Titre: Title:	Deep learning-based interference detection, classification, and forecasting algorithm for ESM radar systems
Auteurs: Authors:	Hamda Bouzabia, Georges Kaddoum, & Tri Nhu Do
Date:	2024
Type:	Article de revue / Article
Référence: Citation:	Bouzabia, H., Kaddoum, G., & Do, T. N. (2024). Deep learning-based interference detection, classification, and forecasting algorithm for ESM radar systems. IEEE Access, 12, 148120-148142. https://doi.org/10.1109/access.2024.3475732

Titre de la revue: Journal Title:	IEEE Access (vol. 12)
Maison d'édition: Publisher:	Institute of Electrical and Electronics Engineers
URL officiel: Official URL:	https://doi.org/10.1109/access.2024.3475732
Mention légale: Legal notice:	©2024 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 9 September 2024, accepted 3 October 2024, date of publication 7 October 2024, date of current version 18 October 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3475732

RESEARCH ARTICLE

Deep Learning-Based Interference Detection, Classification, and Forecasting Algorithm for ESM Radar Systems

HAMDA BOUZABIA[1], GEORGES KADDOUM[1,2], (Senior Member, IEEE), AND TRI NHU DO[3], (Member, IEEE)

[1]Department of Electrical Engineering, École de Technologie Supérieure (ÉTS), Université du Québec, Montreal, QC H3C 1K3, Canada
[2]Artificial Intelligence and Cyber Systems Research Center, Lebanese American University, Beirut 03797751, Lebanon
[3]Department of Electrical Engineering, Polytechnique Montreal, Montreal, QC H3T 1J4, Canada

Corresponding author: Hamda Bouzabia (hamda.bouzabia.1@ens.etsmtl.ca)

ABSTRACT In this study, aiming to address the challenges posed by interference from communication systems and jammers, we investigate the application of deep learning (DL) in electronic support measures (ESM) radar systems. Our primary objective is to detect, classify, and forecast interference that can disrupt detection of low probability of intercept (LPI) and low probability of detection (LPD) signals. The proposed algorithm uses a time-frequency distribution (TFD) and received interference strength (RIS) to detect and predict interference. To ensure high precision,we develop a new DL-based outlier detection (OD) technique that is based on the relationship between true positive rate (TPR) and latent space. More specifically, the OD technique applies a new dual-threshold mechanism to the TFD representation for interference detection. We also introduce a DL-enabled classifier designed using the OD architecture to identify the source of interference. Finally, we forecast the RIS by proposing a new DL autoregressive (AR) model through a sliding window designed using the classifier's output. By integrating OD in classifier design and using its output for forecasting, our approach achieves superior accuracy as compared to independent models. Simulation results demonstrate that the proposed algorithm outperforms others, particularly in low signal-to-interference plus noise ratio (SINR) conditions. Specifically, in terms of interference detection, our algorithm achieves 0.9978 TPR, 0.9415 recall, and 0.0004 false positive ratio (FPR). With regard to classification, it records 0.9784 precision and 0.7847 recall. In forecasting, it achieves a 0.2100 mean average error (MAE), thus significantly enhancing ESM radar awareness. The TFD feature also proves to be more accurate than in-phase and quadrature features. These strengths, coupled with an optimal balance of cost and accuracy, make our framework robust and resistant to interference.

INDEX TERMS Radar systems, LPI/LPD, FMCW, interference detection and classification, forecasting, deep learning.

I. INTRODUCTION

The need for low probability of intercept/low probability of detection (LPI/LPD) communications has been recognized by tactical wireless networks (TWNs) and sixth generation (6G) networks. LPI/LPD is considered an essential tactical requirement and is being used by many military radar systems [1]. Electronic support measures (ESM) radar sends LPI/LPD signals and then receives radiation from other sources, targets, communication systems, and other electronic devices. In this framework, short-range ESM radars are particularly interesting because of their robustness against adverse weather conditions. ESM receivers search passively for LPI/LPD radar transmissions within a very large bandwidth, without any a priori knowledge except for a few common characteristics of usually employed radars [2]. The main type of LPI radar is based on frequency modulated-continuous wave (FMCW) technology, which is

The associate editor coordinating the review of this manuscript and approving it for publication was Angelo Trotta.

the result of the use of continuous wave (CW) combined with linear frequency modulation (LFM) [1]. In TWNs, the attempt to degrade and/or deny the target's detection is known as jamming and is a common practice in electronic warfare. The utilization of ESM radars has been extended to civil applications when the carrier frequency of wireless communications started moving toward the radar bands. Existing inter-radar interference has been proven to raise overall clutter levels in ESM radar systems, possibly resulting in the appearance of ghost targets and disrupting the detection of LPI/LPD signals reflected by the targets. According to the literature, the co-existence of wireless communications and radar systems increases the likelihood of interference in the frequency band [3], [4]. The combination of two or more distributed and incoherently operated ESM radars is often called a cooperative radar system. Traditional interference detection techniques depend on prior information like transmitted power and channel state information (CSI), but they are not efficient for ESM radar systems in modern time-sensitive networks and do not achieve high situational awareness (SA). The SA application can be considered as a part of the cognitive radio (CR) and Networking paradigm because it involves the use of intelligent and adaptive techniques to optimize ESM radar performance [5], by identifying the presence of interferences and jammers in real-time, thereby improving SA. Deep learning (DL) addresses this issue by extracting intrinsic features from input data without explicit parameter computation, offering superior performance in signal detection and classification compared to classical methods [6]. To meet the requirements of TWNs and ESM radar systems, an intelligent CR design utilizing DL algorithms for interference detection, classification, and forecasting is necessary.

A. RELATED WORKS

ESM radars' detection of LPI/LPD radiations under existing jammers and interferes has gained increasing attention from academia and industry. A range of solutions have been proposed to address this issue, including LPI/LPD waveform design and interference detection, to enable the accurate detection and analysis of received radiations. In [7], the authors proposed an LPI radar waveform recognition technique (LWRT) based on a single-shot multi-box detector (SSD) and a supplementary classifier to recognize both pulse waves (PW) and continuous wave (CW) LPI radar waveforms. In [8], the author eliminated the interference by replacing the period during which it appeared in the full signal with zeros. By monitoring the interference signal and lowering its amplitude, the advanced weighted-envelope normalization approach successfully suppressed the mutual interference in [9]. In these works, the threshold setting for traditional methods is determined manually, making interference detection and forecasting a time-consuming process. Given recent advances in DL, intelligent techniques that allow radars to learn and adapt to their environment have been proposed in the literature. In this context, DL techniques are known for their capacity to model high-dimensional distributions. In [10], the authors proposed a two-stage autoencoder model for automotive radar interference mitigation based on separate neural networks for detection. The results showed that their proposed approach outperforms traditional solutions. However, in the solution, the interference caused by inband nodes is not considered.

