

KAN-FIF: Spline-Parameterized Lightweight Physics-based Tropical Cyclone Estimation on Meteorological Satellite

Jiakang Shen

Shandong University
School of Control Science and
Engineering
Jinan, Shandong, China
202300171054@mail.sdu.edu.cn

Chenrui Xu

Shandong University
School of Control Science and
Engineering
Jinan, Shandong, China
202300171055@mail.sdu.edu.cn

Jinglin Zhang*

Qinghui Chen

Shandong University
School of Control Science and
Engineering
Jinan, Shandong, China
202420785@mail.sdu.edu.cn

Feng Zhang

Fudan University
Department of Atmospheric and
Oceanic Sciences and Institutes of
Atmospheric Sciences
Shanghai, China
fengzhang@fudan.edu.cn

Runtong Wang

Shandong University
School of Control Science and
Engineering
Jinan, Shandong, China
202300171170@mail.sdu.edu.cn

Cong Bai

Zhejiang University of Technology
College of Computer Science
Hangzhou, Zhejiang, China
congbai@zjut.edu.cn

Abstract

Tropical cyclones (TC) are among the most destructive natural disasters, causing catastrophic damage to coastal regions through extreme winds, heavy rainfall, and storm surges. Timely monitoring of tropical cyclones is crucial for reducing loss of life and property, yet it is hindered by the computational inefficiency and high parameter counts of existing methods on resource-constrained edge devices. Current physics-guided models suffer from linear feature interactions that fail to capture high-order polynomial relationships between TC attributes, leading to inflated model sizes and hardware incompatibility. To overcome these challenges, this study introduces the Kolmogorov–Arnold Network-based Feature Interaction Framework (KAN-FIF), a lightweight multimodal architecture that integrates MLP and CNN layers with spline-parameterized KAN layers. For Maximum Sustained Wind (MSW) prediction, experiments demonstrate that the KAN-FIF framework achieves a 94.8% reduction in parameters (0.99MB vs 19MB) and 68.7% faster inference per sample (2.3ms vs 7.35ms) compared to baseline model Phy-CoCo, while maintaining superior accuracy with 32.5% lower MAE. The offline deployment experiment of the FY-4 series meteorological satellite processor on the Qingyun-1000 development

*Corresponding author

board achieved a 14.41ms per-sample inference latency with the KAN-FIF framework, demonstrating promising feasibility for operational TC monitoring and extending deployability to edge-device AI applications. The code is released at <https://github.com/Jinglin-Zhang/KAN-FIF>.

CCS Concepts

- Applied computing → Environmental sciences; • Computer systems organization → Embedded software; • Computing methodologies → Neural networks; Image manipulation.

Keywords

Tropical Cyclone Estimation; FY-4 series meteorological satellite; Kolmogorov–Arnold Network; Edge-device AI applications; Multimodal Feature

ACM Reference Format:

Jiakang Shen, Qinghui Chen, Runtong Wang, Chenrui Xu, Jinglin Zhang, Cong Bai, and Feng Zhang. 2026. KAN-FIF: Spline-Parameterized Lightweight Physics-based Tropical Cyclone Estimation on Meteorological Satellite. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1 (KDD '26), August 09–13, 2026, Jeju Island, Republic of Korea*. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3770854.3783929>

1 Introduction

Tropical cyclones (TC), recognized as highly destructive meteorological phenomena, inflict severe damage upon coastal regions globally. Characterized by extreme wind velocities, intense precipitation, and elevated sea levels, these recurrent events primarily



This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

KDD '26, Jeju Island, Republic of Korea

© 2026 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-2258-5/2026/08

<https://doi.org/10.1145/3770854.3783929>

devastate infrastructure and communities through destructive sustained winds, extreme rainfall inducing catastrophic flooding, and devastating storm surges that overwhelm coastal defenses and cause widespread inundation. Consequently, accurate and timely monitoring of Tropical Cyclone (TC) intensity and the radius of peak winds is essential for effective disaster risk management, reliable warnings, urgent protective actions, and the guidance of evacuations.

Ground-centralized inference and edge-device inference are the two most popular TC prediction methods. The ground-centralized method involves multiple data transmission steps among satellite-to-ground and ground networks, resulting in high latency and computational costs. Edge-device inference faces stringent constraints in processing capability, memory, and operator compatibility. These limitations hinder the adoption of state-of-the-art MLP-based neural networks with a large number of CNN layers included, which are computationally intensive and require specialized hardware support.

Early TC estimation methods relied on satellite imagery combined with empirical algorithms such as the Dvorak technique[24]. These approaches suffered from subjectivity and limited accuracy due to insufficient feature representation. With advances in deep learning, CNN-based models [25, 28] have dominated the estimation of TC attributes using spatial patterns in satellite data. However, most related works focus on the prediction of a single task, neglecting the inherent physical relationships between the TC attributes. Recent multitask learning (MTL) methods[13] attempt to share parameters among tasks, but they risk negative transfer and oversimplified feature interactions. Some works introduced physics-based constraints to model task correlations[30], yet their linear feature concatenation fails to capture high-order polynomial relationships between variables while inflating parameter counts, further straining edge-device inference. Meanwhile, multiscale Xception networks with dual attention[15] and YOLO-NAS architectures[17] achieve breakthroughs in fast satellite data processing by infrared-water vapor fusion. However, these deep learning approaches have advanced TC estimation accuracy at the cost of substantial computational demand.

Integrating multimodal auxiliary data with satellite imagery has emerged as a promising trend [8]. However, related studies usually treat temporal features as static inputs rather than modeling sequential dependencies. Recurrent neural networks (RNN) and long-short-term memory networks (LSTM)[5, 12] have been applied to track prediction. The FuXi-based perturbation generator demonstrates improved TC track prediction by optimizing initial uncertainties [21]. It operates independently of satellite data processing and requires extensive historical data for testing, creating notable constraints for field-deployed edge devices with limited memory capacity and demanding temporal requirements for timely tropical cyclone disaster monitoring.

