A Cross-Scenario GNSS Spoofing Detection Method Based on Transfer Learning

Jun Xu, Liangyu Qin, Chao Sun (*corresponding author*), Lu Bai, Shuai Zhang, Wenquan Feng, Lei Xu
Department Electronic and Information Engineering, Beihang University

BIOGRAPHY

Jun Xu received his B.E. degree in 2025 in Artificial Intelligence from China University of Mining and Technology(BeiJing), China. Now he is pursuing his Master's. degree at Beihang University. His research focuses on GNSS anti-spoofing and GNSS-5G hybrid positing methods based on machine learning.

Liangyu Qin is pursuing his B.E. degree in Electronic and Information Engineering at Beihang University. His research focuses on GNSS anti-spoofing methods based on machine learning.

Chao Sun received a B.S. degree in Electronic and Information Engineering from Beihang University in July 2013 and Ph.D. degree in Communication and Information system from Beihang University in June 2019. Now he is a Postdoc in the Department of Electronic and Information Engineering, Beihang University. From 2017 to 2018, he was studying as a visiting Ph.D. student at the Australian Centre for Space Engineering Research, University of New South Wales, Sydney, NSW, Australia. His research focuses on GNSS spoofing detection and interference mitigation techniques.

Lu Bai received her B.E. degree in 2014 in Electronic and Information Engineering from Beihang University, China. Now she is pursuing her Ph.D. degree at Beihang University. Her research focuses on GNSS anti-spoofing and GNSS-5G hybrid positing methods.

Shuai Zhang received his B.E. degree in 2023 in Electronic and Information Engineering from Beihang University, China. Now he is pursuing his Master's. degree at Beihang University. His research focuses on GNSS anti-spoofing and GNSS-5G fusion positioning methods.

Wenquan Feng received Ph.D. degree in communication and information system from Beihang University. Now he is a professor at the Department of Electronic and Information Engineering, Beihang University, Beijing, China. His current research interests include satellite navigation, satellite communication, and complex system fault diagnosis.

Lei Xu received his B.E. degree in 2020 in Electronic and Information Engineering from Beihang University, China. Now he is pursuing his Master's. degree at Beihang University. His research focuses on GNSS anti-spoofing methods based on machine learning.

ABSTRACT

As a critical information infrastructure, the reliability of positioning, navigation, and timing (PNT) services from Global Navigation Satellite Systems (GNSS) is directly related to national security and economic operation. However, the open nature of satellite signals and significant propagation path loss make them susceptible to spoofing attacks. Existing spoofing detection methods exhibit poor cross-scenario generalization, causing performance degradation with scarce labeled data and high false alarm rates in multipath environments. To address these issues, we propose a cross-scenario detection method that employs a Transformer to capture long-term signal dynamics and a pre-training/fine-tuning transfer learning framework to ensure rapid adaptation to new data domains. Experimental results demonstrate that the proposed transfer learning framework improve detection accuracy by up to 10.54% and reduce training time by 75.6% using only 5% of labeled data from a new domain. Furthermore, the model demonstrates high robustness in complex scenarios, achieving a 99.75% detection rate against a 0.03% false alarm rate in concurrent spoofing and multipath environments. This work provides a highly robust solution with cross-scenario adaptability for anti-spoofing protection in complex electromagnetic environments.

Keywords: GNSS Spoofing Detection, Multipath Effect, Transformer, Transfer Learning

1. INTRODUCTION

The Global Navigation Satellite System (GNSS) serves as the spatiotemporal benchmark for modern society, with its positioning, navigation, and timing (PNT) services being integral to critical sectors like military reconnaissance, intelligent transportation, and precision agriculture. However, as GNSS signals travel from satellites to ground-based receivers, they are highly vulnerable to spoofing attacks [1]. Because spoofing signals replicate authentic signals in the time, frequency, and code domains, identifying it has become a central challenge in ensuring GNSS security [2].

Traditional spoofing detection techniques include spatial processing [3] [4], Signal Quality Monitoring (SQM) [5], [6], [7], [8], and multi-source information verification [9], [10], [11]. Early methods relied on spatial processing, exploiting the distinct propagation path characteristics between authentic and spoofing signals. For example, a Cornell University team led by Mark Psiaki proposed a method in 2013 based on carrier-phase time-variance analysis using a single moving antenna [3], but it was limited by motion control precision. In 2014, Saeed Daneshmand et al. at the University of Calgary developed an antenna array system that used beamforming to nullify spoofing signals [4], though it required complex calibration procedures.

SQM-based methods operate on the principle that spoofing signals often exhibit quality differences from authentic GNSS signals [5], [6], [7], such as code phase jitter or carrier phase noise. These methods monitor such parameters to detect anomalies. In 2022, Yang Bin et al. designed a correlation peak feature map analysis framework to distinguish between multipath, jamming, and spoofing attacks [8]. While SQM shows potential, its effectiveness against low-complexity spoofing is limited, and it can suffer from false positives or negatives in high-dynamic environments. Multi-source verification methods can also detect spoofing by cross-referencing information [9], [10], [11], as spoofers struggle to perfectly replicate all signal characteristics, but this approach introduces additional hardware costs.

Data-driven machine learning methods have shown superior generalization capabilities in complex scenarios by mining features from vast amounts of signal data. In 2022, a team from Sun Yat-sen University proposed a multi-parameter GNSS spoofing detection method using a Support Vector Machine (SVM) [12]. This approach significantly improved detection performance over single-parameter methods but was sensitive to SVM parameter and kernel function selection [13], [14]. That same year, Sung et al. from South Korea's Agency for Defense Development introduced a 1D Convolutional Neural Network (CNN) based on a residual network architecture [15], which demonstrated high accuracy and low computational cost for real-time spoofing detection on small UAVs.

