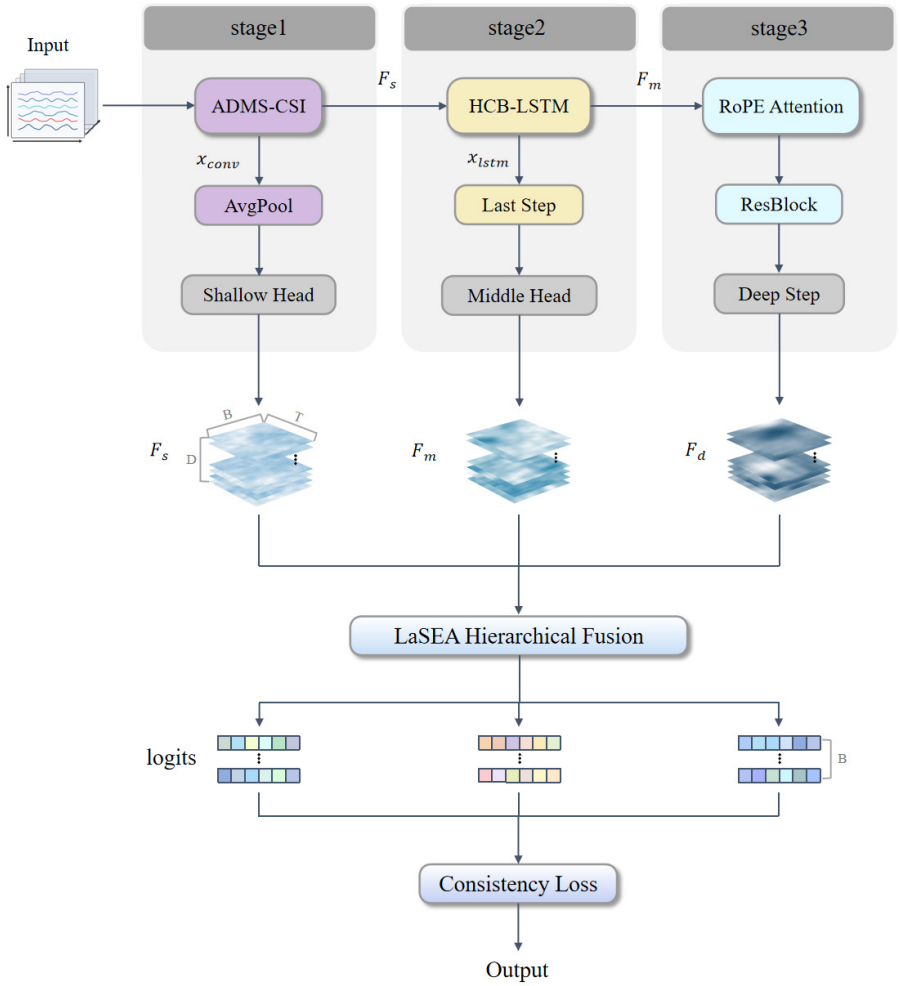


Figure 1. Complete framework of the proposed HCUNet. The input flight sequence is first processed through input normalization and projection, then passed through the ADMS-CSI module for multi-scale temporal feature extraction. Subsequently, a hierarchical context bridging LSTM captures sequential dependencies, followed by RoPE-based temporal attention for global context aggregation. The LaSEA module performs hierarchical fusion across shallow, middle, and deep features, producing the final classification with uncertainty estimates.

3.3. Adaptive Data-Driven Multi-Scale Convolution with Cross-Scale Interaction (ADMS-CSI)

3.3.1. Motivation for Adaptive Multi-Scale Convolution

Fault signals in UAV flight data span a wide frequency range. High-frequency vibrations require small receptive fields, while low-frequency drifts demand larger ones. Existing multi-scale methods use fixed weights and ignore signal-dependent patterns. To address this limitation, the ADMS-CSI module integrates five components, as shown in Figure 2. These components include the following four modules, with the extracted features being further refined by a fifth module. Here, "multi-level classification" refers to predictions at different network depths, where each level outputs a single-label prediction over K fault categories.

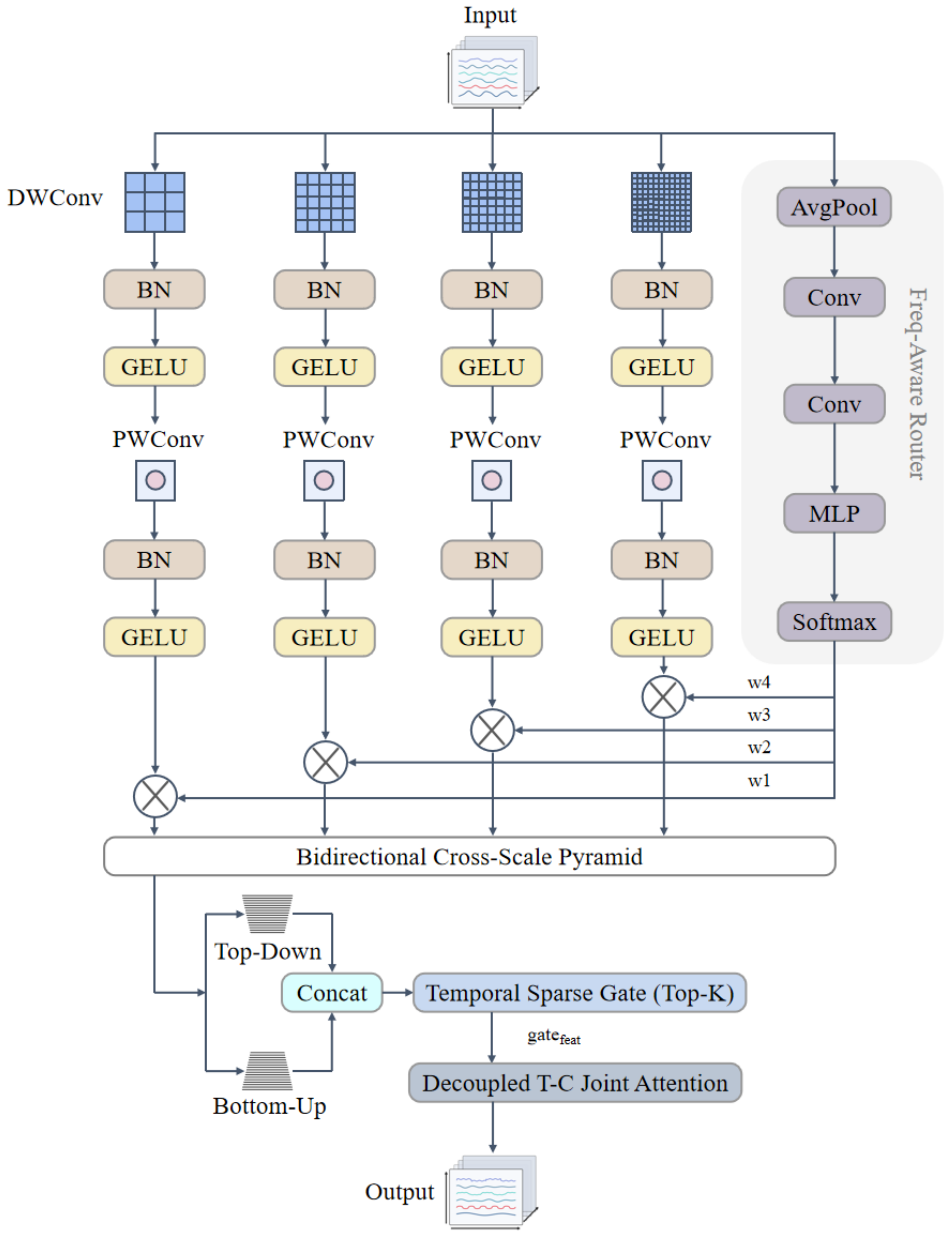


Figure 2. Architecture of the ADMS-CSI module, comprising data-driven scale routing, multi-scale depthwise separable convolutions, a bidirectional cross-scale feature pyramid, temporal sparse gating, and decoupled temporal–channel joint attention.

3.3.2. Data-Driven Adaptive Scale Routing

Given an input feature map $\mathbf{X} \in \mathbb{R}^{B \times C \times T}$, where B is the batch size, C is the number of channels, and T is the temporal sequence length, a lightweight spectral analyzer first extracts frequency-related features through two stacked convolutions over an adaptively pooled representation. The resulting frequency features are then mapped to data-driven scale weights $\mathbf{w}_{\text{data}} \in \mathbb{R}^{B \times S}$ via a two-layer MLP followed by softmax, where $S = 4$ is the number of scales.

To encode the prior that high-frequency content should elicit stronger responses at smaller scales, a learnable bias vector $\boldsymbol{\pi} \in \mathbb{R}^S$ is defined as:

$$\pi_i = \pi_{\max} - \frac{\Delta_\pi (i-1)}{S-1}, \quad i = 1, \ldots, S \tag{3}$$

where $\pi_{\max} = 1.0$ is the maximum prior weight assigned to the smallest scale, $\Delta_\pi = 0.5$ controls the decay rate, S denotes the total number of scales, and i indexes the scale branch in ascending order of kernel size. Smaller kernels receive higher initial weights to enhance high-frequency sensitivity.

Given the data-driven weights $\mathbf{w}^{\text{data}} \in \mathbb{R}^{B \times S}$ produced by the frequency analyzer, the final scale weights $\mathbf{w}^{\text{scale}} \in \mathbb{R}^{B \times S}$ are computed via elementwise modulation followed by normalization:

$$\tilde{w}_i = w_i^{\text{data}} \cdot \sigma(\pi_i), \quad w_i^{\text{scale}} = \frac{\tilde{w}_i}{\sum_{i=1}^{S} \tilde{w}_i + \epsilon} \tag{4}$$

where $\sigma(\cdot)$ denotes the sigmoid function, $\epsilon = 10^{-8}$ prevents division by zero, and w_i^{data} denotes the i-th element of $\mathbf{w}^{\text{data}} = \{w_i^{\text{data}}\}_{i=1}^{S}$. This formulation ensures differentiable routing while allowing strong data evidence to override the prior.

3.3.3. Multi-Scale Depthwise Separable Convolution

Different fault types exhibit distinct temporal patterns. Short-duration anomalies such as sudden vibrations require small kernels to capture rapid changes, while slow drifts need larger kernels to model long-term trends. To handle these diverse patterns, four parallel branches with kernel sizes $k \in \{3, 5, 7, 11\}$ are employed to extract features at different temporal resolutions. Each branch applies a depthwise separable convolution [45] to reduce computation while maintaining feature quality. When denoting the per-branch transformation by the composite operator $\phi_k = \text{GELU} \circ \text{BN} \circ \text{PW} \circ \text{GELU} \circ \text{BN} \circ \text{DW}_k$, where $\circ$ denotes function composition and DW_k and PW are the depthwise and pointwise convolutions, respectively, the output of each branch is written as

$$\mathbf{Y}_k = \phi_k(\mathbf{X}), \qquad \tilde{\mathbf{Y}}_k = w_{\text{scale},k}\, \mathbf{Y}_k \tag{5}$$

The scale-weighted output $\tilde{\mathbf{Y}}_k$ is then passed to the cross-scale pyramid.

3.3.4. Bidirectional Cross-Scale Feature Pyramid

Simply concatenating multi-scale features discards the structural relationships between scales. Inspired by feature pyramid networks [46], we employ a bidirectional pathway that propagates information both bottom-up and top-down. The bottom-up and top-down pathways are initialized as $\mathbf{F}_1^{\uparrow} = \tilde{\mathbf{Y}}_1$ and $\mathbf{F}_S^{\downarrow} = \mathbf{F}_S^{\uparrow}$, respectively. The propagation process is formulated as:

$$\mathbf{F}_i^{\uparrow} = \tilde{\mathbf{Y}}_i + \alpha_{\text{prop}} \cdot \text{Conv}(\mathbf{F}_{i-1}^{\uparrow}), \quad i = 2, \ldots, S \tag{6}$$

$$\mathbf{F}_i^{\downarrow} = \mathbf{F}_i^{\uparrow} + \alpha_{\text{prop}} \cdot \text{Conv}(\mathbf{F}_{i+1}^{\downarrow}), \quad i = S{-}1, \ldots, 1 \tag{7}$$

where $\alpha_{\text{prop}} = 0.5$ is the propagation coefficient that balances the contribution of adjacent scale features, ensuring stable gradient flow while preventing feature dominance from any single scale.

The bottom-up pass propagates fine-grained details from small to large scales, while the top-down pass transmits contextual information in the reverse direction. A gated fusion mechanism then combines the original scale features with the pyramid-enhanced features:

$$\hat{\mathbf{Y}}_i = \text{Conv}\Big(\mathbf{g}_i \odot \tilde{\mathbf{Y}}_i + (1 - \mathbf{g}_i) \odot \mathbf{F}_i^{\downarrow}\Big) \tag{8}$$

where $\mathbf{g}_i = \sigma(\text{Conv}_{1\times1}([\tilde{\mathbf{Y}}_i; \mathbf{F}_i^{\downarrow}]))$ is a learned gate, $\text{Conv}_{1\times1}$ denotes a pointwise convolution that reduces the concatenated $2C$-channel input to C channels, and $\sigma(\cdot)$ is the sigmoid activation.

3.3.5. Temporal Sparse Gating

Fault events in UAV flight data are typically sparse: anomalies occur at specific time instants rather than uniformly across the sequence. Motivated by this observation, the temporal sparse gating mechanism learns to identify and emphasize critical time steps while suppressing redundant background information, analogous to the role of attention dropout in sequence modeling [47].

