GNSS Spoofing Detection Based on LSTM-TNN-CVAE Network

Chaowen Tang*, Tian Qin
Xiangtan University, Xiangtan, China[1]
E-mail: 202205570416@smail.xtu.edu.cn

***Abstract*— In the realm of Global Navigation Satellite System (GNSS), detecting spoofing is critical to ensure the reliability and security of the system. At present, most of the detection methods have deficiencies in terms of environmental adaptability, data dependence, and detection stability. This paper presents a satellite navigation signal deception detection method that considers abnormal variations in multiple signal features. Compared with traditional machine learning methods, the proposed method introduces additional mechanisms to enhance detection performance. The experimental results show that our loop detection mechanism and signal compensation mechanism introduced based on the overall network further improve the detection accuracy compared with the single twin neural network.**

Keywords—GNSS, spoofing detection, loop detection mechanism, signal compensation mechanism

I. INTRODUCTION

GNSS is used in multiple professional fields such as military navigation, surveying and mapping technology, and autonomous driving, with the demand for GNSS increasing daily. However, due to the considerable distance between satellite navigation and the ground, the propagation process is affected by environmental factors, making it susceptible to interference. Additionally, the open channel modulation methods and low signal strength make GNSS signals extremely vulnerable to spoofing interference, posing significant challenges to the security of navigation systems and related applications [1][2][3][4]. Therefore, signal spoofing detection is crucial for the stable and reliable operation of navigation systems.

To date, navigation signal spoofing detection methods have undergone several iterations, generally categorized into signal feature-based spoofing detection techniques, integrated navigation-based spoofing detection techniques, and encryption authentication-based spoofing detection techniques. Akos D.M. et al. [5] proposed a GNSS spoofing detection method using signal strength information, achieving preliminary identification of spoofing signals. However, its strong dependence on specific receiver hardware limits its generalization across different devices. Lee D.K. et al. [6] realized spoofing identification based on standard protocol information using NMEA messages. Still, this method relies on the accuracy and integrity of NMEA messages and is vulnerable to message tampering or forgery. Kuusniemi et al. [7] identified spoofing signals by monitoring abnormal drifts in GNSS signals. Yet, the detection accuracy and real-time performance of this method still require further improvement in complex environments. Most of these methods rely on a single signal signature, and some require specific hardware.

Furthermore, many scholars have proposed machine learning-based spoofing detection methods that extract and learn multi-dimensional data features from received signals. M.Sun et al. [8]proposed a multi-parameter spoofing detection method based on support vector machines (SVM), Input to SVM using singular values of wavelet transform coefficients of spoofed and real signals, but its detection performance was not satisfactory. L. Junzhi et al. [9] proposed an improved random forest algorithm, which achieves high detection accuracy but exhibits strong dependence on datasets. It requires sufficient datasets from corresponding spoofing scenarios to train the model before conducting spoofing detection in those scenarios. Additionally, D. Miralles et al. [10] proposed a spoofing detection method combining signal strength, noise power, and carrier-to-noise ratio in their research, aiming to improve detection accuracy through multi-parameter fusion. These machine learning-based methods for deception detection have higher detection accuracy but are highly dependent on the dataset.

Aiming at the deficiencies of existing spoofing detection methods regarding environmental adaptability, lack of real-time mechanism and detection stability, this paper proposes a spoofing detection method based on compensated cycle neural network for satellite navigation signals.

In this study, we consider the effects of power, carrier-to-noise ratio, pseudorange, Doppler frequency and carrier phase on real GNSS signals. We analyze and extract features by multi-feature fusion analysis and feature engineering, then use TNN to fit the similarity between real GNSS signals and spoofed GNSS signals. Specifically, we improve TNN for better performance: we use LSTM to predict signal features from historical data and perform TNN classification based on actual received signals versus current window predicted signals to detect spoofing.

Considering that using spoofed signals for subsequent LSTM prediction reduces the learning effect of normal signal features, a compensation model is introduced. The idea of the compensation model is to consider the received spoofed signals as false satellite signals at the far end and then compensate them as received noise signals. Specifically, we refined this module using CVAE: We improved the classification detection module by cycling the LSTM prediction mechanism and the signal compensation mechanism to realize a new spoofing detection mechanism.