These accumulated limitations pose three critical challenges for timely tropical cyclone prediction: Firstly, current prediction methodologies predominantly rely on computationally intensive MLP/CNN architectures augmented with specialized detection heads and preprocessing filters, which substantially inflate parameter counts and necessitate high-bandwidth data transmission. Secondly,

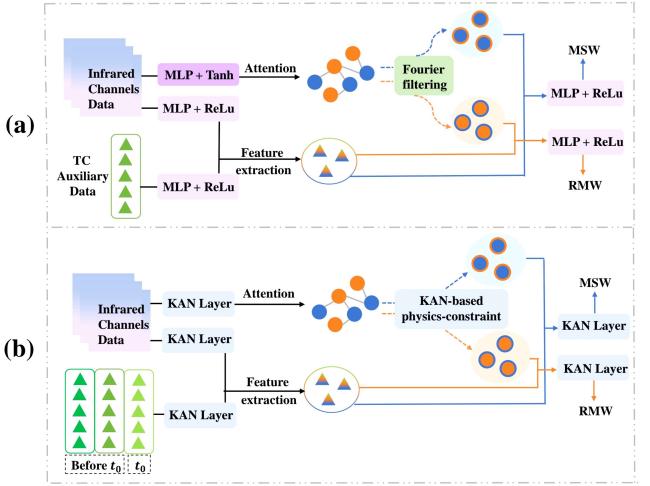


Figure 1: The comparison of frameworks between the MLP-based and KAN-based models. (a) Conventional MLP-based model (Phycoco) (b) our KAN-based lightweight model

satellite data downlinking faces intrinsic constraints, such as asynchronous data accessibility due to confinement to orbital passes over specific regions, coupled with severe bandwidth limitations that restrict transmission rates. Thirdly, although edge-device inference offers potential solutions to downlink delays by enabling inter-satellite data transfer, processing, and inference entirely in-orbit with only minimal results relayed to ground stations, fundamental hardware barriers persist. These barriers encompass restricted computational capacity, insufficient memory resources, and framework-device compatibility issues, which collectively demand ultra-lightweight model architectures and preclude sophisticated multimodal preprocessing algorithms.

To address these challenges, the proposed **KAN-FIF** framework targets efficient and accurate tropical cyclone (TC) estimation on resource-constrained edge devices, with three core contributions:

- **Lightweight deployment via Kolmogorov-Arnold network (KAN) layer substitution.** Traditional multilayer perceptrons (MLPs), convolutional neural networks (CNNs), and filtering operations in feature extraction, physical constraints, and decoding stages are replaced by computationally efficient KAN layers, significantly compressing model complexity while retaining superior accuracy, as shown in Figure 1.
- **Physics-based fusion for cross-modal dependencies.** Sequential features and infrared imagery are fused through a hybrid encoder to capture nonlinear cross-modal couplings. Simultaneously, a differentiable physics-constraint module fits high-dimensional polynomial relationships between intensity and size by embedding domain-specific equations.
- **On-Orbit Edge-device Inference Capability Validation.** Experimental offline deployment of the FY-4 meteorological satellite processor on the Qingyun-1000 development board yielded a **14.41ms** latency per inference sample, with

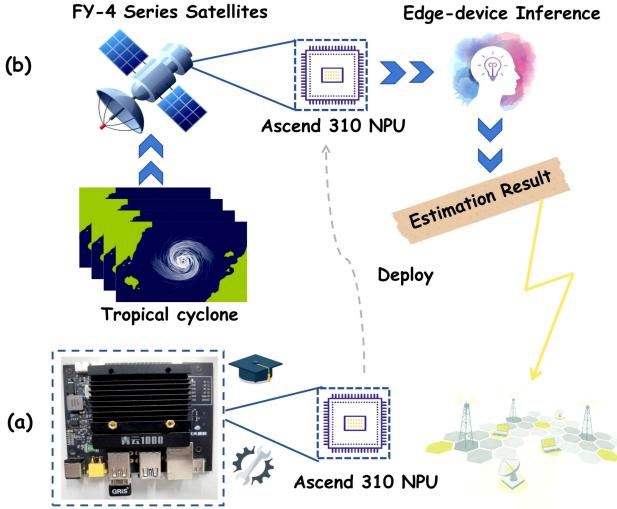


Figure 2: The offline deployment verification process of the FY-4 series satellite processor on the Qingyun-1000 development board. (a) Deployment verification on Ascend 310 NPU (b) Edge-device inference process of tropical cyclone estimation on FY-4 series satellite

the verification process depicted in Figure 2. Through architectural refinements, edge-computing constraints were mitigated, enabling high-performance, timely tropical cyclone monitoring and demonstrating promising potential for operational deployment onboard satellites.

2 Related Work

Despite advances in developing hardware-aware computational methods, persistent constraints continue to challenge multimodal data processing in tropical cyclone estimation. Consequently, advancing lightweight architectures must become a research priority to meet edge-device inference demands.

2.1 Tropical Cyclone Estimation

Tropical cyclone estimation has long been a focal point of extensive research, encompassing key attributes such as Maximum Sustained Wind (MSW), the radius of maximum winds (RMW), the spatial extent of impact, and associated precipitation volumes. Related models primarily utilize two data sources: infrared (IR) channel data from satellites, and auxiliary information comprising geographical metadata, seasonal context, and temporal characteristics. Although track prediction relies heavily on spatiotemporal sequences, the estimation of other attributes of TC usually depends on high-dimensional IR satellite imagery, like DeepMicroNet, TCIEnet, and TCICEnet[25, 28, 29]. Recently, multi-task prediction frameworks have emerged that incorporate physical priors to constrain the output from multiple tasks[9, 23]. Some studies employ encoder-decoder fusion architectures or integrate frequency domain features through filtering operations[4, 26, 27]. However, these approaches

usually rely on substantial computational resources and extensive parameters to capture non-linear relationships.

2.2 Edge-device inference

Edge-device inference offers a promising alternative to ground-centralized processing by mitigating limitations such as intermittent connectivity and restricted data transmission capacity. Early research targeted basic inference tasks, exemplified by CubeSat cloud segmentation studies [16]. Subsequent missions, exemplified by NASA’s IPEX mission, achieving real-time classification [1], revealed persistent computational constraints onboard. The rise of deep learning substantially advanced onboard capabilities, with recent missions validating high-accuracy cloud detection using optimized deep learning models [7]. Researchers subsequently explored efficient architectures through three pathways: First, lightweight models deployed on microcontrollers [18, 19]; Second, quantized networks achieving real-time performance on FPGAs [20]; and Third, specialized designs such as row-wise stream processors [2].

Separately in meteorological forecasting, lightweight models have optimized efficiency-accuracy tradeoffs. Tian et al. developed a multitask network termed TC-MTLNet with adaptive loss balancing for joint tropical cyclone intensity and size estimation, reducing errors while minimizing computational overhead [23]. Similarly, Shang et al. proposed CDC-Net, which leverages channel dilation combined with feature copying to enable rapid satellite image classification under strict computational constraints [22]. Building on these advances, our method integrates Kolmogorov-Arnold Network layers as substitutes for conventional CNN and MLP components. This design eliminates preprocessing filtration requirements while achieving dramatic parameter reduction, substantially enhancing feasibility for edge deployment.

