RESEARCH ARTICLE

Enhancing UAV Security with GPS Spoofing and Jamming Anomaly Detection

Ying-Chen Liu[1], **Lin-Fa Lee**[2] **and Kuo-Hui Yeh**[1,2,*]

[1] Department of Information Management, National Dong Hwa University, Hualien 974301, Taiwan
[2] Institute of Artificial Intelligence Innovation, National Yang Ming Chiao Tung University, Hsinchu 300093, Taiwan

Abstract

Unmanned aerial vehicles face GPS spoofing and jamming that can compromise navigation and safety. We present an anomaly detection method that achieves both high accuracy and interpretability, enabling UAV operators to understand why an alert is triggered, which enables timely responses and builds trust in autonomous detection systems operating in safety-critical environments. We use five classifiers, including XGBoost, Support Vector Machine, K-Nearest Neighbor, Random Forest, and Naive Bayes, trained on a UAV dataset containing 3622 samples for spoofing detection and 6445 for jamming detection made in PX4 and Gazebo with benign flight and attack cases. After feature scaling and reduction, XGBoost reaches F1 near 0.998 for both attacks and runs fast enough for small onboard computers. Our main goal is to explain what the models learn. We study feature importance in four ways using gain in XGBoost, impurity decrease in Random Forest, permutation tests for Support Vector Machine and K-Nearest Neighbor, and a closed form score for Naive Bayes. The results point to the same key signals across models. Spoofing shows up as position drift and a mismatch between speed and course. Jamming shows up as sharp growth in position and velocity errors and poor satellite geometry. These insights help operators watch the right signals and trust the alerts.

Keywords: anomaly detection, extreme gradient boosting (XGBoost), network security, unmanned aerial vehicle (UAV).

1 Introduction

Unmanned Aerial Vehicles (UAVs) are now common in logistics, emergency response, disaster monitoring, environmental protection, and aerial imaging. Market reports forecast a multi-billion-dollar global UAV market by 2025. The widespread use of UAVs raises safety concerns because most platforms depend on Global Navigation Satellite Systems (GNSS), especially GPS, for navigation and control. Two threats are particularly important, GPS spoofing where an attacker transmits forged signals to mislead the position estimate [1, 2], and GPS jamming where noise blocks legitimate signals [3, 4]. These attacks can cause route deviation, loss of control, mission failure, or public-safety risks.

In recent years, GNSS jamming and spoofing has

Submitted: 21 October 2025
Accepted: 20 December 2025
Published: 28 December 2025

Vol. 1, **No.** 1, 2025.
10.62762/JRSC.2025.368867

***Corresponding author:**
✉ Kuo-Hui Yeh
khyeh@nycu.edu.tw

Citation
Liu, Y. C., Lee, L. F., & Yeh, K. H. (2025). Enhancing UAV Security with GPS Spoofing and Jamming Anomaly Detection. *Journal of Reliable and Secure Computing*, 1(1), 54–67.

evolved from a research topic into a significant risk that is widely observed in actual airspace and capable of affecting civil aviation operations. Since 2022, the European Union Aviation Safety Agency (EASA) has issued Safety Information Bulletin SIB 2022-02 and its subsequent revisions, noting that GNSS jamming and spoofing events have shown an upward trend in the severity of impact, intensity of occurrence, and technical sophistication. These events are particularly concentrated in airspace surrounding conflict zones. The bulletin emphasizes that, compared to jamming, spoofing is more difficult for flight crew to detect in a timely manner and thus poses a greater risk to flight safety, potentially causing avionics systems to display inconsistent or misleading aircraft position and GNSS altitude, triggering spurious Terrain Awareness and Warning System (TAWS) alerts, route divergence, and even airspace infringements [29].

In the unmanned aircraft domain, similar vulnerabilities have been empirically demonstrated. In 2012, the Radio navigation Laboratory at the University of Texas at Austin, on invitation of the U.S. Department of Homeland Security, conducted a civil GPS spoofing field test at White Sands Missile Range. Using a civil GPS spoofer located approximately 0.62 KM away, researchers successfully influenced the GPS position and velocity solution of a Hornet Mini civilian UAV, causing its autopilot system to adjust the aircraft's flight attitude based on the falsified navigation solution. The aircraft was tested in its stock configuration without any additional hardware modifications. The study explicitly concluded that civil UAVs can indeed be hijacked through civil GPS spoofing [30], and noted that existing effective defenses largely rely on multi-antenna architectures, expensive receivers, or cryptographic signals, which are not easily adoptable for small UAV platforms constrained by battery, computation, and payload weight. In this context, designing software-only anomaly detection mechanisms deployable on lightweight autopilot systems has become one of the key needs for enhancing UAV navigation integrity and flight safety.

Many defenses exist, such as multi-band anti-interference antennas, encrypted and authenticated signals, military-grade receivers, and multi-source fusion with cross-validation. However, these solutions often require extra hardware, higher cost, and more power. Small and commercial drones have limited batteries, CPU, and memory, which makes hardware-heavy defenses difficult to deploy in practice. This gap motivates software-only detection that can run in real time on resource-constrained autopilots.

This study focuses on a telemetry-only anomaly detection method for GPS spoofing and jamming. The goal is to deliver accurate and fast detection under tight compute and energy budgets. We turn flight logs into learning-ready data and evaluate classical machine-learning models on two feature variants, a compact processed set and a high-dimensional raw set. Results are reported using Cross-validation with flight-level splits to avoid temporal leakage. In experiments, tree-based ensembles demonstrate robust detection performance and effectively capture sudden signal variations characteristic of attacks. These results suggest that a lightweight software detector can improve flight integrity without extra sensors or specialized radios.

In summary, this work builds a practical pipeline from data generation and feature engineering to model validation, compares multiple supervised learners on both raw and processed feature sets, and analyzes errors and feature importance to identify signals that are most sensitive to spoofing and jamming.

2 Related Work

UAVs increasingly operate in safety-critical tasks, which exposes confidentiality, integrity, and availability risks. Attackers can intercept the link between the UAV and the Ground Control Station, manipulate navigation through GNSS and GPS spoofing, or disrupt control with jamming and network attacks. Hardware based defenses such as multi-antenna processing, direction finding, and encrypted and authenticated signals improve resilience but raise cost, weight, and power. Software based detection builds on telemetry or spectrum features with machine learning and deep sequence models, and recent work explores reinforcement learning for adaptive response within an intrusion detection system. We follow these lines and focus on methods that remain practical for resource-constrained autopilots.

2.1 Hardware Defenses

UAVs exchange data with the ground control station over wireless links. Attackers can break in and intercept traffic. Reported tools include malware, trojans, and keyloggers. Stolen data can be live video feeds, navigation coordinates, mission commands, or other payloads [5, 6]. Firmware in the supply chain is another weak point. A hijacked image can bypass later protections. Attackers can watch, limit, or take over core functions [7].

Manipulation of sensing or control also appears in prior work. GPS spoofing sends fake signals so the UAV accepts wrong positions. The drone can drift from its route or land at a location chosen by the attacker. Civil GPS is not encrypted, which helps adoption but also makes spoofing an easier target [8, 9]. Denial-of-service attacks pose a separate risk for consumer Wi-Fi drones such as AR.Drone 2.0 and 3DR SOLO. Studies have shown vulnerabilities to TCP and UDP floods as well as 802.11 de-authentication attacks [10, 11]. Jamming can also block the control link or the GNSS band, which leads to loss of lock and abnormal flight [12].

Physical-layer and device-side defenses appear across the literature. Multi-antenna or multi-receiver designs compare phase, power, or direction to detect a single spoofing source. Arrays estimate the angle of arrival and flag signals that come from one direction only [13, 14]. Signal-quality checks monitor SNR or carrier power. Large, sudden power rises can trigger alerts [15]. Integrity checks such as RAIM compare ranges from multiple satellites and test consistency. Time and frequency continuity can expose attacks, and receivers can monitor out-of-lock events in their phase-locked loops [16]. Sensor fusion is also common. INS or IMU data provide a short-term motion estimate that can be cross-checked against GPS. Divergence beyond a threshold signals possible spoofing [17–19]. These defenses work in many reports, but they often need extra antennas, precise synchronization, or careful calibration, which raises weight, cost, and power on small UAVs.