This work was supported by HunanStudents'Platform for innovation and entrepreneurship training program under Grant No. S202410530405.

II. MODELING

Since the GNSS navigation signal will inevitably be affected by the environment during reception, such as fading, multipath effect, and the impact is larger in the dynamic environment. Disregarding dynamic conditions, the signal under the receiver can be expressed as:

$$S(t) = S_a(t) + S_S(t) + n(t) \tag{1}$$

Where $S_a(t)$ denotes real satellite signals, $S_S(t)$ denotes spoofed signals, and $n(t)$ denotes noise.

In dynamic environments, current spoofing devices cannot perfectly replicate all the characteristics of a real GNSS signal, so inevitable differences between key signal parameters still exist. Given this, we can detect spoofing by observing the fluctuations of key parameters of the received signal.

The GNSS spoofing detection algorithm based on the LSTM-TNN-CVAE network that we proposed is shown in Figure 1. It is elaborated as follows:

(1) First, observation data from the TEXBAT [11] dataset is used, which contains five key parameters: power, carrier-to-noise ratio, pseudo-distance, Doppler frequency and carrier phase. Subsequently, data preprocessing is performed to normalize the last four observational data metrics of different satellite signals with the power and average carrier-to-noise ratio of the received signals to ensure consistency and reliability of the analysis.

(2) The segmented TEXBAT-Clean dataset is trained and tested using LSTM to verify the prediction performance of LSTM and output the predicted data.

(3) Train the TNN model and the CVAE model. The TNN model is trained by classifying labels, using the dataset in clean scenarios versus the dataset in spoofing scenarios , so that the TNN is able to classify the judgment of the received signals; And the CVAE model is trained, so that the TNN is able to compensate the received data to be normal after detecting spoofing at a specific point in time.

(4) Assuming that the initially received signal is an authentic signal, the partial time data segments of the received dataset (clean) are input into the LSTM to predict the authentic signal data for the next time window. Subsequently, the freshly received signal is combined with the predicted authentic signal data and input into the TNN classification network to detect the freshly received signal data.

(5) Regarding step (4): When no spoofing signal is detected, the received signal data is fused with the initial authentic signal data and fed into the LSTM for the next round of prediction and detection modeling. Otherwise, if a spoofing signal is detected, the received signal data is input into the CVAE compensation network, which compensates the data to be treated as authentic signal data. This compensated data is then fused with the initial authentic signal data to enter the subsequent processing cycle.

(6) Repeating the execution of steps (4) and (5) forms a complete "prediction-detection-compensation" cyclic neural network framework.

As illustrated in Fig. 1, through the analysis of distinct features between authentic signals and spoofing signals, combined with sliding temporal window-based LSTM prediction, a novel approach for spoofing detection has been successfully developed.

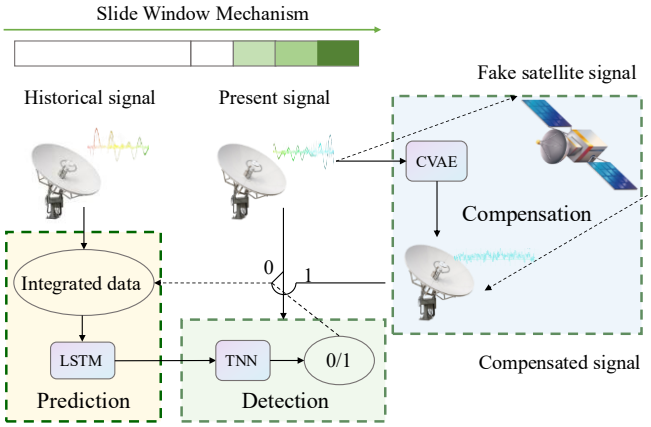

Fig. 1. Spoofing Detection Flowchart

III. METHODS

A. LSTM prediction

LSTM is a special variant of Recurrent Neural Network (RNN) that excels in processing and predicting time series related data. Structurally, it consists of a storage unit, an input gate, an output gate and a forgetting gate.