In [11], the authors proposed a solution for signal classification by leveraging convolutional neural networks (CNN). They applied six well-known CNN models to train for ten signal classes, including long term evolution (LTE), radar, wireless fidelity (WiFi), and filter bank multicarrier (FBMC), as well as their interference combinations, such as LTE + Radar, LTE + WiFi, FBMC + Radar, FBMC + WiFi, WiFi + Radar, and Noise. To extract features, they introduced a novel representation called the Quarter-spectrogram (Q-spectrogram), which compresses temporal and frequency information for input to CNN models. The study showed that the highest classification accuracy of 98% was achieved by the ResNet18 model. Although the authors considered interference from inband systems, they did not address the issue of jammers, and the study did not account for LPI/LPD systems which can significantly impact signal classification. In [12], the authors introduced a novel data-driven approach to automatic modulation classification using the long short-term memory (LSTM) model. They utilized time domain amplitude and phase information of the modulation schemes as the features for their proposed model. Their analysis showed that the proposed model achieved an average classification accuracy of almost 90%. However, the authors in [12] only considered features in the time domain that rely on the CSI. Additionally, they did not account for the presence of jammers or inband interferences, which may impact the accuracy of the classification model. In [4], the author proposed a fully connected deep neural network (FCDNN) in communications systems subject to interference from radar systems. Their numerical results showed that the learning-based detector achieves comparable performance in the radar-communication system to the traditional detector. However, they did not consider interference from jammers and interference from inband systems.

Several works have previously addressed the wireless signal prediction problem using AI techniques. In [13], a preliminary investigation towards channel prediction in body area networks (BANs) was conducted. It was demonstrated that an LSTM-based framework outperforms conventional methods, like moving average and adaptive prediction, on BAN measurements. In [14] the authors designed DeepChannel, an encoder-decoder based sequence-to-sequence DL model that is capable of predicting future signal strength variations based on past signal strength data. However, these techniques have not yet fully addressed the complexities posed by heterogeneous sources of interference in ESM radar systems. It is important to note that a preliminary part of this research was published in [15],

which focused on interference detection and classification in the presence of a single jammer. The current work expands on this by detecting, classifying, and forecasting heterogeneous interference involving multiple jammers and inband communications. Furthermore, this study provides a more comprehensive and advanced analysis of interference in ESM radar systems, incorporating dynamics reflective of real-world scenarios. We introduce a novel adaptive forecasting algorithm that markedly improves the prediction of interference, accommodating the changing landscape of threats and environmental conditions. By accurately predicting interference, our approach allows ESM radar systems to proactively adjust scanning strategies and adapt to potential threats, thereby enhancing SA and defensive capabilities.

B. RESEARCH QUESTION, PROPOSED SOLUTION, AND CONTRIBUTIONS

In contrast to previous works, in this paper, we raise three research questions: *(i) how to extract significant features from received radiation, (ii) how to detect interference, and (iii) how to classify and forecast the interference type and value.* To answer these questions, we propose a new hybrid DL framework based on cognitive techniques for interference detection, classification, and forecasting *(IDCF) algorithm*. Our proposed IDCF algorithm consists of three phases, each with a unique DL method. By optimizing each phase and improving the quality of input data for subsequent phases, our system achieves higher accuracy, robustness, and SA, compared to existing approaches. The approach relies on the results of each phase as input for the next, resulting in an integrated system. In Phase I, a residual autoencoder, i.e., CWD-Res-AE combined with kernel density estimation (KDE) in the latent space is applied to the Choi William distribution (CWD) representations to detect interference. In Phase II, interference is classified using the encoder of Phase I with a classifier block, i.e., CWD-Res-CNN. In Phase III, the received interference strength (RIS) is predicted using an autoregressive model with LSTM, RIS-AR-LSTM, designed using the output of CWD-Res-CNN on past RIS levels. The IDCF algorithm, which focuses on scalability and efficiency, is built for real-world radar systems. It effectively handles complex interference, such as in air traffic control and military radar, enhancing detection and response. Its modular design allows for an easy integration into existing systems, offering a cost-effective way to improve SA and operational efficiency. The key contributions of the paper are summarised as follows:

- The front-end processing design incorporates the direct spread spectrum technique (DSSS) into the LFM waveform to design LPI/LPD radiation. Our approach uses a window function to reduce cross-term interference and extracts time-frequency distribution (TFD) representations of a received signal, known as the CWD, enabling no-reference interference detection and classification.
- To determine the most accurate and optimal architecture for the first phase, we studied the effect of the CWD-Res-AE architecture on the latent space dimension and the effect of the latent space dimension on true positive ratio (TPR). Using curve fitting and least square regression (LSR) methods, we derived a quadratic function between the dimension of latent space and the TPR. This allowed us to design an optimal architecture that enables more informed decision-making.
- We propose a novel dual-threshold anomaly detection (AD) system to enhance interference detection performance at low signal-to-interference plus noise ratio (SINR). To this end, we first compute the reconstruction threshold error after the CWD-Res-AE. However, relying solely on this threshold is insufficient, as some anomalies may have errors similar to normal samples. Therefore, we also perform KDE to evaluate the probability density function of samples in the latent space and determine a KDE threshold. By using two thresholds, we can control the system's sensitivity to interference, thereby significantly improving accuracy and reliability. This dual-threshold approach ensures robust and precise detection in real-world applications that require maintaining high performance under varying signal conditions, such as military radar systems and tactical networks.
- After detecting the presence of interference, the ESM radar applies the CWD-Res-CNN to the CWD feature to classify the source of interference. Our proposed solution uses the optimally designed architecture of Phase I to design the CWD-Res-CNN and the result of Phase I to update the CWD dataset. The updated CWD dataset, which contains samples classified as interference in Phase I, is used to train and test the CWD-Res-CNN. Our simulation results show that our proposed two-phase interference classification system outperforms a standalone system, achieving higher accuracy in low and high SINR scenarios.
- We design a forecasting model, RIS-AR-LSTM, that combines an autoregressive (AR) model with an LSTM model and a sliding window. By capturing both linear and non-linear relationships using AR and LSTM respectively, our approach offers a unique and effective solution for forecasting interference strength that has not been explored before. Additionally, by classifying the source of interference, we can select an appropriate window length and improve the quality of input data, leading to better prediction accuracy and improved system performance. This approach increases reliability and SA in ESM radar systems, thus enabling proactive interference prevention and mitigation.
- In practical simulation and implementation, our proposed method demonstrates high classification accuracy using existing datasets. We also create a synthesized dataset through rigorous scattering parameter network analysis. This dataset incorporates practical settings,

including waveforms and measurements from diverse locations and interference powers. We study sensitivity of the IDCF algorithm against unseen data and data generated with different setting parameters, finding that the IDCF has low sensitivity and high robustness. A comparative analysis is conducted, demonstrating the superiority of TFD over the in-phase/quadrature (I/Q) features for training DL networks. Furthermore, we show that the proposed algorithm outperforms other relevant DL algorithms for interference detection, such as those in [4], [10], [11], [12], [13], [14], and [16], in terms of TPR, false positive rate (FPR), mean absolute error (MAE), and root mean square error (RMSE), especially in the low SINR regime. These characteristics, with a great trade-off between computational cost and accuracy, make our proposed IDCF algorithm robust and resistant to jammers and interference.

II. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a co-existence scenario where both cooperative ESM radars and communication systems are deployed in a neighborhood. The network is composed of N_{R} ESM radars, $\mathcal{R} = \{\mathrm{R}_k, k = 1, \ldots, N_{\mathrm{R}}\}$, N_{S} legitimate inband transmitters, $\mathcal{S} = \{\mathrm{S}_m, m = 1, \ldots, N_{\mathrm{S}}\}$, and N_{J} jammers, $\mathcal{J} = \{\mathrm{J}_n, n = 1, \ldots, N_{\mathrm{J}}\}$, as well as one target, T. In the network, the cartesian coordinates of R_k, S_m, T, and J_n are denoted by $(x_{\mathrm{R}_k}, y_{\mathrm{R}_k})$, $(x_{\mathrm{S}_m}, y_{\mathrm{S}_m})$, $(x_{\mathrm{T}}, y_{\mathrm{T}})$, and $(x_{\mathrm{J}_n}, y_{\mathrm{J}_n})$, respectively. Thus, the Euclidean distance from node A to the R_k radar is given by $d_{\mathrm{A}\to\mathrm{R_k}} = \sqrt{(x_{\mathrm{A}} - x_{\mathrm{R}_k})^2 + (y_{\mathrm{A}} - y_{\mathrm{R}_k})^2}$, where $\mathrm{A} \in \{\mathrm{S}_m, \mathrm{J}_n, \mathrm{T}\}$. The echoes from T and signals from the communication nodes are received by the k-th ($k = 1, \ldots, N$) ESM radar. The inband S_m and the jammer J_n generate signals that interfere with the target detection. In the present study, we assume that, under the given scenario, the received signals at the radar can be classified into the following four distinct categories: received echoes from the target (C_0), received echoes from the target alongside multiple inband nodes (C_1), received echoes from the target in the presence of multiple jammers (C_2), and a combination of received echoes from the target, jammers, and inband nodes (C_3).

A. NODE DISTRIBUTION

In the considered network, to provide a more realistic model, we let nodes $A \in \{\mathrm{S}_m, \mathrm{J}_n, \mathrm{T}\}$ be stochastically distributed across the network area through the homogeneous poisson point process (HPPP). In an HPPP, node points are uniformly and independently distributed within the target area $\mathcal{A}$ [m^2]. An HPPP Φ_A with density per meter square λ_A is given by $\Phi_A = \{(\mathbf{p}_A, \mathcal{A}) : \mathbf{p}_A \in \mathbb{R}^3, \mathcal{A} \subseteq \mathbb{R}^2\}$, where $\mathbf{p}_A$ is the position of node A and $\mathcal{A}$ is its serving area. Furthermore, the random number of node points, $X(\mathcal{A})$, follows a Poisson distribution with $\mu(\mathcal{A}) = \mathcal{A}\lambda$, which is $X(\mathcal{A}) \sim \text{Poisson}(\mathcal{A}\lambda)$, where $\mu(\mathcal{A})$ is the average size of $\mathcal{A}$. However, the distribution of node $B \in \{\mathrm{R}\}$ is characterized by random process $\phi_B = \{(\mathbf{p}_B, \mathcal{A}) : \mathbf{p}_B \in \mathbb{R}^3, \mathcal{A} \subseteq \mathbb{R}^2\}$ within the same serving area $\mathcal{A}$, where $\mathbf{p}_B$ denotes the position of node B.

B. LFM WAVEFORM FOR FMCW LPI/LPD RADARS

An ESM radar transmits an *LFM chirp signal with pseudo-noise (PN) code* to reduce the transmitted power of the FMCW waveforms [17]. The idea of LFM is to sweep the frequency band linearly during the phase pulse duration T. At the k-th radar, the transmitted LFM lowpass chirp signal can be modeled as

$$x_{k,\mathrm{LP}}^{\mathrm{LFM}}(t) = A(t)\exp(j\phi(t)), \quad -T/2 \le t \le T/2, \tag{1}$$

where $A(t)$ is the normalized envelope of the signal, $\phi(t) = 2\pi\int_t f(t)dt = 2\pi\left(\frac{\Gamma}{2}t^2\right) + \phi_0$, is the instantaneous phase of the radio frequency (RF) waveform with an initial phase ϕ_0, $f(t) = \Gamma \times t$ is the instantaneous frequency, $\Gamma = \pm B/T$ is the chirp rate (the frequency sweep rate), where B is the pulse bandwidth [1], [18]. Γ is positive for an up-sweep, where the magnitude of the carrier frequency increases over time, and negative for a down-sweep, where the magnitude decreases over time. Plugging $\phi(t)$ in (1), the transmit lowpass chirp signal is expressed as

$$x_{k,\mathrm{LP}}^{\mathrm{LFM}}(t) = \mathrm{rect}\left(\frac{t}{T}\right)A(t)\exp\left(j2\pi\left(\frac{\Gamma}{2}t^2\right) + \phi_0\right), \tag{2}$$

where $\mathrm{rect}(t)$ is the rectangle function over $[-\frac{1}{2}, \frac{1}{2}]$. The complex modulated LFM chirp waveform can be expressed as

$$x_{k,\mathrm{c}}^{\mathrm{LFM}}(t) = A(t)\exp\left(j2\pi\left(f_{\mathrm{c}}t + \frac{\Gamma}{2}t^2\right) + \phi_0\right), \tag{3}$$

where f_{c} is the carrier frequency. Using Euler's formula, we have $x_{k,\mathrm{BP}}^{\mathrm{LFM}}(t) = A(t)\cos\left(2\pi\left(f_{\mathrm{c}}t + \frac{\Gamma}{2}t^2\right) + \phi_0\right)$. To achieve LPI/LPD characteristics in the transmitted waveforms, DSSS is used to spread the narrow band signal over a wide bandwidth. The idea is to combine the PN code with the LFM chirp signal, as explained above [19]. Specifically, the PN code signal, $c(t)$, consisting of P chips, is expressed as $c(t) = \sum_{l=0}^{P-1} c_l \mathrm{rect}\,(t - lT_{\mathrm{c}})$. In this manner, a processing gain P of the spread spectrum is exploited to provide a significant performance gain to the chirp modulated signal. The transmitted LFM chirp signal with PN code can be expressed as

$$x_{k,\mathrm{BP}}^{\mathrm{PN,LFM}}(t) = \sum_{l=0}^{P-1} c_l A(t)\,\mathrm{rect}\,(t - lT_{\mathrm{c}}) \times \cos\left(2\pi\left(f_{\mathrm{c}}t + \frac{\Gamma}{2}t^2\right) + \phi_0\right). \tag{4}$$

