Onboard Implementation of Machine Learning for Autonomous Vortex Detection on Mars

Anita Amofah
Department of Computer Science
Fayetteville State University
Fayetteville, NC, USA
aamofah@broncos.uncfsu.edu
0009-0005-8932-6860

Daxell T Wells
Department of Computer Science
Fayetteville State University
Fayetteville, NC, USA
dwells11@broncos.uncfsu.edu
0009-0002-0532-6125

Rajil S Vembe Sajila
Department of Computer Science
Fayetteville State University
Fayetteville, NC, USA
rvembesajila@broncos.uncfsu.edu
0009-0009-2253-9449

Gary Doran
Jet Propulsion Laboratory,
California Institute of Technology
Pasadena, CA, USA
gary.b.doran.jr@jpl.nasa.gov
0000-0003-1233-2224

Sambit Bhattacharya
Department of Computer Science
Fayetteville State University
Fayetteville, NC, USA
sbhattac@uncfsu.edu
0000-0001-6407-0217

***Abstract*—Atmospheric vortices on Mars, such as dust devils, play a critical role in the planet's climate, dust transport, and surface–atmosphere interactions. Capturing high-rate wind, pressure, and dust sensor data during vortex encounters provides valuable scientific insight; however, such events are rare and unpredictable. Continuous high-rate data collection is therefore impractical due to strict power and onboard data storage constraints, particularly for low-cost landers. Prior work has proposed using machine learning algorithms to detect approaching vortices from low-rate pressure measurements and autonomously trigger high-rate data acquisition during the most scientifically relevant portion of the event. In this work, we extend prior vortex detection studies by evaluating both classical machine learning models and a two-stage deep learning pipeline for early-warning detection under precision-critical deployment constraints. Specifically, we assess Random Forest and Extreme Gradient Boosting (XGBoost) classifiers trained on statistically engineered features, alongside a two-stage autoencoder-gated Temporal Convolutional Network (AE→TCN) designed to filter nominal atmospheric behavior prior to classification. Model performance is evaluated using precision, recall, and F1-score metrics appropriate for rare-event detection. We also frame these results in the context of onboard deployment by qualitatively considering the computational requirements of the candidate models. Direct benchmarking of inference latency and power consumption on representative edge hardware is left for future work. Overall, this study provides a comparative assessment of detection performance and deployment-relevant trade-offs for autonomous onboard triggering and highlights the potential of learning-enabled intelligent sensing to enhance Martian atmospheric science under mission resource constraints.**

***Index Terms*—Vortex detection, Mars atmosphere, time-series classification, Random Forest, XGBoost, Temporal Convolutional Networks, onboard machine learning**

I. Introduction

Atmospheric vortices on Mars, commonly observed as dust devils, are short-lived but scientifically significant phenomena that play an important role in the planet's climate system. These vortices contribute to dust lifting and transport, influence near-surface atmospheric dynamics, and shape surface–atmosphere interactions that are relevant to both scientific investigation and future surface operations. Observations from landers and rovers have shown that dust devils can dominate local dust flux and produce substantial atmospheric changes over short timescales.

Despite their scientific importance, atmospheric vortices are difficult to observe using conventional data collection strategies. Sparse, pre-scheduled sampling may miss vortex encounters entirely, while continuous high-rate acquisition can exceed the power, storage, and downlink capabilities of spacecraft, particularly for small or resource-constrained platforms. These limitations motivate the development of intelligent observing strategies that prioritize data collection during scientifically valuable events while minimizing resource usage during quiescent periods.

Recent work has proposed environment-responsive data acquisition approaches in which onboard algorithms monitor low-rate sensor measurements and autonomously trigger higher-rate, higher-power observations when signatures of an approaching vortex are detected. Proof-of-concept studies using pressure time-series data from the Mars 2020 Perseverance rover have shown that dust devils often exhibit characteristic pressure drops that can serve as early indicators of vortex encounters [1]. These studies have explored threshold-based methods, expert systems, and supervised machine learning approaches, including recurrent neural networks, to enable autonomous detection under constrained mission resources.

While prior investigations have demonstrated the promise of machine learning for vortex detection, important questions remain regarding model selection, feature design, and the

practical trade-offs associated with onboard deployment. In particular, detection algorithms must operate reliably under strict constraints on inference latency, memory usage, and power draw, which are especially relevant for future small lander and edge-computing platforms.

In this work, we extend prior research by systematically evaluating both classical and deep learning approaches for early vortex detection using pressure time-series data. Specifically, we compare feature-based classical models, including Random Forest and Extreme Gradient Boosting (XGBoost), against a two-stage deep learning pipeline that combines anomaly-based autoencoder gating with a Temporal Convolutional Network (TCN) classifier. This unified evaluation enables direct comparison of detection performance, operational robustness, and computational considerations across model families under precision-critical deployment constraints.