3 Problem Definition

Our model targets predictions of two critical tropical cyclone attributes: Maximum Sustained Wind (MSW) and Radius of Maximum Wind (RMW), learning parametric mappings that:

$$\begin{aligned} f_{\text{msw}} : (X_{\text{seq}}, X_{\text{img}}) &\rightarrow \hat{Y}_{\text{msw}} \\ f_{\text{rmw}} : (X_{\text{seq}}, X_{\text{img}}) &\rightarrow \hat{Y}_{\text{rmw}} \end{aligned} \quad (1)$$

The TC estimation problem is formulated as:

$$\theta^* = \arg \max_{\theta} \left[\alpha \cdot L(f_{\text{msw}}(X_{\text{seq}}, X_{\text{img}}; \theta), Y_{\text{msw}}) + \beta \cdot L(f_{\text{rmw}}(X_{\text{seq}}, X_{\text{img}}; \theta), Y_{\text{rmw}}) \right] \quad (2)$$

where θ represents all the learnable parameters. α and β represent the adjustable weights of the tasks. $L(\cdot)$ denotes the MAE loss function. Y_{msw} and Y_{rmw} represent ground truth values. $\hat{Y}_{\text{msw}} \in [19, 170]$ knots and $\hat{Y}_{\text{rmw}} \in [5, 200]$ nmi are denormalized through min-max scaling. The input data consists of two modalities: 1) Temporal sequence data $X_{\text{seq}} \in \mathbb{R}^{T \times 5}$ containing the evolution features of the TC (latitude x^{lat} , longitude x^{lon} , the time since the TC was named x^t , previous category x^{cat} and central pressure x^{pre}) over $T = 3$ consecutive time steps, with a temporal resolution of 3 hours

between steps. 2) Satellite image data $X_{img} \in \mathbb{R}^{8 \times H \times W}$ representing multichannel infrared observations with spatial dimensions $H = 156$, $W = 156$. The first four channels $X_{img}^{ch_1-4}$ correspond to observations from 3 hours before the target prediction time, while the last four channels $X_{img}^{ch_5-8}$ represent current-time imagery.

4 Methods

In this study, we integrate KAN layers in four critical aspects of our architecture, as shown in Figure 3, in order to design a framework that takes into account both lightweight deployability and prediction accuracy: a) **Shared Feature Extraction**: We employ KAN-LSTM for temporal feature extraction and KAN-CNN for spatial feature extraction from multi-spectral satellite imagery. b) **Attention Encoding**: When computing center-aware spatial attention, we utilize KAN layers to encode both distance features and spatial patterns, replacing traditional linear+tanh encoding. c) **Physical Constraints**: We directly implement inter-task physical constraints through KAN layers to fit polynomial relationships among task features, eliminating conventional convolution, MLP, Fourier transform, and Kalman filtering operations. d) **Feature Fusion and Decoding**: We utilize the KAN layers to fuse and decode shared features, task-specific features, and physical constraint features to obtain the final prediction outputs. Note that the full KAN-FIF architecture employs an LSTM layer followed by KAN projection for temporal features, while the deployment variant (Section 5) removes the LSTM due to NPU incompatibility.

4.1 Kolmogorov-Arnold Networks

Kolmogorov-Arnold networks (KANs) introduce a novel neural network framework inspired by the Kolmogorov-Arnold representation theorem [14]. Unlike traditional Multi-Layer Perceptrons (MLPs) that employ fixed node activation functions, KANs utilize spline-approximated learnable activations parameterized on edges. This architectural innovation enables KANs to model complex non-linear relationships with exceptional parameter efficiency, offering distinct advantages for high-dimensional data modeling. The foundation of KANs lies in the Kolmogorov-Arnold theorem, which asserts that any continuous multivariate function can be decomposed into a finite composition of univariate functions and additive operations. Mathematically, for a function $f : [0, 1]^n \rightarrow \mathbb{R}$, this is expressed as:

$$f(x_1, \dots, x_n) = \sum_{q=1}^{2n+1} \varphi_q \left(\sum_{p=1}^n \varphi_{q,p}(x_p) \right) \quad (3)$$

where $\varphi_{q,p}$ and φ_q are univariate functions.

KAN layers fundamentally replace linear weight matrices with spline-parameterized edges. This architectural shift enables adaptive modeling of high-order polynomial relationships, where MLP and CNN architectures struggle due to fixed activation functions constraining expressiveness. By leveraging locally adaptive polynomial segments through edge-based parameterization, KAN layers efficiently capture intricate nonlinearities without requiring excessive network depth. Crucially, as KAN layers directly optimize inter-layer function parameters, their outputs form strictly expressible composite polynomials, facilitating significantly higher interpretability than MLPs through explicit exploration of variable

relationships. Each edge in a KAN layer, as shown in Figure 3(e), represents a univariate function $\varphi_{q,p}$ parameterized as:

$$\varphi_{q,p}(x) = w \cdot (\text{silu}(x) + \text{spline}(x)) \quad (4)$$

where $\text{silu}(x) = \frac{x}{1+e^{-x}}$ is a self-gating activation function and $\text{spline}(x)$ is a piecewise polynomial curve optimized via gradient descent.

In subsequent implementations, the KAN layer can be treated as a linear layer, employing fixed polynomial basis functions with $\text{grid_size}=5$ and $\text{spline_order}=3$. Here, grid_size specifies the number of intervals used to partition the input domain for the spline approximation, where each interval corresponds to a locally defined polynomial segment. The spline_order parameter controls the degree of the polynomial used in each interval, with $\text{spline_order}=3$ corresponding to cubic splines.

4.2 KAN-based Shared Feature Extraction

The Shared Feature Extraction Module implements parallel temporal-spatial feature extraction through dual pathways. **KAN-LSTM** processes sequential TC evolution data and **KAN-CNN** extracts multi-scale spatial patterns from infrared imagery.

4.2.1 Temporal Feature Extraction with KAN-LSTM. For temporal input $X_{seq} \in \mathbb{R}^{B \times T \times 5}$ containing B samples of 3-step TC evolution features, we first employ an LSTM layer to capture temporal dependencies, and then a KAN projection extracts the features:

$$F_{seq} = \text{KAN}_{\text{Linear}}^{64 \rightarrow 32}(F_{\text{LSTM}}) \in \mathbb{R}^{B \times 32} \quad (5)$$

4.2.2 Spatial Feature Extraction with KAN-CNN. For satellite imagery $X_{img} \in \mathbb{R}^{B \times 8 \times H \times W}$, we design a Multi-Scale ConvBlock with residual enhancement:

$$\begin{aligned} F_{\text{conv}}^{(1)} &= \text{ReLU}(\text{Conv2D}_{8 \rightarrow 16}^{5 \times 5}(X_{img})) \\ F_{\text{conv}}^{(2)} &= \text{MaxPool}(\text{ReLU}(\text{Conv2D}_{16 \rightarrow 32}^{3 \times 3}(F_{\text{conv}}^{(1)}))) \\ F_{\text{res}} &= \text{Conv2D}_{32 \rightarrow 64}^{1 \times 1}(F_{\text{conv}}^{(2)}) \quad (\text{Residual path}) \\ F_{\text{multiscale}} &= \text{concat} \left[F_{\text{res}}, \sum_{d=1}^3 \text{DilatedConv}_{32 \rightarrow 32}^{3 \times 3}(F_{\text{conv}}^{(2)}) \right] \end{aligned} \quad (6)$$

The compressed spatial features are obtained as follows:

$$F_{img} = \text{KAN}_{\text{Linear}}^{256 \rightarrow 32}(\text{Flatten}(F_{\text{multiscale}})) \in \mathbb{R}^{B \times 32} \quad (7)$$

4.2.3 Final Shared Representation. The final shared representation combines temporal and spatial features:

$$F_{\text{shared}} = \text{concat}[F_{\text{seq}}, F_{img}] \in \mathbb{R}^{B \times 64} \quad (8)$$

4.3 KAN-based Task-Specific Feature Extraction

Traditional attention mechanisms for TC estimation often employ linear layers with static activation functions (ReLU or tanh) to encode spatial distances and cloud patterns. However, these methods struggle to model high-order polynomial relationships between annular cloud features and TC attributes. In our KAN-Attention, we replace linear layers with KAN layers in spatial distance encoding and content-aware feature projection(Algorithm 1).