$$I_t = \sigma\left(X_t W_{xi} + H_{t-1} W_{hi} + b_i\right) \tag{2}$$

$$F_t = \sigma\left(X_t W_{xf} + H_{t-1} W_{hf} + b_f\right) \tag{3}$$

$$O_t = \sigma\left(X_t W_{xo} + H_{t-1} W_{ho} + b_o\right) \tag{4}$$

B. Twin neural network classification

The twin neural network, a specialized neural network architecture, is composed of a weight-sharing base network and a similarity calculation head network [12] . The base network comprises two sub-networks with identical weights and parameters. In this study, these sub-networks employ LSTMs to project input time series into a low-dimensional feature space, yielding the feature vectors of the two input sequences (as defined in Equation 5). Following the extraction of temporal features from both input sequences, the Euclidean distance between these two feature vectors is computed (as presented in Equation 6). The head network integrates a 32-unit fully connected layer (with ReLU activation) and a 1-unit fully connected layer (with Sigmoid activation) to construct a classifier and output the similarity score (as detailed in Equation 7). The loss function employs binary cross-entropy (as illustrated in Equation 8).

$$f(x,\theta) = Dense_p\left(LSTM_j\left(LSTM_j(x)\right)\right) \tag{5}$$

$$d = \sqrt{\sum_{k=1}^{n}(f_{1k} - f_{2k})^2} \tag{6}$$

$$\hat{y} = \sigma(W \cdot [f_1; f_2; d] + b) \tag{7}$$

$$L = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(p_i) + (1-y_i)\log(1-p_i)\right] \tag{8}$$

Here, x represents the input time series, where m and n denote the length of the time series and the feature dimension, respectively. f_{1k} and f_{2k} are the k elements of the n-dimensional feature vectors output by the two base networks, respectively.

W and b are the parameters of the 32-unit fully connected layer in the head network; $\sigma(\bullet)$ denotes the sigmoid activation function, which outputs a probability value−if the probability exceeds the threshold, the pair is classified as the same class; otherwise, it is classified as different classes. y_i represents the label of the sample pair, and p_i is the probability predicted by the model that the two samples are dissimilar.

Based on the TNN model, we set a threshold (0.5) for the output probability to classify the predicted signal data and the actual traversal signal data, thereby deriving the classification result of the actual traversal signal data.

C. CVAE compensation

The Conditional Variational Autoencoder (CVAE) model primarily comprises an encoder and a decoder. The encoder encodes input data x and conditional information c into distribution parameters of the latent space, which are assumed to follow a simple standard authentic distribution to ensure the continuity and completeness of the latent space (Equation 9).

$$z \sim N(0, I) \tag{9}$$

Since the posterior probability is intractable to compute, CVAE introduces variational inference, utilizing a recognition model to approximate the true posterior probability distribution. The KL divergence measures the discrepancy between these two distributions, and the likelihood function is maximized to solve for unknown parameters (Equation 10).

$$q(z \mid x, c) = N(z; \mu(x), \sigma^2(x)I) \tag{10}$$

To achieve effective approximation, minimizing the KL divergence between the two distributions is required. Given the non-negativity of KL divergence, maximizing the likelihood function can be reformulated as maximizing the Evidence Lower Bound (ELBO).

The ELBO consists of a reconstruction term for data fitting and the KL divergence term. Balancing these two components enables the model to simultaneously possess generative capability and generalization performance (Equation 11).

$$ELBO = E_{q(z|x)}[logp(x \mid z, c)] - KL(q(z \mid x, c) \,||\, p(z)) \tag{11}$$

Where μ and σ are output by the encoder neural network.

When the KL divergence is slight, the approximate a posteriori is similar to a priori, and the KL divergence is simplified and calculated as:

$$KL(q(z \mid x) \,||\, p(z)) = \frac{1}{2}\sum\left(\sigma_i^2 + \mu_i^2 - 1 - \log \sigma_i^2\right) \tag{12}$$

Here, d denotes the dimension of the latent variable z, and i represents the dimension of the input samples.

Based on the CVAE model, we utilize the encoder to receive the deceptive signal segments and conditional information, mapping them to the mean and variance vectors in the latent space. The decoder, based on the sampled values from the latent space and conditional information, generates the compensated normal signals.