Model performance is evaluated using precision-, recall-, and F1-based metrics under severe class imbalance. We also discuss onboard deployment feasibility by qualitatively considering the computational requirements of the candidate models. Direct benchmarking of inference latency and power consumption on a Qualcomm Snapdragon–class processor is left for future work.

A. Operational Constraints and Precision-Critical Deployment

Early vortex detection for onboard triggering is inherently a rare-event detection problem. During continuous surface pressure monitoring, atmospheric vortices occur infrequently relative to nominal background behavior, resulting in extreme class imbalance under realistic deployment conditions. Although balanced datasets can be useful during training to support learning from limited positive examples, operational performance is ultimately governed by the overwhelming prevalence of negative windows.

Under flight-relevant constraints, false-positive detections carry substantial cost. Each spurious trigger initiates higher-rate sensing, increases onboard data storage, and consumes additional power, directly affecting mission lifetime and resource availability. Accordingly, vortex detection is treated here as a *precision-critical early-warning task*, in which minimizing false activations is prioritized over exhaustive event capture. In this setting, a single early trigger is generally sufficient to activate high-power instruments and capture the scientifically relevant portion of a vortex encounter, whereas repeated or unnecessary activations provide limited additional scientific benefit.

Model outputs are therefore interpreted through tunable decision thresholds that explicitly control the trade-off between precision and recall. Lower thresholds increase sensitivity to vortex precursors but can rapidly elevate false-trigger rates, effectively pushing the system toward near-continuous operation. Higher thresholds yield more conservative behavior by suppressing false activations at the cost of missed events, reflecting a more appropriate operating regime for power- and resource-constrained platforms.

Consistent with this deployment philosophy, evaluation throughout this work emphasizes *event-level operational metrics* rather than sliding-window accuracy alone. In particular, we focus on precision, event recall, and trigger behavior under severe class imbalance, as these more directly reflect mission decision-making requirements. *Event recall* is defined as the fraction of labeled vortex events for which at least one positive prediction falls within the event's detection window, reflecting the operational requirement that a single early trigger suffices to capture each encounter. We additionally report an *adjusted precision* metric that discounts detections occurring during the event core following a successful early trigger, since the system would already be active during that interval; formal definitions are given in Section V. Together, these metrics provide a more realistic assessment of onboard detection performance in scenarios where intelligent sensing must balance scientific return against strict resource constraints. This deployment framing also motivates future benchmarking of inference latency and power consumption on representative edge hardware.

II. Dataset

This study uses time-series atmospheric pressure measurements collected by the Mars Environmental Dynamics Analyzer (MEDA) instrument onboard the Mars 2020 Perseverance rover. The MEDA pressure sensor records near-surface atmospheric pressure at a nominal sampling rate of 1 Hz, enabling the observation of short-duration pressure drops associated with passing atmospheric vortices.

Raw MEDA data were obtained from the NASA Planetary Data System (PDS) and compiled into a unified time-series dataset indexed by spacecraft clock (SCLK). The compiled dataset spans multiple Martian sols and contains over one million pressure samples, providing extensive temporal coverage across varying environmental conditions. The low sampling rate and continuous streaming nature of the data reflect realistic onboard sensing conditions, where computational and storage constraints limit the feasibility of continuous high-rate acquisition.

Ground-truth vortex encounters were derived from a catalog of manually identified dust devil events reported in prior work [2]. These events were originally provided as computer-readable tables and subsequently aligned with MEDA timestamps. Each detection is represented as a temporal window centered around a characteristic pressure minimum, capturing the evolution of a vortex encounter.

The resulting dataset is highly imbalanced, with vortex events comprising a very small fraction of the total observations (305 events across over one million samples). This extreme class imbalance reflects the operational reality of rare-event detection in continuous sensor streams, where long periods of nominal conditions are punctuated by infrequent but scientifically valuable events. As a result, the dataset provides a realistic testbed for evaluating precision-critical detection systems that must minimize false triggers while maintaining sufficient event coverage.

Using the compiled MEDA data and associated annotations, an ML-ready dataset was constructed as part of prior work [1]. The dataset includes time-indexed pressure measurements, sol and local time metadata, and binary indicators denoting whether a given timestamp lies within or near a vortex event. This unified dataset serves as the common input for all models evaluated in this study.

While this study is conducted using data from a single mission (Mars 2020 MEDA), this reflects the current state of planetary atmospheric datasets, where high-resolution in-situ measurements are inherently limited due to mission constraints. The proposed framework is designed to operate under such data-scarce conditions and does not rely on mission-specific features. As future planetary missions expand the availability and diversity of atmospheric sensing data, the methods presented here can be readily extended to multi-mission datasets, enabling improved generalization and cross-environment validation.