C. SIGNAL AND CHANNEL MODELING

1) CHANNEL MODELING BETWEEN INTERFERENCE SOURCE AND RADAR

In channel modeling, the stochastic nature of the channels and noise are accounted as a random process. Moreover, at a

specific time instant t, the channel coefficient and noise at the R_k radar introduce additional randomness to the system, which can be viewed as random variables. An appropriate and practical assumption is to use the Nakagami propagation model as the fading channel model [20]. Let $h_{I\to R_k}$, where $I \in \{S_m, J_n\}$, be the channel fading coefficient between the ESM radar R_k and the source of interference I, $h_{I\to R_k} = l_{I\to R_k} g_{I\to R_k} \exp(j\theta_{I\to R_k})$, where $l_{I\to R_k}$ is the large-scale shadowing is modeled as an inverse-Gaussian (IG) random variable (RV), $\theta_{I\to R_k}$ is the channel phase, and $g_{I\to R_k}$ is the small-scale fading, an RV following a Nakagami distribution with the probability density function (PDF), parameterized by the shape parameter $m_{I\to R_k}$ and the spread parameter $\Omega_{I\to R_k} = \mathbb{E}\left[g^2_{I\to R_k}\right]$. The PDF of $g_{I\to R_k}$ is given by

$$f_{g_{I\to R_k}}(x; m_{I\to R_k}, \Omega) = \frac{2m_{I\to R_k}{}^{m_{I\to R_k}}}{\Gamma(m_{I\to R_k})\Omega^{m_{I\to R_k}}} x^{2m_{I\to R_k}-1} \times \exp\left(-\frac{m_{I\to R_k}}{\Omega_{I\to R_k}} x^2\right), \tag{5}$$

where $\Gamma(\cdot)$ is the gamma function. The IG PDF of $l_{I\to R_k}$ is given by

$$f_{l_{I\to R_k}}(x; \alpha_{I\to R_k}, \mu_{I\to R_k}) = \frac{\mu_k^{\alpha_{I\to R_k}}}{\Gamma(\alpha_{I\to R_k})} x^{(-\alpha_{I\to R_k}-1)} e^{\frac{-\mu_{I\to R_k}}{x}}, x > 0, \tag{6}$$

where $\alpha_{I\to R_k} > 1$ and $\mu_{I\to R_k}$ are the shape and the scale parameters of the distribution, respectively [21].

2) PATH-LOSS OF THE k-TH ECHOED LINK

The transmitted waveform of the radar $R_{k'}$ is reflected from a target, and the echo signal is received at the radar R_k. The directional power density is $S_{\texttt{direct}} = S_{R_{k'}} G_{t,R_{k'}} = \frac{P^{\texttt{tx}}_{R_{k'}}}{4\pi d^2_{R_{k'}}} G_{t,R_{k'}}$, where $G_{t,R_{k'}}$ is the transmitted gain density, $P^{\texttt{tx}}_{R'_k}$ and $d_{R'_k}$ denote the transmitted power and the range from radar R'_k to the target [22]. Let us denote by κ the radar cross-section, the reflected power $P^{\texttt{rfl}}_{R_k}$ from the target to the radar R_k is given by, $P^{\texttt{rfl}}_{R_k} = \frac{P^{\texttt{tx}}_{R_{k'}}}{4\pi d^2_{R_{k'}}} G_{t,R_{k'}} \kappa$. Because echoes are subjected to the same conditions as the transmitted power, the power density at the radar R_k becomes $S_{R_k} = \frac{P^{\texttt{rfl}}_{R_k}}{4\pi d^2_{R_k}}$. At radar R_k, the received power is given by $P^{\texttt{rx}}_{R_{k'}\to T\to R_k} = S_{R_k} A_W$, where A_W is the effective antenna aperture, i.e., $A_W = \frac{G_{R_k}\lambda^2}{4\pi}$ and λ is the wavelength associated with the carrier frequency f_c. Thus, the power received is given by [22]

$$P^{\texttt{rx}}_{R_{k'}\to T\to R_k} = \frac{P^{\texttt{rfl}}_{R_k}}{4\pi d^2_{R_k}} A K_a. \tag{7}$$

Thus, the deterministic path-loss model of the radar R'_k to target and the target to radar R_k link can be expressed as

$$\text{PL}_{R_{k'}\to T\to R_k} = \left(\frac{c^2 G_{r,R_k} G_{t,R_{k'}} \kappa}{(4\pi)^3 f^2 d^2_{R_{k'}} d^2_{R_k}}\right)^{-1}. \tag{8}$$

3) PATH-LOSS BETWEEN THE INTERFERENCE SOURCE AND THE r_k RADAR

The path loss between R_k and the interferer I is given by $\text{PL}_{I\to R_k} = \left(\frac{G_I G_{R_k} \lambda^2}{(4\pi)^2 (d_{I\to R_k})^2}\right)^{-1}$, with $P^{\texttt{tx}}_I$ and G_I being the transmit power and the antenna gain of the interference source, respectively, G_{R_k} the antenna receiver gain of the radar R_k, and $d_{I\to R_k}$ the range between the interference source and R_k radar. In case C_0, the received waveform, which is reflected from a target at a distance $d_{k'}$ with a velocity $v_{k'}$, can be expressed as a delayed copy of the transmitted LFM chirp signal $r^{C_0}_{k,\text{BP}}(t) = \sum_{K'=0}^{N_R-1} \sqrt{2P^{\texttt{rx}}_{R_{k'}\to T\to R_k}} x^{\text{PN,LFM}}_{k',\text{BP}}(t - \tau_{k'}) + w_{R_k}(t)$ [1], where the round-trip time delay is $\tau_{k'} = \frac{2(d_{k'}+v_T t)}{c} = \tau_0 + \frac{2v_T t}{c}$, v_T is the target velocity, $w_{R_k}(t)$ is the complex bandpass white Gaussian noise random process over the time domain, at a specific time instance t, each realization of the random process, $w_{R_k} \sim \mathcal{N}(0, \sigma^2_{R_k})$ is an additive white gaussian noise (AWGN) with zero mean and variance $\sigma^2_{R_k}$, and $P^{\texttt{rx}}_{R_{k'}\to T\to R_k}$ is the received power from radar k' to radar k [1]. The received and transmitted signals are mixed resulting in the generation of an intermediate-frequency signal given by (9), as shown at the bottom of the next page, where $w^{\texttt{mix}}_{R_k,\text{BP}}(t) = x^{\text{PN,LFM}}_{k,\text{BP}} \times w_{R_k}(t)$ is the noise after de-chirping the received signal, and $\phi'(t) = 2\pi\left(f_c t + \frac{\Gamma}{2} t^2\right) + \phi_0$. We have