We selected channel 7 ($10.4 \mu\text{m}$ infrared band from Himawari-8) as spatial input because it optimally captures cold cloud tops in

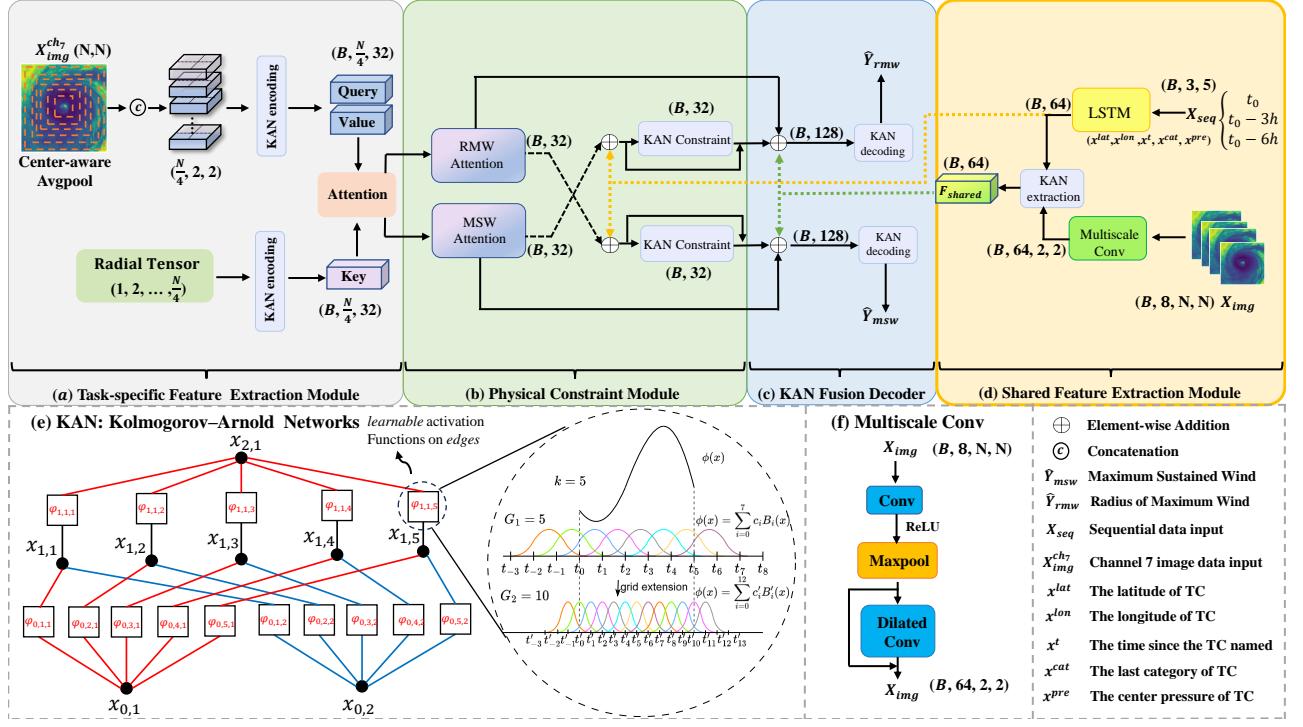


Figure 3: The architecture of the KAN-FIF learning framework: (a) The Task-specific Feature Extraction Module uses KAN layer and center-aware attention to extract the task features of MSW and RMW respectively (b) The Physical Constraint Module is designed to conduct constraints among task-specific features ; (c) The KAN Fusion Decoder fuse the features from(a)(b)(d) and obtain the final output ; (d) The Shared Feature Extraction Module take the multi-channel image and the temporal sequence data as input and obtain the shared feature between TC tasks; (e) the architecture of KAN layers; (f) the architecture of Multiscale Conv

TC eye-wall regions, where lower brightness temperatures correlate strongly with convective intensity and MSW/RMW values[27]. Our preprocessing employs adaptive annular pooling across 39 concentric rings to extract hierarchical cloud features, generating content-aware queries (Q) and values (V). Radial distances are encoded through a KAN layer to produce position-sensitive keys (K).

4.4 Physics-Guided Constraint Modeling

To establish meteorologically consistent relationships between MSW and RMW predictions, we design bidirectional residual connections governed by KAN layers. Let $A_{msw} \in \mathbb{R}^{B \times 32}$ and $A_{rmw} \in \mathbb{R}^{B \times 32}$ be the task-specific features. The physics-guided interaction is formulated as follows.

$$\begin{aligned} \Gamma_{msw \rightarrow rmw}(A_{msw}) &= A_{rmw} + K_{msw2rmw}(A_{msw}) \\ \Gamma_{rmw \rightarrow msw}(A_{rmw}) &= A_{msw} + K_{rmw2msw}(A_{rmw}) \end{aligned} \quad (9)$$

where $K(\cdot)$ represents a KAN layer that implements a dimensional mapping. $K_{msw2rmw}$ and $K_{rmw2msw}$ employ independent spline bases to model distinct physical mechanisms: $K_{msw2rmw}$ learns wind-driven radius expansion or contraction patterns and $K_{rmw2msw}$ captures radius-modulated wind intensification. This formulation preserves the primary task features through residual connections while injecting physically plausible interactions between TC attributes.

4.5 Multimodal Feature Fusion

Final predictions fuse three categories of features:

- (1) Task-specific features (A_{msw}, A_{rmw})
- (2) Physics-constrained features ($\Gamma_{msw \rightarrow rmw}, \Gamma_{rmw \rightarrow msw}$)
- (3) Shared spatiotemporal embeddings F_{shared}

The fusion and decoding process is defined as follows:

$$\begin{aligned} \bar{Y}_{msw} &= D_{msw}(\text{cat}[A_{msw}, \Gamma_{rmw \rightarrow msw}, F_{shared}]) \\ \bar{Y}_{rmw} &= D_{rmw}(\text{cat}[A_{rmw}, \Gamma_{msw \rightarrow rmw}, F_{shared}]) \end{aligned} \quad (10)$$

where $D(\cdot)$ denotes a KAN decoder which can be configured as follows.