III. Random Forest Model

We implement a Random Forest (RF) classifier as a computationally efficient baseline for early vortex detection using engineered pressure-based features. Properties of RF models [3], [4] are attractive for onboard deployment scenarios, where inference latency, memory usage, and power consumption are tightly constrained [5].

Unlike sequence-based deep learning models that operate directly on raw time-series inputs, the RF model operates on fixed-length precursor windows represented through engineered features. These features are designed to encode physically meaningful characteristics of vortex-related pressure behavior. While Random Forests are not inherently transparent in the same way as a single decision tree or linear model, their feature-based formulation enables post hoc inspection through tools such as feature importance analysis. This provides a practical means of diagnosing which engineered characteristics contribute most strongly to model decisions while preserving low inference overhead [6].

Temporal Windowing and Feature Representation: To capture precursor signatures of vortex encounters, we extract fixed-length temporal windows of 60 samples (approximately 60 seconds at 1 Hz) from continuous surface pressure measurements. Each window ends one sample prior to the annotated vortex onset time, ensuring that no future information is available to the classifier. This sliding-window formulation is widely used in time-series classification because it converts sequential sensor measurements into fixed-dimensional inputs while preserving local temporal context [4]. The selected window length balances temporal coverage with computational efficiency and is consistent with prior observational studies suggesting that meaningful pressure deviations associated with vortex formation can emerge within the minute preceding vortex passage [2].

From each pressure window, we compute 15 engineered features intended to capture physically meaningful characteristics of precursor pressure behavior. These include trend-based descriptors, pressure-drop magnitude and timing metrics, statistical summaries of local variability, and anomaly-based indicators derived from global pressure statistics. Feature normalization is performed using z-score normalization, with normalization statistics computed exclusively on the training set to prevent information leakage.

Training Strategy and Model Configuration: Under realistic deployment conditions, vortex encounters are rare, leading to extreme class imbalance. To support effective learning while maintaining realistic evaluation conditions, class balancing is applied only during training, consistent with established practices for imbalanced classification [7], [8]. The training set contains 188 positive precursor windows and an equal number of negative windows sampled from background regions that are carefully separated in time from known vortex events, resulting in a balanced training dataset of 376 windows.

The RF classifier is implemented using the `RandomForestClassifier` from the `scikit-learn` library. The ensemble consists of 100 decision trees, each with a maximum depth of 15. A minimum of 10 samples is required to split an internal node, and a minimum of 5 samples per leaf is enforced to improve statistical robustness. At each split, a subset of features is selected using the square-root heuristic. Residual imbalance is further addressed through class-weighted training. All experiments are conducted with a fixed random seed to ensure reproducibility.

Summary: The training procedure is designed to help the RF model learn meaningful class boundaries despite the rarity of vortex events. Balancing is introduced only during training so that the model can learn effectively, while evaluation remains representative of the highly imbalanced operational setting.

Operational Evaluation and Results: RF performance is evaluated under both idealized and deployment-relevant conditions. In a fixed-window evaluation using temporally aligned precursor windows, the model achieves 71.4% precision, 30.0% recall, a 42.3% F1-score, and an ROC AUC of 0.696. These results indicate that the model can identify precursor signatures when the evaluation windows are favorably aligned to the event onset.

To evaluate performance under continuous monitoring, we also perform a sliding-window test over the full test timeline using a step size of 10 samples. This produces 85,925 evaluation windows with a natural class imbalance of approximately 225:1. Table I reports results across a range of decision thresholds, illustrating how operational precision and recall vary under severe imbalance and emphasizing the importance of threshold selection in deployment-oriented use.

At the threshold maximizing F1-score (0.80), the RF model achieves 5.10% precision and 8.42% recall, corresponding to 32 true positives and 596 false positives.

RF performance is further evaluated within the autoencoder-gated pipeline introduced in Section V to assess the extent to which anomaly-based pre-filtering can suppress false triggers during continuous operation. As summarized in Table II, autoencoder-based gating improves operational effectiveness by increasing both precision and F1-score relative to the base-

TABLE I
SLIDING-WINDOW EVALUATION RESULTS FOR THE AE-GATED RANDOM FOREST MODEL.