Despite progress in GNSS spoofing detection, existing research is limited by two main factors: a strong dependency on labeled data that restricts adaptability to new target scenarios, and the high false alarm rates commonly encountered in multipath environments. Therefore, this paper proposes a cross-scenario spoofing detection method based on Transformer and transfer learning [16]. This method designs a spoofing detection model using a Transformer to capture long-term dynamic signal features via a multi-head self-attention mechanism, thereby accurately distinguishing multipath from spoofing signals. It then introduces transfer learning to pre-train the model on abundant data from existing scenarios, effectively addressing the bottleneck of data scarcity and enhancing the model's adaptability to new scenarios.

2. SIGNAL MODEL

2.1 Spoofing Attack Signal Model

For a spoofing attack targeting the GPS L1 signal, the attacker generates a local C/A code sequence based on the public PRN number. The navigation message's ephemeris and timestamp data are altered to reflect a predetermined false position. The resulting spoofing signal for the i-th satellite can be represented by Equation (1):

$$s_{spoof}^{i}(t) = C^{i}(t-\tau^{i})D^{i}(t)\cos(2\pi(f_c + f_d^{i}) + \varphi^{i}(t)) + n^{i}(t) \tag{1}$$

where $C^{i}(t)$ is the PRN code, τ^{i} is the code phase offset, $D^{i}(t)$ is the counterfeit navigation message, $n^{i}(t)$ represents the additive noise. f_c and f_d^{i} are the carrier frequency and Doppler shift respectively, and φ^{i} is the initial phase.

The attack is typically executed using a towed spoofing strategy[17], as illustrated in Figure 1, which consists of three phases.

The first phase is Initialization and Synchronization (a). The spoofer first captures the authentic satellite signal to estimate its power, code phase, and Doppler shift. It then generates a corresponding spoofing signal with a modified navigation message and transmits it at a power level slightly below the authentic signal to avoid immediate detection while allowing the receiver to track it.

The second phase is Power Towing and Signal Transition (b). The spoofer gradually increases the power of the spoofing signal. The receiver's tracking loop begins to lock onto the stronger spoofing signal, whose correlation peak starts to separate from the authentic signal's peak, causing the receiver's position solution to deviate.

The third phase is Spoofing Completion (c). As the spoofing signal power continues to increase, its correlation peak completely separates from the authentic signal's peak. The receiver fully locks onto the spoofing signal, and its positioning is now entirely controlled by the attacker.

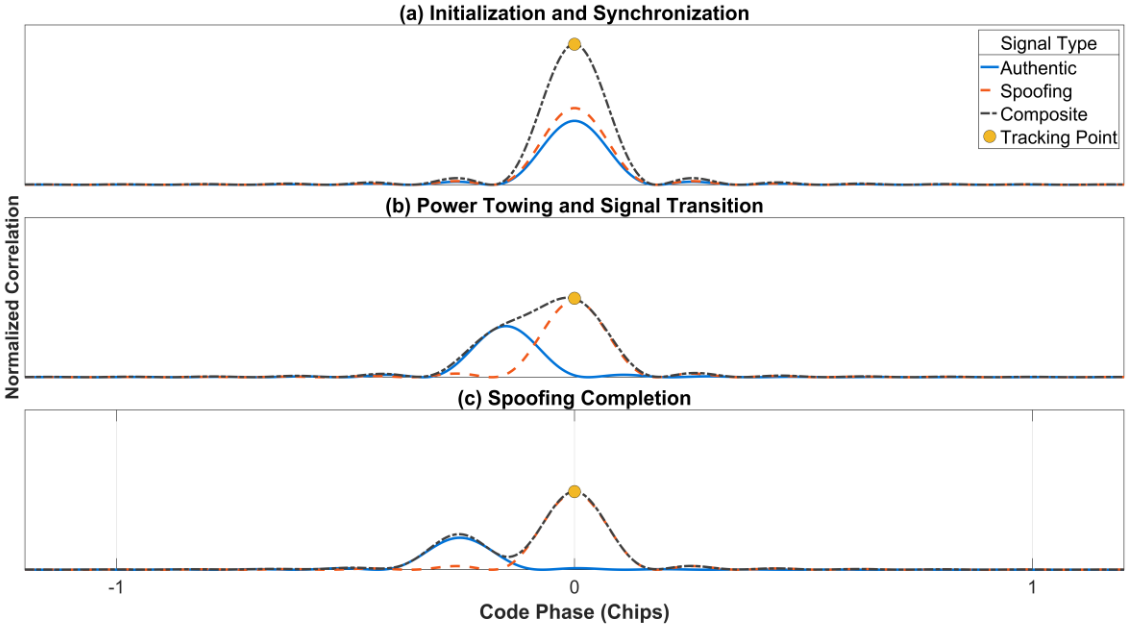


FIGURE 1 Towed Spoofing Process

2.2 Multipath Signal Model

In a real-world environment, when satellite signals are reflected by obstacles such as buildings and the ground, the receiver's antenna simultaneously receives a direct signal and multiple delayed reflected signals, as shown in Equations (2) to (4). The receiver must correlate a locally generated C/A code with the incoming signal, which is a composite of the direct wave and multiple reflected waves. Consequently, the local C/A code correlates separately with the direct and reflected waves. This distorts the correlation peak, which is intended to reflect the code phase of the direct wave, and in severe cases, can lead to a loss of lock in the code tracking loop.

$$s_{authentic}(t) = Ap(t)\sin(2\pi ft) + n(t) \tag{2}$$

$$s_i(t) = \alpha_i Ap(t-\tau_i)\sin(2\pi f(t-\tau_i) + \Delta\varphi_i) + n(t) \tag{3}$$

$$s(t) = s_{authentic}(t) + \sum_i s_i(t) = Ap(t)\sin(2\pi ft) + \sum_i [\alpha_i Ap(t-\tau_i) sin(2\pi ft + \varphi_i)] + n(t) \tag{4}$$

where A is the authentic signal amplitude, f is the carrier frequency, $p(t)$ is the pseudorandom noise code, $n(t)$ is the additive noise, α_i , τ_i and φ_i are the amplitude attenuation, propagation delay, and phase shift of the reflected signals, respectively.