$$D_{task}(x) = \text{KAN}_{\text{Linear}}^{128 \rightarrow 1}(x) \quad (11)$$

5 Deployment Preparation for TC Estimation

5.1 Hardware and Software Environment

The proposed KAN-FIF framework was experimentally deployed on the Qingyun-1000 development board for the FY-4 meteorological satellite processor, and three edge-deployment constraints were overcome: limited operator compatibility for neural network layers, hardware acceleration dependencies requiring static computation graphs, and memory limitations necessitating model compression to prevent runtime failures. The edge-device deployment lever-

Algorithm 1 KAN-Attention

Require: $F_{\text{seq}} \in \mathbb{R}^{B \times 32}$: Temporal features
Require: $X_{\text{img}}^{ch_7} \in \mathbb{R}^{B \times 1 \times H \times W}$: Channel 7 image
Output $A_{\text{task}} \in \mathbb{R}^{B \times 32}$

```

1: // Center-aware Avgpool
2: Initialize  $r_{\text{center}} \leftarrow 77$ 
3: for  $i \in \{0, \dots, 38\}$  do
4:    $L \leftarrow r_{\text{center}} - 2i, R \leftarrow r_{\text{center}} + 2i$ 
5:    $R_i \leftarrow X_{\text{img}}[:, :, L : R, L : R]$ 
6:    $P_i \leftarrow \text{AdaptiveAvgPool}(R_i, (2, 2))$ 
7:    $P \leftarrow \text{Concat}(P, \text{Flatten}(P_i))$ 
8: end for
9: Initialize  $G \leftarrow \text{linspace}(0, 1, 39)$ 
10: // KAN-based encoding
11:  $K \leftarrow \text{KAN\_Linear}(G)$  {Spatial distance encoding}
12:  $Q, V \leftarrow \text{Split}(\text{KAN\_Linear}(P), 2)$  {Content encoding}
13: // Multi-head attention
14:  $Q_h, K_h, V_h \leftarrow \text{SplitHeads}(Q, K, V, \text{num\_heads} = 4)$ 
15:  $\text{Attn} \leftarrow \text{Softmax}\left(Q_h @ K_h^T / \sqrt{d}\right)$ 
16:  $C_h \leftarrow \text{Attn} @ V_h, C \leftarrow \text{MergeHeads}(C_h)$ 
17: // Temporal fusion
18:  $A_{\text{task}} \leftarrow \text{KAN\_Linear}(\text{Concat}(C_{\text{avg}}, F_{\text{seq}}))$ 

```

Table 1: Hardware and Software Deployment Specifications

Category	Specification
Hardware Platform	Qingyun-1000 board
Acceleration Module	Atlas 200I A2
Processor	Huawei Ascend 310 NPU
Memory	8GB LPDDR4
Compute Capacity	22 TOPS @ INT8/ 11 TOPS @ FP16
Power Budget	<10W
Acceleration Stack	CANN 6.0.1
Environment	Python 3.7

ages Huawei’s AscendCL framework to replace standard PyTorch inference with a static execution paradigm. Key adaptations involve initializing inference sessions via InferSession for loading precompiled OM models, enforcing FP16 tensor precision to ensure NPU compatibility, and implementing asynchronous I/O queues to minimize latency. Unlike GPU-based inference relying on dynamic computation graphs, this deployment pre-allocates fixed-size input buffers for satellite imagery while eliminating runtime branching operations.

5.2 Model Conversion Workflow

The original KAN-LSTM hybrid architecture underwent redesign to eliminate NPU-incompatible LSTM operators. In its deployment version, flattened temporal sequences are processed directly through an expanded KAN network, bypassing recurrent computations entirely.

To ensure compliance with the NPU’s static computation graph requirements, all conditional branching operations were removed to establish a fixed execution path. Key hyperparameters—including the spline grid_size (fixed at 5) and spline_order (fixed at 3)—were hardcoded, while B-spline basis functions underwent precomputation during model initialization. Adaptive pooling layers were replaced by fixed-stride average pooling, preventing dynamic output shapes. Collectively, these modifications guarantee deterministic memory allocation and operator sequencing during inference.

For addressing the ATC compiler’s 63-kernel-size limit on pooling operations, the deployment version employs a two-stage pooling strategy.

While our offline validation demonstrates promising performance on hardware identical to the FY-4 satellite processor, actual on-orbit deployment may face additional challenges including data pipeline integration, radiation hardening, power management, and command-control systems that require further engineering validation.

6 Experiments

6.1 Experimental Setup

6.1.1 Dataset and Preprocessing. The model is evaluated using the Tropical Cyclone Multi-Modal Dataset (TCMM), which shares data sources with the baseline study [10, 27] but incorporates temporal sequencing. This dataset integrates infrared brightness temperature from Himawari-8 satellite channels 7, 8, 13, 15 [3] and tropical cyclone track records from the IBTrACS repository [6, 11], covering the period 2015–2022. Each sample comprises an 8-channel 156×156 infrared image normalized to $[0, 1]$, paired with five temporal features: (a) center position (latitude/longitude), (b) hours since cyclone naming, (c) prior storm category (0-5 scale with -1 for unnamed systems), and (d) central pressure (hPa). Samples are structured as 3-hour interval sequences $(t_0 - 6, t_0 - 3, t_0)$ to capture short-term evolution, with this 6-hour lookback window enabling operational forecasts 6 hours post-naming—critical for timely disaster prevention. To enhance dataset robustness, we applied 90° clockwise, 90° counterclockwise, and 180° rotational augmentation to the original images. Augmented data are stratified by cyclone name and rotation type to preserve spatiotemporal consistency during training and evaluation.

6.1.2 Implementation Details. Model training is conducted on an NVIDIA RTX 4090 GPU using PyTorch, employing Stochastic Gradient Descent (SGD) with an initial learning rate of 0.001 and batch size of 128. Training proceeds for a maximum of 200 epochs, with early stopping typically triggered at approximately epoch 20 upon validation loss plateauing. To mitigate overfitting and accelerate convergence, we implement learning rate scheduling (reducing by a factor of 0.5 after 5 epochs of validation loss stagnation) and early stopping (activated after 10 epochs without improvement). Adhering to a strict temporal holdout strategy to prevent information leakage, the data partition comprises: training set (2015–2020, 46,285 samples), validation and test set (2021–2022, 1,140 and 1,158 samples respectively).

6.1.3 Evaluation Metrics. In alignment with standard tropical cyclone estimation protocols, we evaluate two key metrics for both

Maximum Sustained Wind (MSW) and Radius of Maximum Winds (RMW): the mean absolute error $L_{MAE}(\hat{Y}, Y) = \frac{1}{N} \sum_{i=1}^N |\hat{Y}_i - Y_i|$ and root mean square error $L_{RMSE}(\hat{Y}, Y) = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{Y}_i - Y_i)^2}$. All metrics are computed on denormalized predictions using operational units—MSW in knots (range: [19, 170 kt]) and RMW in nautical miles (range: [5, 200 nmi])—ensuring direct comparability with established literature and operational forecasting systems.