$$\cos\left(\phi'(t)\right) \times \cos\left(\phi'(t - \tau_{k'})\right) = \frac{1}{2} \underbrace{\cos(\phi'(t) + \phi'(t - \tau_{k'}))}_{\text{high-frequency component}} + \underbrace{\cos\left(\phi'(t) - \phi'(t - \tau_{k'})\right)}_{\text{low-frequency component}}, \tag{10}$$

and since it is typically only taken into account in the phase and is incredibly small in comparison to the sweep duration, the term $\tau_{k'}$ is ignored in the *rect* function [1]. Thus, the equation (9) can be rewritten as equation (11), as shown at the bottom of the next page. Then the complex bandpass signal is given by $r^{C_0,\texttt{mix}}_{k,c}(t) = r^{C_0,\texttt{mix}}_{k,\text{BP}}(t) + j\widehat{r}^{C_0,\texttt{mix}}_{k,\text{BP}}(t)$ [23], where $\widehat{r}^{C_0,\texttt{mix}}_{k,\text{BP}}(t)$ denotes the Hilbert transform of $r^{C_0,\texttt{mix}}_{k,\text{BP}}(t)$, as a result, the signal is obtained as

$$r^{C_0,\texttt{mix}}_{k,c}(t) = \sum_{K'=0}^{N_R-1} \sqrt{P^{\texttt{rx}}_{R_{k'}\to T\to R_k}} \sum_{i=0}^{M-1} b_i \sum_{l=0}^{P-1} c_l A \times \text{rect}\left(t - iT - lT_c\right) \times \exp\Big(2\pi j\big(\Gamma\tau_0 + 2f_c\Gamma\tau_{k'} + \Gamma\tau_{k'}t - \frac{\Gamma}{2}\tau^2_{k'}\big)\Big) + w^{\texttt{mix}}_{R_k,c}(t), \tag{12}$$

where $w^{\texttt{mix}}_{R_k,c}(t) = w^{\texttt{mix}}_{R_k,\text{BP}}(t) + j\widehat{w}^{\texttt{mix}}_{R_k,\text{BP}}(t)$. Replacing $\tau_{k'}$ in $r^{C_0,\texttt{mix}}_{k,c}(t)$, we have

$$r^{C_0,\texttt{mix}}_{k,c}(t) = \sum_{K'=0}^{N_R-1} \sqrt{P^{\texttt{rx}}_{R_{k'}\to T\to R_k}} \sum_{i=0}^{M-1} b_i \sum_{l=0}^{P-1} c_l A$$

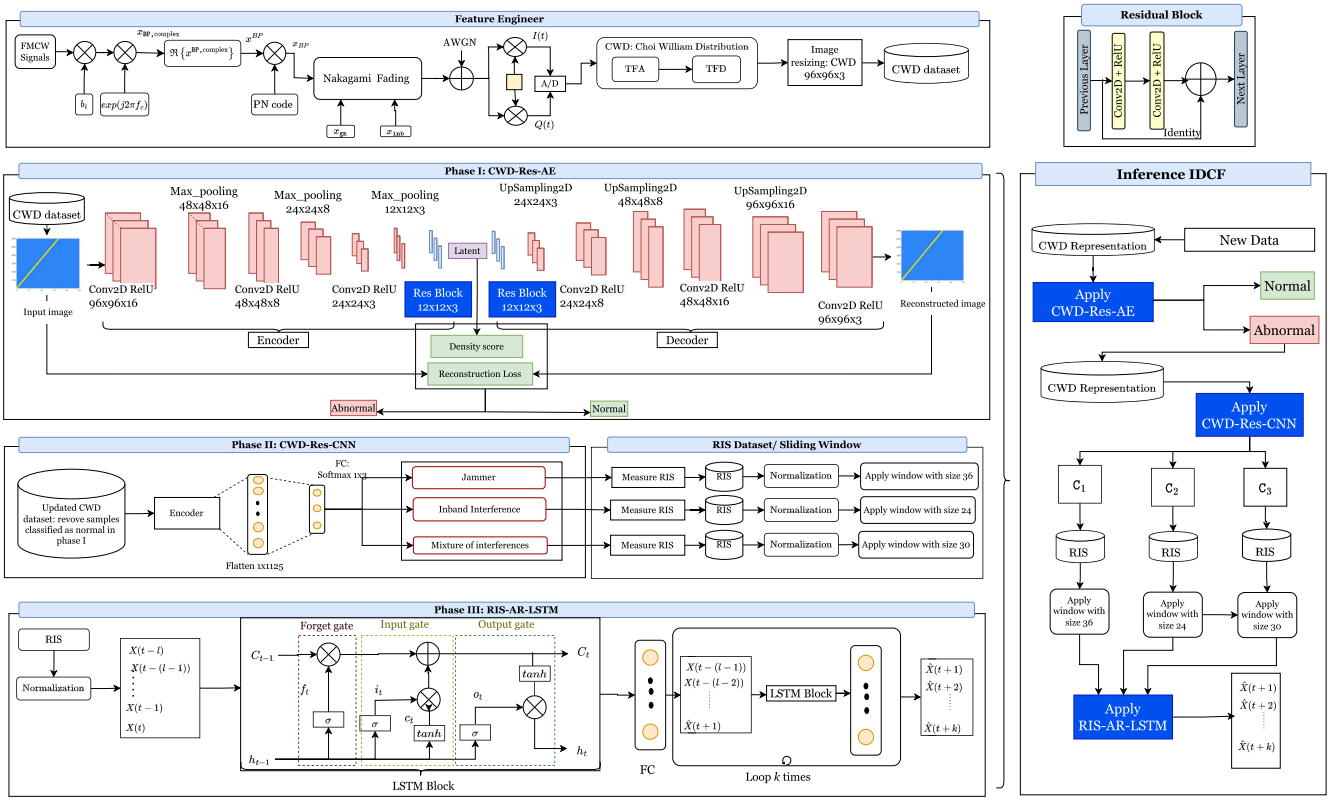


FIGURE 1. An illustration of the architecture of the proposed algorithm using CWD-Res-AE, CWD-Res-CNN, and RIS-AR-LSTN.

$$\times \operatorname{rect}\left(t - iT - lT_{\mathrm{C}}\right) \exp\left(2\pi j\left(\left(\Gamma\tau_0 + \frac{2\nu_{\mathrm{T}} f_{\mathrm{C}}}{c} - \frac{\Gamma\nu_{\mathrm{T}}\tau_0}{c}\right)t - \frac{\Gamma\tau_0^2}{2} + \frac{2\Gamma\left(\nu_{\mathrm{T}} - \frac{\nu_{\mathrm{T}}^2}{c}\right)t^2}{c}\right)\right) + w_{\mathrm{R}_k,\mathrm{C}}^{\mathtt{mix}}(t). \tag{13}$$