Both multipath and spoofing signals can cause correlation peak distortion. However, multipath signals typically exhibit discontinuous fluctuations in the correlation peak because they are caused by reflections from fixed environmental objects and complex factors. In contrast, spoofing signals, being artificially manipulated, often show distinct, phased, and abrupt changes, including the three stages of initialization, partial separation, and complete separation of the correlation peak. Non-sequential machine learning models, such as SVM and Random Forest, tend to misclassify multipath signals as spoofing signals, leading to false alarms. This wastes resources in scenarios where the objective is solely to detect spoofing. Since these models typically treat signal features at each moment as independent samples, they ignore the temporal continuity of the signal. Therefore, this paper employs a Transformer-based model to capture global dependencies in the signal through its self-attention mechanism, effectively distinguishing the non-continuous fluctuations of multipath from the phased changes characteristic of spoofing signals.

3. CROSS-SCENARIO DETECTION FRAMEWORK DESIGN

To address the issue of capturing long-term temporal features of GNSS signals in real-world environments with multipath effects, this paper first designs a Transformer-based spoofing detection model. This model uses a multi-head self-attention mechanism to model long-sequence dependencies. Secondly, a two-stage transfer learning strategy involving pre-training and fine-tuning is adopted to effectively enhance the model's feature extraction capabilities and detection accuracy in new scenarios.

3.1 Transformer-Based Spoofing Detection Model

The proposed GNSS spoofing detection model is shown in Figure 2. It consists of three main components: a signal embedding layer, a Transformer encoder, and a classification output layer, designed for end-to-end processing of input signal sequences.

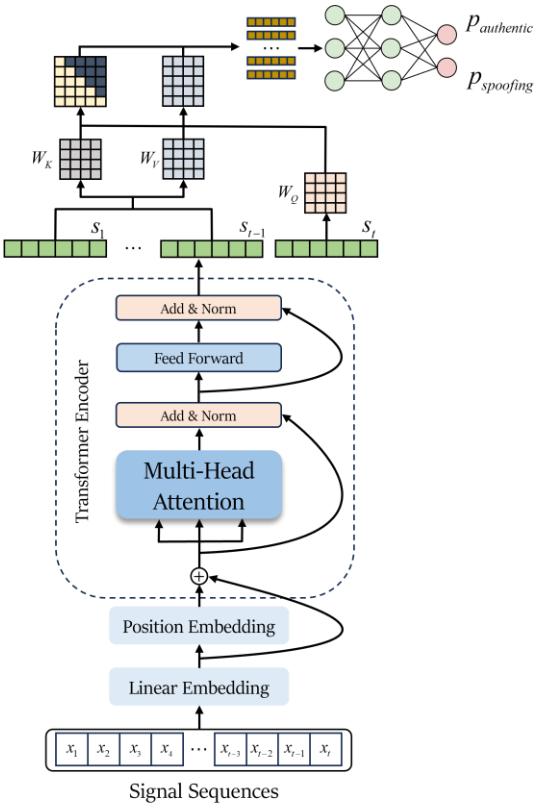


Figure 2 Transformer-based Spoofing Detection Model

Input observation data is first mapped to a high-dimensional space via a linear transformation, as shown in Equation (5). This linear embedding process uses a learnable weight matrix W_e to map the input features X_i of each time step into a hidden layer vector, enhancing the model's ability to represent complex features. The resulting vectors are then combined with positional encoding information, obtained through Equations (6) and (7), and fed into the Transformer encoder.

$$Z = XW_e + PE \tag{5}$$

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{\frac{2i}{d}}}\right) \tag{6}$$

$$PE_{(pos,2i+1)} = cos\left(\frac{pos}{10000^{\frac{2i}{d}}}\right) \tag{7}$$

Within the encoder, after each attention calculation, the output is added to the original input (a residual connection) and then passed through a layer normalization step. This helps mitigate the vanishing gradient problem during the training of deep networks. The modules within the encoder are stacked, meaning the output of each layer serves as the input for the next, creating a deep feature abstraction. This allows the model to capture both static signal characteristics and the dynamic patterns of a spoofing attack over time.

The subsequent classification output layer transforms the abstract features extracted by the encoder into a spoofing detection result. Notably, to enhance the model's representation of key stages of a spoofing signal, learnable weight parameters are introduced in the Transformer output layer.

$$\beta_t = \frac{\exp(-\gamma \cdot \frac{N-t}{N})}{\sum_{i=1}^{N} \exp(-\gamma \cdot \frac{N-i}{N})}, \quad t = 1, 2, ..., N \tag{8}$$

where γ is a learnable decay coefficient, N is the number of time steps, and $\beta_t \in [0,1]$ is the normalized weight for the t-th time step, with more recent time steps receiving higher weights.