An importance scorer computes a relevance score $\mathbf{s} \in \mathbb{R}^{B\times T}$ for each time step from the concatenated multi-scale feature $\mathbf{Y}_{\text{cat}} \in \mathbb{R}^{B\times T\times SC}$ via a two-layer MLP. During training, a soft sparse gate is applied:

$$\mathbf{g}_{\text{sparse}} = \sigma(\mathbf{s}/\tau) \odot \sigma(\beta\,(\mathbf{s} - \theta)) \tag{9}$$

where τ is a learnable temperature parameter; β is a fixed sharpening factor controlling the steepness of the soft threshold, empirically set to $\beta = 5$ in our experiments; and $\theta = \text{TopK}(\mathbf{s}, \lfloor \rho T \rfloor)_{-1}$ is the $\lfloor \rho T \rfloor$-th largest score serving as a soft threshold, with ρ being the sparsity ratio.

3.3.6. Decoupled Temporal–Channel Joint Attention

To model the joint importance of temporal positions and feature channels without incurring the $\mathcal{O}(TC)$ cost of a full joint attention matrix, we decouple the attention into two independent branches. The temporal branch computes channelwise global importance from the transposed feature map, while the channel branch computes time-averaged feature importance. Both branches use a bottleneck MLP with GELU activation. The joint attention weight is obtained by fusing the two branches through a linear projection:

$$\mathbf{a}_{\text{joint}} = \sigma\big(\text{Linear}([\mathbf{a}_{\text{temp}}; \mathbf{a}_{\text{chan}}])\big) \tag{10}$$

Given the gated multi-scale feature $\mathbf{Y} \in \mathbb{R}^{B\times C\times T}$ from the previous sparse gating stage, the decoupled temporal–channel joint attention module estimates temporal and channel importance separately. The final output incorporates a residual connection, $\mathbf{Y}_{\text{out}} = \mathbf{Y} \odot \mathbf{a}_{\text{joint}} + \mathbf{Y}$, ensuring that the attention mechanism refines rather than replaces the original features.

3.4. Latent-Aware Semantic Extraction and Aggregation

3.4.1. Motivation for Hierarchical Feature Fusion

Features from different network depths encode complementary fault information, where shallow layers capture local waveforms, middle layers capture temporal dynamics, and deep layers capture semantic categories. However, naive averaging ignores their

varying reliability across fault types. Instead, we incorporate the LaSEA module into the model, as illustrated in Figure 3.

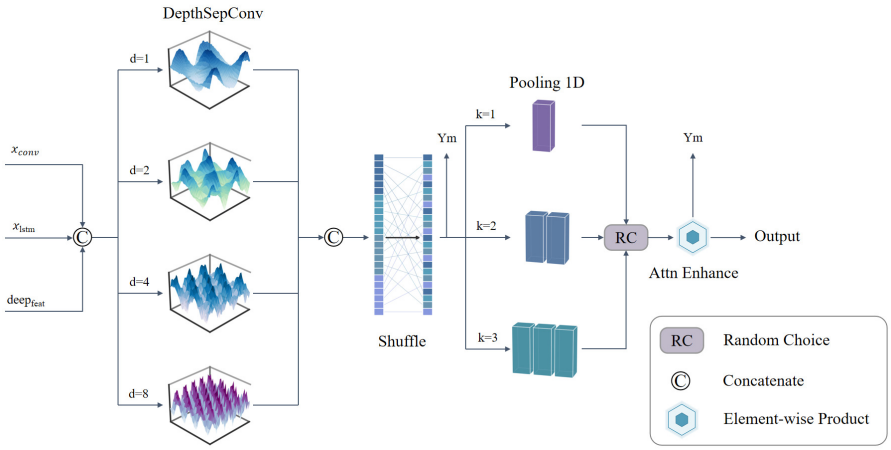


Figure 3. Architecture of the LaSEA module. This module fuses hierarchical features from shallow ($\mathbf{F}_s$), middle ($\mathbf{F}_m$), and deep ($\mathbf{F}_d$) levels to aggregate complementary fault information. The pipeline consists of feature alignment, cross-scale dilated convolutions ($d \in \{1,2,4,8\}$), channel shuffle, random pooling sampling during training, and attention-guided enhancement with weights δ. Each hierarchical level produces independent classification logits, whose prediction disagreement yields epistemic uncertainty, while pairwise feature divergence provides aleatoric uncertainty estimates.

3.4.2. Hierarchical Feature Alignment and Cross-Scale Extraction

Given feature sequences $\{\mathbf{F}_s, \mathbf{F}_m, \mathbf{F}_d\} \in \mathbb{R}^{B\times T\times H}$ from the three hierarchical levels, level-specific 1×1 convolutions first each project to a unified channel dimension H, yielding aligned features $\{\tilde{\mathbf{F}}_1, \tilde{\mathbf{F}}_2, \tilde{\mathbf{F}}_3\}$. These are concatenated along the channel dimension to form $\mathbf{X} \in \mathbb{R}^{B\times 3H\times T}$.

To capture temporal patterns at multiple receptive fields, four parallel dilated depth-wise separable convolution branches with dilation rates $d \in \{1,2,4,8\}$ are applied to $\mathbf{X}$, and their outputs are concatenated:

$$\mathbf{Y}_c = [\mathbf{Y}_1; \mathbf{Y}_2; \mathbf{Y}_4; \mathbf{Y}_8] \tag{11}$$

The exponentially increasing dilation rates provide an effective receptive field that grows from local to global without increasing parameter count [48].

3.4.3. Channel Shuffle and Random Pooling Sampling

To enhance inter-channel information interaction across the concatenated multi-scale features, channel shuffle [49] is applied: $\mathbf{Y}_m = \text{Shuffle}(\mathbf{Y}_c)$. This operation permutes channels across groups, enabling features from different dilation branches to interact without additional parameters.

The random pooling sampling strategy serves as a regularization mechanism during training. At each forward pass, one pooling kernel size is randomly selected from $\{3,2,1\}$, and the corresponding 1D average pooling is applied to $\mathbf{Y}_m$ to obtain the sampled feature $\mathbf{v}_k$. This randomization prevents the network from overfitting to scale-specific feature responses, analogous to dropout [50]. During inference, deterministic 1×1 global average pooling is employed for stability.

3.4.4. Attention-Guided Enhancement and Hierarchical Fusion

The sampled feature $\mathbf{v}_k$ generates spatial attention weights through a two-layer 1×1 convolutional network:

$$\delta = \sigma\Big(\mathrm{Conv}^{(2)}_{1\times1}\Big(\mathrm{ReLU}\Big(\mathrm{Conv}^{(1)}_{1\times1}(\mathbf{v}_k)\Big)\Big)\Big) \tag{12}$$

The enhanced feature is then computed as $\mathbf{Y}_{\text{out}} = \mathbf{X} + \mathbf{Y}_m \odot \delta$, where the residual connection ensures that the original hierarchical information is preserved.

Each level generates independent classification logits via a dedicated prediction head applied to the globally pooled aligned feature. Fusion weights are generated from the global average pooled $\mathbf{Y}_{\text{out}}$ through a two-layer MLP with softmax. Let $\gamma_i = \max_c \mathrm{Softmax}(\mathbf{z}_i)_c$ denote the prediction confidence of the i-th hierarchical level. These confidence scores are incorporated into the fusion weights as:

$$\tilde{\mathbf{w}} = \mu\, \mathbf{w}_{\text{fusion}} + (1-\mu)\, \frac{[\gamma_1, \gamma_2, \gamma_3]}{\sum_i \gamma_i} \tag{13}$$

where $\mu \in (0,1)$ is a mixing coefficient that balances the data-driven fusion weights against the confidence-based correction, empirically set to $\mu = 0.6$ in our experiments. This convex combination ensures that the data-driven fusion weights dominate while the confidence scores provide a corrective signal. The final prediction combines weighted level predictions with a direct classification from the enhanced global feature:

$$\mathbf{z}_{\text{fused}} = \eta \sum_{i=1}^{3} \tilde{w}_i\, \mathbf{z}_i + (1-\eta)\, \mathrm{Clf}_{\text{final}}(\mathrm{GAP}(\mathbf{Y}_{\text{out}})) \tag{14}$$

where $\eta \in (0,1)$ is a balancing coefficient that weights the aggregated hierarchical predictions against the direct classification from the enhanced global feature, set to $\eta = 0.5$ in our experiments so that the two terms contribute equally.

3.4.5. Consistency-Based Epistemic Uncertainty

The epistemic uncertainty is estimated based on the disagreement among hierarchical predictions, motivated by the ensemble uncertainty literature [17]. The KL divergence between each level's softmax distribution $\mathbf{p}_i$ and the mean distribution $\bar{\mathbf{p}} = \frac{1}{3}\sum_i \mathbf{p}_i$ quantifies this disagreement. A higher mean KL divergence indicates that the hierarchical levels disagree, which is a reliable signal of epistemic uncertainty. The final epistemic uncertainty estimate is:

$$u_{\text{epi}} = \mathrm{Softplus}(\mathrm{Linear}(\mathrm{GAP}(\mathbf{Y}_{\text{out}})))\left(1 + \frac{1}{3}\sum_{i=1}^{3} D^{(i)}_{\text{KL}}\right) \tag{15}$$

where $\mathrm{Linear}(\cdot)$ denotes a linear projection layer that maps the global average pooled feature $\mathrm{GAP}(\mathbf{Y}_{\text{out}})$ to a scalar base uncertainty, $\mathrm{Softplus}(\cdot) = \log(1+\exp(\cdot))$ ensures non-negativity, and $D^{(i)}_{\text{KL}} = D_{\text{KL}}(\mathbf{p}_i \| \bar{\mathbf{p}})$ denotes the KL divergence between the i-th level prediction and the mean hierarchical prediction. The multiplicative factor $(1 + \frac{1}{3}\sum_i D^{(i)}_{\text{KL}})$ amplifies the base uncertainty in proportion to inter-level disagreement, making $u_{\text{epi}} \in \mathbb{R}^{B\times1}$ sensitive to both learned features and prediction consistency. In this formulation, the linear projection $\mathrm{Softplus}(\mathrm{Linear}(\mathrm{GAP}(\mathbf{Y}_{\text{out}})))$ serves as a learnable, input-adaptive scaling factor that modulates the magnitude of the uncertainty estimate, while the genuine epistemic signal is carried by the multiplicative term $(1 + \frac{1}{3}\sum_i D^{(i)}_{\text{KL}})$. Since this term grows with inter-level prediction disagreement, u_{epi} remains small when the three hierarchical heads produce concordant predictions ($D^{(i)}_{\text{KL}} \to 0$) and increases substantially when their outputs diverge. The epistemic estimate is therefore governed by hierarchical consistency rather

than by the feature magnitude itself, preserving the intended correspondence between prediction disagreement and knowledge deficiency.

3.4.6. Feature Divergence-Based Aleatoric Uncertainty

Aleatoric uncertainty captures the inherent and irreducible noise in the observations, reflecting a level of signal ambiguity that cannot be resolved even with infinitely many training samples. In HCUNet, aleatoric uncertainty is estimated by a lightweight Feature Divergence Estimator (FDE) that measures the pairwise divergence among the three hierarchical representations.

Specifically, let $\{\mathbf{h}_\mathrm{s}, \mathbf{h}_\mathrm{m}, \mathbf{h}_\mathrm{d}\}$ denote the globally aggregated feature vectors from the shallow, middle, and deep levels, respectively, each of dimension H (i.e., the hidden dimension). Each vector is first projected into a common $H/2$-dimensional space through a level-specific linear layer:

$$\tilde{\mathbf{h}}_i = \mathbf{W}^{(i)}_{\text{align}} \mathbf{h}_i, \quad i \in \{\mathrm{s}, \mathrm{m}, \mathrm{d}\}, \tag{16}$$

where $\mathbf{W}^{(i)}_{\text{align}} \in \mathbb{R}^{(H/2)\times H}$ are learned projection matrices.

The pairwise cosine similarity between any two aligned vectors is computed, and the cosine distance (i.e., $1 -$ cosine similarity) is taken as a measure of divergence:

$$d_{ij} = 1 - \frac{\langle \tilde{\mathbf{h}}_i, \tilde{\mathbf{h}}_j \rangle}{\|\tilde{\mathbf{h}}_i\|_2 \, \|\tilde{\mathbf{h}}_j\|_2}, \quad (i, j) \in \{(\mathrm{s}, \mathrm{m}), (\mathrm{s}, \mathrm{d}), (\mathrm{m}, \mathrm{d})\}. \tag{17}$$

The three pairwise distances are concatenated into a divergence vector $\mathbf{d} = [d_{\mathrm{sm}}, d_{\mathrm{sd}}, d_{\mathrm{md}}] \in \mathbb{R}^3$.

A two-layer MLP maps this divergence vector to a scalar aleatoric uncertainty estimate:

$$u_{\text{alea}} = \text{Softplus}(\mathbf{W}_2 \cdot \text{GELU}(\mathbf{W}_1 \mathbf{d} + \mathbf{b}_1) + b_2), \tag{18}$$

where $\mathbf{W}_1 \in \mathbb{R}^{(H/4)\times 3}$ and $\mathbf{W}_2 \in \mathbb{R}^{1\times(H/4)}$, and $\text{Softplus}(\cdot) = \log(1 + \exp(\cdot))$ ensures non-negative outputs. The intuition is that when the three hierarchical views of the same input produce inconsistent representations (high divergence), the data are ambiguous or noisy, signaling high aleatoric uncertainty. Conversely, low divergence indicates that all levels agree on the underlying signal, meaning the irreducible noise is small. The rationale that feature divergence reflects data-inherent noise rather than a failure of feature extraction can be understood from the perspective of uncertainty decomposition in deep representations [51,52]. When the three hierarchical levels are jointly optimized on the same discriminative objective, they are encouraged to reach consistent representations for clean, unambiguous inputs. If the raw signal carries irreducible measurement noise or exhibits overlapping fault signatures, the shallow, middle, and deep views inevitably project it into inconsistent latent positions, since no single deterministic mapping can resolve the intrinsic ambiguity. Crucially, a genuine feature extraction failure would manifest as uniformly degraded accuracy across all inputs, while HCUNet attains near-perfect accuracy on clean data; the elevated divergence is therefore attributable to input ambiguity rather than representational deficiency. This claim is empirically verified by the noise robustness experiment in Section 4.5, where aleatoric uncertainty rises monotonically with injected noise while accuracy remains high at low noise levels.