IV. Numerical Validation

A. Data preprocessing

Considering that signal data from different satellite IDs are collected in the TEXBAT dataset simultaneously, processing satellite IDs might cause the model to learn incorrect dimensions, further reducing accuracy. Therefore, we adopt one-hot encoding to convert different TXID numbers into binary vectors, where each category corresponds to an independent dimension. This approach avoids issues related to numerical order while clearly distinguishing between different categories. Based on one-hot encoding, the feature dimension of the dataset comprises six navigation signal features (carrier-to-noise ratio, pseudorange, Doppler frequency, carrier phase, power, average carrier-to-noise ratio) plus a 14-dimensional TXID one-hot encoding.

For the LSTM prediction network in this study, the static (clean) dataset is used: the first portion of the data, with 80% of the dataset allocated as the training set and the remaining data used as the test set to validate the prediction performance of the LSTM.

For the single Twin neural network classification network, the TEXBAT mixed dataset is employed: 80% of the dataset serves as the training set, and the rest is used as the test set to validate the classification performance of the TNN.

For the Twin neural network classification network and CVAE compensation network in the integrated network, spoofing scenarios (ds3, ds7) and the static (clean) dataset are utilized. Specifically, the datasets correspond to data collected after the start of spoofing time points in the two scenarios (i.e., datasets after 105s and 110s, respectively). Here, 80% of the data is used as the training set, and the remaining data is used as the test set to validate the classification performance of the TNN and compensation performance of the CVAE. For the overall detection process of this study, data from the spoofing scenario (ds2) and static (clean) scenario are used to validate the deception detection performance of the overall model.

B. LSTM model training

The preprocessed navigation signal feature sequence is received, with an input shape of (100, 20). An LSTM layer composed of bidirectional LSTM units is used to capture the time-series features of the navigation signals. It outputs predictions with the same feature dimension as the input, using a linear activation function to predict the subsequent part of the navigation signal. The front segment of normal signals is used as the training set input, and the corresponding true back segment serves as the target output. Mean Squared Error (MSE) is adopted as the loss function, and the Adam optimization algorithm is used for training. A batch size of 32 and a learning rate of 0.01 are set, with ten epochs of iterative training conducted until the model's performance stabilizes on the validation set. Two Dropout layers with rates of 0.2 and 0.3 are utilized to prevent overfitting.

C. TNN model training

We construct two types of sample pairs: positive sample pairs and negative sample pairs. Through two LSTM networks with identical weights, we extract temporal features and compute the Euclidean distance between the outputs of the two networks. The feature vectors and the distance are concatenated and then passed through two fully connected layers with ReLU and Sigmoid activation functions, respectively, to calculate the probability of similarity or dissimilarity. Binary cross-entropy is adopted as the loss function, and the Adam optimization algorithm is used for training. A batch size of 32 and a learning rate of 0.01 are set, with ten training epochs performed until the model's performance stabilizes on the validation set. A Dropout layer with a rate of 0.3 is employed to prevent overfitting.

D. *CVAE model training*

We utilize the TXID numbers from different satellite signals as conditional information and extract spoofed signal samples and corresponding normal signal samples from the TEXBAT dataset. The encoder layer comprises a 64-unit LSTM layer and an 8-unit Dense layer, while the decoder layer includes a 64-unit LSTM layer, a 20-unit Dense layer, and finally a Dense layer with Sigmoid activation. Spoofed signal samples are used as the model's input, and normal signal samples serve as the target output. A combination of reconstruction loss and KL divergence loss is adopted. The reconstruction loss measures the discrepancy between the generated signals and the target normal signals, while the KL divergence loss ensures that the latent space distribution approximates the standard normal distribution. During training, the loss function uses -ELBO, and reparameterization is employed to address backpropagation. The Adam optimization algorithm is used for training, with multiple epochs of iterative training conducted until the model's performance stabilizes on the validation set.

V. RESULT ANALYSIS

To evaluate the classification performance of the algorithm, we use the confusion matrix and ROC curves and F1 scores as the evaluation metrics, and the four cells of the confusion matrix correspond to TP, FP, TN, and FN, respectively.