The output feature vectors from the encoder across multiple time steps are initialized with weights according to Equation (8), giving higher importance to more recent steps. Since real-world predictions are typically most related to the most recent observations, the output of the final time step Z^L is used as the query vector (Q). The outputs from all other time steps serve as the key (K) and value (V) vectors. These are passed through the learnable parameter matrices $\mathbf{W_Q}$, $\mathbf{W_K}$, and $\mathbf{W_V}$ of the self-attention module to produce a time-weighted feature output vector(Equation 11):

$$K_{weighted} = \beta \odot Z^L, \quad V_{weighted} = \beta \odot Z^L \tag{9}$$

$$\mathbf{Q} = Z_N^L \mathbf{W_Q}, \quad \mathbf{K} = K_{weighted} \mathbf{W_K}, \quad \mathbf{V} = V_{weighted} \mathbf{W_V} \tag{10}$$

$$\text{Output} = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathbf{T}}}{\sqrt{d_{model}}} \right) \mathbf{V} \tag{11}$$

where $Z^L \in \mathbb{R}^{N \times d_{model}}$ is the output of the final encoder layer, $\odot$ denotes element-wise multiplication, and d_{model} is the feature dimension.

The vector is passed to a fully connected network with a softmax function, which outputs confidence scores for the presence of an authentic signal $P_{authentic}$ and a spoofing signal $P_{spoofing}$, and a final decision is made based on a predefined threshold.

3.2 Cross-Scenario Spoofing Detection Strategy

To address the performance degradation caused by differing data distributions across various real-world application scenarios, this paper employs the cross-scenario transfer learning framework shown in Figure 3. This framework retains the original model architecture but introduces a two-stage pre-training and fine-tuning strategy.

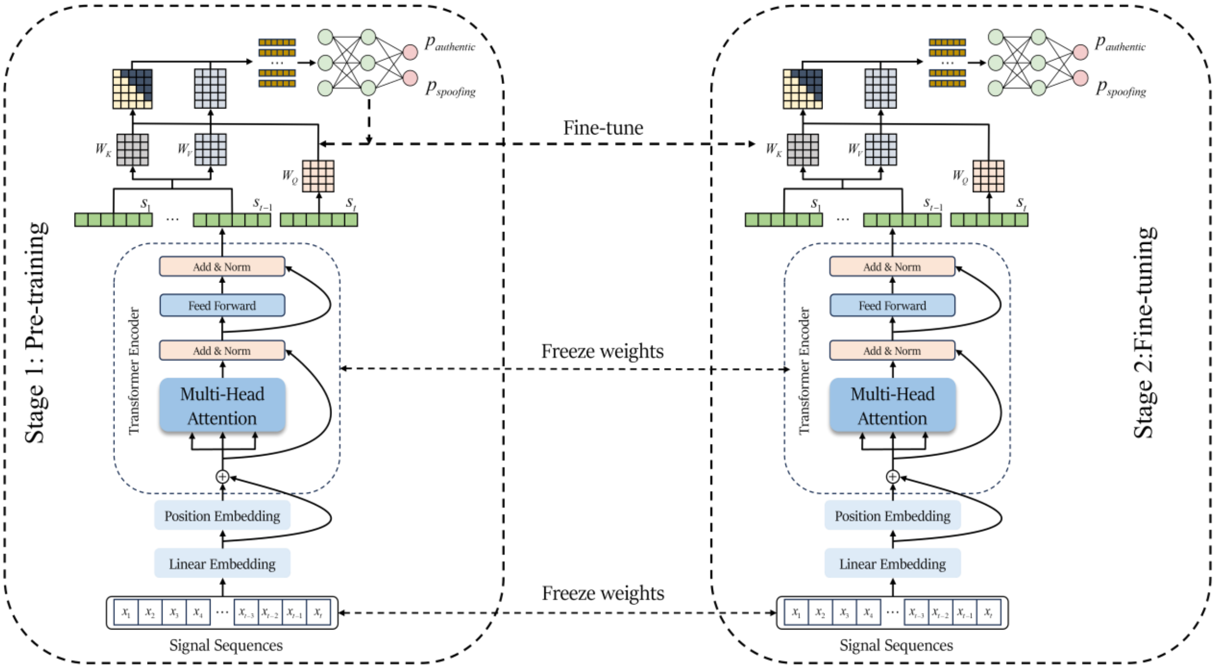


Figure 3 Cross-Scenario Spoofing Detection Strategy Based on Pre-training and Fine-tuning

By learning a general representation of spoofing features from multi-scenario data and then adapting the parameters using a small amount of labeled data from the target scenario, the model preserves its fundamental ability to discriminate spoofing signals while quickly adapting to the characteristics of a new environment. This effectively solves the model generalization problem in data-scarce situations.

After the pre-training stage establishes the model's fundamental ability to recognize various types of spoofing signals, model optimization is carried out in the fine-tuning stage for specific application scenarios. To prevent overfitting on the small dataset of the target domain, a layered parameter adjustment strategy is adopted. First, the majority of the parameters in the lower layers of the model are frozen to retain the general feature extraction capabilities acquired during pre-training. Then, only a portion of the parameters in the top layers, including the final encoder layer and the classifier layer, are fine-tuned. During this stage, a dynamic learning rate strategy is used, with the initial learning rate set to one-fifth of that used in the pre-training phase and adjusted adaptively based on performance on a validation set. By utilizing this two-stage training mechanism, the model can achieve rapid adaptation to specific scenarios based on a general spoofing discrimination model obtained from the source domain, even when only a small number of samples are available in the target scenario.

4. SIMULATION RESULTS

4.1 Dataset Collecting

The spoofing data was sourced from the public TEXBAT (Texas Spoofing Test Battery) dataset[18], which includes data from eight different spoofing scenarios covering a range of GPS L1 spoofing attacks from simple to complex. Multipath signal data was collected using a Beidou-3 new-system multi-functional configurable dual-channel sampler (SIS800), as shown in Figure 4. The collection took place between two dormitory buildings, with the antenna aimed at the nearest building to capture GPS radio frequency signals that had undergone one or more reflections.