2.2 Software Detection

Machine learning is widely studied as a way to improve UAV security. Traditional protection based only on encryption and basic rules is not enough for complex and changing threats. ML learns patterns from diverse attack data and flags anomalies in time to react [20].

Supervised learning is a common path. Studies use telemetry such as position, velocity, acceleration, GNSS quality, and other onboard signals. XGBoost is often reported with good results after feature engineering from GPS and IMU logs. Feng et al. [19] train on flight logs with feature selection and report about 96 percent accuracy for binary detection of attacks vs benign flights. These models are fast and interpretable, but they need enough varied samples to avoid overfitting.

Deep learning appears in several forms. Convolutional networks and LSTM models learn temporal patterns from navigation data and from receiver-level features such as Doppler sequences. They detect abnormal flight behavior and spoofed patterns in prior studies [13, 21–23]. One-dimensional CNNs work on time series with low compute cost and have been shown to run in real time on embedded boards while outperforming SVM baselines [24].

Unsupervised and one-class methods reduce labeling needs. Autoencoders learn to reconstruct benign behavior. When an attack shifts the distribution, the reconstruction error grows and triggers an alert [25].

Ensemble and adaptive methods also appear. Dynamic selection across multiple trained classifiers reduces misses and false alarms in reported tests [13]. Reinforcement learning tunes thresholds, allocates resources, and plans trajectories that avoid interference. Prior work shows benefits for intrusion detection and autonomous navigation in dynamic settings [15, 17, 18, 26]. These studies support the view that software approaches can raise resilience without adding radios or antennas.

2.3 Data and Simulation

Data for UAV security research comes from two sources. Real flight logs capture true radio noise, multipath, weather, and operator behavior. They reflect deployment conditions but are costly and risky to collect. Simulated logs are safe and economical during development and testing. They let researchers script spoofing, jamming, and network abuse in a controlled way. The main concern is the domain gap between simulation and field conditions, so studies should document scenarios, attack scripts, parameter ranges, and labeling rules to improve reproducibility and transferability.

PX4 Autopilot is a widely used open platform that supports multirotor and fixed-wing aircraft. It offers SITL(Software In The Loop) and HITL(Hardware In The Loop). In SITL, the flight control code runs on a host computer and interacts with a simulated vehicle and world. This enables rapid testing of control, navigation, and mission logic without physical risk or airspace constraints. PX4 integrates well with Gazebo, a popular open-source robotics simulator that provides realistic rigid-body physics, built-in sensors such as cameras, lidar, and IMU, and ready-made worlds. Gazebo allows configurable sensor noise, weather, and

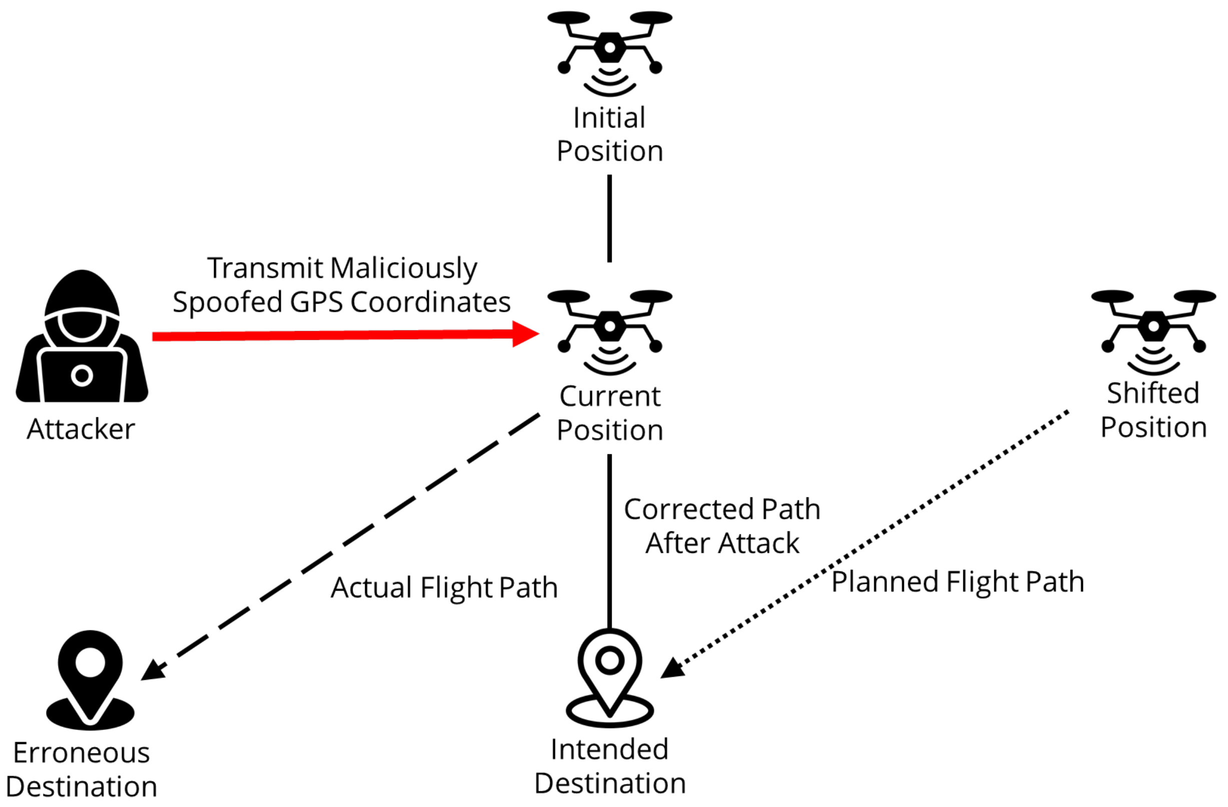

Figure 1. Conceptual diagram of a GPS spoofing attack on a UAV.

environment details, which helps generate repeatable datasets and ablation variants. In HITL, the real flight controller runs the actual firmware while the simulator provides vehicle dynamics and sensor feedback.

AirSim is an open-source simulator for aerial and ground robots built on Unreal Engine. It offers high-fidelity rendering and physics and includes rich sensor models and ground truth. AirSim provides APIs for C++, Python, and other languages and runs on Windows, macOS, and Linux. It can connect with PX4, including HITL, which enables realistic testing while keeping cost and safety under control. Researchers use AirSim to create labeled datasets for deep learning, computer vision, and reinforcement learning, including attack scenarios needed for security studies such as spoofing and jamming.

3 Methodology

We define the GPS spoofing threat and use Figure 1 [27] to show how forged signals affect telemetry and how our data flow is organized. We then describe the dataset and introduce two feature sets. The raw set keeps 829 fields. The processed set has 46 features after normalization. Next, we outline the learners and the training scheme. On the 829 features table we train a single model for three classes. On the 46 features table we train two binary models for benign vs spoofing and benign vs jamming. Evaluation uses flight level Cross-validation with a held out test split. The experiments highlight feature importance to indicate which signals matter most.

Figure 1 shows the GPS spoofing threat considered in this study. A spoofer transmits counterfeit GNSS-like signals that look valid to the receiver and are slightly stronger at the UAV antenna. The receiver locks onto the forged signals and the navigation solution shifts away from the truth. The shift can be a slow drift or a sudden jump. Unlike jamming, which raises noise and breaks lock, spoofing aims to be accepted as legitimate. In the logs this produces consistent changes in common telemetry, for example satellite availability, signal quality, dilution of precision, and mismatches between GNSS and inertial motion. Our detector uses windowed features from these logs to decide among benign, spoofing, and jamming.