Threshold	Precision	Recall	F1	TP	FP	FN	TN	ROC AUC
0.20	0.59%	85.79%	1.17%	326	54,810	54	30,735	0.7366
0.30	1.07%	60.26%	2.11%	229	21,137	151	64,408	0.7366
0.40	1.41%	50.00%	2.75%	190	13,249	190	72,296	0.7366
0.50	1.80%	38.68%	3.45%	147	8,001	233	77,544	0.7366
0.60	2.28%	22.63%	4.15%	86	3,683	294	81,862	0.7366
0.70	2.93%	15.26%	4.92%	58	1,920	322	83,625	0.7366
0.80	**5.10%**	**8.42%**	**6.35%**	32	596	348	84,949	0.7366
0.90	0.00%	0.00%	0.00%	0	0	380	85,545	0.7366

TABLE II
PERFORMANCE COMPARISON BETWEEN BASELINE AND AE-GATED RANDOM FOREST MODELS AT OPTIMAL THRESHOLDS.

Model	Threshold	Precision	Recall	F1	ROC AUC
Baseline	0.90	3.78%	6.58%	4.80%	0.7457
AE-Gated	0.80	5.10%	8.42%	6.35%	0.7366

line RF model. This result highlights the value of anomaly-aware gating as a practical strategy for improving trigger reliability in precision-critical onboard detection settings.

IV. EXTREME GRADIENT BOOSTING (XGBOOST) MODEL

A. Overview

Extreme Gradient Boosting (XGBoost) is a scalable implementation of gradient-boosted decision trees designed to achieve strong predictive performance through additive ensemble learning and explicit regularization [9], [10].

XGBoost optimizes a regularized objective function of the form

$$\mathcal{L} = \sum_i \ell(y_i, \hat{y}_i) + \sum_k \Omega(f_k), \tag{1}$$

where $\ell(\cdot)$ denotes a differentiable loss function and $\Omega(\cdot)$ penalizes model complexity [9].

In addition to its learning formulation, XGBoost incorporates implementation-level optimizations such as sparsity-aware split finding, column subsampling, and efficient memory access [9]. These properties make it attractive as a lightweight feature-based detector for early vortex precursor signatures under resource-constrained deployment conditions [11].

B. Specific Windowing and Feature Preparation

The XGBoost model operates on low-rate atmospheric pressure measurements segmented into overlapping sliding windows of 10 and 25 samples. These window lengths are selected to capture short- and intermediate-scale precursor dynamics while remaining compatible with real-time onboard execution.

Consistent with the Random Forest evaluation framework, each window is labeled according to whether a vortex event occurs within a predefined future horizon, framing the task as early-warning detection rather than post-event classification. Because XGBoost operates on fixed-dimensional feature vectors rather than raw temporal sequences, each window is represented using physically motivated summary statistics that encode pressure trends, variability, and short-term instability.

All features are standardized using z-score normalization computed exclusively from the training data within each cross-validation fold to prevent information leakage and ensure fair evaluation.

C. Training Strategy

XGBoost is trained under severe class imbalance, reflecting the rarity of vortex events during continuous atmospheric monitoring [12]. Rather than globally rebalancing the full dataset, the natural class distribution is preserved to better reflect operational deployment conditions.

Model training and evaluation are performed using five-fold Stratified Cross-Validation. This strategy is appropriate because XGBoost operates on window-level feature vectors and does not explicitly model temporal continuity, so temporal ordering between folds does not introduce information leakage. By contrast, the sequence-based AE$\rightarrow$TCN pipeline (Section V) uses chronological splits to preserve temporal causality. Stratification ensures that each fold contains representative vortex samples, yielding more stable and repeatable performance estimates while avoiding the high variance that could arise from a single split under extreme imbalance.

Model performance is evaluated using precision, recall, F1-score, and ROC-AUC, which are more informative than accuracy in rare-event detection settings. To support precision-critical deployment, decision thresholds are swept over the range $[0.01, 1.0]$, and the operating point that maximizes F1-score is selected.

D. Model Configuration

The XGBoost classifier is implemented as an ensemble of gradient-boosted decision trees trained using logistic loss for binary classification [9]. Tree depth and learning rate are selected to balance expressive power and generalization, while subsampling of both training instances and feature dimensions is used to reduce sensitivity to background pressure variability [10], [11].

Additional regularization is applied to discourage overly complex tree structures and improve robustness in the presence of noisy atmospheric signals [9]. All models are trained using a fixed random seed to ensure reproducibility across cross-validation folds.

Summary: The chosen configuration is intended to preserve predictive flexibility while limiting overfitting to noisy background structure. This is important in the present setting, where vortex precursor patterns are subtle and false triggers are costly.

E. XGBoost Results

At the optimized operating point, corresponding to a decision threshold of 0.174, the XGBoost model exhibits strong class separability, achieving an ROC-AUC of 0.986. Precision reaches 0.9137, indicating a low false-trigger rate, while recall is lower (0.2724), reflecting a conservative early-warning regime that prioritizes false-positive avoidance. The resulting F1-score of 0.4196 summarizes this precision–recall trade-off.