6.2 Comparison with State-of-the-Art Methods

We compare KAN-FIF with seven established tropical cyclone estimation methods across four accuracy metrics, as shown in Table 2. The complete KAN-FIF model achieves a 32.5% reduction in MSW MAE (3.21 kt vs. 4.76 kt) and a 31.9% RMSE reduction (4.31 kt vs. 6.33 kt) compared to the state-of-the-art multi-task model Phy-CoCo. RMW prediction shows consistent improvement (MAE: 8.83 nmi vs. 8.89 nmi), validating KAN-FIF’s ability to model cross-task physical relationships without negative transfer. Furthermore, while DeepMicroNet, TCIEnet, TCICEnet, and the Xception model also demonstrate competitive accuracy, these are single-task models focused solely on intensity prediction. It is evident that even when compared to these specialized models, KAN-FIF achieves superior predictive performance. Further Results demon-

Table 2: Performance comparison of state-of-the-art TC estimation methods

Model	MSW		RMW	
	MAE	RMSE	MAE	RMSE
TC-MTLNet[23]	9.99	13.77	11.03	14.53
DeepCNet[31]	6.84	9.25	11.21	15.10
DeepMicroNet[25]	3.94	5.47	(single-task)	
TCIEnet[29]	3.61	4.93	(single-task)	
TCICEnet[28]	3.47	4.75	(single-task)	
Xception[15]	3.88	4.50	(single-task)	
Phy-CoCo[27]	4.76	6.33	8.89	12.24
KAN-FIF	3.21	4.31	8.83	11.66

strate significant improvements in computational efficiency and estimation accuracy (Table 3), where KAN-FIF reduces parameter count by 94.8% versus Phy-CoCo (from 19MB to 0.99MB) and decreases per-sample inference time by 68.7% (7.35ms to 2.3ms), enabling lightweight edge-device deployment. To comprehensively

Table 3: Comparison of model size and inference time with multi-task model

Model	TC-MTLNet[23]	DeepCNet[31]	Phy-CoCo[27]	KAN-FIF
Size(M)	170	86	19	0.99
Infer Time(ms)	10.17	8.91	7.35	2.3

evaluate model performance, we selected four representative tropical cyclone structures from the test set for quantitative assessment. Figure 4 provides visual comparisons between KAN-FIF predictions and the baseline Phy-CoCo model, where circle sizes scale proportionally with RMW values and MSW magnitudes are annotated along corresponding guidelines.

6.3 Deployment metrics

Prior to deployment experiments, we executed operator compatibility refactoring, constructed static computation graphs, and performed offline model conversion (as detailed in Section 5) to produce the deployable KAN-FIF variant. Following these architectural simplifications, predictive accuracy exhibited marginal degradation within expected tolerances (Table 4, deploy(GPU)). Although static computation graph enforcement and LSTM removal contribute to this marginal accuracy loss, the model largely preserved baseline precision, demonstrating the robustness of KAN layers under operator compatibility constraints.

Leveraging the Ascend 310 NPU processor onboard FY-4 series meteorological satellites, we conducted offline deployment experiments on the Qingyun-1000 development board using multispectral remote sensing data and temporal auxiliary inputs. The edge-device deployment achieved a per-sample inference latency of 14.41ms, validating the promising potential for operational tropical cyclone monitoring.

Table 4: Metrics of deployment

Model	Size (M)	Infer Time (ms)	MSW		RMW	
			MAE	RMSE	MAE	RMSE
deploy(GPU)	0.92	2.28	3.63	4.93	8.95	11.77
deploy(AscendNPU)	0.92	14.41	6.66	9.78	9.37	12.22
KAN-FIF	0.99	2.30	3.21	4.31	8.83	11.66

6.4 Expanded Results Analysis on Compressed Model

Further demonstrating the scalability and deployment potential of KAN-FIF, experiments with substantially reduced hidden units achieved a 77.8% parameter reduction (from 0.99M to 0.22M) while maintaining competitive performance (Table 5 KAN-FIF-s). The resulting model exhibits minimal accuracy degradation, evidenced by marginal increases in MSW MAE (3.21 to 3.39 kt), RMW MAE (8.83 to 8.98 nmi), MSW RMSE (4.31 to 4.49 kt), and RMW RMSE (11.66 to 11.79 nmi). Crucially, this parameter efficiency reduces memory footprint while reducing on-board inference energy consumption.

Table 5: Performance comparison with compressed model

Model	Size (M)	Infer Time (ms)	MSW		RMW	
			MAE	RMSE	MAE	RMSE
KAN-FIF	0.99	2.30	3.21	4.31	8.83	11.66
KAN-FIF-s	0.22	2.30	3.39	4.49	8.98	11.79

6.5 Ablation study

6.5.1 Ablation Analysis of Temporal Feature Integration. To underscore the critical role of temporal feature organization in KAN-FIF, we first analyze the performance gap between KAN-FIF and the deployment variant, wherein the LSTM layer is substituted with KAN layers, resulting in marginal accuracy degradation (MSW MAE: +13.1%, RMW MAE: +1.4%). Conversely, when temporal features

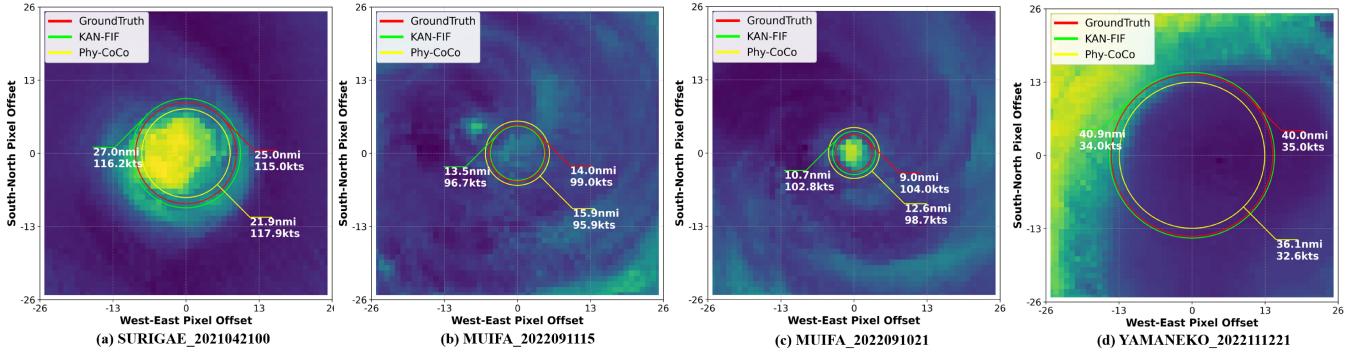


Figure 4: Visual comparison between KAN-FIF and Phy-CoCo models under different tropical cyclone structures

are removed and single-time-step inputs encoded via linear layers with tanh activation, severe performance deterioration occurs: MSW MAE increases by 32.08% and RMSE by 32.01%, while RMW MAE rises 7.70% with RMSE increasing 8.58%. As shown in Table 6, the stark contrast demonstrates that while edge deployment necessitates LSTM removal, preserving temporal sequencing through KAN projections largely maintains accuracy, whereas abandoning temporal context catastrophically compromises prediction accuracy.