3.1 Workflow

Figure 2 summarizes the workflow used in this study. PX4 ULog [28] flights are exported to CSV and then merged into a single table. From this table we prepare two feature sets. The raw set retains 829 fields from

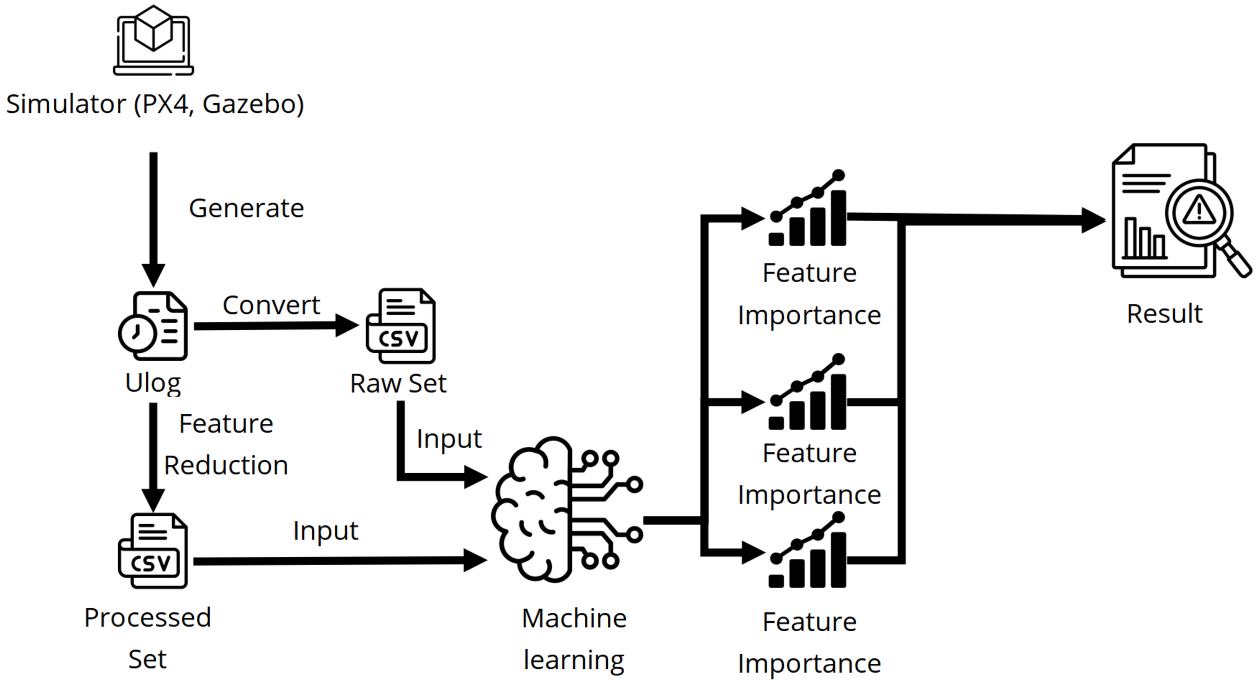

Figure 2. Research workflow.

the logs, while the processed set applies normalization and keeps 46 compact features. The data are labeled as benign, spoofing, or jamming. We train five standard learners, that is Naïve Bayes, KNN, Random Forest, an SVM with a Nystroem map for the RBF kernel, and XGBoost.

We compute feature importance for each model and rank the features. We then compare rankings between the raw 829-field table and the processed 46-feature set. We also merge the top lists across models to form a union that captures complementary signals, and we give higher priority to features that recur across models.

3.2 Data and feature construction

The dataset used in this study is the UAV Attack Dataset from IEEE DataPort [31], which contains 10,067 samples: 3,622 for spoofing detection and 6,445 for jamming detection. GPS spoofing attacks were conducted using a HackRF software-defined radio with the GPS-SDR-SIM tool, broadcasting falsified coordinates to induce a sudden position jump. GPS jamming attacks were performed by broadcasting white Gaussian noise with an amplitude of 0.3 and a gain of -48 dB. All experiments utilized a Holybro S500 UAV frame equipped with a Pixhawk 4 flight controller running PX4 Autopilot v1.11.3 and a Pixhawk GPS receiver.

PX4 ULog flights are exported to CSV and then merged into a single table with three labels. They are benign, spoofing, and jamming. From this table we form two binary datasets so that each model focuses on one attack type. The first contains benign and spoofing records, and the second contains benign and jamming records. Each dataset is prepared in two feature variants to compare the 829-feature raw inputs with a 46-feature processed set. The raw variant keeps all 829 log fields. The processed variant applies normalization and PCA (Principal Component Analysis) following the preprocessing pipeline provided in the original dataset, which reduces the inputs to 46 features. Column names are cleaned for consistency, and missing or infinite values are handled before training using the same steps for both datasets and both feature variants.

3.3 Feature importance

To understand how features affect model predictions, we add a feature-importance analysis to the workflow. For XGBoost, each split computes the drop in the overall loss after regularization. This drop is the Gain. It equals the sum of the objective values at the two child nodes minus the value at the parent node, then minus the regularization term. When a split makes the loss decrease according to first- and second-order statistics, the split separates the data well and reduces prediction error. Gain is therefore the key signal of split value. During training the algorithm selects the candidate split with the largest Gain. Summing Gain across all splits gives a natural way to rank features by their contribution. A higher total Gain means a larger influence on the final model.

In Random Forest, feature importance uses the mean decrease in impurity, also known as Gini importance. Each tree measures how mixed the classes are at a node. Lower Gini means a purer node and a lower misclassification chance. When a feature is chosen to split a parent node, the impurity is distributed to the two children. If the weighted impurity of the children is much smaller than that of the parent, the feature has effectively reduced noise. The forest sums these impurity reductions over all nodes and all trees with sample weights and then normalizes the totals so that they sum to one. A larger value shows that the feature appears in many useful splits and helps the forest make better decisions.

For non-tree models such as SVM and KNN, we estimate feature importance with permutation tests by first recording the baseline on the test set, then shuffling one feature at a time, re-evaluating the model, and taking the average performance drop as the score, so larger drops indicate stronger influence while negligible change suggests little contribution.

For Naïve Bayes, we use a closed-form score derived from class-conditional normal distributions, where between-class mean differences are scaled by variances to measure separation; larger values imply better class separation and values near zero imply minimal utility, and this method is fast and transparent because it requires neither retraining nor shuffling.

We then produce a ranked list for each model. We compare rankings between the raw 829-field table and the processed 46-feature set. We form a union across models to collect complementary signals, and we give higher priority to features that recur in several models.

4 Experiments

4.1 Raw data result

Table 1 summarizes the results on the raw dataset with 829 features. In overall accuracy, shown in Table 1, XGBoost and Random Forest clearly lead at about 0.679 and 0.671. Both values are roughly double the random baseline of 0.333 for a three-class task, which indicates that tree models can still extract useful structure from high-dimensional telemetry. By contrast, KNN, SVM, and Naïve Bayes reach 0.409, 0.392, and 0.416. These numbers are only slightly above chance, suggesting that these methods are constrained in this feature space.

F1 scores follow the same pattern and better reflect balance across classes. XGBoost achieves 0.666 and Random Forest 0.653, which shows that both models maintain relatively even performance across the three classes rather than overfitting to a single label. KNN and SVM produce 0.305 and 0.256, and Naïve Bayes drops to 0.230. Without feature selection or dimensionality reduction, these approaches struggle to capture enough discriminative signal, leading to low precision and recall. Taken together, accuracy and F1 consistently highlight the advantage of tree-based models on the raw dataset.