Feature importance analysis indicates that long-horizon pressure variability, particularly rolling standard deviation over extended windows, is the dominant contributor to model decisions. This aligns with physical expectations that vortex formation is associated with sustained atmospheric instability rather than isolated pressure perturbations. Trend-based features provide secondary contributions, while diurnal encodings capture known daily modulation of vortex activity.

Overall, these results show that XGBoost provides a strong feature-based detection capability under severe class imbalance. When considered alongside the Random Forest and sequence-based models, it offers a complementary lightweight approach for early vortex precursor detection using low-rate atmospheric pressure data.

V. Methodology: Two-Stage TCN Vortex Detection

A. Overview

We frame vortex detection as a time-series classification problem characterized by extreme class imbalance and precision-critical operating requirements. To reduce false triggers while conserving onboard power, we implement a two-stage detection pipeline consisting of (1) an autoencoder (AE) that filters nominal background behavior, followed by (2) a Temporal Convolutional Network (TCN) classifier that operates only on AE-flagged segments [13]. This two-stage AE$\rightarrow$TCN design explicitly targets precision-critical onboard deployment by decoupling anomaly filtering from classification, enabling aggressive false-trigger suppression while preserving event-level recall under severe class imbalance.

Concretely, the pipeline operates as follows: (1) the autoencoder learns to reconstruct nominal pressure behavior and flags windows whose reconstruction error exceeds a coverage-preserving threshold as anomalous; (2) only flagged windows are forwarded to the TCN classifier, which produces a window-level vortex probability; and (3) a tunable decision threshold converts this probability into a trigger or no-trigger decision. This design aligns with deployment constraints in which high-power sensors and imaging systems should activate infrequently, but must reliably capture scientifically relevant vortex events when triggered. Accordingly, vortex detection is treated as an early-warning problem rather than retrospective event classification.

B. Data Preparation and Windowing

We construct fixed-length, no-peek windows over the pressure signal and assign labels based on the final sample of each window, ensuring that the classifier never accesses future information. Ground-truth vortex events are encoded using two temporal regions: a *Detection Window*, representing the pre-event interval during which an early trigger would be operationally useful, and an *Event Core*, defined by the full width at half maximum (FWHM) of the pressure drop corresponding to the vortex center.[1]

Detection-only variants are trained using labels derived from the Detection Window, while combined variants use the union of the Detection Window and the Event Core.

C. Stage 1: Autoencoder Gating

We employ a TCN-based autoencoder (TCN-AE) trained exclusively on nominal pressure behavior to gate candidate windows prior to classification. The autoencoder consists of a stack of residual temporal convolutional blocks with exponentially increasing dilation factors, using strictly causal convolutions to preserve temporal ordering. A 1×1 convolution maps the encoder output to a latent representation, which is decoded by a mirrored causal TCN to reconstruct the input sequence.

The autoencoder is trained using mean-squared reconstruction error over full windows, with early stopping based on validation loss. Training data are restricted to *pure nominal* windows that exclude (i) all labeled vortex intervals and (ii) automatically detected pressure artifacts, ensuring that the model learns only background atmospheric dynamics rather than vortex-related structure.

Operationally, the autoencoder functions as a front-end filter that forwards only anomalous pressure segments to the classifier. In this context, artifacts refer to transient pressure deviations not associated with vortex formation, including non-vortex atmospheric fluctuations and sensor-induced transients. This gating mechanism limits the downstream classifier's scope to anomalous regions, reducing exposure to overwhelming nominal background behavior.

To avoid catastrophic suppression of true vortex events at the gating stage, autoencoder thresholds are selected to prioritize coverage rather than performance optimization. Specifically, for each autoencoder variant, the reconstruction-error threshold is chosen such that at least one window associated with every labeled vortex event exceeds the threshold when applied to the full timeline. This constraint guarantees that all vortex events are represented in the classifier's candidate set while still discarding the majority of nominal pressure windows. Threshold selection is performed independently of downstream classification metrics and is designed solely to prevent irreversible false negatives at the gating stage.

[1]These regions correspond to the variables `gt_detection_win` and `gt_fwhm` in the underlying dataset.

D. Stage 2: TCN Classifier

We employ a Temporal Convolutional Network (TCN) with dilated causal convolutions to model local and long-range temporal dependencies while preserving strict temporal causality and enabling parallelizable training [14]. The classifier operates on fixed-length, no-peek windows that have passed the autoencoder gating stage and outputs a window-level probability of vortex presence.