Table 6: Ablation study on temporal feature integration

Methods		MSW		RMW	
LSTM	Seq feature	MAE	RMSE	MAE	RMSE
		4.24	5.69	9.51	12.66
✓		3.63	4.93	8.95	11.77
KAN-FIF		3.21	4.31	8.83	11.66

6.5.2 Hyperparameter Sensitivity Analysis. Ablation studies on KAN hyperparameters revealed moderate sensitivity to `spline_order` but limited sensitivity to `grid_size`. The fixed configuration (`grid_size=5`, `spline_order=3`) was selected based on this analysis to balance accuracy and deployment constraints.

6.5.3 Ablation Study on Multi-Stage KAN Integration. This ablation study establishes the indispensable role of systematically integrating KAN layers across four architectural stages: shared feature extraction, attention-based encoding, physics-constrained modeling, and prediction decoding. By implementing KAN layers at each stage and benchmarking against the full KAN-FIF model (Table 7), we quantify their contribution to accuracy gains through KAN layers.

Five ablated variants were evaluated: (a) Full substitution of KAN layers with MLPs (bilinear layers + ReLU/tanh activations) or Fourier transform modules (b) Replacement of KAN layers in the Shared Feature Extraction Module (KAN-LSTM/KAN-CNN) with linear-ReLU blocks (c) Substitution of KAN-Attention encoding with linear-tanh blocks (d) Exchange of the KAN-based Physics-Constrained Module for the Fourier Transform Module from Phy-CoCo [27] (e) Application of linear layers instead of KAN layers for final prediction decoding

Table 7: Ablation Study on Multi-Stage KAN Integration

Extract	Attention	Constraint	Decoder	KAN Ablation Stage		MSW	RMW
				MAE	RMSE	MAE	RMSE
				3.63	4.85	9.96	13.24
✓		✓		3.46	4.62	9.15	11.94
✓	✓			3.41	4.54	8.84	11.87
✓	✓	✓		3.76	4.94	8.93	11.99
✓	✓	✓	✓	3.53	4.66	9.12	11.69
KAN-FIF				3.21	4.31	8.83	11.66

The full KAN-FIF model achieves optimal performance in all metrics. The greatest degradation occurs when removing KAN layers from the physics-guided constraint stage (MSW MAE: +17.1%), validating KANs' superiority in modeling high-order polynomial wind-radius relationships. While removing KAN layers from the attention-based encoding stage exhibits minimal degradation of RMW (+0.11%), its MSW MAE increases by 6.2%. The marginal impact on RMW estimation may suggest that the radius prediction relies more on shared spatial features than on task-specific ones. Removing KAN layers from the shared feature extraction stage degrades both tasks (MSW MAE: +7.8%, RMW MAE: +3.6%).

The cumulative degradation when removing KAN layers from all four stages (+13.1% MSW MAE) reflects nonlinear interactions between ablation stages rather than simple additive effects. In particular, full-stage removal induces less severe degradation than isolated removal in the physics-guided constraint stage (which degrades MSW MAE by 17.1%). This counterintuitive result suggests that MLP-based substitutions in non-critical modules partially compensate for errors introduced by removing physics-constrained KAN layers, masking the full impact of individual substitutions. However, such compensation is task-specific and unstable: RMW MAE degradation (+12.8%) disproportionately exceeds isolated ablation impacts, highlighting the role of KAN layers in harmonizing multitask predictions.

7 Conclusion

In this study, we propose a resource-efficient framework KAN-FIF for tropical cyclone estimation of intensity and size, which integrates MLP, CNN, and filtering operations with spline-parameterized KAN layers to resolve computational and deployment constraints.

The framework achieves state-of-the-art accuracy with a 94.8% parameter reduction compared to physics-guided baselines, leveraging the mathematical foundation of KAN layers in multivariate function decomposition for adaptive high-order nonlinear modeling. Offline deployment on the Qingyun-1000 development board equipped with the Ascend 310 NPU, identical to the FY-4 satellite processor, achieved a 14.41ms per-sample inference latency, validating hardware readiness for orbital deployment through static computation graph optimization and memory-aware compression, establishing a scalable edge-device inference paradigm for high-precision geophysical modeling, offering significant societal value for disaster monitoring in infrastructure-limited regions. Future work will expand multimodal spatiotemporal datasets and optimize dynamic KAN configurations for meteorological satellite deployment scenarios.

Acknowledgments

This work was supported in part by the National Key R&D Program of China under Grant 2022YFB3206900, Key R&D Program of Shandong Province of China under Grant 2023CXGC010112, the joint funds of the National Natural Science Foundation of China under Grant U24A20221, Distinguished Young Scholar of Shandong Province under Grant ZR2023JQ025, Taishan Scholars Program under Grant tstop20250708, Major Basic Research Projects of Shandong Province under Grant ZR2022ZD32.