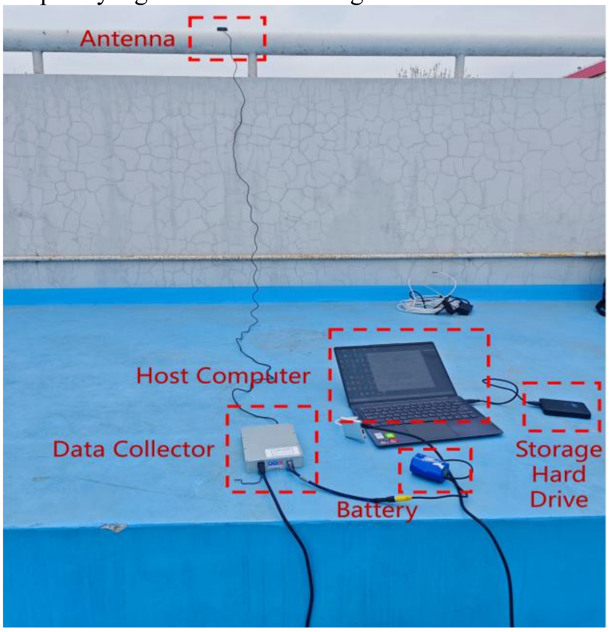
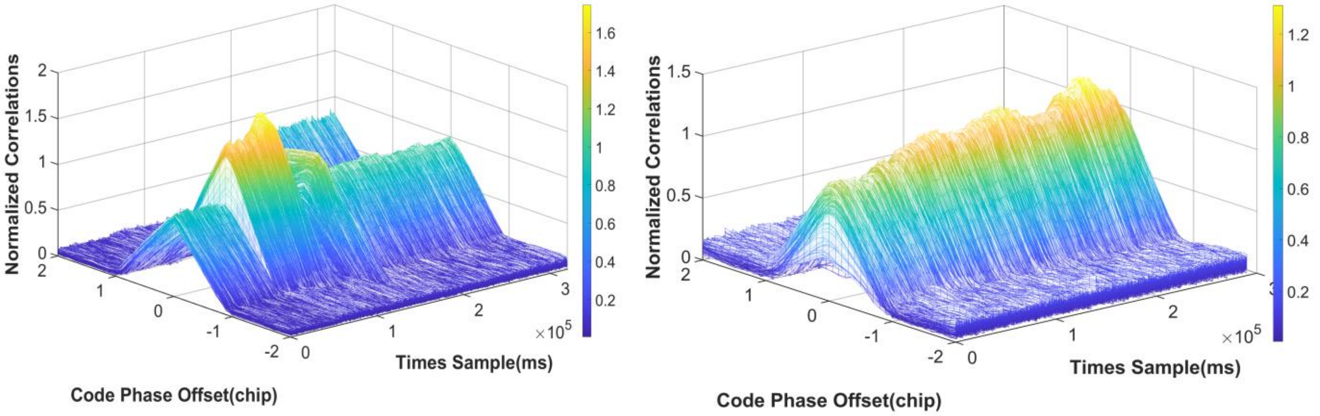


Figure 4 Multipath GPS Signal Acquisition in Practical Scenarios

4.2 Data Preprocessing

Both the spoofing and multipath data were processed using the Borre Matlab GPS software receiver. To capture the temporal evolution of the correlation peak shape for both authentic and spoofing signals, 20 sets of early-late correlators were configured in the I/Q signal tracking loop. These correlators were evenly distributed within a range of [-2, 2] chips, with a spacing of 0.2 chips.

As shown in Figure 5, which displays the receiver's correlation peak changes in spoofing and multipath environments, the spoofing signal exhibits clear phased-change characteristics, whereas the multipath signal merely causes distortion in the correlation peak shape. Therefore, the features selected for subsequent model training were the outputs of the 20 sets of I/Q channel early-late correlators IE_i , QE_i and the corresponding prompt correlator outputs IP, QP. Considering the signal characteristic variations across different scenarios and satellites, a separate standardization strategy was applied to each individual satellite within each scenario.



(a) Correlation peak change of PRN 23 satellite in-phase branch under ds3 spoofing scenario

(b) Correlation peak change of PRN 6 satellite in-phase branch under multipath scenario

Figure 5 Correlated Peak Changes in Spoofing and Multipath Environments

4.3 Simulation Results and Analysis

The concurrent multipath-spoofing scenario was configured as described in Table 1. Spoofing and authentic data were taken from the ds2 and cleanStatic scenarios of the TEXBAT dataset, respectively. Multipath data was from the real-world collection. In the training phase, data from PRN 13 and PRN 23 satellite signals were used. The cleanStatic data was treated as the negative class, and the ds2 static spoofing data was the positive class. 70% of this data about 28,750 samples was used for training. The testing phase simulated a real-world environment where GNSS signals are affected by multipath. Data for PRN 6 was from the multipath collection and was treated as a negative class, since multipath is a natural interference, not deliberate spoofing. The spoofing data was from the same ds2 scenario as the training set. The remaining 30% of the data about 7,182 samples was used for testing.

Table 1 Multipath-Spoofing Concurrent Scenario Setup

Scene	Data Source	PRN
Training Scene	cleanStatic、ds2	13、23
Testing Scene	cleanStatic、Multipath、ds2	6、16

The transfer learning task was set up as shown in Table 2 to evaluate the framework's capability across different scenarios. In this experiment, the "train from scratch" strategy for the target domain involved retraining the model using a small amount of target domain data, followed by testing on the test set. For the transfer learning strategy, 90% of the source domain data about 12,805 time windows was used for pre-training, with the remaining 10% used as a validation set to monitor the process. After pre-training, 5% of the target domain data about 117 time windows was used to fine-tune the model for the specific scenario, with the rest of the target data used for final testing.