The main contribution is the Doppler shift induced in the beat frequency $f_b \approx \Gamma\tau_0 + \frac{2\nu_{\mathrm{T}} f_{\mathrm{C}}}{c} = \alpha\tau_0 + f_D$, where f_D represents the Doppler frequency component. The power of the desired mixed signal at time t can be expressed as $P_{\mathrm{R}} = \sum_{k'=0}^{N_{\mathrm{R}}-1} \sqrt{P_{\mathrm{R}_{k'}\to\mathrm{T}\to\mathrm{R}_k}^{\mathtt{rx}}}$. Mathematically, the synthesized received signal at the radar R_k can be expressed as (14), shown at the bottom of the next page, where $P_{\mathrm{J}_n}^{\mathtt{tx}}$ is the transmitted power from the jammer J_n to the radar R_k, $\mathrm{PL}_{\mathrm{J}_n\to\mathrm{R}_k}$ and $x_{\mathrm{J}_n,\mathrm{BP}}(t)$ are the path loss and transmitted signal between the jammer J_n and R_k radar, respectively. $P_{\mathrm{S}_m}^{\mathtt{tx}}$ is the transmitted power from the legitimate inband interferer to the radar R_k, $\mathrm{PL}_{\mathrm{S}_m\to\mathrm{R}_k}$ and $x_{\mathrm{S}_m,\mathrm{BP}}(t)$ are the path loss and transmitted signal between the inband interferer S_m and radar R_k, respectively. The jammer radiates random jamming power, i.e., the jamming power is uniformly distributed in $[P_{\mathrm{J}}^{\min}, P_{\mathrm{J}}^{\max}]$, where $P_{\mathrm{J}}^{\min}$, $P_{\mathrm{J}}^{\max}$ are minimum and maximum jamming powers, respectively, in other words, $P_{\mathrm{J}_n}^{\mathtt{tx}} \sim \mathcal{U}\{P_{\mathrm{J}}^{\min}, P_{\mathrm{J}}^{\max}\}$. From (14) the SINR is given by

$$\mathrm{SINR} = \frac{\mathbb{I}(\mathrm{C}_0)\sum_{K'=0}^{N_{\mathrm{R}}-1} P_{\mathrm{R}_{k'}\to\mathrm{T}\to\mathrm{R}_k}^{\mathtt{rx}}}{X_t^{\mathrm{RIS}}}, \tag{15}$$

$$r_{k,\mathrm{BP}}^{\mathrm{C}_0,\mathtt{mix}}(t) = x_{k,\mathrm{BP}}^{\mathrm{PN,LFM}}(t) \times r_{k,\mathrm{BP}}^{\mathrm{C}_0}(t) = \sum_{K'=0}^{N_{\mathrm{R}}-1} \sqrt{2P_{\mathrm{R}_{k'}\to\mathrm{T}\to\mathrm{R}_k}^{\mathtt{rx}}} \left(\sum_{i=0}^{M-1} b_i \sum_{l=0}^{P-1} c_l A \operatorname{rect}\left(t - iT - lT_c\right)\right) \times \left(\sum_{i=0}^{M-1} b_i \sum_{l=0}^{P-1} c_l A \operatorname{rect}\left(t - iT - lT_{\mathrm{C}} - \tau_{k'}\right)\right) \cos\left(\phi'(t)\right) \cos\left(\phi'(t-\tau_{k'})\right) + w_{\mathrm{R}_k,\mathrm{BP}}^{\mathtt{mix}}(t), \tag{9}$$

$$r_{k,\mathrm{BP}}^{\mathrm{C}_0,\mathtt{mix}}(t) = \sum_{K'=0}^{N_{\mathrm{R}}-1} \sqrt{\frac{P_{\mathrm{R}_{k'}\to\mathrm{T}\to\mathrm{R}_k}^{\mathtt{rx}}}{2}} \times \left(\sum_{i=0}^{M-1} b_i \sum_{l=0}^{P-1} c_l A \operatorname{rect}\left(t - iT - lT_{\mathrm{C}}\right)\right) \times \cos\left(2\pi\left(f_{\mathrm{C}}\Gamma\tau_{k'} + \Gamma\tau_{k'}t - \frac{\Gamma}{2}\tau_{k'}^2\right)\right) + w_{\mathrm{R}_k,\mathrm{BP}}^{\mathtt{mix}}(t). \tag{11}$$

where $X_t^{\mathrm{RIS}} = \sigma^2_{\mathrm{R}_k} + \mathbb{I}(\mathtt{C}_1)\sum_{n=1}^{N_\mathrm{J}} P^{\mathtt{tx}}_{\mathrm{J}_n}\mathrm{PL}_{\mathrm{J}_n\to\mathrm{R}_k}|h_{\mathrm{J}_n\to\mathrm{R}_k}|^2(t) + \mathbb{I}(\mathtt{C}_2)\sum_{m=1}^{N_\mathrm{I}} P^{\mathtt{tx}}_{\mathrm{S}_m}\mathrm{PL}_{\mathrm{S}_m\to\mathrm{R}_k}|h_{\mathrm{S}_m\to\mathrm{R}_k}|^2(t)$, is the interference power at time t.

D. PROBLEM FORMULATION

In this study, our main objective is to detect, classify, and forecast interference to enhance the performance and reliability of radar systems and take proactive measures against upcoming interference. We reformulate the considered problem as three sub-problems. Consider a set $\mathcal{D}$ containing $|\mathcal{D}|$ different samples and $\psi^{\mathtt{phs}}$, a set of different class labels with $\mathtt{phs} \in \mathtt{phsI}, \mathtt{phsII}$. Here, $\mathtt{phsI}$ is for Phase I, i.e., interference detection, and $\mathtt{phsII}$ is for Phase II, i.e., interference classification. For instance, $|\psi^{\mathtt{phsII}}| = 3$ with $\psi^{\mathtt{phsII}} = \mathtt{C}_1, \mathtt{C}_2, \mathtt{C}_3$, and $\psi^{\mathtt{phsI}} = \mathrm{H}_1, \mathrm{H}_0$, where H_1 is the scenario with interference, while H_0 indicates no interference.