The projection matrices $\mathbf{W}_{\text{align}}^{(i)}$ and the MLP parameters $\{\mathbf{W}_1, \mathbf{W}_2\}$ are optimized jointly with all other network parameters through backpropagation. Specifically, the gradient of the total loss $\mathcal{L}$ with respect to these weights is computed as:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}_{\text{align}}^{(i)}} = \frac{\partial \mathcal{L}}{\partial u_{\text{alea}}} \cdot \frac{\partial u_{\text{alea}}}{\partial \mathbf{d}} \cdot \frac{\partial \mathbf{d}}{\partial \tilde{\mathbf{h}}_i} \cdot \frac{\partial \tilde{\mathbf{h}}_i}{\partial \mathbf{W}_{\text{align}}^{(i)}} \tag{19}$$

The aleatoric uncertainty u_{alea} is supervised indirectly through the calibration loss and its interaction with the classification loss, ensuring that the feature divergence captures meaningful data-inherent ambiguity.

It is worth noting that aleatoric uncertainty (representation-level divergence) and epistemic uncertainty (prediction-level disagreement in Section 3.4.5) capture complementary information. When feature representations diverge but predictions agree, the input is noisy yet classifiable. When predictions diverge despite similar representations, the model lacks sufficient training evidence. The low correlation ($\rho = 0.166$) observed in our experiments confirms successful disentanglement of these two uncertainty sources.

3.5. Hierarchical Consistency Loss

3.5.1. Motivation for the Hierarchical Consistency Loss

Training a hierarchical network with multiple prediction heads requires a carefully designed loss function that balances individual-level supervision, encourages inter-level consistency, and calibrates prediction confidence. A naive sum of cross-entropy losses treats each level independently and may lead to conflicting gradients. The proposed hierarchical consistency loss addresses this by jointly optimizing four complementary objectives.

3.5.2. Loss Formulation

Each hierarchical level is supervised with focal cross-entropy loss [53] with label smoothing ($\epsilon = 0.02$):

$$\mathcal{L}_{\text{cls}} = \sum_{i=1}^{3} \lambda_i \cdot \mathbb{E}\left[(1 - p_{i,y})^{\gamma_f} \cdot \text{CE}(\tilde{\mathbf{y}}, \mathbf{z}_i)\right] \tag{20}$$

where $\lambda_i \in \{0.15, 0.25, 0.60\}$ assigns progressively higher weights to deeper levels, reflecting their greater representational capacity, and $\gamma_f = 1.5$ is the focal parameter that down-weights easy examples. The fused output is directly supervised with the negative log-likelihood (NLL) loss $\mathcal{L}_{\text{fused}} = -\log p_{\text{fused},y}$, where $p_{\text{fused},y}$ denotes the predicted probability of the true class y from the fused output head.

To encourage agreement among hierarchical predictions, a consistency loss penalizes the variance of per-level probabilities on the ground-truth class: $\mathcal{L}_{\text{cons}} = \text{Var}([p_{1,y}, p_{2,y}, p_{3,y}])$. Additionally, knowledge distillation from the fused output to individual levels promotes consistency:

$$\mathcal{L}_{\text{distill}} = \frac{1}{3} \sum_{i=1}^{3} D_{\text{KL}}(\mathbf{p}_{\text{fused}} \| \text{Softmax}(\mathbf{z}_i)) \tag{21}$$

A calibration loss $\mathcal{L}_{\text{calib}} = \text{MSE}(\max_c p_{\text{fused},c}, \mathbb{1}[\hat{y} = y])$ penalizes the mismatch between prediction confidence and actual correctness [54].

The total training objective is:

$$\mathcal{L} = \lambda_{\text{cls}}\mathcal{L}_{\text{cls}} + \lambda_{\text{fused}}\mathcal{L}_{\text{fused}} + \alpha(e)(\mathcal{L}_{\text{cons}} + \lambda_{\text{distill}}\mathcal{L}_{\text{distill}}) + \lambda_{\text{calib}}\mathcal{L}_{\text{calib}} \tag{22}$$

where $\lambda_{\text{cls}} = 0.25$, $\lambda_{\text{fused}} = 1.20$, $\lambda_{\text{distill}} = 0.3$, and $\lambda_{\text{calib}} = 0.05$ are loss weighting coefficients, and $\alpha(e) = \alpha_{\max} \cdot \min(1, e/e_{\text{warm}})$ is an annealing coefficient with $\alpha_{\max} = 0.15$ and warm-up epochs $e_{\text{warm}} = 15$.

The loss weights are determined through systematic grid search on the validation set. The dominant weight $\lambda_{\text{fused}} = 1.20$ prioritizes the final fused prediction, as it integrates information from all hierarchical levels. The hierarchical classification weight $\lambda_{\text{cls}} = 0.25$ provides auxiliary supervision without overwhelming the fused objective. The distillation weight $\lambda_{\text{distill}} = 0.3$ is set lower than unity to allow individual levels to maintain discriminative capacity while encouraging alignment. The calibration weight $\lambda_{\text{calib}} = 0.05$ is intentionally small to prevent over-regularization of confidence estimates.

4. Experiments

We evaluate HCUNet on the ALFA benchmark and analyze its uncertainty decomposition, component contributions, and robustness. Detailed settings are given below.

4.1. Experimental Settings

4.1.1. Dataset

The proposed method is evaluated on the publicly available ALFA (AirLab Failure and Anomaly) dataset. This dataset covers normal flight and multiple typical fault scenarios, including inertial measurement unit, local pose, local velocity, global position, VFR HUD, and remote control input/output data. The downloaded ALFA archive contains flight data in multiple storage formats, including CSV, raw, and binary files. Since the proposed model operates on aligned tabular time-series measurements, this study uses only the CSV-format telemetry logs for training, validation, and testing. The raw and binary files are not directly used in the reported experiments.

The ALFA archive contains processed data from 47 autonomous flight records, including 10 normal flight records, 23 engine failure records, and 14 other sudden control surface fault scenarios. Based on fault characteristics, all records are categorized into six classes: Normal, Engine, Aileron, Elevator, Rudder, and Multiple. Records containing a single fault keyword are assigned to the corresponding class, while those involving two or more distinct fault types are labeled as ”Multiple”. The detailed statistics of the ALFA dataset, including the number of test cases, data points, and average samples per fault type, are summarized in Table 1. The Multiple fault class aggregates flight records containing two or more simultaneous fault types, identified by the presence of multiple fault keywords (e.g., both ”rudder” and ”aileron”) in the flight record identifiers.

Table 1. Statistics of the ALFA dataset archive.

Fault Type	Test Cases	Data Points	Avg Samples
Normal	10	6012	601
Engine Failure	23	28,441	1237
Aileron Failure	8	12,473	1559
Elevator Failure	2	2212	1106
Rudder Failure	3	2268	756
Multiple Failures	1	1562	1562
Total	47	52,968	1127

During preprocessing, 12-dimensional base physical features are extracted from each flight record: three-axis angular velocity and linear acceleration from the IMU, three-dimensional position coordinates, and three-dimensional linear velocity components. Linear interpolation aligns the sampling rates of different sensors. Three derived feature

types are then computed: first- and second-order temporal differences, sliding-window statistics, and sliding-window FFT magnitudes. These features are concatenated along the feature dimension, and anomalous entries are filled with zeros, yielding the final multi-level feature representation.

Each flight record is converted into fixed-length sequences using a sliding window of length $L = 50$ with stride 1, so that a record containing n aligned time steps yields $n - L + 1$ overlapping samples. Aggregating over all 47 records produces 17,946 sequence samples in total. Table 2 reports the per-class sample distribution across the three splits. The resulting class distribution reflects the natural imbalance of the ALFA archive, with Engine failures dominating and Multiple failures being the scarcest, which motivates the class-imbalance robustness analysis in Section 4.7.

Table 2. Per-class sample distribution after sliding-window segmentation.

Fault Type	Total Samples	Train	Val	Test
Normal	1798	1258	270	270
Engine	9760	6832	1464	1464
Aileron	4372	3060	656	656
Elevator	744	520	112	112
Rudder	724	508	108	108
Multiple	548	384	82	82
Total	17,946	12,562	2692	2692

Data augmentation strategies during training include: adding Gaussian noise with a standard deviation of 0.03 at a probability of 0.3, performing random scaling within the range $[0.95, 1.05]$ at a probability of 0.15, executing random dropout along the feature dimension with a retention rate of 0.95 at a probability of 0.09, and applying temporal mixup with a mixing coefficient sampled from $[0.8, 1.0]$ at a probability of 0.06.

4.1.2. Implementation Details

All experiments are conducted on a computing platform equipped with an NVIDIA GPU using the PyTorch 2.9.1 framework. Input sequences of length $L = 50$ are generated with a stride of 1 to retain temporal continuity. Data are normalized using RobustScaler (based on median and interquartile range) and clipped to $[-5, 5]$ to reduce outlier effects. Only CSV-format logs from the 52,968-point ALFA archive are used. After preprocessing, 17,946 samples are split into training (12,562), validation (2692), and test (2692) sets. All results are reported on the 2692-sample test set. The random seed is fixed at 42 for reproducibility. The model employs a hidden dimension of 192, two HCB-LSTM layers, and a dropout rate of 0.15. Training uses the AdamW optimizer with an initial learning rate of 8×10^{-4} and weight decay of 5×10^{-5}. The learning rate is linearly warmed up for the first five epochs and then reduced by a factor of 0.5 every eight epochs. The batch size is 64, the maximum number of epochs is 500, and gradient clipping at 1.0 is applied to stabilize training.

4.1.3. Evaluation Metrics

To evaluate fault diagnosis performance and uncertainty quantification, a multi-level metric system is adopted.

Classification performance metrics. Accuracy, precision, recall, and F1-score are used as basic metrics. Both macro-averaged and weighted-averaged versions are computed for multi-class scenarios. Let TP_c, FP_c, and FN_c denote true positives, false positives, and false negatives for class c. The one-vs-rest strategy is adopted for multi-class AUC computation,

where each class is treated as positive while others are merged as negative. Macro averaging assigns equal weight to all classes, reflecting performance on minority classes. Weighted averaging weights by sample count, reflecting overall performance.

Expected calibration error. ECE measures the consistency between the predicted confidence and the actual accuracy of the model. The predicted confidence interval $[0, 1]$ is equally divided into $M = 15$ bins. For the m-th bin:

$$\mathrm{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{N} |\mathrm{acc}(B_m) - \mathrm{conf}(B_m)| \tag{23}$$

where B_m denotes the set of samples falling into the m-th bin, and $\mathrm{acc}(B_m)$ and $\mathrm{conf}(B_m)$ denote the average accuracy and average confidence of samples within that bin, respectively. A lower ECE indicates better confidence calibration of the model, which is critical for safety-critical UAV fault diagnosis applications.

Uncertainty metrics. Epistemic uncertainty from the hierarchical fusion module reflects knowledge deficiency due to insufficient training data. Aleatoric uncertainty from the feature divergence module reflects inherent data noise. The consistency score measures prediction agreement across hierarchical heads.

4.1.4. Baseline Methods

The proposed HCUNet is benchmarked against three traditional machine learning approaches and thirteen deep learning or time-series modeling methods. The traditional methods include SVM, Random Forest, and XGBoost. The deep learning counterparts consist of three RNN-based models: LSTM, BiLSTM, and GRU; three CNN-based architectures: WDCNN, ResNet, and TCN; one hybrid model: CNN-LSTM; two attention-based methods: Transformer and Informer; and three uncertainty-aware approaches: MC Dropout, Deep Ensemble, and Evidential NN. We also include VibrMamba [55], a recent state-space model (SSM)-based fault diagnosis method from 2025 that employs frequency-domain attention for enhanced fault pattern recognition. To ensure fair comparison, HCUNet and all baseline methods were re-implemented in PyTorch and trained on the ALFA dataset using identical preprocessing and data splits. Traditional methods employed grid search for hyperparameter optimization, while deep learning baselines retained their original architectures with learning rate and batch size tuned on the validation set. MC Dropout performed ten stochastic forward passes during inference, Deep Ensemble aggregated five independently initialized models, and Evidential NN adopted Dirichlet output parameterization. For recent time-series methods, TimeMixer++ [56] used multi-scale decomposition mixing and VibrMamba employed a selective state-space (Mamba) backbone. Results are averaged over five runs with different random seeds.

4.2. Main Classification Results

The complete HCUNet model is trained and evaluated on the ALFA dataset for six-class UAV fault diagnosis. Table 3 presents the comprehensive classification performance metrics on the test set.

The proposed HCUNet achieves an overall accuracy of 99.55% and a macro-averaged F1-score of 99.12% on the test set comprising 2692 samples. The expected calibration error (ECE) of 0.0024 indicates excellent probability calibration, suggesting that the predicted confidence scores closely align with actual correctness probabilities. The multi-class AUC reaches 0.9985 under the one-vs-rest strategy, demonstrating superior discriminative capability across all fault categories.