XGBoost (Figure 3) shows steady three class performance as benign remains strong and jamming is often correct, while spoofing is split between the two and therefore becomes the most challenging class. SVM (Figure 4) is biased toward the jamming class, so many samples from all classes are mapped to jamming and benign and spoofing are rarely identified, which suggests an unstable boundary in the original high dimensional space. Naïve Bayes (Figure 5) collapses toward the benign class, so the apparent success on benign reflects a majority class bias while spoofing and jamming are almost never detected, which indicates that the model assumptions do not match these signals.

Random Forest (Figure 6) follows the pattern of XGBoost as benign remains solid and jamming is frequently correct, yet spillover from benign to jamming persists and spoofing again shows the weakest recall. KNN (Figure 7) degenerates to a single dominant output as benign takes most predictions and spoofing and jamming are rarely identified, which implies that distance is not informative in this high dimensional setting and useful neighborhoods do not form.

The five machine learning models on the raw dataset show that the main errors lie on the boundary between benign and jamming, and spoofing remains the hardest class, so improvements should target a clearer separation between benign and jamming and stronger cues for spoofing.

4.2 Processed data result

After normalization and PCA, all five classifiers improve compared with the raw dataset. As shown in Table 2 for accuracy and F1-score, XGBoost and Random Forest approach near-perfect performance in both spoofing and jamming. XGBoost reaches 0.9965

Table 1. Accuracy and F1-Score performance on the raw set.

	XGBoost	SVM	Naïve Bayes	Random Forest	KNN
F1 Score	0.6661	0.2563	0.2300	0.6532	0.305
Accuracy	0.6794	0.3917	0.4163	0.6708	0.409

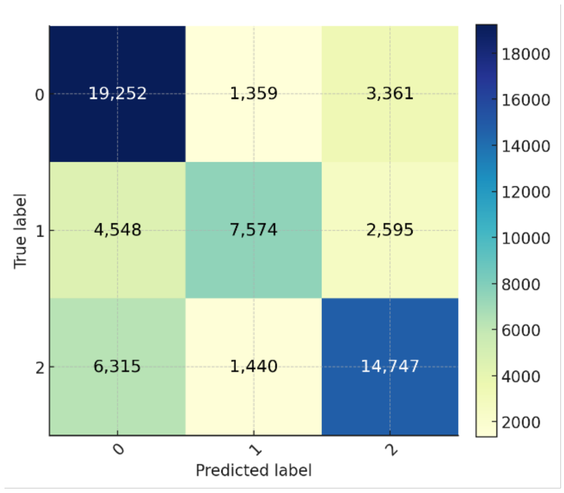

Figure 3. XGBoost Confusion Matrix (Raw Data).

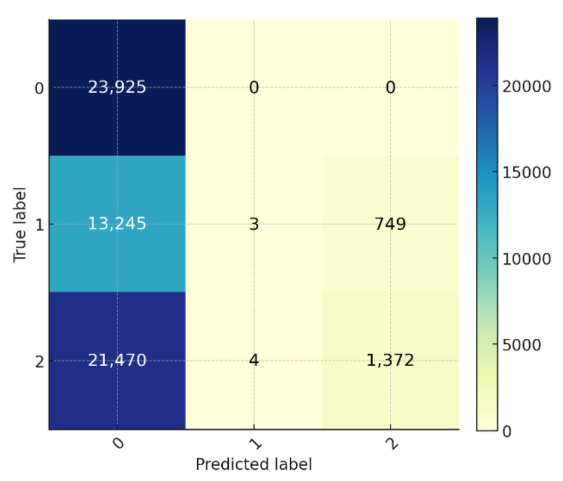

Figure 5. Naïve Bayes Confusion Matrix (Raw Data).

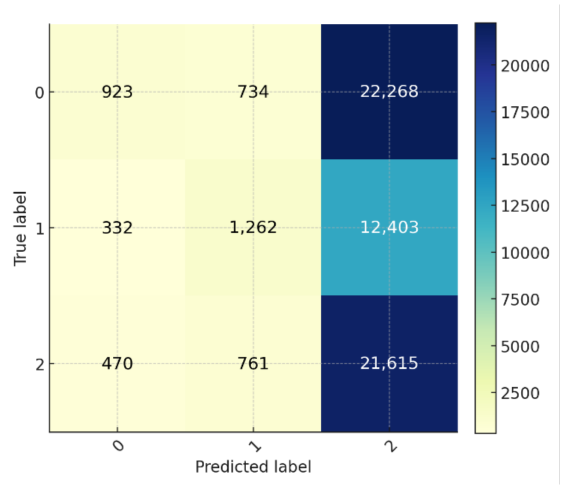

Figure 4. SVM Confusion Matrix (Raw Data).

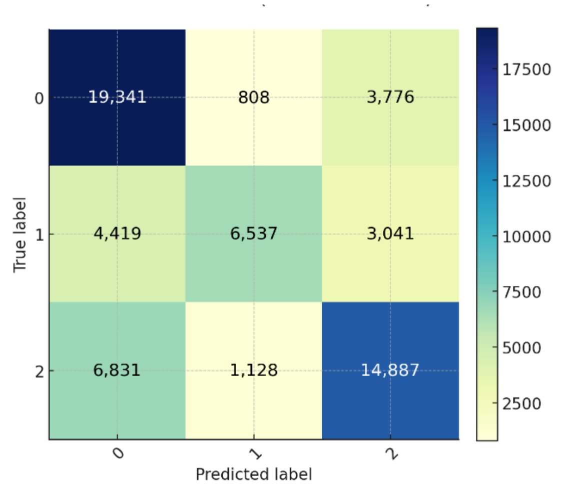

Figure 6. Random Forest Confusion Matrix (Raw Data).

and 0.9984 in accuracy and 0.9979 and 0.9989 in F1 for spoofing and jamming respectively. Random Forest achieves 0.9896 and 0.9981 in accuracy and 0.9939 and 0.9987 in F1 for the same two classes. The results indicate that tree ensembles capture the discriminative structure even after aggressive feature compaction.

KNN benefits from reduced dimensionality and better feature scaling. Its accuracy rises to 0.9248 for spoofing and 0.9806 for jamming, and its F1-score reaches 0.9582 and 0.9873. SVM also becomes more effective in the lower-dimensional space. It attains 0.9579 and 0.9267 in accuracy and 0.9761 and 0.9502 in F1 for spoofing and jamming. Naïve Bayes remains limited by its Gaussian assumption. While it records 0.9551 in accuracy and 0.9733 in F1 for spoofing, its jamming performance drops to 0.7785 in accuracy and 0.8328 in F1, which suggests sensitivity to variance estimation. Overall, the comparison in Table 2 shows that feature normalization and PCA make distance and margin methods more competitive, while tree-based models still lead across classes.

XGBoost (Figures 8 and 9) separates benign from both attacks almost perfectly, since in the spoofing split all 1,994 benign samples are correct and 584 spoofing samples yield only 4 misses, while in the jamming split only 4 of 1,246 benign samples are flagged as attacks

Table 2. Accuracy and F1-Score performance on the processed set.

	XGBoost	SVM	Naïve Bayes	Random Forest	KNN
F1 Score (spoofing)	0.9979	0.9761	0.9733	0.9939	0.9582
F1 Score (jamming)	0.9989	0.9502	0.8328	0.9987	0.9873
Accuracy (spoofing)	0.9965	0.9579	0.9551	0.9896	0.9248
Accuracy (jamming)	0.9984	0.9267	0.7785	0.9981	0.9806

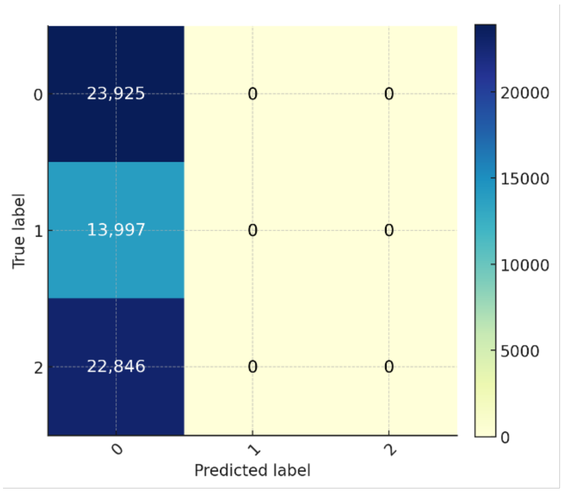

Figure 7. KNN Confusion Matrix (Raw Data).

and just 1 of 199 jamming windows is missed, so the residual confusion is small and localized.SVM (Figures 10 and 11) is highly sensitive yet imbalanced, because in the spoofing split 181 attacks produce only 18 misses but 1,117 benign samples are reported as attacks, and in the jamming split all 584 attacks are detected while 189 benign samples are still flagged, so the boundary continues to favor sensitivity over specificity even after preprocessing.