Table 2 Data Distribution of Source and Target Domains for Multi-Task Learning

Transfer Learning Scenario	Domain Type	Data Source	PRN
A	Source	cleanStatic、cleanDynamic、ds2、ds4、ds6、ds7、ds8	6、7、16、19、22
	Target	cleanStatic、ds3	23
B	Source	cleanStatic、cleanDynamic、ds2、ds3、ds6、ds7、ds8	6、7、16、19、22
	Target	cleanStatic、ds4	23
C	Source	cleanStatic、ds2、ds3、ds4、ds7、ds8	6、7、16、19、23
	Target	cleanDynamic、ds6	22

4.3.1 Detection Results in a Concurrent Multipath-Spoofing Scenario

As presented in Table 3 and Table 4, most models, including traditional non-sequential and deep learning approaches, exhibit strong performance in an ideal environment without multipath interference. An exception exists for tree-based models, which show sensitivity to inter-satellite noise variations. Under these conditions, the proposed Transformer model excels, with an accuracy of 99.78%, a detection rate of 99.75%, and a false alarm rate of only 0.19%. These metrics underscore its robust capability to discriminate between authentic and spoofing signals.

However, in the concurrent multipath-spoofing scenario, as shown in Figure 6, model performances diverge significantly. Non-sequential models like SVM, RF, and XGBoost struggle to distinguish between multipath and spoofing features, leading to considerably higher false alarm rates. Even the 1D-CNN, while maintaining a high 99.69% detection rate, exhibits a sharp 46.82% increase in its false alarm rate, causing a wasteful level of false alarms. In stark contrast, the proposed Transformer model leverages its self-attention mechanism to model global temporal features. This method proves highly effective, as the model maintains a 99.75% detection rate with a minimal 0.03% false alarm rate and achieves an F1-score of 99.85%. This superior performance highlights the model's robust interference suppression capability and practical deployment value.

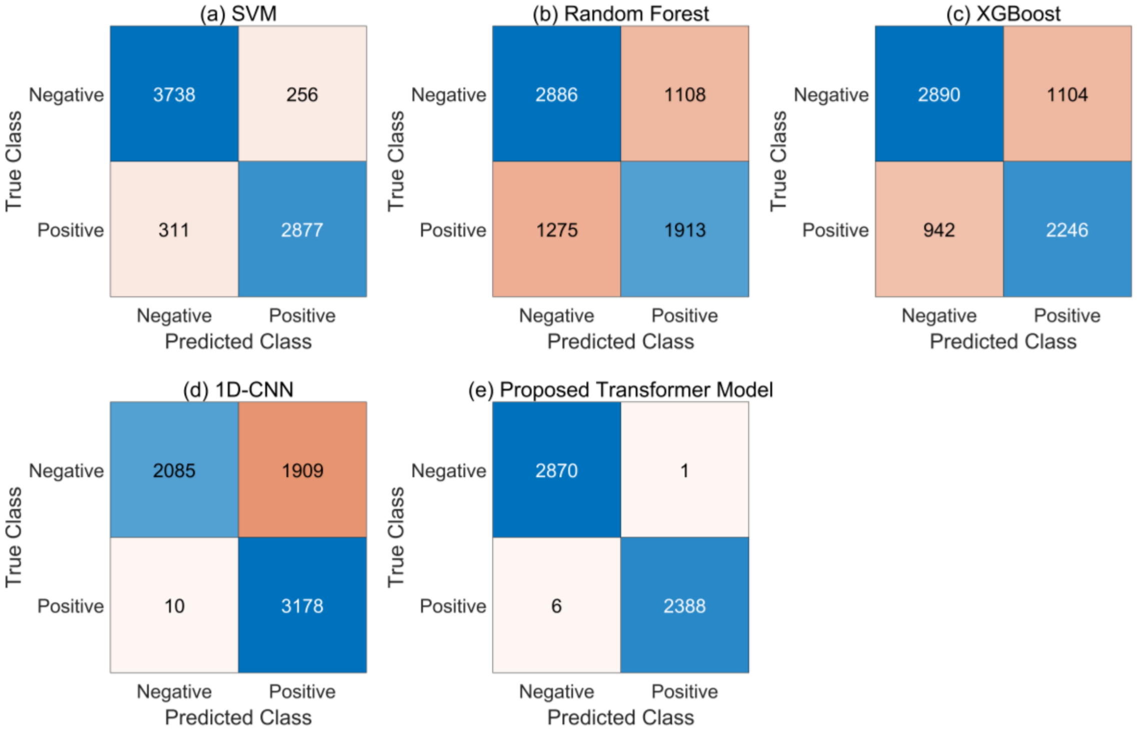


Figure 6 Confusion Matrix in Multipath-Spoofing Concurrency Scenarios

Table 3 Spoofing Detection Results without Multipath

Model	Accuracy	Detection Rate	False Alarm Rate	F1-Score
SVM	0.9299	0.9024	0.0483	0.9195
RF	0.7048	0.6001	0.2118	0.6432
XGBoost	0.7579	0.7045	0.1995	0.7208
1D-CNN	0.9901	0.9899	0.0098	0.9889
Proposed Transformer Model	0.9978	0.9975	0.0019	0.9975

Table 4 Spoofing Detection Results under Concurrent Multipath-Spoofing Conditions

Model	Accuracy	Detection Rate	False Alarm Rate	F1-Score
SVM	0.9211	0.9024	0.0641	0.9103
RF	0.6682	0.6001	0.2774	0.6162
XGBoost	0.7151	0.7045	0.2764	0.6871
1D-CNN	0.7328	0.9969	0.4780	0.7681
Proposed Transformer Model	0.9986	0.9975	0.0003	0.9985

4.3.2 Cross-Scenario Spoofing Detection Results

The confusion matrices in Figure 7 show the results of training the model from scratch on the target domain for each scenario. It is clear that the model rarely produces false alarms; therefore, the subsequent evaluation metrics selected were accuracy, detection rate, and F1-score.