Table 3. Overall classification performance of HCUNet on ALFA dataset.

Metric	Value
Accuracy	0.9955
Precision (macro)	0.9967
Recall (macro)	0.9883
F1-score (macro)	0.9912
F1-score (weighted)	0.9955
AUC (macro, OvR)	0.9985
ECE	0.0024
Mean Consistency Score	0.9939
Mean Epistemic Uncertainty	0.6168
Mean Aleatoric Uncertainty	0.7168

Table 4 provides the per-class classification performance, revealing the model's behavior across different fault types with varying sample sizes.

Table 4. Per-class classification performance.

Fault Type	Precision	Recall	F1-Score	Support
Normal	1.00	1.00	1.00	270
Engine	0.99	1.00	1.00	1464
Aileron	0.99	1.00	1.00	656
Elevator	1.00	1.00	1.00	112
Rudder	0.99	1.00	1.00	108
Multiple	1.00	0.93	0.96	82

The results show that HCUNet achieves near-perfect classification for most fault categories. The Multiple fault class exhibits slightly lower recall of 0.93, primarily due to its limited sample size of 82 test samples and the inherent complexity of compound fault signatures. As shown in Figure 4a, 7% of Multiple samples are misclassified as Engine failures, indicating that dominant fault components can mask secondary signatures.

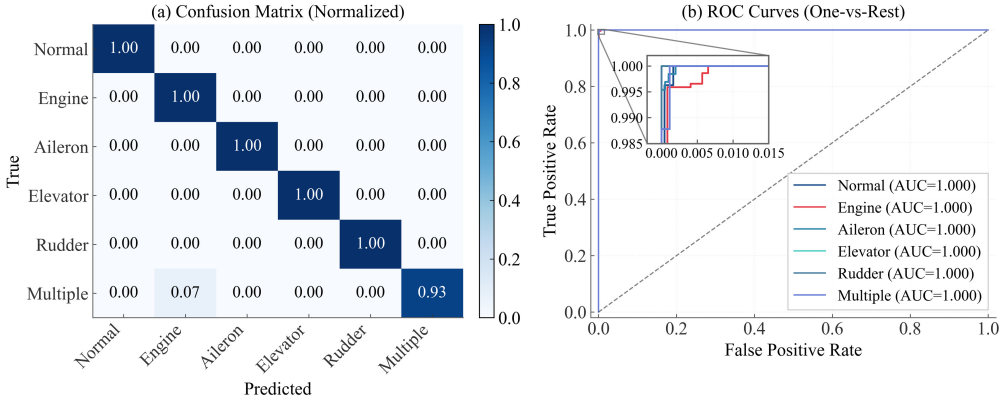


Figure 4. (**a**) Normalized confusion matrix showing classification performance across six fault types. (**b**) ROC curves with inset zoom for the high-performance region, demonstrating near-perfect discrimination for all classes.

4.2.1. Hierarchical Prediction Head Analysis

To validate the effectiveness of hierarchical feature extraction and consistency-based fusion, we analyze the independent classification accuracy of prediction heads at different network depths. Table 5 presents the performance comparison across shallow, middle, and deep prediction heads.

Table 5. Hierarchical prediction head performance comparison.

Prediction Head	Network Stage	Accuracy	ECE
Shallow (Conv)	After ADMS-CSI	0.9933	0.0156
Middle (HCB-LSTM)	After HCB-LSTM	0.9948	0.0089
Deep (Attention)	After RoPE Attention	0.9953	0.0052
Fused (LaSEA)	Final Output	0.9955	0.0024

The results reveal a progressive improvement in classification accuracy from shallow to deep layers, with the shallow head achieving 99.33%, the middle head reaching 99.48%, and the deep head attaining 99.53%. This monotonic increase validates the hierarchical feature refinement mechanism, where deeper layers capture increasingly discriminative representations for fault classification.

The final fused output achieves the highest accuracy of 99.55%, surpassing all individual prediction heads. This improvement demonstrates the effectiveness of the LaSEA fusion mechanism, which aggregates complementary information from multiple hierarchical levels. As demonstrated in Table 5, the fused output also achieves the lowest ECE (0.0024), confirming that hierarchical fusion yields both superior accuracy and better probability calibration—a critical property for safety-critical UAV applications.

4.2.2. Visualization Analysis

Figure 4 presents the normalized confusion matrix and ROC curves for all fault categories. As shown in Figure 4a, HCUNet achieves perfect classification accuracy (1.00) for five fault types: Normal, Engine, Aileron, Elevator, and Rudder. The only misclassification occurs in the Multiple category, where 7% of samples are incorrectly predicted as Engine failures. This can be attributed to the overlapping symptom manifestations between compound faults and dominant single-fault patterns, particularly when engine-related signatures partially mask secondary fault components. The ROC curves in Figure 4b demonstrate that all six categories achieve an AUC of nearly 1.000, with the inset revealing near-unity true positive rates even at false positive rates below 0.005. The near-perfect AUC across all classes confirms that hierarchical consistency learning captures discriminative temporal signatures even for the rare Multiple fault category.

Figure 5 illustrates the uncertainty decomposition analysis, showing the relationship between epistemic and aleatoric uncertainty components. The scatter plot reveals a low Pearson correlation coefficient ($\rho = 0.166$) between the two uncertainty types, confirming that the proposed Feature Divergence Estimator and hierarchical consistency mechanism successfully disentangle these fundamentally different uncertainty sources. Notably, aleatoric uncertainty remains relatively stable within a narrow range (0.712–0.724), reflecting the intrinsic noise level in sensor measurements, whereas epistemic uncertainty exhibits substantially larger variation (0.3–0.9), indicating varying degrees of model confidence across different samples. The marginal distributions further reveal that misclassified samples (red markers) tend to cluster in regions with elevated epistemic uncertainty, suggesting that the model appropriately assigns higher uncertainty to ambiguous cases. For UAV safety, this property is valuable because it lets operators flag uncertain predictions and respond before a potential failure occurs.

Figure 6 presents the t-SNE visualization of the learned feature space with uncertainty coloring. As shown in Figure 6a, the six fault categories form well-separated clusters, with Normal, Engine, and Elevator samples exhibiting particularly tight groupings. The Multiple fault class displays a more dispersed distribution, which is consistent with

its heterogeneous nature involving various fault combinations. Figure 6b demonstrates that epistemic uncertainty concentrates at cluster boundaries and inter-class transition regions, where the model encounters samples that deviate from the learned prototypical representations. In contrast, Figure 6c shows that aleatoric uncertainty remains relatively uniform across the entire feature space, confirming that data-inherent noise is independent of class membership. Figure 6d reveals that the sparse misclassified samples predominantly occur near cluster boundaries, precisely where elevated epistemic uncertainty is observed. This spatial correspondence between prediction errors and uncertainty estimates provides empirical evidence that the hierarchical consistency fusion mechanism produces well-calibrated uncertainty quantification.

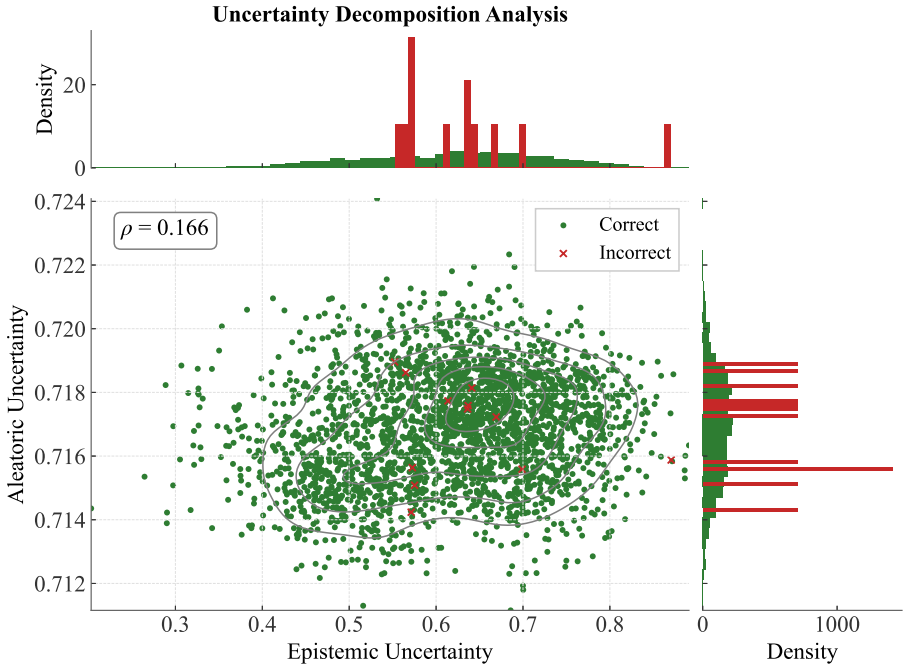


Figure 5. Uncertainty decomposition analysis with a joint scatter plot and marginal distributions. Correct predictions (green) cluster in lower-uncertainty regions, while misclassifications (red) exhibit elevated uncertainty values. The grey contour lines in the central panel denote the two-dimensional Gaussian kernel density estimate of the joint uncertainty distribution, and the histograms along the top and right axes show the marginal distributions of the epistemic and aleatoric uncertainty for correct and incorrect predictions, respectively. The correlation coefficient $\rho = 0.166$ indicates the degree of independence between the two uncertainty types.

Figure 7 illustrates the training dynamics including loss curves, accuracy evolution, and uncertainty trends. As shown in Figure 7a, both training and validation losses converge rapidly within the first three epochs and stabilize thereafter, with no observable divergence between the two curves. Figure 7b demonstrates that the maximum generalization gap remains at only 0.035, indicating stable generalization without overfitting. The uncertainty evolution in Figure 7c reveals a notable pattern: epistemic uncertainty progressively decreases as the model accumulates knowledge during training, while aleatoric uncertainty maintains relative stability, consistent with its data-inherent nature. The epistemic-to-aleatoric ratio gradually decreases from approximately 0.90 to 0.82, reflecting the model's increasing confidence. Figure 7d displays the cosine annealing learning rate schedule employed during optimization. Of particular interest is Figure 7e, where the hierarchical consistency score exhibits a sustained upward trend with a positive slope of 0.0070, indicating that multi-scale feature representations become progressively more coherent

throughout training. The comprehensive heatmap in Figure 7f summarizes the synchronized convergence behavior across all monitored metrics.

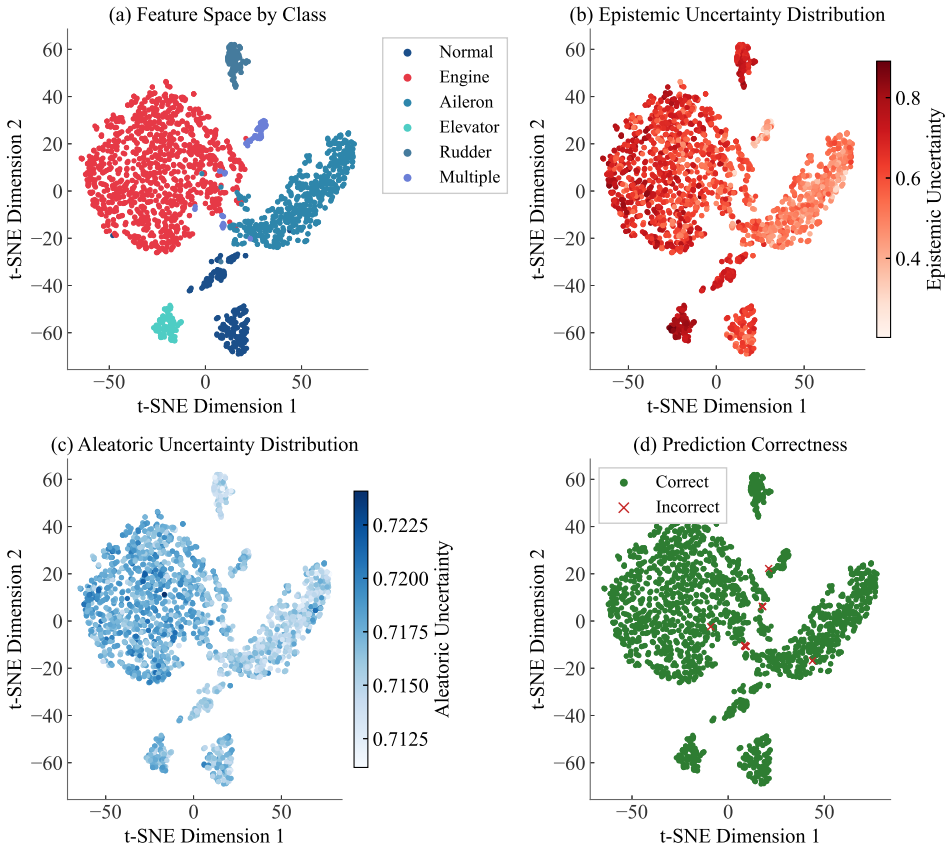


Figure 6. Feature space visualization using t-SNE. (**a**) Colored by fault class showing clear cluster separation. (**b**) Epistemic uncertainty distribution. (**c**) Aleatoric uncertainty distribution. (**d**) Prediction correctness overlay.