As shown in Table 2, LSTM-TNN-CVAE has better spoof detection performance compared to other methods (e.g., SQM[13], ANC[14]). In the ds3 scenario, we use a clean static dataset and ds2 for training and ds3 for testing, which alleviates the dependence on the dataset in the new environment to some extent.

On the basis of TNN network, we take the prediction data obtained by using historical real signal prediction as a benchmark and then determine whether it is to spoofing or not, here we use clean static dataset and ds2 as well as ds3 as training and ds7 as well as clean static dataset as testing, as shown in Tab. 3, the LTCN network has a better performance in spoofing detection performance.

In the TNN network, two satellite signal data are deemed to be the same class and labeled as 0, and two satellite signal data are deemed to be different classes and labeled as 1. The related indexes are shown in Fig. 2. The TNN network pair has better classification performance.

While in the LTCN network, the satellite signal data received during the test is deemed as normal and labeled as 0, and is deemed as spoofing and labeled as 1, and the related indexes are shown in Fig. 3.

In experiments using the TEXBAT dataset, the twin neural network achieved 91.48% classification accuracy on the mixed dataset. In the recurrent detection test, the neural network achieved an overall detection accuracy of 94.92% on ds7.

Here are some limitations in our experiments :

In step (4), if the signal initially received a spoofed signal, we must convert it within the compensation model using the normal signal dataset as a reference. However, this signal may contain significant errors and requires subsequent reception of normal signals for correction.

Due to the high complexity of experiments in real-time dynamic environments, this study only tested the dataset. For cost reasons, testing in the real-world dynamic GNSS environments was not conducted.

Tab. 2. Performance comparison of LTCN with other methods

Algorithm	Accuracy	Precision	Recall	F1
LTCN	95.32%	94.76%	99.61%	96.91%
ANC	91.22%	92.34%	99.06%	95.58%
SQM	69.86%	87.77%	75.35%	81.08%

Tab. 3. Performance comparison of Twin Neural Networks and LTCN

Algorithm	Accuracy	Precision	Recall	F1
TNN	91.48%	90.69%	99.52%	94.89%
LTCN	94.92%	94.36%	99.61%	96.91%

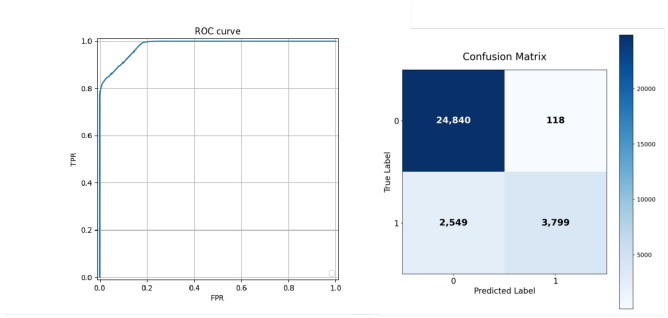

Fig. 2. Confusion Matrix and ROC curve of Twin Neural Networks

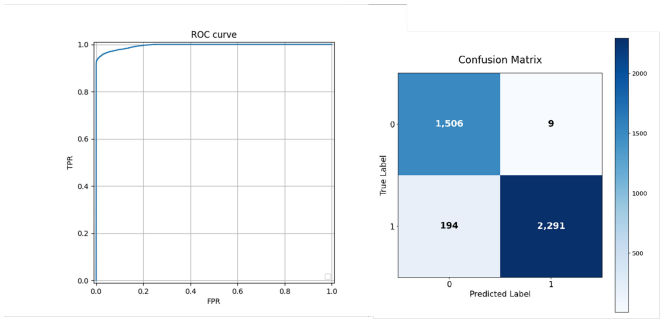

Fig. 3. Confusion Matrix and ROC curve of LTCN

VI. CONCLUSIONS

Given the lack of real-time detection mechanism and insufficient detection progress of existing methods, this study explores a spoofing detection compensation cycle network structure based on GNSS multi-signal features and proposes the LSTM-TNN-CVAE algorithm. Under the condition that the historically acquired signals are not spoofed, LSTM is utilized to develop new reference data, and the predicted and actual received signal features are input into the TNN classification network for spoofing detection, and finally the introduction of the CVAE compensation network is used to advance the spoofing detection process.