The classifier architecture consists of a residual TCN backbone with three temporal convolutional blocks (64 channels per block, kernel size 3, dropout rate 0.2), followed by global average pooling over time and a lightweight two-layer multilayer perceptron (64→32→1) with a sigmoid output. Inputs consist of sequences of normalized pressure measurements; for reconstruction-augmented variants, the autoencoder reconstruction error is concatenated as an additional input channel.

We evaluate classifier variants across multiple experimental axes, including window length (15, 30, and 60 samples), label definition (detection-only using `gt_detection_win` versus combined labels using `gt_detection_win` ∪ `gt_fwhm`), training balance (balanced and unbalanced class distributions), and optional reconstruction-augmented inputs. In total, 60 classifier configurations are evaluated across these dimensions.

Unlike the feature-based models evaluated with stratified cross-validation, all AE→TCN models are trained on chronologically ordered splits to preserve temporal causality, since the TCN operates directly on raw temporal sequences where future information leakage across folds would be a concern. Decision thresholds are selected post hoc on the validation set to optimize precision subject to minimum event-recall constraints, reflecting precision-critical deployment scenarios in which minimizing false activations is prioritized over exhaustive event capture.

E. Evaluation

We report AE-gated, detection-only metrics as the primary operational view, focusing on precision and event-level recall.

Event recall is computed at the event level: a vortex event is considered detected if the model produces at least one positive prediction within the event's labeled span (for detection-only evaluation, within `gt_detection_win`). This definition reflects deployment intent, where a single early trigger is sufficient to activate the high-power sensor and capture the scientifically relevant portion of the event.

For precision-critical analysis, we additionally report a modified precision metric that ignores `gt_fwhm` timesteps following a successful `gt_detection_win` hit. This adjustment reflects the operational reality that, once triggered, the system would already be active during the event core, and subsequent detections should not be penalized as false positives.

VI. Results

Figure 1 summarizes deployable AE-gated TCN performance under **detection-only evaluation**, which constitutes the primary operational metric in this work. The figure reports adjusted precision achieved under minimum event-recall constraints and at selected operating points, reflecting precision-critical deployment scenarios in which false activations must be minimized while preserving sufficient event-level coverage. These results are used for model comparison and decision-threshold selection.

Across all evaluated recall constraints, detection-only–trained models exhibit more stable precision behavior, while combined-label training can yield higher precision at low-recall operating points but degrades more rapidly as recall requirements increase. Precision-optimized operating points achieve the highest adjusted precision overall, while Opt-F1 selections balance precision and event recall for scenarios in which limited recall degradation is acceptable. These results demonstrate that the AE→TCN pipeline can meaningfully suppress false triggers relative to standalone classifiers while preserving operationally useful event coverage, confirming the value of anomaly-based gating for precision-critical deployment.

Additional analyses report extended detection-window sensitivity to provide insight into model behavior under relaxed evaluation assumptions. These results are included for interpretability and timing analysis only and are not used to inform deployment decisions.

a) Overall trends.: Across window sizes and autoencoder thresholds, models exhibit the expected precision–recall trade-off, with higher precision achieved at the cost of reduced event recall.

b) Impact of training balance.: Unbalanced training consistently shifts models toward higher precision, typically with only moderate losses in event recall. This behavior suggests that, within the AE-gated regime, preserving the natural class imbalance during training is beneficial for deployment-oriented performance.

c) Reconstruction-augmented inputs.: Including reconstruction error as an additional input channel yields mixed results. While some variants show modest recall improvements, gains in precision are inconsistent, indicating limited additional discriminative value without further tuning.

d) Precision-optimized thresholds.: Optimizing decision thresholds for precision improves AE-gated precision while reducing event recall, as expected. This trade-off is acceptable for precision-critical operation, where minimizing false activations is prioritized over exhaustive event capture.

VII. Discussion

Our results demonstrate that a two-stage AE→TCN pipeline substantially improves precision relative to an always-on baseline, which is critical for power-constrained planetary deployment. Reducing false activations directly conserves power and storage on rover platforms.

a) Label-extension sensitivity.: Table III is included as a sensitivity analysis only and is not part of the primary results. These results show that if the triggering requirement is relaxed from *strictly pre-onset* detection to *early-in-event* detection (i.e., allowing limited leeway into the vortex core),