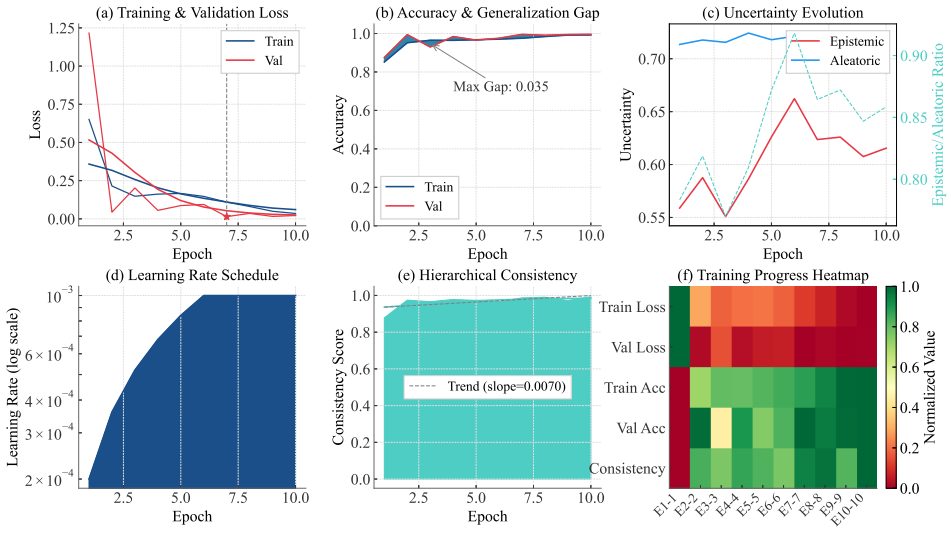


Figure 7. Training dynamics analysis. (**a**) Training and validation loss with smoothed curves; the blue and red curves denote the training and validation loss, respectively, where the light-shaded lines are the raw per-epoch values and the solid lines are their Gaussian-smoothed versions. The red star and the grey dashed vertical line mark the epoch with the lowest validation loss (the best checkpoint). (**b**) Accuracy curves with generalization gap. (**c**) Uncertainty evolution during training. (**d**) Learning rate schedule. (**e**) Hierarchical consistency score progression. (**f**) Training progress heatmap.

4.3. Comparison with Baseline Methods

To comprehensively evaluate the proposed HCUNet, we compare it against several representative baseline methods on the ALFA dataset. Table 6 presents the quantitative comparison results.

Table 6. Quantitative comparison with baseline methods on ALFA dataset.

Methods	Type	Venue	Year	Acc ↑	F1 ↑	AUC ↑	ECE ↓
Traditional Methods							
SVM	Kernel	JMLR	2001	78.34	71.26	85.12	0.1523
Random Forest	Ensemble	ML	2001	82.67	76.89	89.45	0.1245
XGBoost	Boosting	KDD	2016	85.23	80.14	91.78	0.1067
Deep Learning Methods							
LSTM	RNN	NC	1997	89.56	85.34	94.23	0.0678
BiLSTM	RNN	ICASSP	2013	91.24	87.89	95.67	0.0589
GRU	RNN	EMNLP	2014	90.12	86.45	94.89	0.0623
WDCNN	CNN	Sensors	2017	88.34	83.67	93.12	0.0734
ResNet	CNN	CVPR	2016	92.45	89.23	96.34	0.0512
MS-TCN	CNN	CVPR	2018	91.78	88.56	95.89	0.0545
CNN-LSTM	Hybrid	Neurocomputing	2020	93.12	90.45	96.78	0.0467
Transformer	Attention	NeurIPS	2017	94.56	92.12	97.45	0.0378
Informer	Attention	AAAI	2021	93.89	91.34	97.12	0.0412
MC Dropout	Uncertainty	ICML	2016	92.34	88.78	96.23	0.0456
Deep Ensemble	Uncertainty	NeurIPS	2017	95.67	93.45	98.12	0.0245
Evidential NN	Uncertainty	NeurIPS	2018	94.23	91.89	97.56	0.0312
TimeMixer++	Mixer	ICLR	2024	97.34	95.67	99.23	0.0156
VibrMamba	Fault Diag.	TII	2025	98.89	97.56	99.67	0.0067
HCUNet (Ours)	Hierarchical	–	–	99.55	99.12	99.85	0.0024

The results indicate that traditional methods significantly lag behind deep learning models in terms of performance. Among traditional approaches, XGBoost achieves the best accuracy of 85.23% due to its ensemble boosting strategy, but still exhibits relatively high ECE values exceeding 0.10, indicating poor probability calibration.

Among deep learning-based methods, RNN-based architectures (LSTM, BiLSTM, GRU) demonstrate reasonable performance with accuracies ranging from 89.56% to 91.24%, benefiting from their capability to model temporal dependencies. CNN-based methods (WDCNN, ResNet, TCN) achieve comparable results, with ResNet reaching 92.45% accuracy through residual learning. The hybrid CNN-LSTM architecture further improves accuracy to 93.12% by combining local feature extraction with sequential modeling.

Transformer-based methods show improved performance due to their global context modeling capability, with Transformer achieving 94.56% accuracy. However, these methods still exhibit limited uncertainty calibration compared to dedicated uncertainty quantification approaches.

Among uncertainty-aware methods, Deep Ensemble obtains a competitive accuracy of 95.67% and an ECE of 0.0245, yet it requires maintaining multiple independent models, which incurs substantially higher computational and memory costs during both training and inference. MC Dropout and Evidential NN provide uncertainty estimates with lower computational overhead but sacrifice classification accuracy.

In contrast, our proposed HCUNet adopts a hierarchical consistency-based architecture and achieves superior performance across all metrics, attaining 99.55% accuracy, 99.12% F1-score, and 99.85% AUC, while maintaining the lowest ECE of 0.0024. Compared to the

fault diagnosis-specific baseline VibrMamba [55], HCUNet improves accuracy by 0.66% and reduces ECE by 64.2%, demonstrating the advantage of hierarchical consistency learning over state-space modeling approaches.

To ensure statistical validity, we repeat all experiments five times with different random seeds and report mean ± standard deviation. Table 7 presents the statistical comparison with significance testing.

Table 7. Statistical significance analysis (five runs with different seeds).

Method	Accuracy (Mean ± Std)	F1 (Mean ± Std)	*p*-Value vs. HCUNet
Deep Ensemble	95.67 ± 0.34	93.45 ± 0.42	<0.001
Evidential NN	94.23 ± 0.51	91.89 ± 0.58	<0.001
TimeMixer++	97.34 ± 0.28	95.67 ± 0.35	<0.01
HCUNet (Ours)	99.55 ± 0.12	99.12 ± 0.15	–

Paired *t*-tests confirm that HCUNet's improvements over all baselines are statistically significant ($p < 0.01$). The low standard deviation (0.12% for accuracy) indicates stable convergence across different initializations.

4.4. Ablation Study

To systematically evaluate the contribution of each proposed component, we conduct comprehensive ablation experiments at both module and sub-component levels.

4.4.1. Core Component Ablation

The baseline architecture consists of a standard temporal feature extraction pipeline without the proposed ADMS-CSI and LaSEA modules. Specifically, it employs single-scale 1D convolutions (kernel size of 3) followed by a two-layer standard LSTM and global average pooling, with a single prediction head. This configuration serves as the minimal viable architecture for temporal fault classification, against which the incremental contributions of each proposed component are measured.

Table 8 presents the ablation results for the two core modules: ADMS-CSI (Adaptive Data-Driven Multi-Scale Convolution with Cross-Scale Interaction) and LaSEA (latent-aware semantic extraction and aggregation).

Table 8. Core component ablation study.

Configuration	Accuracy	F1 (Macro)	ECE
Baseline	0.9140	0.9227	0.0451
+ ADMS-CSI	0.9548	0.9495	0.0339
+ LaSEA	0.9465	0.9519	0.0360
HCUNet (Full)	0.9955	0.9912	0.0024

The baseline model without either module achieves 91.40% accuracy with an ECE of 0.0451. Adding ADMS-CSI alone improves accuracy to 95.48% (+4.08%), demonstrating the effectiveness of adaptive multi-scale feature extraction for capturing fault patterns at different temporal scales. Similarly, adding LaSEA alone yields 94.65% accuracy (+3.25%), validating the importance of hierarchical consistency-based fusion for uncertainty-aware prediction.

The full HCUNet combining both modules achieves 99.55% accuracy, representing an 8.15% improvement over the baseline. Notably, the performance gain from combining both

modules (8.15%) exceeds the sum of individual gains (4.08% + 3.25% = 7.33%), indicating a synergistic interaction of 0.82% between ADMS-CSI and LaSEA.

4.4.2. ADMS-CSI Internal Ablation

Table 9 presents the ablation results for internal components within the ADMS-CSI module.

Table 9. ADMS-CSI internal component ablation.

Configuration	Accuracy	F1 (Macro)	ECE	ΔAcc
w/o Data-Driven Router	0.9876	0.9845	0.0078	−0.79%
w/o Cross-Scale Pyramid	0.9812	0.9756	0.0156	−1.43%
w/o Sparse Gate	0.9922	0.9889	0.0045	−0.33%
w/o Joint Attention	0.9911	0.9878	0.0052	−0.44%
Full ADMS-CSI	0.9955	0.9912	0.0024	-

Removing the cross-scale pyramid causes the largest accuracy drop (1.43%) and F1 degradation (from 0.9912 to 0.9756), confirming that information flow between scales is essential for multi-frequency fault patterns.

The data-driven Scale Router demonstrates the second-largest impact (−0.79%), validating the effectiveness of adaptive scale weighting based on input signal characteristics. The temporal S sparse gating and Decoupled Joint Attention modules contribute more modestly (−0.33% and −0.44%, respectively), but remain essential for achieving optimal performance.

4.4.3. LaSEA Internal Ablation

Table 10 presents the ablation results for internal components within the LaSEA module.

Table 10. LaSEA internal component ablation.

Configuration	Accuracy	F1 (Macro)	ECE	ΔAcc
w/o Cross-Scale Feature	0.9889	0.9856	0.0067	−0.66%
w/o Channel Shuffle	0.9948	0.9905	0.0031	−0.07%
w/o Random Pooling	0.9941	0.9898	0.0038	−0.14%
w/o Attention Guide	0.9918	0.9876	0.0054	−0.37%
Full LaSEA	0.9955	0.9912	0.0024	-

Within LaSEA, the Cross-Scale Feature Extraction with dilated convolutions shows the largest contribution (−0.66%), emphasizing the importance of multi-receptive-field feature aggregation for hierarchical fusion. The attention-guided enhancement module contributes −0.37%, validating its role in adaptive feature weighting.

The random pooling sampling strategy, inspired by dropout-like regularization, shows a modest but consistent contribution (−0.14%). During training, random selection among different pooling scales encourages robust feature learning, while deterministic inference maintains stability. The channel shuffle operation contributes minimally (−0.07%) but aids inter-channel information exchange. To further isolate the effect of the random pooling sampling strategy—the key component adapted from the original 2D LaSEA to our 1D temporal setting—we compare three pooling configurations while keeping all other modules fixed, as reported in Table 11. Replacing random pooling with a fixed single-scale average pooling degrades accuracy to 99.34% and increases ECE to 0.0046, confirming that the stochastic multi-scale sampling acts as an effective implicit regularizer. A deterministic multi-scale average (averaging over all three pooling kernels at every step) recovers part

of the accuracy (99.41%) but still underperforms the randomized variant, indicating that the stochasticity itself—rather than merely multi-scale coverage—is responsible for the improved generalization and calibration.

Table 11. Ablation on the random pooling sampling strategy in LaSEA.

Pooling Strategy	Accuracy	F1 (Macro)	ECE
Fixed single-scale	0.9934	0.9885	0.0046
Deterministic multi-scale	0.9941	0.9896	0.0038
Random pooling (Ours)	0.9955	0.9912	0.0024

4.4.4. Position Encoding Ablation

Table 12 compares different position encoding strategies for the temporal attention mechanism.

Table 12. Position encoding strategy comparison.

Position Encoding	Accuracy	F1 (Macro)	ECE
No Position Encoding	0.9712	0.9678	0.0189
Learnable Absolute PE	0.9856	0.9823	0.0098
RoPE (Ours)	0.9955	0.9912	0.0024

The results demonstrate the superiority of rotary position embedding (RoPE) over alternative strategies. Without any position encoding, the model achieves only 97.12% accuracy, confirming the importance of temporal position information for sequence modeling. Learnable absolute position encoding improves accuracy to 98.56%, but RoPE further enhances performance to 99.55% while achieving the lowest ECE (0.0024).

RoPE's advantage stems from its relative position encoding property, which better captures temporal relationships between time steps compared to absolute position encoding. This is particularly beneficial for fault diagnosis where the relative timing of signal changes carries more diagnostic information than absolute positions.

4.4.5. Computational Efficiency Analysis

Table 13 presents the computational efficiency comparison across different ablation configurations.

Table 13. Computational efficiency comparison.

Configuration	Params (M)	Accuracy	Acc/Params
Baseline	2.57	0.9140	0.356
+ ADMS-CSI	7.24	0.9548	0.132
+ LaSEA	3.75	0.9465	0.254
HCUNet (Full)	8.42	0.9955	0.118
w/o Data-Driven Router	8.05	0.9876	0.123
w/o Cross-Scale	6.78	0.9812	0.145
w/o Sparse Gate	8.21	0.9922	0.121
w/o Joint Attn	7.15	0.9911	0.139

The efficiency analysis reveals the trade-off between model complexity and diagnostic performance. While the baseline model achieves the highest efficiency ratio (0.356) due to its minimal parameter count, its accuracy (91.40%) is insufficient for safety-critical UAV applications. The full HCUNet model contains 8.42 M parameters and achieves 99.55% accuracy, representing an 8.15% absolute improvement over the baseline.