After incorporating the loop detection mechanism and signal compensation mechanism, detection accuracy saw a certain degree of improvement. Although the initial training of the relevant neural network takes time, and is slightly complex, the whole neural network proposed in this study can be to some extent adapted to the data trends of GNSS signals in new environments. provides several potentially useful new mechanisms for detecting deception signals.

REFERENCES

[1] Mark Psiaki; Todd Humphreys, "Civilian GNSS Spoofing, Detection, and Recovery," in *Position, Navigation, and Timing Technologies in the 21st Century: Integrated Satellite Navigation, Sensor Systems, and Civil Applications* , IEEE, 2021, pp.655-680.

[2] L. Xiao, C. Xie, M. Min and W. Zhuang, "User-Centric View of Unmanned Aerial Vehicle Transmission Against Smart Attacks," in *IEEE Transactions on Vehicular Technology*, vol. 67, no. 4, pp. 3420-3430, April 2018.

[3] K. Jansen, M. Schäfer, D. Moser, V. Lenders, C. Pöpper and J. Schmitt, "Crowd-GPS-Sec: Leveraging Crowdsourcing to Detect and Localize GPS Spoofing Attacks," *2018 IEEE Symposium on Security and Privacy (SP)*, San Francisco, CA, USA, 2018, pp. 1018-1031.

[4] Wang-Xun Z , Hong-Tao H , Wei-Ping W .Research on GNSS's security-protection[J].Computer Engineering & Science, 2013.

[5] Akos, D.M. Who's Afraid of the Spoofer? GPS/GNSS Spoofing Detection via Automatic Gain Control (AGC). Navigation 2012,59, 281−290.

[6] D. -K. Lee *et al*., "Detection of GNSS Spoofing using NMEA Messages," *2020 European Navigation Conference (ENC)*, Dresden, Germany, 2020, pp. 1-10.

[7] Kuusniemi, Heidi, Blanch, Juan, Chen, Yu-Hsuan, Lo, Sherman, Innac, Anna, Ferrara, Giorgia, Honkala, Salomon, Bhuiyan, M. Zahidul H., Thombre, Sarang, Söderholm, Stefan, Walter, Todd, Phelts, R. Eric, Enge, Per, "Feasibility of Fault Exclusion Related to Advanced RAIM for GNSS Spoofing Detection," *Proceedings of the 30th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2017)*, Portland, Oregon, September 2017, pp. 2359-2370.

[8] M. Sun, Y. Qin, J. Bao, and Y. Li, "GPS spoofing detection based on decision fusion with a K-out-of-N rule," Int. J. Netw. Secur., vol. 19, no. 5, pp. 670–674, Sep. 2017.

[9] L. Junzhi et al., "Performance Testing and Analysis of a New GNSS Spoofing Detection Method in Different Spoofing Scenarios," in IEEE Access, vol. 13, pp. 54779-54793, 2025.

[10] D. Miralles, N. Levigne, and A. Bornot, "An assessment of GPS spoofing detection via radio power and signal quality monitoring for aviation safety operations," IEEE Aerospace and Electronic Systems Magazine, vol. 12, no. 3, pp. 136-146, 2020.

[11] T. E. Humphreys, J. A. Bhatti, and D. P. Shepard, "The Texas spoofing test battery: toward a standard for evaluating GNSS signal authentication techniques," in Proceedings of the ION GNSS Meeting, Nashville, TN, USA, 2012, pp. 3569–3583. https://rnl data.ae.utexas.edu/datastore/texbat/

[12] Pant H, Sharma M, Soman S. Twin neural networks for the classification of large unbalanced datasets[J]. Neurocomputing, 2019, 343: 34-49.

[13] C. Sun, JW. Cheong, and AG. Dempster, "Robust spoofing detection for GNSS instrumentation using Q-Channel signal quality monitoring metric," IEEE Transactions on Instrumentation and Measurement, vol. 70, 2021.

[14] M. Deng, H. Wang, D. Ming and Y. Chen, "GNSS Spoofing Detection Based on Abnormal Receiver Noise and Carrier-to-Noise Ratio Metric," 2022 IEEE 22nd International Conference on Communication Technology (ICCT), Nanjing, China, 2022, pp. 1306-1311.