References

- [1] Alphan Altinok, David R Thompson, Benjamin Bornstein, Steve A Chien, Joshua Doubleday, and John Bellardo. 2016. Real-Time Orbital Image Analysis Using Decision Forests, with a Deployment Onboard the IPEX Spacecraft. *Journal of Field Robotics* 33, 2 (2016), 187–204.
- [2] Gaëtan Bahl and Florent Lafarge. 2022. Scanner neural network for on-board segmentation of satellite images. In *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 3504–3507.
- [3] Kotaro Bessho, Kenji Date, Masahiro Hayashi, Akio Ikeda, Takahito Imai, Hidekazu Inoue, Yukihiko Kumagai, Takuya Miyakawa, Hidehiko Murata, Tomoo Ohno, et al. 2016. An introduction to Himawari-8/9—Japan’s new-generation geostationary meteorological satellites. *Journal of the Meteorological Society of Japan. Ser. II* 94, 2 (2016), 151–183.
- [4] Daniel R Chavas, Kevin A Reed, and John A Knaff. 2017. Physical understanding of the tropical cyclone wind-pressure relationship. *Nature communications* 8, 1 (2017), 1360.
- [5] Boyo Chen, Buo-Fu Chen, and Yun-Nung Chen. 2021. Real-time tropical cyclone intensity estimation by handling temporally heterogeneous satellite data. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 14721–14728.
- [6] J. Gahtan, K. R. Knapp, C. J. Schreck, H. J. Diamond, J. P. Kossin, and M. C. Kruk. 2024. International Best Track Archive for Climate Stewardship (IBTrACS) Project, Version 4r01. NOAA National Centers for Environmental Information. doi:10.25921/82ty-9e16 [indicate subset used].
- [7] Gianluca Giuffrida, Luca Fanucci, Gabriele Meoni, Matej Batić, Léonie Buckley, Aubrey Dunne, Chris Van Dijk, Marco Esposito, John Hefele, Nathan Vercruyssen, et al. 2021. The Ph-Sat-1 mission: The first on-board deep neural network demonstrator for satellite earth observation. *IEEE Transactions on Geoscience and Remote Sensing* 60 (2021), 1–14.
- [8] Cheng Huang, Cong Bai, Sixian Chan, and Jinglin Zhang. 2022. MMSTN: A Multi-Modal Spatial-Temporal Network for Tropical Cyclone Short-Term Prediction. *Geophysical Research Letters* 49, 4 (2022), e2021GL096898.
- [9] Cheng Huang, Cong Bai, Sixian Chan, Jinglin Zhang, and YuQuan Wu. 2023. MGTCF: multi-generator tropical cyclone forecasting with heterogeneous meteorological data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 5096–5104.
- [10] Cheng Huang, Pan Mu, Jinglin Zhang, Sixian Chan, Shiqi Zhang, Hanting Yan, Shengyong Chen, and Cong Bai. 2025. Benchmark dataset and deep learning method for global tropical cyclone forecasting. *Nature Communications* 16, 1 (2025), 5923.
- [11] Kenneth R Knapp, Michael C Kruk, David H Levinson, Howard J Diamond, and Charles J Neumann. 2010. The international best track archive for climate stewardship (IBTrACS) unifying tropical cyclone data. *Bulletin of the American Meteorological Society* 91, 3 (2010), 363–376.
- [12] J Senthil Kumar, V Venkataraman, S Meganathan, and Kannan Krishivasan. 2023. Tropical cyclone intensity and track prediction in the bay of Bengal using LSTM-CSO method. *IEEE Access* 11 (2023), 81613–81622.
- [13] Juhyun Lee, Cheolhee Yoo, Jungho Im, Yeji Shin, and Dongjin Cho. 2020. Multi-task learning based tropical cyclone intensity monitoring and forecasting through fusion of geostationary satellite data and numerical forecasting model output. *Korean journal of remote sensing* 36, 5_3 (2020), 1037–1051.
- [14] Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y Hou, and Max Tegmark. 2024. Kan: Kolmogorov-arnold networks. *arXiv preprint arXiv:2404.19756* (2024).
- [15] Zhaoyang Ma, Yunfeng Yan, Jianmin Lin, and Dongfang Ma. 2024. A multiscale and multilayer feature extraction network with dual attention for tropical cyclone intensity estimation. *IEEE Transactions on Geoscience and Remote Sensing* 62 (2024), 1–15.
- [16] Chandrasekhar Nagarajan, Rodney Gracian D’souza, Sukumar Karumuri, and Krishna Kinger. 2014. Design of a cubesat computer architecture using COTS hardware for terrestrial thermal imaging. In *2014 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology*. IEEE, 67–76.
- [17] Priyanka Nandal, Prerna Mann, Navdeep Bohra, Ghadah Aldehim, Asma Abbas Hassan Elnour, and Randa Allafi. 2025. Tropical cyclone intensity estimation based on YOLO-NAS using satellite images in real time. *Alexandria Engineering Journal* 113 (2025), 227–241.
- [18] Shindi Marlina Oktaviani, Aipujana T Santoso, Yasir MO Abbas, Mark Angelo C Purio, Galuh Mardiansyah, et al. 2023. The development of experimental remote sensing cubesat payload integrated with on-board classification feature: the progress and educational aspect. In *IGARSS 2023-2023 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 253–256.
- [19] Ji Hyun Park, Takaya Inamori, Ryuhei Hamaguchi, Kensuke Otsuki, Jung Eun Kim, and Kazutaka Yamaoka. 2020. Rgb image prioritization using convolutional neural network on a microprocessor for nanosatellites. *Remote Sensing* 12, 23 (2020), 3941.
- [20] Radoslav Pitonak, Jan Mucha, Lukas Dobis, Martin Javorka, and Marek Marusin. 2022. Cloudsatnet-1: Fpga-based hardware-accelerated quantized cnn for satellite on-board cloud coverage classification. *Remote Sensing* 14, 13 (2022), 3180.
- [21] Jingchen Pu, Mu Mu, Jie Feng, Xiaohui Zhong, and Hao Li. 2025. A fast physics-based perturbation generator of machine learning weather model for efficient ensemble forecasts of tropical cyclone track. *npj Climate and Atmospheric Science* 8, 1 (2025), 128.
- [22] Shuyao Shang, Jinglin Zhang, Xing Wang, Xinghua Wang, Yuanjun Li, and Yuanjiang Li. 2023. Faster and lighter meteorological satellite image classification by a lightweight channel-dilation-concatenation net. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 16 (2023), 2301–2317.
- [23] Wei Tian, Xinxin Zhou, Xianhua Niu, Linhong Lai, Yonghong Zhang, and Kenny Thiam Choy Lim Kam Sian. 2022. A lightweight multitask learning model with adaptive loss balance for tropical cyclone intensity and size estimation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 16 (2022), 1057–1071.
- [24] Christopher Velden, Bruce Harper, Frank Wells, John L Beven, Ray Zehr, Timothy Olander, Max Mayfield, Charles “CHIP” Guard, Mark Lander, Roger Edson, et al. 2006. The Dvorak tropical cyclone intensity estimation technique: A satellite-based method that has endured for over 30 years. *Bulletin of the American Meteorological Society* 87, 9 (2006), 1195–1210.
- [25] Anthony Wimmers, Christopher Velden, and Joshua H Cossuth. 2019. Using deep learning to estimate tropical cyclone intensity from satellite passive microwave imagery. *Monthly Weather Review* 147, 6 (2019), 2261–2282.
- [26] Dazhi Xi, Ning Lin, and James Smith. 2020. Evaluation of a physics-based tropical cyclone rainfall model for risk assessment. *Journal of Hydrometeorology* 21, 9 (2020), 2197–2218.
- [27] Hanting Yan, Pan Mu, Cheng Huang, Jinglin Zhang, and Cong Bai. 2024. PhyCoCo: Physical Constraint-Based Correlation Learning for Tropical Cyclone Intensity and Size Estimation. In *ECAI 2024*. IOS Press, 2226–2233.
- [28] Chang-Jiang Zhang, Xiao-Jie Wang, Lei-Ming Ma, and Xiao-Qin Lu. 2021. Tropical cyclone intensity classification and estimation using infrared satellite images with deep learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14 (2021), 2070–2086.
- [29] Rui Zhang, Qingshan Liu, and Renlong Hang. 2019. Tropical cyclone intensity estimation using two-branch convolutional neural network from infrared and water vapor images. *IEEE Transactions on Geoscience and Remote Sensing* 58, 1 (2019), 586–597.
- [30] Ziheng Zhou, Ying Zhao, Yiyu Qing, Wenming Jiang, Yihan Wu, and Wenguang Chen. 2023. A physics-guided nn-based approach for tropical cyclone intensity estimation. In *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*. SIAM, 388–396.
- [31] Jing-Yi Zhuo and Zhe-Min Tan. 2021. Physics-augmented deep learning to improve tropical cyclone intensity and size estimation from satellite imagery. *Monthly Weather Review* 149, 7 (2021), 2097–2113.