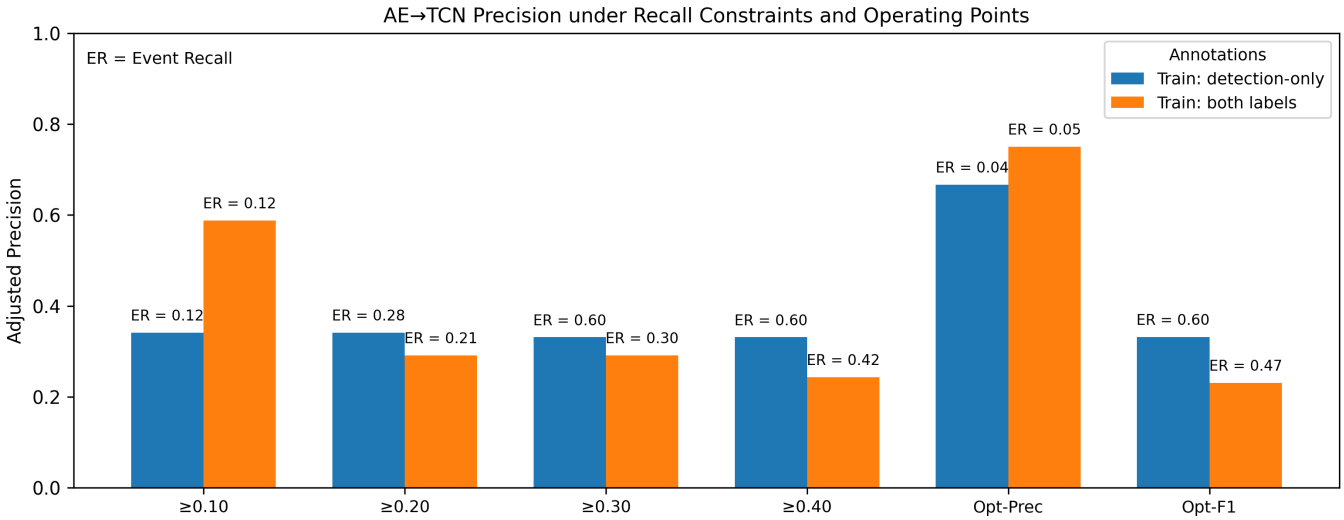

Fig. 1. AE→TCN adjusted precision under event-recall constraints and operating points. Bars compare models trained with detection-only labels versus both labels. For each recall constraint or operating point, the bar corresponds to the best-performing AE-gated configuration satisfying that criterion.

the AE→TCN pipeline achieves substantially higher precision and event recall. This suggests that the most discriminative pressure signatures often strengthen near onset and in the earliest portion of the encounter, rather than exclusively in the pre-event window. While such leeway may not satisfy the most conservative early-warning definition, it can still be operationally meaningful for platforms where triggering shortly after onset is sufficient to capture the scientifically valuable core. These findings motivate future work on mission-aligned trigger horizons and labeling definitions that explicitly trade earlier activation against false-trigger cost.

b) Limitations and future work.: Several limitations remain. First, rolling post-processing (e.g., suppressing isolated triggers) is not yet incorporated and could further reduce false activations. Second, reconstruction-error augmentation produced mixed benefits and warrants more systematic tuning. Finally, event-level evaluation currently relies on a single detection-window definition; exploring alternative pre-event horizons may better align evaluation with operational goals. Future work will integrate these extensions and quantify their impact on trigger rate and event capture.

A. Comparison of Detection Approaches and Deployment Trade-offs

This study evaluates multiple vortex detection configurations spanning classical machine learning and deep temporal architectures, each reflecting a distinct trade-off between interpretability, computational efficiency, and operational robustness under precision-critical deployment constraints.

The *standalone Random Forest (RF)* serves as a lightweight baseline that supports post hoc analysis through feature importance using engineered pressure-based features. While computationally efficient and well suited for resource-constrained platforms, RF performance degrades under continuous sliding-window deployment due to extreme class imbalance, resulting in elevated false-trigger rates unless conservative thresholds are applied.

The *Extreme Gradient Boosting (XGBoost)* model strengthens the classical feature-based approach by capturing higher-order feature interactions through boosted decision trees. Compared to RF, XGBoost achieves improved discriminative performance and higher precision at conservative operating points while maintaining efficient inference. However, like RF, it operates on fixed-length windows and lacks explicit temporal modeling, limiting robustness under sustained background-dominated monitoring.

The *AE-gated Random Forest* introduces anomaly-based pre-filtering to suppress nominal atmospheric behavior prior to classification. This hybrid approach significantly reduces false activations relative to standalone RF and XGBoost, improving operational precision while preserving feature-level diagnostic capability and low inference overhead. AE-gated RF represents a pragmatic compromise between classical efficiency and deployment robustness.

The *two-stage AE→TCN pipeline* provides the strongest deployment-oriented performance by combining anomaly gating with a temporally expressive classifier. By directly modeling precursor dynamics from raw pressure sequences, the TCN achieves superior precision–recall trade-offs under realistic operating conditions. Although this approach incurs greater architectural complexity and reduced interpretability, restricting inference to AE-flagged segments maintains feasibility for onboard triggering.