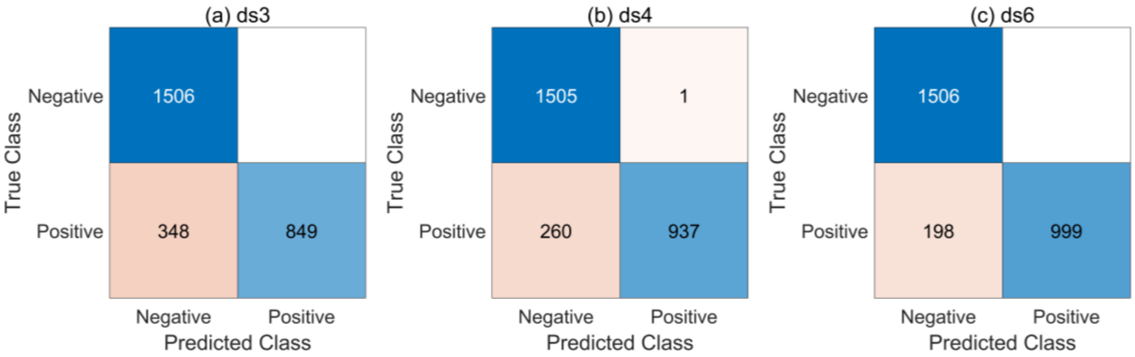


Figure 7 Confusion Matrix for Target-Domain Training from Scratch

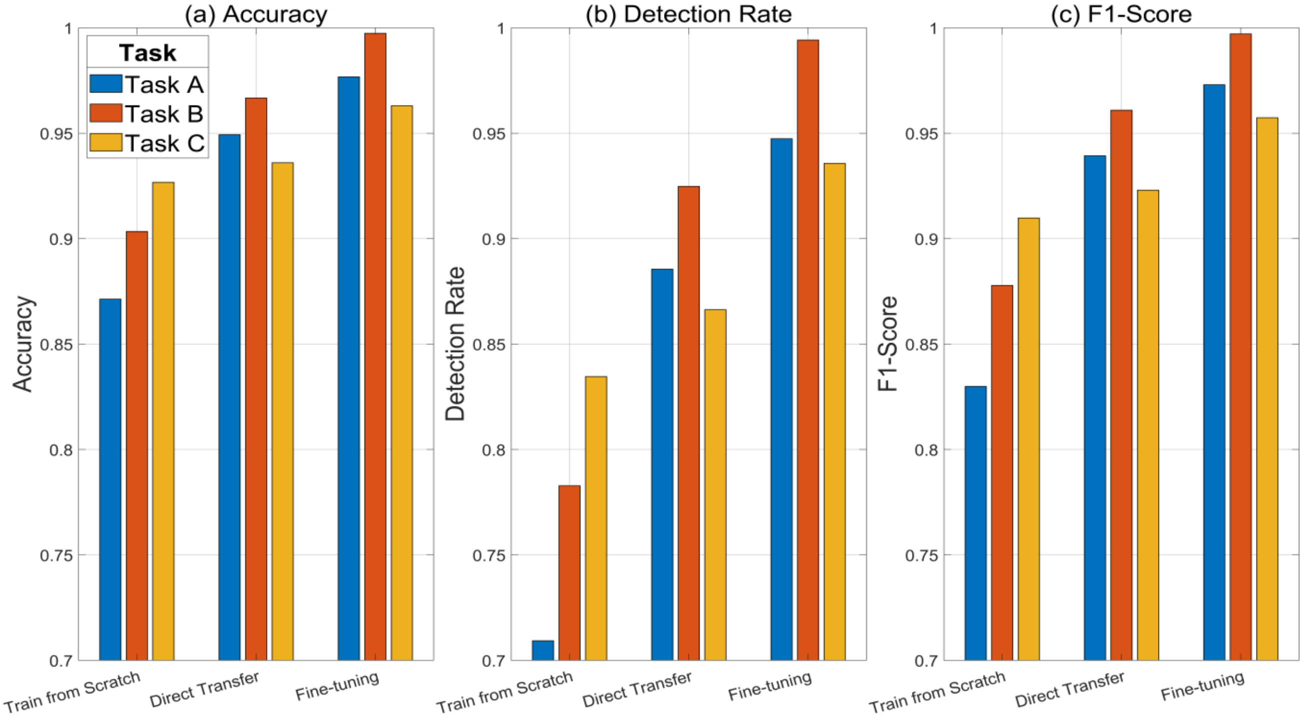


Figure 8 Multi-Task Spoofing Detection Results

Table 5 and Figure 8 compare the detection performance and training efficiency of three strategies across multiple scenarios: training from scratch on the target domain, direct transfer after pre-training, and pre-training followed by fine-tuning of the linear/classification layers.

Taking task A as an example, the benefits of the proposed transfer learning approach are evident. The baseline strategy of training from scratch results in a modest 87.13% accuracy and a 70.93% detection rate, limited by insufficient target domain data. While direct transfer of the pre-trained model improves accuracy to 94.93%, it still falls short of full adaptation due to distributional shifts between domains. The crucial fine-tuning step resolves this, elevating the model's performance to 97.67% accuracy and a 94.74% detection rate. Critically, this fine-tuning strategy also enhances efficiency, slashing the training time from 14.0571 seconds to 3.4262 seconds by optimizing only a fraction of the model's parameters.