1) INTERFERENCE AND/OR JAMMER DETECTION PROBLEM

In this phase, our first objective is to detect devices that generate interference and disrupt the detection of the target by the radar. Radar signals are sensitive to interference, where the presence of interference strongly affects the CWD representation and generates unforeseen representations (abnormalities). An abnormality is defined as a defect in the TFD representation. These unforeseen TFD representations have no pattern and represent abnormalities compared to TFD representations corresponding to the normal case ($\mathtt{C}_0$). Thus, the interference detection problem can be formulated as two hypotheses, H_0 (normal), and an alternative hypothesis, H_1 (abnormal)

$$\mathrm{H}_0: \quad r^{\mathtt{mix}}_{k,\mathtt{c}}(t) = r^{\mathtt{C}_0,\mathtt{mix}}_{k,\mathtt{c}}(t) \tag{16}$$

$$\begin{aligned}\mathrm{H}_1: \quad r^{\mathtt{mix}}_{k,\mathtt{c}}(t) = r^{\mathtt{C}_0,\mathtt{mix}}_{k,\mathtt{c}}(t) &+ \sum_{m=1}^{N_\mathrm{S}} \sqrt{P^{\mathtt{tx}}_{\mathrm{S}_m}}\sqrt{\mathrm{PL}_{\mathrm{S}_m\to\mathrm{R}_k}}h_{\mathrm{S}_m\to\mathrm{R}_k} \\ & x_{\mathrm{S}_m,\mathrm{BP}}(t)x^{\mathtt{PN,LFM}}_{k,\mathrm{BP}}(t) + \sum_{n=1}^{N_\mathrm{J}} \sqrt{P^{\mathtt{tx}}_{\mathrm{J}_n}}\sqrt{\mathrm{PL}_{\mathrm{J}_n\to\mathrm{R}_k}}h_{\mathrm{J}_n\to\mathrm{R}_k} \\ & x_{\mathrm{J}_n,\mathrm{BP}}(t)x^{\mathtt{PN,LFM}}_{k,\mathrm{BP}}(t).\end{aligned} \tag{17}$$

Our interference detection approach is based on analyzing the CWD representation of a received signal, denoted as $\mathbf{X}_{\mathtt{CWD}}$ with double decision thresholds. We design two decision-making statistics, $\Lambda(\mathbf{X}_{\mathtt{CWD}})$ and $\Lambda'(\mathbf{X}_{\mathtt{CWD}})$, and compare them with two thresholds, λ and λ', to make decisions regarding the presence of interference. The hypothesis for interference detection can be expressed as follows, H_1 = True if $\Lambda(\mathbf{X}_{\mathtt{CWD}}) > \lambda$ and $\Lambda'(\mathbf{X}_{\mathtt{CWD}}) < \lambda'$ and H_0 = True if $\Lambda(\mathbf{X}_{\mathtt{CWD}}) < \lambda$ or $\Lambda'(\mathbf{X}_{\mathtt{CWD}}) > \lambda'$. The tests statistic $\Lambda(\mathbf{X}_{\mathtt{CWD}})$ and $\Lambda'(\mathbf{X}_{\mathtt{CWD}})$, and the thresholds λ and λ' are determined by the ML model and KDE in the latent space being considered. The objectives of the detection algorithm are to minimize the false alarm probability and increase the detection probability, which are defined as $P_d = \Pr\left(\Lambda(\mathbf{X}_{\mathtt{CWD}}) > \lambda \text{ and } \Lambda'(\mathbf{X}_{\mathtt{CWD}}) < \lambda' \mid \mathrm{H}_1\right)$ and $P_f = \Pr\left(\Lambda(\mathbf{X}_{\mathtt{CWD}}) > \lambda \text{ and } \Lambda'(\mathbf{X}_{\mathtt{CWD}}) < \lambda' \mid \mathrm{H}_0\right)$, respectively. In the binary classification problem, the TPR represents the P_d, and FPR represents the P_f.

2) HETEROGENEOUS INTERFERENCE TYPE CLASSIFICATION PROBLEM

The objective of interference classification is to identify the source of the interference. In this task, there are three classes $\mathtt{C}_1$, $\mathtt{C}_2$, and $\mathtt{C}_3$. Therefore we have a multi-classification problem. Since we have $|\psi^{\mathtt{phsII}}|$ distinct types of interference, we have $|\psi^{\mathtt{phsII}}|$ hypotheses $\mathtt{C}_i, i \in \{0, \ldots, |\psi^{\mathtt{phsII}}| - 1\}$, where each hypothesis denotes a type of interference. Given a sample $\mathbf{X}^{\mathrm{H}_1}_{\mathtt{CWD}}$ classified as H_1 in Phase I and its label $y \in \psi^{\mathtt{phsII}}$, the objective of the interference classification model is to find the class with the greatest posterior probability, which is expressed as

$$\begin{aligned} y^* &= \underset{0\le i\le|\psi^{\mathtt{phsII}}|-1}{\arg\max} \Pr\left(\mathtt{C}_i \mid \mathbf{X}_{\mathtt{CWD}}\right) \\ &= \underset{0\le i\le|\psi^{\mathtt{phsII}}|-1}{\arg\max} \frac{\Pr\left(\mathbf{X}_{\mathtt{CWD}} \mid \mathtt{C}_i\right) \times \Pr\left(\mathtt{C}_i\right)}{\Pr\left(\mathbf{X}_{\mathtt{CWD}}\right)}, \end{aligned} \tag{18}$$

where $\Pr(\mathbf{X}_{\mathtt{CWD}})$ is the evidence and does not influence the argmax, the prior probability is given by $\Pr(\mathtt{C}_i) = \frac{1}{|\psi^{\mathtt{phsII}}|}\forall i$, and $\Pr(\mathbf{X}_{\mathtt{CWD}} \mid \mathtt{C}_i)$ is the likelihood. Determining the likelihood mathematically can be difficult because the relationships between variables are complex and the data is noisy. Thus, we need machine learning(ML) to automatically learn patterns and relationships in the data to make probabilistic predictions with high accuracy.

3) INTERFERENCE STRENGTH FORECASTING PROBLEM

Let us formulate the interference prediction problem as time series forecasting problem. At time, t the goal is to predict the interference level in the next k time steps, $\widehat{\mathbf{X}}^{\mathrm{RIS}}_t = \left[\widehat{X}^{\mathrm{RIS}}_{t+1}, \widehat{X}^{\mathrm{RIS}}_{t+2}, \ldots, \widehat{X}^{\mathrm{RIS}}_{t+(k-1)}, \widehat{X}^{\mathrm{RIS}}_{t+k}\right]$, based on past interference strength measurement values in a window of L