Naïve Bayes (Figures 12 and 13) leans toward attack labels under fluctuation, as in the spoofing split only 3 of 196 attacks are missed but 62 of 1,188 benign samples are marked as attacks, and in the jamming split all 584 attacks are found while 571 of 1,423 benign samples are mislabeled, so many benign windows are still pulled into the attack side. Random Forest (Figures 14 and 15) remains balanced, since in the spoofing split only 10 of 1,240 benign samples are flagged and only 2 of 199 attacks are missed, and in the jamming split just 1 of 1,993 benign samples is mislabeled while only 4 of 580 attacks are missed, so errors are rare across both scenarios. K-nearest neighbors (Figures 16 and 17) improves after preprocessing but stays uneven, because in the spoofing split all 1,250 benign samples are correct while 109 spoofing samples are labeled as benign, and in the jamming split all 584 attacks are detected while only 50 benign samples are flagged, so mild spoofing drifts still escape detection whereas larger jamming changes separate more clearly.

Across the five models on the processed dataset, tree ensembles keep the error counts low, SVM and Naïve Bayes trade many benign false alarms for high attack hits, and k-nearest neighbors remains asymmetric between spoofing and jamming.

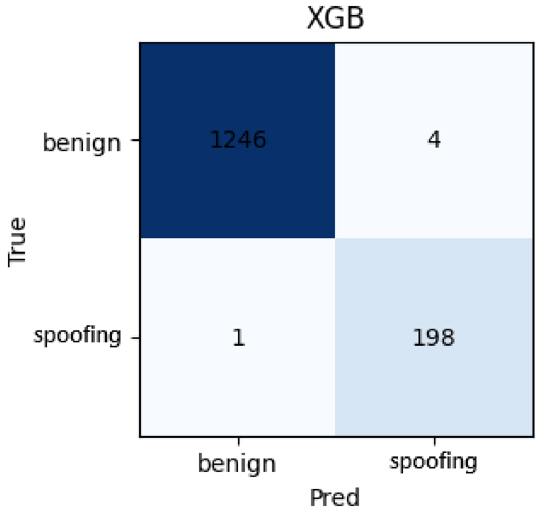

Figure 8. XGBoost Spoofing Confusion Matrices (Processed Data).

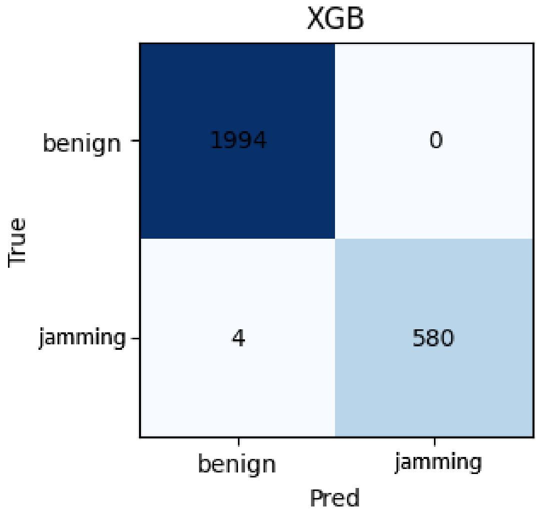

Figure 9. XGBoost Jamming Confusion Matrices (Processed Data).

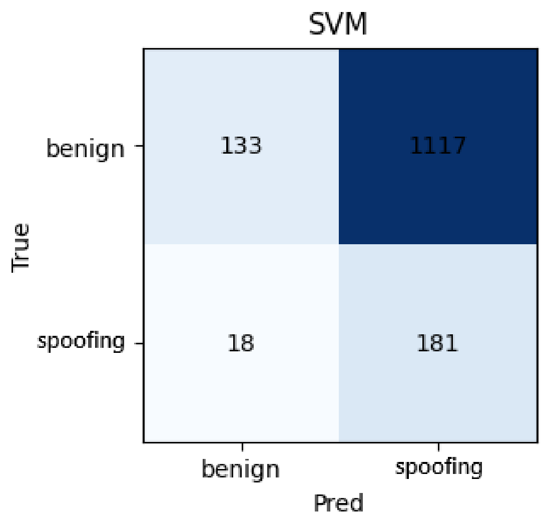

Figure 10. SVM Spoofing Confusion Matrices (Processed Data).

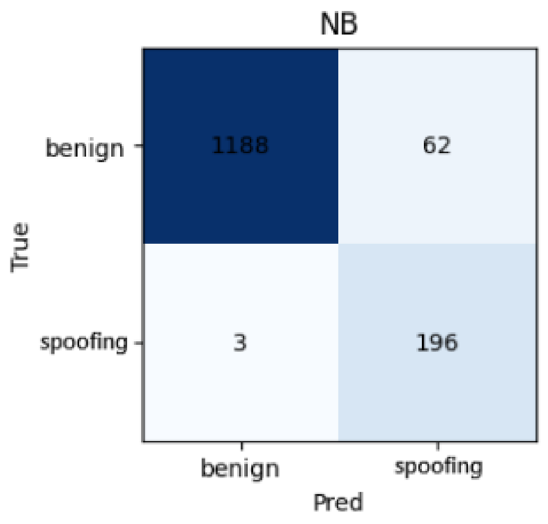

Figure 12. Naïve Bayes Spoofing Confusion Matrices (Processed Data).

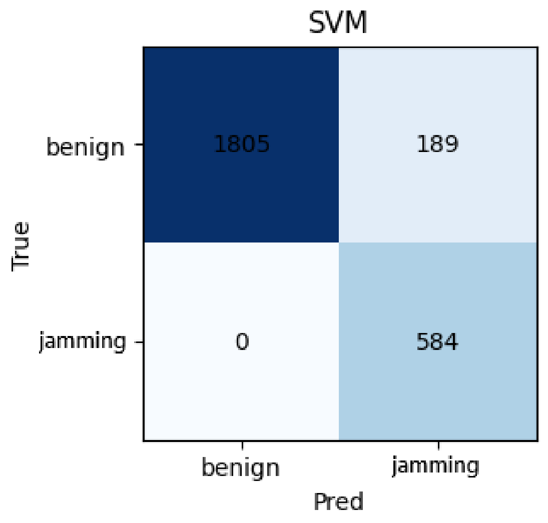

Figure 11. SVM Jamming Confusion Matrices (Processed Data).

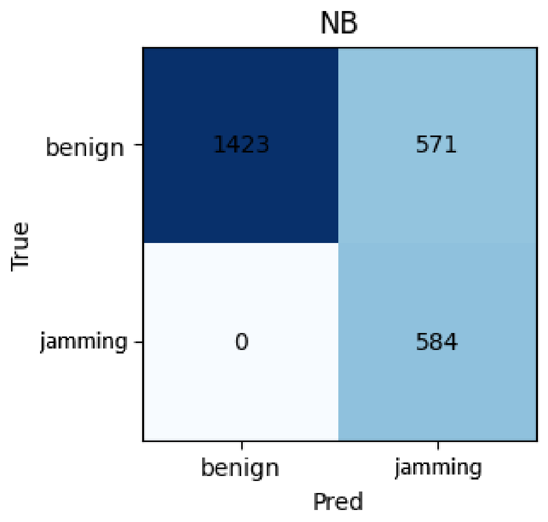

Figure 13. Naïve Bayes Jamming Confusion Matrices (Processed Data).