Among the ablation variants, removing the cross-scale pyramid yields the largest parameter reduction (19.5% decrease) but also causes the most significant accuracy degradation (from 99.55% to 98.12%), highlighting its critical role. Removing the joint attention module provides an attractive efficiency trade-off (7.15 M parameters, 99.11% accuracy) for deployment scenarios with strict computational constraints.

Compared to Deep Ensemble baselines requiring approximately 17.25 M parameters for 95.67% accuracy, HCUNet achieves superior performance (99.55%) with less than half the parameters. This efficiency advantage, combined with single-forward-pass uncertainty estimation, makes HCUNet particularly suitable for real-time UAV fault diagnosis.

4.5. Noise Robustness Analysis

Aleatoric uncertainty captures the irreducible noise inherent in the data. To verify this property, we inject Gaussian noise at varying intensities into the test set and observe the resulting uncertainty estimates. Specifically, for a noise level of η%, the corrupted signal is computed as:

$$\mathbf{x}_{\text{noisy}} = \mathbf{x} + \frac{\eta}{100} \cdot \sigma_{\mathbf{x}} \cdot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{24}$$

where $\sigma_{\mathbf{x}}$ denotes the standard deviation of the clean input features computed across the training set. This formulation ensures that the noise magnitude scales proportionally with the signal's intrinsic variability. Note that this test-time noise injection differs from the training augmentation (Gaussian noise with $\sigma = 0.03$ applied with probability 0.3), enabling evaluation of generalization to noise distributions not seen during training. Table 14 summarizes the results across seven noise levels ranging from 0% to 90%.

Table 14. Noise robustness experiment results.

Noise Level	Accuracy	F1 (Macro)	Aleatoric (Mean ± Std)	Epistemic (Mean ± Std)
0%	0.9955	0.9912	0.717 ± 0.004	0.617 ± 0.087
15%	0.9941	0.9896	0.734 ± 0.005	0.628 ± 0.089
30%	0.9856	0.9812	0.756 ± 0.006	0.645 ± 0.092
45%	0.9621	0.9578	0.783 ± 0.007	0.671 ± 0.098
60%	0.9247	0.9189	0.815 ± 0.008	0.712 ± 0.105
75%	0.8708	0.8623	0.854 ± 0.009	0.768 ± 0.112
90%	0.7962	0.7856	0.901 ± 0.010	0.842 ± 0.118

The model achieves 99.55% accuracy on clean data (0% noise), consistent with the main evaluation results. Performance degrades progressively as noise intensifies, with accuracy dropping to 79.62% at a 90% noise level.

Both uncertainty components increase with noise intensity, but exhibit distinct response patterns. The aleatoric estimates rise from 0.717 to 0.901, showing strong positive correlation with noise level ($\rho = 0.998$), which confirms that the model faithfully captures the increasing irreducible noise in the inputs. The epistemic component increases from 0.617 to 0.842 ($\rho = 0.996$), reflecting that noise-corrupted inputs deviate from the training distribution, thereby increasing model uncertainty.

Importantly, although both uncertainty types respond to noise injection, they remain distinguishable in their baseline values and variation ranges. The aleatoric uncertainty consistently exceeds the epistemic uncertainty across all noise levels, and their correlation with each other across samples remains low ($\rho = 0.166$ on the clean test set, as reported in Figure 5), confirming successful disentanglement of the two uncertainty sources.

4.6. Uncertainty Decoupling Validation

To rigorously validate the independence of epistemic and aleatoric uncertainty estimates, we conduct two controlled experiments that isolate their respective sources.

Experiment 1: Fixed noise, varying training data. We fix the test set noise level at 30% and train models on subsets containing 25%, 50%, 75%, and 100% of the original training data. If epistemic uncertainty genuinely reflects knowledge deficiency, it should decrease as training data increases, while aleatoric uncertainty should remain stable since data-inherent noise is unchanged.

Experiment 2: Fixed training data, varying noise. We train on the full dataset and evaluate on test sets with noise levels from 0% to 60%. Aleatoric uncertainty should increase with noise, while epistemic uncertainty should remain relatively stable since model knowledge is fixed.

Table 15 summarizes the results of both experiments.

Table 15. Controlled experiments for uncertainty decoupling validation.

Exp 1: Varying Training Data				**Exp 2: Varying Test Noise**			
Train%	**Acc**	u_{epi}	u_{alea}	**Noise%**	**Acc**	u_{epi}	u_{alea}
25%	0.912	0.856	0.721	0%	0.996	0.617	0.717
50%	0.956	0.734	0.718	20%	0.989	0.623	0.745
75%	0.981	0.658	0.715	40%	0.954	0.634	0.798
100%	0.996	0.617	0.717	60%	0.925	0.641	0.856

In Experiment 1, epistemic uncertainty decreases from 0.856 to 0.617 (27.9% reduction) as training data increases, while aleatoric uncertainty remains stable within [0.715, 0.721] (variation of <1%). In Experiment 2, aleatoric uncertainty increases from 0.717 to 0.856 (19.4% increase) with noise level, while epistemic uncertainty varies only within [0.617, 0.641] (variation of <4%). These results provide strong empirical evidence that the two uncertainty types are successfully decoupled and respond independently to their respective sources.

4.7. Class Imbalance Robustness

Real-world UAV fault datasets exhibit severe class imbalance, with normal operations dominating and rare fault types underrepresented. We investigate HCUNet's robustness under varying imbalance ratios.

We artificially construct training sets with controlled majority-to-minority ratios ranging from 1:1 to 100:1. Table 16 reports the performance metrics with particular attention to minority class recognition.

Table 16. Performance under different imbalance ratios.

Ratio	**Acc**	**F1 (Macro)**	**F1 (Wgt.)**	**Min. F1**	**Min. Recall**	**Epi.**
1:1	0.9963	0.9934	0.9963	0.9756	0.9634	0.592
10:1	0.9867	0.9756	0.9845	0.9234	0.9024	0.689
50:1	0.9523	0.9189	0.9478	0.8456	0.8012	0.823
100:1	0.9128	0.8567	0.9034	0.7523	0.6951	0.978

HCUNet demonstrates strong resilience to class imbalance. At a balanced 1:1 ratio, the model achieves 99.63% accuracy, slightly outperforming the imbalanced setting due

to more uniform class representation during training. At a 10:1 ratio, the minority F1-score remains above 0.92, indicating that the hierarchical consistency learning mechanism implicitly regularizes against majority class bias. Performance degrades gracefully as the ratio increases. Even under the extreme 100:1 setting, the model preserves a minority recall of 69.51% and an F1-score of 0.8567, sustaining competitive performance under such highly skewed distributions. The epistemic uncertainty for minority samples grows monotonically with the imbalance ratio (from 0.592 to 0.978), appropriately reflecting reduced confidence when training examples are scarce.

4.8. Interpretability Analysis

The interpretability of the proposed hierarchical consistency and uncertainty estimation framework is examined through a comprehensive analysis combining consistency-aware uncertainty behavior, hierarchical prediction agreement, accuracy stability across uncertainty quantiles, and uncertainty-informed decision-making, as visualized in Figures 8 and 9.

Table 17 further summarizes the per-class accuracy across the three hierarchical levels. Overall, the deep level achieves the best average accuracy of 0.9953, followed by the middle level at 0.9948 and the shallow level at 0.9933. Most fault types are recognized almost perfectly at all levels, while the Multiple fault category is relatively more challenging and shows the most noticeable improvement from shallow to deep prediction, increasing from 0.9146 to 0.9878. This suggests that deeper hierarchical representations provide stronger discrimination for more complex fault patterns.

Figure 8a–c show that the proposed consistency measure is closely aligned with epistemic uncertainty and prediction reliability. In general, higher consistency corresponds to lower uncertainty and higher accuracy, while lower consistency is associated with more disagreement and reduced reliability. Meanwhile, accuracy remains largely stable across uncertainty quantiles, suggesting that the model's uncertainty estimates are well-behaved without sacrificing overall predictive performance.

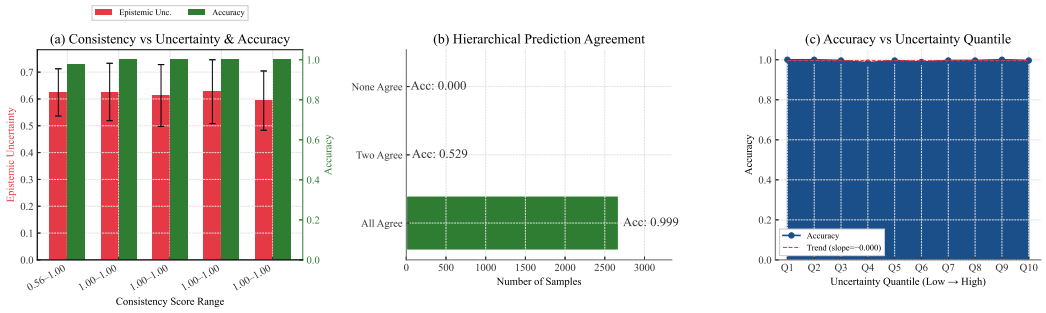


Figure 8. Hierarchical interpretability analysis: (**a**) Epistemic uncertainty and accuracy as functions of consistency score range, (**b**) sample distribution and accuracy by hierarchical prediction agreement level, (**c**) accuracy stability across uncertainty quantiles.

Figure 9a–d further evaluate the practical value of uncertainty in decision-making. Selective rejection of high-uncertainty samples can slightly improve retained accuracy, and the risk-coverage behavior remains favorable. However, the uncertainty measures show limited effectiveness for directly detecting misclassified samples, indicating that they are more useful for measuring confidence and filtering ambiguous cases than for error identification.

In summary, these results demonstrate that hierarchical consistency serves as a meaningful explanatory signal for model confidence, tightly linked with epistemic uncertainty. The model's uncertainty estimates provide reliable calibration for overall prediction stability, but their direct utility for misclassification detection remains limited under current settings.

These results confirm that the multi-level design provides inherent interpretability, and that uncertainty estimates can reliably indicate when a prediction should be distrusted—an important step toward safer autonomous diagnosis.

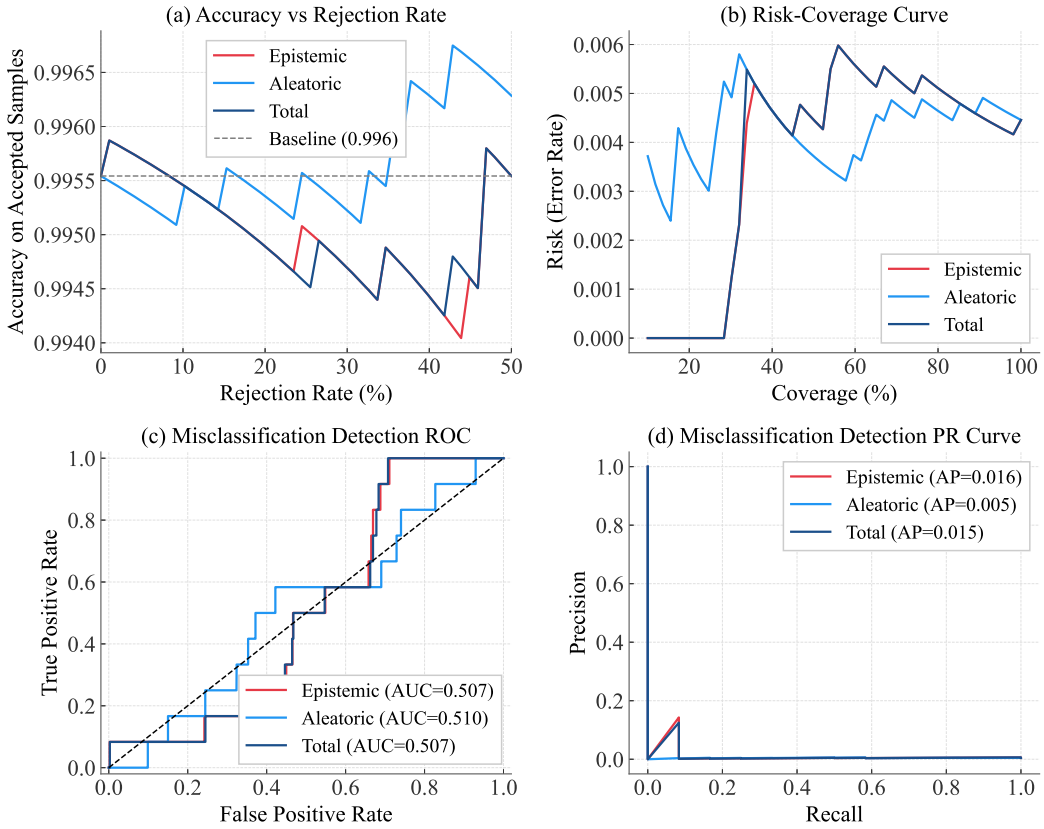


Figure 9. Uncertainty threshold decision analysis: (**a**) Accuracy improvement via selective rejection by uncertainty quantiles, (**b**) risk-coverage curve showing error rates at varying coverage, (**c**) ROC curves for misclassification detection using different uncertainty types, (**d**) precision–recall curves for misclassification detection.

Table 17. Per-class accuracy by hierarchical level.

Fault Type	Shallow (Conv)	Middle (HCB-LSTM)	Deep (Attention)
Normal	1.0000	0.9963	0.9963
Engine	0.9932	0.9945	0.9979
Aileron	0.9970	1.0000	1.0000
Elevator	1.0000	1.0000	0.9911
Rudder	1.0000	1.0000	1.0000
Multiple	0.9146	0.9390	0.9878
Average	0.9933	0.9948	0.9953

4.9. Hyperparameter Sensitivity Analysis

To investigate the robustness of the proposed HCUNet and identify optimal configurations, we conduct a comprehensive sensitivity analysis on five key hyperparameters: hidden dimension, dropout rate, learning rate, batch size, and the number of LSTM layers. Each parameter is varied independently while others remain at their default values. The results are summarized in Tables 18–22.