Overall, these approaches form a clear progression: classical feature-based models offer efficient and interpretable baselines, AE-gated hybrids improve robustness with minimal added cost, and the full AE→TCN architecture delivers the most reliable operational performance for precision-critical autonomous vortex detection.

TABLE III
EXTENDED DETECTION-WINDOW SENSITIVITY (FUTURE WORK): BEST OPT-F1 AE-GATED METRICS FOR EXTEND10 AND EXTEND5.

Dataset	Selection	Label	Model	Precision	Event Recall	Threshold
extend10	Opt-F1	detection-only	`w15_detection_58 unbalanced`	0.7533	0.5263	0.35
extend10	Opt-F1	both	`w60_both_80 unbalanced`	0.8263	0.7176	0.55
extend5	Opt-F1	detection-only	`w15_detection_58 unbalanced`	0.6440	0.4737	0.35
extend5	Opt-F1	both	`w60_both_80 unbalanced`	0.8263	0.7176	0.55

VIII. CONCLUSIONS AND FUTURE WORK

This work demonstrates the feasibility of deploying learning-enabled vortex detection pipelines for precision-critical onboard triggering under realistic planetary mission constraints. By systematically evaluating classical machine learning models alongside a two-stage AE→TCN architecture, we show that model selection and system design must be guided not only by detection accuracy, but by operational metrics such as false-trigger behavior, event-level recall, and resource efficiency. In particular, anomaly-based gating is shown to play a critical role in suppressing nominal background behavior and enabling reliable early-warning detection in the presence of extreme class imbalance.

The comparative analysis highlights a clear progression in capability: classical feature-based models provide efficient and interpretable baselines, AE-gated hybrids substantially improve operational precision with minimal added complexity, and the full AE→TCN pipeline delivers the most robust performance for deployment-oriented early vortex detection. Together, these results support the use of learning-enabled intelligent sensing as a viable strategy for enhancing science return while respecting strict power, storage, and computational constraints on planetary platforms.

Future work will focus on extending this analysis to additional onboard-relevant dimensions. Planned efforts include more comprehensive benchmarking of inference latency and power consumption across candidate hardware platforms, integration of post-processing strategies such as temporal trigger suppression to further reduce false activations, and exploration of alternative labeling horizons to better capture early vortex formation dynamics. In addition, future studies will investigate generalization across environmental regimes and mission contexts, supporting the broader applicability of these approaches to autonomous sensing in future planetary missions.

ACKNOWLEDGMENT

This research was supported by the National Aeronautics and Space Administration (NASA) under Federal Award No. 80NSSC24K1618, through the project *"Building an FSU–JPL Partnership to Advance Science Productivity through Applications of Deep Learning."* We thank Mark Wronkiewicz, Serina Diniega, and Brian Jackson for providing the ML-ready dataset they compiled as part of their prior work. Part of this research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (80NM0018D0004).

REFERENCES

[1] S. Diniega, M. Wronkiewicz, and B. Jackson, "Quantifying science gain through strategic data acquisition strategies: A pilot study of martian dust devils," in *Proceedings of the Tenth International Conference on Mars*. Pasadena, CA, USA: Lunar and Planetary Institute, 2024, lPI Contribution No. 3007.

[2] B. Jackson, "Vortices and dust devils as observed by the mars environmental dynamics analyzer instruments on board the mars 2020 perseverance rover," *The Planetary Science Journal*, vol. 3, no. 20, 2022.

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

[4] A. Bagnall, J. Lines, A. Bostrom, J. Large, and E. Keogh, "The great time series classification bake off: A review and experimental evaluation of recent algorithmic advances," *Data Mining and Knowledge Discovery*, vol. 31, no. 3, pp. 606–660, 2017.

[5] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8034–8054, 2020.

[6] G. Louppe, L. Wehenkel, A. Sutera, and P. Geurts, "Understanding variable importances in forests of randomized trees," in *Advances in Neural Information Processing Systems*, 2013.

[7] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[8] Y. Sun, M. S. Kamel, A. K. C. Wong, and Y. Wang, "Cost-sensitive boosting for classification of imbalanced data," *Pattern Recognition*, vol. 40, no. 12, pp. 3358–3378, 2007.

[9] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 785–794.

[10] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.

[11] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems*, 2017.

[12] S. Chen and H. He, "Learning from imbalanced data using XGBoost," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2455–2468, 2018.

[13] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: A survey," *ACM Computing Surveys*, vol. 54, no. 2, 2021.

[14] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018. [Online]. Available: https://arxiv.org/abs/1803.01271