4.3 Feature Importance analyze

Table 3 summarizes the most influential features for spoofing across the five models. XGBoost computes feature importance using Gain. Random Forest relies on mean decrease in impurity. SVM and KNN are assessed with permutation importance. Naïve Bayes adopts a closed-form measure derived from class-conditional means and variances.

The feature `lat_y` denotes the absolute latitude, and `hdop` measures the horizontal dilution of precision, which reflects satellite geometry. Under spoofing, an adversary can introduce an abrupt offset to the reported position, producing a visible drift. Because the spoofed signal cannot fully reproduce the true satellite configuration, `hdop` increases. Concurrent large deviations in `lat_y` and abnormal rises in `hdop` are therefore strong indicators of spoofing.

The feature `vel_m_s` is the ground-speed magnitude, and `vel_n_m_s` is its northward component. The feature `cog_rad` is the course over ground in radians and represents the direction of motion, while `c_variance_rad` quantifies short-term variability in that course. During spoofing, forged coordinates may imply a drift in one direction while inertial sensors indicate motion in another, creating inconsistency between the speed vector and the course. If the coordinates are repeatedly adjusted, the course exhibits jumps across consecutive samples and `c_variance_rad` increases. The joint behavior of speed magnitude, directional components, and course

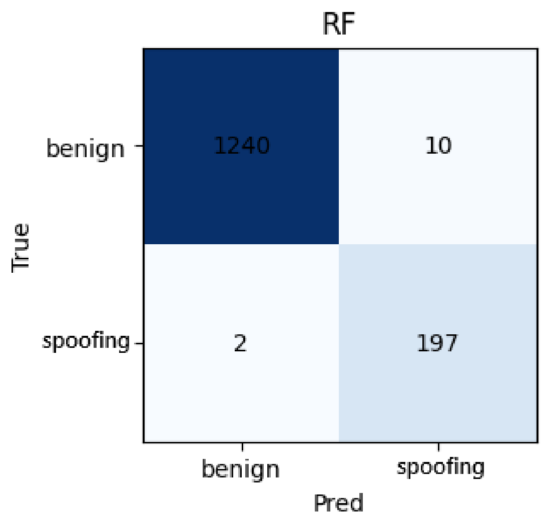

Figure 14. Random Forest Spoofing Confusion Matrices (Processed Data).

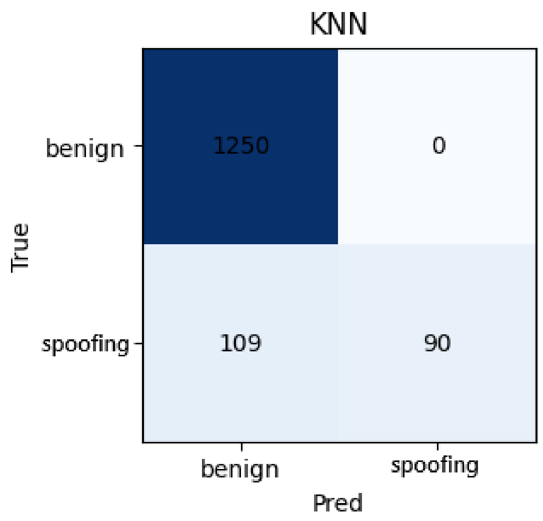

Figure 16. KNN Spoofing Confusion Matrices (Processed Data).

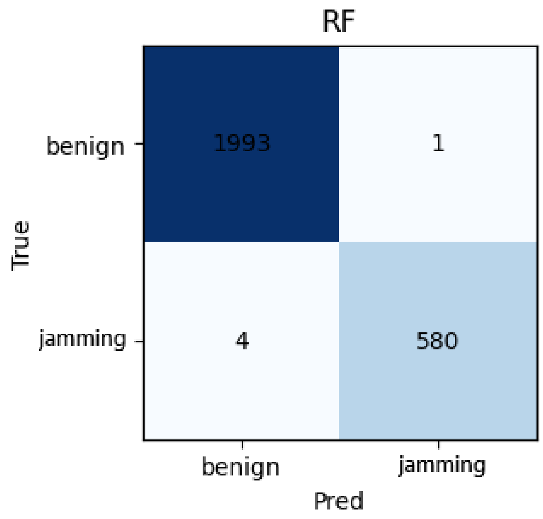

Figure 15. Random Forest Jamming Confusion Matrices (Processed Data).

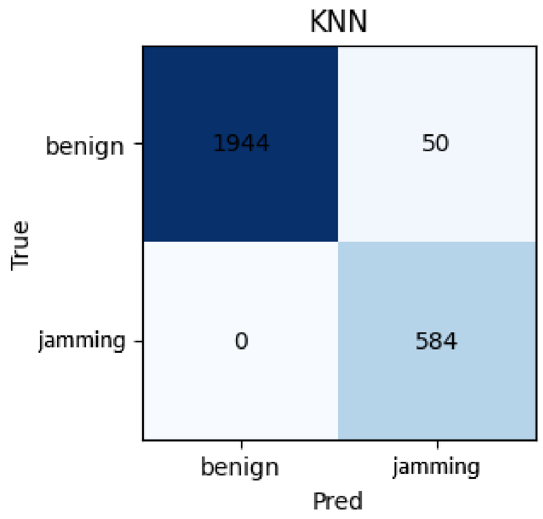

Figure 17. KNN Jamming Confusion Matrices (Processed Data).

stability enables detection even without relying solely on satellite-quality indicators.

XGBoost favors splits that rapidly reduce error and therefore relies on the combination of `vel_m_s`, `cog_rad`, and `vel_n_m_s`. When the reported position is shifted but the motion no longer agrees with the course, splits on these variables separate benign flight from spoofing effectively and yield high Gain values. The feature `c_variance_rad` reinforces this effect by capturing step-like changes in the course when the attacker fine-tunes the forged position.

Random Forest subsamples features per tree and does not depend on a single variable. The magnitude and northward component of speed appear frequently across trees, so their impurity reduction accumulates and they rank highly by MDI. The features `cog_rad` and `c_variance_rad` capture directional shifts and volatility, providing auxiliary split points that reduce impurity in many trees and thus raise their average importance.

SVM with an RBF kernel maps data to a higher-dimensional space and separates classes by a maximum-margin hyperplane. Permutation tests indicate that shuffling any single feature has a limited effect, but jointly disrupting the alignment between `vel_m_s` and `cog_rad` alters the locations of support vectors, narrows the margin, and degrades accuracy. This suggests that SVM relies on geometric consistency between speed and course.

KNN classifies by distances to neighboring samples. After standardization, `vel_m_s` and `vel_n_m_s`

Table 3. Key features of five machine learning algorithms for identifying GPS spoofing.

Feature	XGBoost	SVM	Naïve Bayes	Random Forest	KNN
vel_m_s	16.07284	0.278032	0.025167	0.012208	0.482895
lat_y	23.4295	0.262439	-	-	0.04205
hdop	16.69048	0.177657	-0.00228	0.017517	0.20293
vel_n_m_s	11.11447	0.127849	0.003562	0.048393	0.406282
cog_rad	15.42732	0.024408	-0.08047	0.020101	0.536137
c_variance_rad	15.08844	0.088598	0.029513	0.01675	0.016717
jamming_indicator	3.38069	0.036092	0.009363	-0.0067	0.204817
satellites_used	11.2902	0.004926	-0.00998	0.005863	0.105328

Table 4. Key features of five machine learning algorithms for identifying GPS jamming.