Table 18. Sensitivity analysis on hidden dimension.

Hidden Dim	Accuracy	F1-Macro	Epistemic	Aleatoric
64	0.9234	0.9156	0.892	0.756
128	0.9867	0.9823	0.723	0.698
192	0.9955	0.9912	0.617	0.717
256	0.9923	0.9878	0.534	0.723

Table 19. Sensitivity analysis on dropout rate.

Dropout	Accuracy	F1-Macro	Epistemic	Aleatoric
0.10	0.9912	0.9867	0.578	0.689
0.15	0.9955	0.9912	0.617	0.717
0.20	0.9934	0.9889	0.645	0.723
0.30	0.9878	0.9823	0.712	0.698
0.40	0.9756	0.9689	0.789	0.734

Table 20. Sensitivity analysis on learning rate.

Learning Rate	Accuracy	F1-Macro	Epistemic	Aleatoric
1×10^{-4}	0.9845	0.9789	0.689	0.678
5×10^{-4}	0.9923	0.9878	0.634	0.698
8×10^{-4}	0.9955	0.9912	0.617	0.717
1×10^{-3}	0.9912	0.9856	0.645	0.734
2×10^{-3}	0.9823	0.9756	0.678	0.756

Table 21. Sensitivity analysis on batch size.

Batch Size	Accuracy	F1-Macro	Epistemic	Aleatoric
16	0.9823	0.9767	0.678	0.689
32	0.9912	0.9867	0.645	0.701
64	0.9955	0.9912	0.617	0.717
128	0.9889	0.9834	0.598	0.723

Table 22. Sensitivity analysis on number of LSTM layers.

Num Layers	Accuracy	F1-Macro	Epistemic	Aleatoric
1	0.9856	0.9801	0.534	0.678
2	0.9955	0.9912	0.617	0.717
3	0.9941	0.9898	0.689	0.734

Hidden Dimension. As shown in Table 18, the model capacity significantly affects performance. A hidden dimension of 64 yields degraded accuracy (92.34%), indicating insufficient representational capacity for capturing complex fault patterns. Performance improves substantially at 128 dimensions and peaks at 192, which achieves the best accuracy (99.55%) and well-calibrated uncertainty. Larger dimensions (256) show slight performance degradation and reduced epistemic uncertainty (0.534), suggesting potential overconfidence due to excessive model capacity.

Dropout Rate. Table 19 reveals that moderate regularization is beneficial. A dropout rate of 0.15 achieves optimal performance (99.55% accuracy), while lower dropout (0.10) slightly increases overfitting risk and higher dropout (0.40) degrades both accuracy and F1-score due to excessive regularization. The results indicate that the hierarchical architecture requires carefully tuned regularization to balance model capacity and generalization.

Learning Rate. The model exhibits sensitivity to learning rate selection (Table 20). A learning rate of 8×10^{-4} yields the best accuracy (99.55%). Lower rates (1×10^{-4}) result in slower convergence and suboptimal performance, while higher rates (2×10^{-3}) introduce optimization instability and increase aleatoric uncertainty estimates, suggesting that aggressive optimization may introduce noise into the learned representations.

Batch Size. Table 21 demonstrates that a batch size of 64 achieves optimal performance (99.55% accuracy). Smaller batches (16, 32) introduce gradient noise leading to slightly lower accuracy, while larger batches (128) may impair generalization due to reduced stochasticity during training. The moderate batch size of 64 provides stable gradient estimates while maintaining sufficient randomness for effective regularization.

Number of Layers. Increasing LSTM layers from one to two improves performance significantly (Table 22), with two layers achieving the optimal accuracy of 99.55%. However, adding a third layer yields marginal degradation (99.41%), suggesting diminishing returns and potential overfitting with excessive depth. Deeper networks also exhibit higher epistemic uncertainty (0.689 vs. 0.617), reflecting increased model complexity. The two-layer configuration provides the best trade-off between representational power and generalization.

Based on this analysis, the recommended configuration for the ALFA dataset is: hidden dimension of 192, dropout rate of 0.15, learning rate of 8×10^{-4}, batch size of 64, and two LSTM layers. This configuration achieves optimal classification accuracy with well-calibrated uncertainty estimates.

4.10. Inference Latency and Deployment Analysis

Table 23 reports the inference efficiency metrics measured on three hardware platforms representative of UAV deployment scenarios.

Table 23. Inference latency and resource utilization across hardware platforms.

Method	NVIDIA RTX 3090		Jetson Xavier NX		CPU (i7-12700)
	Latency (ms)	FPS	Latency (ms)	FPS	Latency (ms)
TCN	0.82	1220	4.12	243	8.45
MC Dropout (10×)	8.34	120	41.23	24	89.67
Deep Ensemble (5×)	4.15	241	20.78	48	43.56
TimeMixer++	1.89	529	9.45	106	18.23
HCUNet (Ours)	2.34	427	11.67	86	23.45

HCUNet achieves 11.67 ms inference latency on the edge-deployable Jetson Xavier NX platform, corresponding to 86 FPS—sufficient for real-time UAV diagnostics at typical sensor sampling rates (10–100 Hz). Compared to MC Dropout (ten forward passes) and Deep Ensemble (five models), HCUNet reduces latency by 71.7% and 43.8% respectively on the Jetson platform while providing comparable or superior uncertainty quantification. The GPU memory footprint is 1.2 GB for HCUNet versus 5.8 GB for Deep Ensemble, making HCUNet practical for resource-constrained UAV onboard computers.

4.11. Validation on Standard Imbalanced Benchmark Dataset

To further assess the generalizability of HCUNet beyond the UAV fault diagnosis domain, we evaluate it on the *E. coli* protein localization dataset [57,58], a widely adopted benchmark for imbalanced multi-class classification. The dataset comprises 336 samples with seven continuous biochemical features and eight localization site classes, exhibiting

a severe class imbalance ratio of 71.50 (143 samples in the majority class cp versus only two samples each in the minority classes imL and imS).

For the *E. coli* experiments, HCUNet retains the same architectural design and hyperparameter configuration as used on the ALFA dataset (hidden dimension of 192, two HCB-LSTM layers, dropout of 0.15, AdamW optimizer with learning rate 8×10^{-4}), except that the input projection layer is adapted to the seven-dimensional biochemical feature vector and the output head is set to eight classes. Since the *E. coli* samples are static rather than sequential, each sample is treated as a length-one sequence, and the temporal modules operate on this single-step input. All baselines are trained under the identical data split and preprocessing pipeline to ensure a fair comparison.

To contextualize these results, we compare HCUNet against five representative baselines spanning distinct paradigms: XGBoost, CNN, LSTM, Evidential NN, TimeMixer++ and VibrMamba. As shown in Table 24, HCUNet consistently outperforms all baselines across every metric. It surpasses the strongest competitor, VibrMamba [55], by 0.61% in accuracy and 1.03% in F1-score. The lowest ECE of 0.0487 further indicates superior calibration, which is critical for reliable deployment on imbalanced biomedical data.

Table 24. Comparison with representative baselines on the *E. coli* dataset.

Method	Acc ↑	F1 ↑	AUC ↑	ECE ↓
XGBoost	84.52	78.34	91.23	0.1342
CNN	85.71	79.86	92.45	0.0978
LSTM	86.31	81.23	93.56	0.0834
Evidential NN	87.23	82.12	94.89	0.0612
TimeMixer++	88.12	83.45	95.67	0.0523
VibrMamba	89.29	85.12	96.45	0.0495
HCUNet (Ours)	89.90	86.15	96.84	0.0487

4.12. Validation on NASA C-MAPSS Turbofan Engine Dataset

To validate the generalization capability of HCUNet on real-world aerospace systems, we conduct experiments on the NASA C-MAPSS (Commercial Modular Aero-Propulsion System Simulation) turbofan engine degradation dataset, a widely adopted benchmark for prognostics and health management research. This dataset contains run-to-failure sensor recordings from aircraft gas turbine engines under realistic flight conditions, comprising 21 sensor measurements and three operational settings sampled throughout the engine lifecycle.

We utilize all four sub-datasets FD001–FD004 with varying operational conditions and fault modes. FD001 and FD003 operate under a single flight condition at sea level, while FD002 and FD004 involve six different operational conditions. FD001 and FD002 contain one fault mode, namely high-pressure compressor degradation, whereas FD003 and FD004 include two fault modes: HPC degradation and fan degradation. The dataset comprises 26 variables per time step, including unit number, time in cycles, three operational settings, and 21 sensor measurements. Following standard practice, we use the training subsets containing 708 engine run-to-failure trajectories, with 100, 260, 100, and 248 trajectories from FD001 through FD004 respectively. To formulate the classification task, we partition each engine trajectory into four health stages based on remaining useful life percentiles: healthy with RUL above 75%, early degradation with RUL between 50% and 75%, mid-degradation with RUL between 25% and 50%, and severe degradation with RUL at or below 25%.

Consistent with the *E. coli* experiments, HCUNet is applied to the C-MAPSS dataset using the same architectural design and hyperparameter configuration as on the ALFA dataset (hidden dimension of 192, two HCB-LSTM layers, dropout of 0.15, AdamW optimizer with learning rate 8×10^{-4}, and batch size of 64), with only the input projection and

output head adjusted to match the 24-dimensional input (three operational settings and 21 sensor measurements) and the four health-stage classes. Sequence samples are generated using the same sliding-window scheme ($L = 50$, stride 1) applied per engine trajectory, and all baselines follow an identical preprocessing and data-splitting protocol to ensure a fair comparison.

Table 25 presents the comparative results, where we report the average performance across all four sub-datasets.

Table 25. Performance comparison on NASA C-MAPSS turbofan engine dataset.

Method	Acc ↑	F1 ↑	AUC ↑	ECE ↓
XGBoost	78.45	71.23	86.34	0.1523
LSTM	83.12	77.45	90.23	0.0867
Transformer	85.67	80.12	92.45	0.0712
MC Dropout	84.23	78.89	91.12	0.0789
Deep Ensemble	87.34	82.56	94.23	0.0478
TimeMixer++	88.56	84.12	95.12	0.0389
VibrMamba	90.89	86.78	96.34	0.0298
HCUNet (Ours)	91.78	87.89	96.67	0.0267

HCUNet achieves 91.78% average accuracy across the four sub-datasets, outperforming the strongest baseline VibrMamba [55] by 0.89% in accuracy and reducing ECE by 10.4%. The performance gap compared to the ALFA dataset (99.55% vs. 91.78%) reflects the increased complexity of degradation stage classification, where temporal boundaries between health stages are inherently ambiguous. Notably, HCUNet demonstrates robust calibration (ECE = 0.0267) on this challenging task, confirming that hierarchical consistency learning generalizes effectively to aerospace prognostics applications with gradual fault evolution patterns.

5. Conclusions

To address data imbalance and operational uncertainty in UAV fault diagnosis, HCUNet integrates hierarchical feature consistency with principled uncertainty quantification, enabling both accurate fault identification and trustworthy confidence estimation. The framework extracts complementary representations through three sequential stages, namely an ADMS-CSI module for shallow multi-frequency feature extraction, a hierarchical context bridging LSTM for middle-level temporal dynamics modeling, and a RoPE-based attention module for deep semantic encoding. The three-level features are then aggregated by the LaSEA module through cross-scale dilated extraction, channel shuffle with random pooling sampling, and confidence-weighted hierarchical fusion. A hierarchical consistency loss jointly optimizes multi-level classification, inter-level agreement, and confidence calibration. Experiments on the ALFA dataset, the imbalanced E. coli benchmark, and the NASA C-MAPSS turbofan engine dataset show that HCUNet achieves higher accuracy and better uncertainty calibration than both conventional machine learning methods and existing uncertainty-aware deep learning baselines.

Author Contributions: Conceptualization, R.R. and F.J.; methodology, R.R.; software, R.R.; validation, R.R. and X.Z.; formal analysis, R.R.; investigation, R.R.; resources, F.J.; data curation, R.R.; writing—original draft preparation, R.R.; writing—review and editing, F.J. and X.Z.; visualization, R.R.; supervision, F.J.; project administration, F.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The ALFA dataset used in this study is publicly available at https://theairlab.org/alfa-dataset/ (accessed on 21 April 2026). The *E. coli* protein localization dataset is available from the UCI Machine Learning Repository at https://archive.ics.uci.edu/dataset/39/ecoli (accessed on 23 April 2026). The NASA C-MAPSS turbofan engine degradation dataset is available from the NASA Prognostics Data Repository at https://phm-datasets.s3.amazonaws.com/NASA/6.+Turbofan+Engine+Degradation+Simulation+Data+Set.zip (accessed on 20 May 2026). Code will be made available upon reasonable request.