Table 5 Multi-Task Spoofing Detection Results

Transfer Task	Training Strategy	Accuracy	Detection Rate	F1-Score	Training Time (s)
A	Train from scratch on target domain	0.8713	0.7093	0.8299	14.0571
	Direct transfer after pre-training	0.9493	0.8855	0.9393	—
	Pre-training + linear/classification layer fine-tuning	0.9767	0.9474	0.9730	3.4262

B	Train from scratch on target domain	0.9034	0.7828	0.8778	14.3062
	Direct transfer after pre-training	0.9667	0.9248	0.9609	—
	Pre-training + linear/classification layer fine-tuning	0.9974	0.9942	0.9971	3.4130
C	Train from scratch on target domain	0.9267	0.8346	0.9098	14.6636
	Direct transfer after pre-training	0.9360	0.8663	0.9230	—
	Pre-training + linear/classification layer fine-tuning	0.9630	0.9357	0.9573	3.9124

5. CONCLUSIONS

To address the challenge of performance degradation in new target scenarios with scarce labeled data, this paper develops a spoofing detection framework using a Transformer architecture combined with a two-stage pre-training and fine-tuning strategy. Experimental results demonstrate the power of this approach: the transfer learning strategy improves detection accuracy by up to 10.54% and reduces training time by 75.63% in cross-scenario tasks. In addition to its high adaptability, the proposed Transformer model also exhibits exceptional robustness in complex environments. It achieves a 99.75% detection rate with a false alarm rate of only 0.03% in concurrent spoofing-multipath scenarios, a reduction of over 98% compared to traditional models. By reducing the dependency on large labeled datasets while ensuring high performance, this approach offers an efficient and robust solution for spoofing detection in complex environments.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (NSFC) with No.62573032. It was also supported by the Open Project Program of Shaanxi Key Laboratory of Integrated and Intelligent Navigation (No. SXKLIIN202403002) and the Open Project Program of State Key Laboratory of CNS/ATM (No.2024A02) (Corresponding author: Chao Sun).

REFERENCES

[1] Wu Z, Zhang Y, Yang Y, et al. Spoofing and Anti-Spoofing Technologies of Global Navigation Satellite System: A Survey[J]. IEEE Access, 2020, 8: 165444-165496.

[2] Meng L, Yang L, Yang W, et al. A Survey of GNSS Spoofing and Anti-Spoofing Technology[J]. Remote Sensing, 2022, 14(19): 4826.

[3] Psiaki M L, Powell S P, O'Hanlon B W. GNSS Spoofing Detection using High-Frequency Antenna Motion and Carrier-Phase Data[C]//Proceedings of the 26th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2013). 2013: 2949-2991.

[4] Daneshmand S, Jafarnia-Jahromi A, Broumandan A, et al. A GNSS structural interference mitigation technique using antenna array processing[C]//2014 IEEE 8th Sensor Array and Multichannel Signal Processing Workshop (SAM). 2014: 109-112.

[5] Phelts R E. Multicorrelator techniques for robust mitigation of threats to GPS signal quality[M]. Stanford University, 2001.

[6] ZHANG L D, ZHANG C, GAO Y J. GNSS spoofing and detection(Ⅱ):GNSS spoofing detection technology based on specially adapted receivers[J]. Journal of Navigation and Positioning, 2021,9(4): 1-10.

[7] Mubarak O M, Dempster A G. Analysis of early late phase in single-and dual-frequency GPS receivers for multipath detection[J]. GPS Solutions, 2010, 14(4): 381-388.

[8] Yang B, Dong C, Gao B, et al. Research on GNSS interference recognition based on ROI of correlation peaks[J]. International Journal of Satellite Communications and Networking, 2022, 40(5): 330-342.

[9] Swaszek P F, Pratz S A, Arocho B N, et al. GNSS Spoof Detection Using Shipboard IMU Measurements[C]//Proceedings of the 27th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2014). 2014: 745-758.

[10] Tanıl Ç, Khanafseh S, Pervan B. Detecting Global Navigation Satellite System Spoofing Using Inertial Sensing of Aircraft Disturbance[J]. Journal of Guidance, Control, and Dynamics, 2017, 40(8): 2006-2016.

[11] Ceccato M, Formaggio F, Laurenti N, et al. Generalized Likelihood Ratio Test for GNSS Spoofing Detection in Devices With IMU[J]. IEEE Transactions on Information Forensics and Security, 2021, 16: 3496-3509.

[12] Chen Z, Li J, Li J, et al. GNSS Multiparameter Spoofing Detection Method Based on Support Vector Machine[J]. IEEE Sensors Journal, 2022, 22(18): 17864-17874.

[13] Aissou G, Benouadah S, El Alami H, et al. Instance-based Supervised Machine Learning Models for Detecting GPS Spoofing Attacks on UAS[C]//2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC). 2022: 0208-0214.

[14] Zhu X, Hua T, Yang F, et al. Global positioning system spoofing detection based on Support Vector Machines[J]. IET Radar, Sonar & Navigation, 2022, 16(2): 224-237.

[15] Sung Y H, Park S J, Kim D Y, et al. GPS Spoofing Detection Method for Small UAVs Using 1D Convolution Neural Network[J]. Sensors, 2022, 22(23): 9412.

[16] Panigrahi S, Nanda A, Swarnkar T. A Survey on Transfer Learning[C]//Mishra D, Buyya R, Mohapatra P, et al. Intelligent and Cloud Computing. Singapore: Springer, 2021: 781-789.

[17] 靳睿敏, 甄卫民, 韩超, 等. ICG IDM 情况介绍及卫星导航干扰检测定位技术发展分析[J]. 全球定位系统, 2024, 49(04): 10-21.

[18] Humphreys T E, Bhatti J A, Shepard D, et al. The Texas spoofing test battery: Toward a standard for evaluating GPS signal authentication techniques[C]//Radionavigation Laboratory Conference Proceedings. 2012.