Feature	XGBoost	SVM	Naïve Bayes	Random Forest	KNN
s_variance_m_s	54.86269	0.272648	0.096456	0.235145	0.494284
evh	40.73054	0.131591	0.087847	0.029756	0.306661
eph_x	48.45486	0.224056	0.007857	-	0.109449
vel_d_m_s	19.12197	0.031293	0.067787	0.044473	0.329122
epv	44.60001	0.163062	0.009863	-0.00092	0.127749
epv_x	36.74346	0.155794	-	-	0.039005
vel_n_m_s	10.15541	0.001373	0.049147	0.031344	0.025268
vdop	11.56408	0.020045	0.002591	0.001839	0.108672

dominates the distance metric. When the speed magnitude or its direction deviates from nearby benign samples, the point becomes closer to attack neighborhoods. If c_variance_rad increases due to unstable course, distances to benign clusters grow further, and spoofing samples occupy a distinct region that KNN can identify.

Naïve Bayes models each feature with class-conditional Gaussians. During spoofing, the class means of vel_m_s, vel_n_m_s, and cog_rad diverge from benign flight. Preprocessing reduces within-class variance, which increases their discriminative ratios. The feature c_variance_rad also rises under spoofing and widens the separation. As a result, speed magnitude, directional components, course, and its variability yield the highest closed-form importance for NB in the spoofing scenario.

Table 4 summarizes the most influential features for the jamming scenario across the five models. The same importance measures as in Table 3 are used for XGBoost, Random Forest, SVM, KNN, and Naïve Bayes.

The feature s_variance_m_s is the receiver-reported speed accuracy estimate from PX4. The features eph_x and epv are the one-sigma horizontal and vertical position errors, and evh is the one-sigma horizontal velocity error. During jamming, the pseudorange solution, which estimates the receiver position from satellite range measurements, becomes unstable. The variance in speed estimates grows first, and then position and velocity errors inflate together. Short-term concurrent increases in s_variance_m_s and either eph_x or epv indicate that the band is being covered by high-power noise, which makes this set of accuracy and error indicators a strong basis for detecting jamming.

The features vel_d_m_s and vel_n_m_s are the downward and northward velocity components. The feature c_variance_rad quantifies short-term variability of the course, and vdop measures vertical dilution of precision and thus satellite geometry in the vertical dimension. Under jamming, the pseudorange solution oscillates among competing fixes, velocity components show nonphysical spikes, the course variance rises sharply, and partial satellite masking elevates vdop. Detecting abnormal velocity components together with increased course jitter and elevated vdop enables jamming recognition without relying on absolute coordinates.

For spoofing, the most informative signals are coordinate shifts captured by lat_y together with spikes in hdop, and the loss of alignment between motion and heading reflected by vel_m_s, vel_n_m_s, cog_rad, and c_variance_rad. For jamming, early growth in s_variance_m_s is followed by inflation in eph_x, epv, and evh, while vdop and c_variance_rad rise as satellites are partially masked. XGBoost and Random Forest prioritize these variables as high-gain or high-impurity-reduction splits. SVM relies on the geometric consistency between speed and course after projection, KNN separates samples by standardized distances dominated by speed components, and Naive Bayes exploits shifts in class means relative to within-class variance.

5 Conclusion

This work studies GPS spoofing and jamming detection for UAVs using telemetry-only learning on PX4 with the Gazebo simulator. According to the results, high dimensionality weakens most models on the raw set, whereas normalization and PCA substantially improve all methods. The confusion matrices confirm these trends, because errors on the raw set cluster on the boundary between benign and jamming and spoofing is often split across the two, while after preprocessing the matrices for tree ensembles are nearly diagonal with only small and localized confusions. SVM and Gaussian Naive Bayes raise many benign false alarms, and k-nearest neighbors misses many spoofing windows even when jamming separates cleanly. Tree ensembles achieve the most balanced gains and the best configuration reaches an F1-score of 0.998 on the processed set, which underscores the benefit of careful preprocessing.

Feature-importance analysis explains these outcomes and suggests practical cues for monitoring. For spoofing, coordinate shifts and elevated hdop, together with mismatches among vel_m_s, vel_n_m_s, and cog_rad and increases in c_variance_rad, provide early and reliable signals. For jamming, s_variance_m_s rises first, followed by growth in eph_x, epv, and evh, alongside higher vdop and c_variance_rad. These findings point to a compact set of high-yield features for online checks and to simple temporal consistency rules that can stabilize alerts. Future work will validate these signals in field trials, study adaptive thresholds under changing environments, and explore domain shift aware training to preserve performance across platforms and missions.

Data Availability Statement

Data will be made available on request.

Funding

This work was supported in part by the National Science and Technology Council (NSTC), Taiwan, under Project NSTC 114-2622-E-A49-022, Project NSTC 114-2221-E-A49-210, Project NSTC 114-2634-F-011-002-MBK, and Project NSTC 114-2923-E-194-001-MY3; in part by the Hon Hai Research Institute, Taipei, Taiwan, under Project 114UA90042; in part by the Industry-Academia Innovation School, NYCU, Taiwan, under Project 113UC2N006.

Conflicts of Interest

The authors declare no conflicts of interest.

Ethical Approval and Consent to Participate

Not applicable.

References

[1] Khalil, J. (2024, October 1). *GNSS spoofing threatens airline safety, alarming pilots and aviation officials*. GPS World. Retrieved from https://www.gpsworld.com/gnss-spoofing-threatens-airline-safety-alarming-pilots-and-aviation-officials/ (accessed on 27 December, 2025).

[2] *Iran 'building copy of captured US drone' RQ-170 Sentinel*. (2012, April 22). BBC News. Retrieved from https://www.bbc.com/news/world-middle-east-17805201 (accessed on 27 December, 2025).

[3] Whipple, T. (2024, October 5). *1,000 flights a day have signals jammed over war zones*. Latest news & breaking headlines | The Times and The Sunday Times. Retrieved from https://www.thetimes.com/uk/technology-uk/article/1000-planes-a-day-have-signals-jammed-as-they-fly-over-war-zones-swtdflkqc (accessed on 27 December, 2025).

[4] Tangel, A., & FitzGerald, D. (2024). Electronic warfare spooks airlines, pilots and air-safety officials. *Wall Street Journal*. Retrieved from https://www.wsj.com/business/airlines/electronic-warfare-spooks-airlines-pilots-and-air-safety-officials-60959bbd (accessed on 27 December, 2025).

[5] Javaid, A. Y., Sun, W., Devabhaktuni, V. K., & Alam, M. (2012, November). Cyber security threat analysis and modeling of an unmanned aerial vehicle system. In *2012 IEEE conference on technologies for homeland security (HST)* (pp. 585-590). IEEE. [Crossref]

[6] Samland, F., Fruth, J., Hildebrandt, M., Hoppe, T., & Dittmann, J. (2012). AR. Drone: security threat analysis and exemplary attack to track persons. *Intelligent robots and computer vision XXIX: algorithms and techniques, 8301*, 158-172. [Crossref]

[7] Mansfield, K., Eveleigh, T., Holzer, T. H., & Sarkani,

S. (2013, November). Unmanned aerial vehicle smart device ground control station cyber security threat model. In *2013 IEEE International Conference on Technologies for Homeland Security (HST)* (pp. 722-728). IEEE. [Crossref]

[8] Kerns, A. J., Shepard, D. P., Bhatti, J. A., & Humphreys, T. E. (2014). Unmanned aircraft capture and control via GPS spoofing. *Journal of field robotics, 31*(4), 617-636. [Crossref]

[9] Bonebrake, C., & O'Neil, L. R. (2014). Attacks on GPS time reliability. *IEEE Security & Privacy, 12*(3), 82-84. [Crossref]

[10] Vasconcelos, G., Carrijo, G., Miani, R., Souza, J., & Guizilini, V. (2016, October). The impact of DoS attacks on the AR. Drone 2.0. In *2016 XIII Latin American robotics symposium and IV Brazilian robotics symposium (LARS/SBR)* (pp. 127-132). IEEE. [Crossref]