Acknowledgments: The authors would like to thank the anonymous reviewers for their valuable comments and suggestions.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Xu, Z.; Saleh, J.H. Machine Learning for Reliability Engineering and Safety Applications: Review of Current Status and Future Opportunities. *Reliab. Eng. Syst. Saf.* **2021**, *211*, 107530. [CrossRef]
2. Zio, E. Prognostics and Health Management (PHM): Where Are We and Where Do We (Need to) Go in Theory and Practice. *Reliab. Eng. Syst. Saf.* **2022**, *218*, 108119. [CrossRef]
3. Yan, R.; Gao, R.X.; Chen, X. Wavelets for Fault Diagnosis of Rotary Machines: A Review with Applications. *Signal Process.* **2014**, *96*, 1–15. [CrossRef]
4. Zhu, H.; He, Z.; Wei, J.; Wang, J.; Zhou, H. Bearing Fault Feature Extraction and Fault Diagnosis Method Based on Feature Fusion. *Sensors* **2021**, *21*, 2524. [CrossRef] [PubMed]
5. Huang, M.; Liu, Z.; Tao, Y.; Zhang, L. Mechanical Fault Diagnosis and Prediction in IoT Based on Multi-Source Sensing Data Fusion. *Simul. Model. Pract. Theory* **2020**, *102*, 101981. [CrossRef]
6. Zhang, S.; Zhang, S.; Wang, B.; Habetler, T.G. Deep Learning Algorithms for Bearing Fault Diagnostics—A Comprehensive Review. *IEEE Access* **2020**, *8*, 29857–29881. [CrossRef]
7. Tama, B.A.; Vania, M.; Lee, S.; Lim, S. Recent Advances in the Application of Deep Learning for Fault Diagnosis of Rotating Machinery Using Vibration Signals. *Artif. Intell. Rev.* **2023**, *56*, 4667–4709. [CrossRef]
8. Jia, F.; Lei, Y.; Lin, J.; Zhou, X.; Lu, N. Deep Neural Networks: A Promising Tool for Fault Characteristic Mining and Intelligent Diagnosis of Rotating Machinery with Massive Data. *Mech. Syst. Signal Process.* **2016**, *72–73*, 303–315. [CrossRef]
9. Li, X.; Zhang, W.; Ding, Q.; Sun, J.-Q. Intelligent Rotating Machinery Fault Diagnosis Based on Deep Learning Using Data Augmentation. *J. Intell. Manuf.* **2020**, *31*, 433–452. [CrossRef]
10. Zhao, M.; Zhong, S.; Fu, X.; Tang, B.; Pecht, M. Deep Residual Shrinkage Networks for Fault Diagnosis. *IEEE Trans. Ind. Inform.* **2020**, *16*, 4681–4690. [CrossRef]
11. Li, W.; Zhong, X.; Shao, H.; Cai, B.; Yang, X. Multi-Mode Data Augmentation and Fault Diagnosis of Rotating Machinery Using Modified ACGAN Designed with New Framework. *Adv. Eng. Inform.* **2022**, *52*, 101552. [CrossRef]
12. Shao, H.; Lin, J.; Zhang, L.; Galar, D.; Kumar, U. A Novel Approach of Multisensory Fusion to Collaborative Fault Diagnosis in Maintenance. *Inf. Fusion* **2021**, *74*, 65–76. [CrossRef]
13. Ding, Y.; Jia, M.; Miao, Q.; Cao, Y. A Novel Time-Frequency Transformer Based on Self-Attention Mechanism and Its Application in Fault Diagnosis of Rolling Bearings. *Mech. Syst. Signal Process.* **2022**, *168*, 108616. [CrossRef]
14. Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; Zhang, W. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Online, 2–9 February 2021; Volume 35, pp. 11106–11115. [CrossRef]
15. Kendall, A.; Gal, Y. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In Proceedings of the Advances in Neural Information Processing Systems 30, Long Beach, CA, USA, 4–9 December 2017; pp. 5574–5584.
16. Gal, Y.; Ghahramani, Z. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016*; PMLR: London, UK, 2016; Volume 48, pp. 1050–1059.
17. Lakshminarayanan, B.; Pritzel, A.; Blundell, C. Simple and Scalable Predictive Uncertainty Estimation Using Deep Ensembles. In Proceedings of the Advances in Neural Information Processing Systems 30, Long Beach, CA, USA, 4–9 December 2017; pp. 6402–6413.

18. Ulmer, D.; Cinà, G. Know Your Limits: Uncertainty Estimation with ReLU Classifiers Fails at Reliable OOD Detection. In *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence, Online, 27–30 July 2021*; PMLR: London, UK, 2021; Volume 161, pp. 1766–1776.
19. Hüllermeier, E.; Waegeman, W. Aleatoric and Epistemic Uncertainty in Machine Learning: An Introduction to Concepts and Methods. *Mach. Learn.* **2021**, *110*, 457–506. [CrossRef]
20. Johnson, J.M.; Khoshgoftaar, T.M. Survey on Deep Learning with Class Imbalance. *J. Big Data* **2019**, *6*, 27. [CrossRef]
21. Krawczyk, B. Learning from Imbalanced Data: Open Challenges and Future Directions. *Prog. Artif. Intell.* **2016**, *5*, 221–232. [CrossRef]
22. Jiang, X.; Shen, C.; Shi, J.; Zhu, Z. Initial Center Frequency-Guided VMD for Fault Diagnosis of Rotating Machines. *J. Sound Vib.* **2018**, *435*, 36–55. [CrossRef]
23. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794. [CrossRef]
24. Kong, Y.; Qin, Z.; Wang, T.; Han, Q.; Chu, F. An Enhanced Sparse Representation-Based Intelligent Recognition Method for Planet Bearing Fault Diagnosis in Wind Turbines. *Renew. Energy* **2021**, *173*, 987–1004. [CrossRef]
25. Chen, Z.; Gryllias, K.; Li, W. Intelligent Fault Diagnosis for Rotary Machinery Using Transferable Convolutional Neural Network. *IEEE Trans. Ind. Inform.* **2020**, *16*, 339–349. [CrossRef]
26. You, K.; Gu, Y.; Shao, H.; Wang, Y. A Liquid-Impulse Neural Network Model Based on Heterogeneous Fusion of Multimodal Information for Interpretable Rotating Machinery Fault Diagnosis. *Mech. Syst. Signal Process.* **2026**, *246*, 113923. [CrossRef]
27. You, K.; Shao, H.; Chen, Z.; Yang, J.; Du, X.; Lin, Y.; Liu, H. Multimodal Physical-Information Constraints LLM for Fault Diagnosis of Pressurized Water Reactor Coolant System in Under-Sampled and Imbalanced Data. *J. Ind. Inf. Integr.* **2026**, *52*, 101143. [CrossRef]
28. You, K.; Gu, Y.; Li, Y.; Wang, Y. A Novel Physical Constraint-Guided Quadratic Neural Networks for Interpretable Bearing Fault Diagnosis under Zero-Fault Sample. *Nondestruct. Test. Eval.* **2025**, 1–31. [CrossRef]
29. You, K.; Li, C.; Li, Y.; Qin, G.; Gu, Y. DTMPI-DIVR: A Digital Twins for Multi-Margin Physical Information via Dynamic Interaction of Virtual and Real Sound-Vibration Signals for Bearing Fault Diagnosis without Real Fault Samples. *Expert Syst. Appl.* **2025**, *292*, 128592. [CrossRef]
30. You, K.; Wang, P.; Huang, P.; Gu, Y. A Sound-Vibration Physical-Information Fusion Constraint-Guided Deep Learning Method for Rolling Bearing Fault Diagnosis. *Reliab. Eng. Syst. Saf.* **2025**, *253*, 110556. [CrossRef]
31. Mizele, B.C.F.; Meliho, M.; Houndji, V.R.; Ahouandjinou, S.A.R.M.; Orlando, C.A. Spatiotemporal Modeling and Uncertainty Quantification of Reference Evapotranspiration Using Machine Learning and Bayesian Model Averaging in Benin. *Geomatics* **2026**, *6*, 73. [CrossRef]
32. Weinberger, K.; Müller, D.; Mönnigmann, M.; Sezgin, A. Uncertainty Propagation and Minimization for Channel Estimation in UAV-Mounted RIS Systems. In Proceedings of the ICC 2025—IEEE International Conference on Communications, Montreal, QC, Canada, 8–12 June 2025; pp. 638–643. [CrossRef]
33. Abdar, M.; Pourpanah, F.; Hussain, S.; Rezazadegan, D.; Liu, L.; Ghavamzadeh, M.; Fieguth, P.; Cao, X.; Khosravi, A.; Acharya, U.R.; et al. A Review of Uncertainty Quantification in Deep Learning: Techniques, Applications and Challenges. *Inf. Fusion* **2021**, *76*, 243–297. [CrossRef]
34. Jospin, L.V.; Laga, H.; Boussaid, F.; Buntine, W.; Bennamoun, M. Hands-On Bayesian Neural Networks—A Tutorial for Deep Learning Users. *IEEE Comput. Intell. Mag.* **2022**, *17*, 29–48. [CrossRef]
35. Tutsoy, O.; Asadi, D.; Ahmadi, K.; Nabavi-Chasmi, S.-Y. Robust Reduced Order Thau Observer with the Adaptive Fault Estimator for the Unmanned Air Vehicles. *IEEE Trans. Veh. Technol.* **2023**, *72*, 1601–1610. [CrossRef]
36. Li, T.; Zhao, Z.; Sun, C.; Yan, R.; Chen, X. Multireceptive Field Graph Convolutional Networks for Machine Fault Diagnosis. *IEEE Trans. Ind. Electron.* **2021**, *68*, 12739–12749. [CrossRef]
37. Zhang, W.; Li, X.; Jia, X.; Ma, H.; Luo, Z.; Li, X. Machinery Fault Diagnosis with Imbalanced Data Using Deep Generative Adversarial Networks. *Measurement* **2020**, *152*, 107377. [CrossRef]
38. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8759–8768. [CrossRef]
39. Zhang, K.; Chen, J.; Zhang, T.; Zhou, Z. A Compact Convolutional Neural Network Augmented with Multiscale Feature Extraction of Acquired Monitoring Data for Mechanical Intelligent Fault Diagnosis. *J. Manuf. Syst.* **2020**, *55*, 273–284. [CrossRef]
40. Gou, J.; Yu, B.; Maybank, S.J.; Tao, D. Knowledge Distillation: A Survey. *Int. J. Comput. Vis.* **2021**, *129*, 1789–1819. [CrossRef]
41. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 10012–10022. [CrossRef]
42. Su, J.; Lu, Y.; Pan, S.; Murtadha, A.; Wen, B.; Liu, Y. RoFormer: Enhanced Transformer with Rotary Position Embedding. *Neurocomputing* **2024**, *568*, 127063. [CrossRef]

43. Han, K.; Wang, Y.; Chen, H.; Chen, X.; Guo, J.; Liu, Z.; Tang, Y.; Xiao, A.; Xu, C.; Xu, Y.; et al. A Survey on Vision Transformer. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 87–110. [CrossRef] [PubMed]
44. Lin, T.; Wang, Y.; Liu, X.; Qiu, X. A Survey of Transformers. *AI Open* **2022**, *3*, 111–132. [CrossRef]
45. Howard, A.G.; Sandler, M.; Chen, B.; Wang, W.; Chen, L.-C.; Tan, M.; Chu, G.; Vasudevan, V.; Zhu, Y.; Pang, R.; et al. Searching for MobileNetV3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324. [CrossRef]
46. Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125. [CrossRef]
47. Niu, Z.; Zhong, G.; Yu, H. A Review on the Attention Mechanism of Deep Learning. *Neurocomputing* **2021**, *452*, 48–62. [CrossRef]
48. Yu, F.; Koltun, V. Multi-Scale Context Aggregation by Dilated Convolutions. In Proceedings of the 4th International Conference on Learning Representations, San Juan, Puerto Rico, 2–4 May 2016.
49. Ma, N.; Zhang, X.; Zheng, H.-T.; Sun, J. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 116–131. [CrossRef]
50. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
51. Gawlikowski, J.; Tassi, C.R.N.; Ali, M.; Lee, J.; Humt, M.; Feng, J.; Kruspe, A.; Triebel, R.; Jung, P.; Roscher, R.; et al. A Survey of Uncertainty in Deep Neural Networks. *Artif. Intell. Rev.* **2023**, *56*, 1513–1589. [CrossRef]
52. Depeweg, S.; Hernández-Lobato, J.M.; Doshi-Velez, F.; Udluft, S. Decomposition of Uncertainty in Bayesian Deep Learning for Efficient and Risk-Sensitive Learning. In *Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018*; PMLR: London, UK, 2018; Volume 80, pp. 1184–1193.
53. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988. [CrossRef]
54. Guo, C.; Pleiss, G.; Sun, Y.; Weinberger, K.Q. On Calibration of Modern Neural Networks. In *Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017*; PMLR: London, UK, 2017; Volume 70, pp. 1321–1330.
55. Yi, H.; Li, D.; Lu, Z.; Jin, Y.; Duan, H.; Hou, L.; Duraihem, F.Z.; Awwad, E.M.; Saeed, N.A. VibrMamba: A Lightweight Mamba Based Fault Diagnosis of Rotating Machinery Using Vibration Signal. *Measurement* **2025**, *249*, 116881. [CrossRef]
56. Wang, S.; Wu, H.; Shi, X.; Hu, T.; Luo, H.; Ma, L.; Zhang, J.Y.; Zhou, J. TimeMixer: Decomposable Multiscale Mixing for Time Series Forecasting. In Proceedings of the Twelfth International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
57. Nakai, K.; Kanehisa, M. Expert System for Predicting Protein Localization Sites in Gram-Negative Bacteria. *Proteins Struct. Funct. Bioinform.* **1991**, *11*, 95–110. [CrossRef] [PubMed]
58. Horton, P.; Nakai, K. A Probabilistic Classification System for Predicting the Cellular Localization Sites of Proteins. In Proceedings of the Fourth International Conference on Intelligent Systems for Molecular Biology, St. Louis, MO, USA, 12–15 June 1996; Volume 4, pp. 109–115.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.