[11] de Carvalho Bertoli, G., Pereira, L. A., & Saotome, O. (2021, November). Classification of denial of service attacks on Wi-Fi-based unmanned aerial vehicle. In *2021 10th Latin-American Symposium on Dependable Computing (LADC)* (pp. 1-6). IEEE. [Crossref]

[12] Yaacoub, J. P., Noura, H., Salman, O., & Chehab, A. (2020). Security analysis of drones systems: Attacks, limitations, and recommendations. *Internet of Things, 11*, 100218. [Crossref]

[13] Jansen, K., Tippenhauer, N. O., & Pöpper, C. (2016, December). Multi-receiver GPS spoofing detection: Error models and realization. In *Proceedings of the 32nd Annual Conference on Computer Security Applications* (pp. 237-250). [Crossref]

[14] Magiera, J., & Katulski, R. (2015). Detection and mitigation of GPS spoofing based on antenna array processing. *Journal of applied research and technology, 13*(1), 45-57. [Crossref]

[15] Liu, C., Zhang, Y., Niu, G., Jia, L., Xiao, L., & Luan, J. (2023). Towards reinforcement learning in UAV relay for anti-jamming maritime communications. *Digital Communications and Networks, 9*(6), 1477-1485. [Crossref]

[16] Jafarnia-Jahromi, A., Broumandan, A., Nielsen, J., & Lachapelle, G. (2012). GPS vulnerability to spoofing threats and a review of antispoofing techniques. *International Journal of Navigation and Observation, 2012*(1), 127072. [Crossref]

[17] Masadeh, A. E., Alhafnawi, M., Salameh, H. A. B., Musa, A., & Jararweh, Y. (2022). Reinforcement learning-based security/safety uav system for intrusion detection under dynamic and uncertain target movement. *IEEE Transactions on Engineering Management, 71*, 12498-12508. [Crossref]

[18] Alsumayt, A., Nagy, N., Alsharyofi, S., Al Ibrahim, N., Al-Rabie, R., Alahmadi, R., ... & Alahmadi, A. A. (2024). Detecting Denial of Service Attacks (DoS) over the Internet of Drones (IoD) Based on Machine Learning. *Sci, 6*(3), 56. [Crossref]

[19] Feng, Z., Guan, N., Lv, M., Liu, W., Deng, Q., Liu, X., & Yi, W. (2020). Efficient drone hijacking detection using two-step GA-XGBoost. *Journal of Systems Architecture, 103*, 101694. [Crossref]

[20] Wang, Z., Li, Y., Wu, S., Zhou, Y., Yang, L., Xu, Y., ... & Pan, Q. (2023). A survey on cybersecurity attacks and defenses for unmanned aerial systems. *Journal of Systems Architecture, 138*, 102870. [Crossref]

[21] Ahmad, W., Almaiah, M. A., Ali, A., & Al-Shareeda, M. A. (2024, April). Deep learning based network intrusion detection for unmanned aerial vehicle (uav). In *2024 7th World Conference on Computing and Communication Technologies (WCCCT)* (pp. 31-36). IEEE. [Crossref]

[22] Ma, T., Zhang, X., & Miao, Z. (2024). Detection of UAV GPS spoofing attacks using a stacked ensemble method. *Drones, 9*(1), 2. [Crossref]

[23] Calvo-Palomino, R., Bhattacharya, A., Bovet, G., & Giustiniano, D. (2020, August). Short: LSTM-based GNSS spoofing detection using low-cost spectrum sensors. In *2020 IEEE 21st International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)* (pp. 273-276). IEEE. [Crossref]

[24] Sung, Y. H., Park, S. J., Kim, D. Y., & Kim, S. (2022). GPS spoofing detection method for small UAVs using 1D convolution neural network. *Sensors, 22*(23), 9412. [Crossref]

[25] Park, K. H., Park, E., & Kim, H. K. (2020, August). Unsupervised intrusion detection system for unmanned aerial vehicle with less labeling effort. In *International Conference on Information Security Applications* (pp. 45-58). Cham: Springer International Publishing. [Crossref]

[26] Pham, H. X., La, H. M., Feil-Seifer, D., & Nguyen, L. V. (2018). Autonomous uav navigation using reinforcement learning. *arXiv preprint arXiv:1801.05086*.

[27] Cengiz, K., Lipsa, S., Dash, R. K., Ivković, N., & Konecki, M. (2024). A novel intrusion detection system based on artificial neural network and genetic algorithm with a new dimensionality reduction technique for UAV communication. *IEEE access, 12*, 4925-4937. [Crossref]

[28] Whelan, J., Sangarapillai, T., Minawi, O., Almehmadi, A., & El-Khatib, K. (2020, November). Novelty-based intrusion detection of sensor attacks on unmanned aerial vehicles. In *Proceedings of the 16th ACM symposium on QoS and security for wireless and mobile networks* (pp. 23-28). [Crossref]

[29] European Union Aviation Safety Agency (EASA). (2024). *Safety Information Bulletin: Global Navigation Satellite System Outage and Alterations Leading to Communication / Navigation / Surveillance Degradation* (SIB No. 2022-02R3). Retrieved from https://ad.easa.europa.eu/ad/2022-02R3 (accessed on 27 December, 2025).

[30] Daniel, S., Jahshan, B., & Todd, H. (2012). Drone Hack: Spoofing Attack Demonstration on a Civilian Unmanned Aerial Vehicle. *GPS World, The Business and Technology of GNSS*. Retrieved from https://www.gpsworld.com/drone-hack/ (accessed on 27 December, 2025).

[31] Whelan, J., Sangarapillai, T., Minawi, O., Almehmadi, A., & El-Khatib, K. (2020). UAV Attack Dataset. *IEEE Dataport*. [Crossref]

Ying-Chen Liu received the M.S. degree in Information Management from National Dong Hwa University, Taiwan. Her research interests include unmanned aerial vehicle (UAV) security and related cybersecurity technologies. (Email: 611235109@gmail.com)

Lin-Fa Lee received the B.S. and M.S. degrees from National Dong Hwa University, Hualien, Taiwan. He is currently pursuing the Ph.D. degree with the Institute of Artificial Intelligence Innovation, National Yang Ming Chiao Tung University, Hsinchu, Taiwan. His research interests include AI security, privacy-preserving machine learning, and blockchain systems. (Email: prolugue.li14@nycu.edu.tw)

Kuo-Hui Yeh serves as a professor at the Institute of Artificial Intelligence Innovation, National Yang Ming Chiao Tung University, Hsinchu, Taiwan. Prior to this appointment, he was a professor in the Department of Information Management at National Dong Hwa University, Hualien, Taiwan, from February 2012 to January 2024. Dr. Yeh earned his M.S. and Ph.D. degrees in Information Management from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 2005 and 2010, respectively. He has contributed over 150 articles to esteemed journals and conferences, covering a wide array of research interests such as IoT security, Blockchain, NFC/RFID security, authentication, digital signatures, data privacy and network security. Furthermore, Dr. Yeh plays a pivotal role in the academic community, serving as an Associate Editor (or Editorial Board Member) for several journals, including the Journal of Information Security and Applications (JISA), Human-centric Computing and Information Sciences (HCIS), Digital Health, Symmetry, Journal of Internet Technology (JIT) and CMC-Computers, Materials & Continua. In the professional realm, Dr. Yeh is recognized as a fellow of the British Computer Society (BCS) and a Senior Member of IEEE, and holds memberships with BCS, ISO, ISA, ISACA, CAA, and CCISA. His professional qualifications include certifications like CISSP, CISM, Security+, ISO 27001/27701/42001 Lead Auditor, IEC 62443-2-1 Lead Auditor, and ISA/IEC 62443 Cybersecurity Expert, covering fundamentals, risk assessment, design, and maintenance specialties. (Email: khyeh@nycu.edu.tw)