$$\begin{aligned} r^{\mathtt{mix}}_{k,\mathtt{c}}(t) = \mathbb{I}(\mathtt{C}_0)\sum_{K'=0}^{N_\mathrm{R}-1}\sqrt{P^{\mathtt{rx}}_{\mathrm{R}_{K'}\to\mathrm{T}\to\mathrm{R}_k}}\sum_{i=0}^{M-1} b_i \sum_{l=0}^{P-1} c_l A \times \mathrm{rect}\left(t - iT - lT_\mathtt{c}\right) \times \exp\Big(2\pi j\Big(\big(\Gamma\tau_0 + \frac{2\nu_\mathrm{T} f_\mathtt{c}}{c} - \frac{\Gamma\nu_\mathrm{T}\tau_0}{c}\big)t - \frac{\Gamma\tau_0^2}{2} \\ + \frac{2\Gamma\left(\nu_\mathrm{T} - \frac{\nu_\mathrm{T}^2}{c}\right)t^2}{c}\Big)\Big) + \mathbb{I}(\mathtt{C}_1)\sum_{m=1}^{N_\mathrm{S}}\sqrt{P^{\mathtt{tx}}_{\mathrm{S}_m}}\sqrt{\mathrm{PL}_{\mathrm{S}_m\to\mathrm{R}_k}} \times h_{\mathrm{S}_m\to\mathrm{R}_k}x_{\mathrm{S}_m,\mathrm{BP}}(t)x^{\mathtt{PN,LFM}}_{k,\mathrm{BP}}(t) + \mathbb{I}(\mathtt{C}_2)\sum_{n=1}^{N_\mathrm{J}}\sqrt{P^{\mathtt{tx}}_{\mathrm{J}_n}} \\ \sqrt{\mathrm{PL}_{\mathrm{J}_n\to\mathrm{R}_k}}h_{\mathrm{J}_n\to\mathrm{R}_k}x_{\mathrm{J}_n,\mathrm{BP}}(t)x^{\mathtt{PN,LFM}}_{k,\mathrm{BP}}(t) + w^{\mathtt{mix}}_{\mathrm{R}_k,\mathtt{c}}(t). \end{aligned} \tag{14}$$

steps, $\mathbf{X}_{t-L}^{\mathrm{RIS}} = \left[X_{t-(L-1)}^{\mathrm{RIS}}, \ldots, X_{t-1}^{\mathrm{RIS}}, X_t^{\mathrm{RIS}}\right]$. Thus, to find the optimal sliding window parameters, i.e., (L^*, k^*), we trained the proposed forecasting algorithm with multi-sliding window parameters for each identified interference in Phase II, and we chose the optimal sliding window parameters for each class that give us the minimum mean average error (MAE), thus the optimal slidin window is given by

$$\left(L_i^*, k_i^*\right) = \arg\min_{L_i, k_i}\{\mathtt{MAE}(\mathtt{L_i}, \mathtt{k_i})\} \quad \forall \quad \mathtt{i} = 1 \ldots 3. \tag{19}$$

Given these L_i^* measures of the interference level of $\mathtt{C}_i$, identified in Phase II, our third objective is to predict the sequence of the next k_i^* interference levels using the RIS-AR-LSTM. For $(j = 1, \ldots, k_i^*)$, the problem can be represented as

$$\widehat{X}_{t+j}^{\mathrm{RIS}} = \arg\max_{X_{t+j}^{\mathrm{RIS}}} \Pr\left(X_{t+j}^{\mathrm{RIS}} \mid X_{t-(L_i^*-j)}^{\mathrm{RIS}}, \ldots, X_t^{\mathrm{RIS}}, \widehat{X}_{t+1}^{\mathrm{RIS}}, \ldots, \widehat{X}_{t+j-1}^{\mathrm{RIS}}\right). \tag{20}$$

4) PERFORMANCE EVALUATION METRICS

To evaluate our framework, we considered metrics for interference detection and classification, as well as metrics for interference forecasting. Let us denote by $\mathcal{D}|\mathtt{C}_{j,i}^{\mathrm{phs}}$ the set of samples from the j-th class classified as the i-th class. Let $\mathrm{TP}(\mathtt{C}_j)$ be the number of samples correctly classified as the j-th class, $\mathrm{FP}(\mathtt{C}_j)$ the number of samples incorrectly classified as the j-th class, and $\mathrm{FN}(\mathtt{C}_j)$ the number of samples that belong to the j-th class but are not classified as such. The accuracy of the j-th class is defined as the proportion of correct predictions among all predictions, which is expressed as

$$P_A^{\mathrm{phs}}\left(\mathtt{C}_j^{\mathrm{phs}}\right) = \frac{|\mathcal{D}|\mathtt{C}_{j,j}^{\mathrm{phs}}|}{\sum_{i=1}^{|\mathcal{C}^{\mathrm{phs}}|}\left|\mathcal{D}|\mathtt{C}_{j,i}^{\mathrm{phs}}\right|}, \tag{21}$$

where $\mathcal{C}^{\mathrm{phs}}$ and $|\mathcal{C}^{\mathrm{phs}}|$ are the set of samples and its cardinality, respectively, in each phase. Also, we use the precision of the j-th class, which is defined as the ratio of correctly predicted observations as j-th class to the total predicted observations as j-th class. The precision is given by

$$\mathrm{PPV}\left(\mathtt{C}_j\right) = \frac{\mathrm{TP}\left(\mathtt{C}_j\right)}{\mathrm{TP}\left(\mathtt{C}_j\right) + \mathrm{FP}\left(\mathtt{C}_j\right)} = \frac{|\mathcal{D}|\mathtt{C}_{j,j}^{\mathrm{phs}}|}{\sum_{i=1}^{|\psi^{\mathrm{phs}}|}|\mathcal{D}|\mathtt{C}_{i,j}^{\mathrm{phs}}|}. \tag{22}$$

In addition, we use the true positive ratio of the class $\mathtt{C}_j$, $\mathrm{TPR}(\mathtt{C}_j)$, which is defined as the ratio of correctly predicted observations as j-th class to all observations in the actual class

$$\mathrm{TPR}\left(\mathtt{C}_j\right) = \frac{\mathrm{TP}\left(\mathtt{C}_j\right)}{\mathrm{TP}\left(\mathtt{C}_j\right) + \mathrm{FN}\left(\mathtt{C}_j\right)} = \frac{|\mathcal{D}|\mathtt{C}_{j,j}^{\mathrm{phs}}|}{\sum_{i=1}^{|\psi^{\mathrm{phs}}|}\left|\mathcal{D}|\mathtt{C}_{j,i}^{\mathrm{phs}}\right|}. \tag{23}$$

Another metric is the FPR of class $\mathtt{C}_j$ which is defined as the number of samples from the j-th class that are incorrectly classified, divided by the total number of samples belonging to the j-th class

$$\mathrm{FPR}\left(\mathtt{C}_j\right) = \frac{\mathrm{FP}\left(\mathtt{C}_j\right)}{\mathrm{FP}\left(\mathtt{C}_j\right) + \mathrm{TN}\left(\mathtt{C}_j\right)} = \frac{\sum_{i=1, i\neq j}^{|\psi^{\mathrm{phs}}|}|\mathcal{D}|\mathtt{C}_{i,j}^{\mathrm{phs}}|}{\sum_{i=1}^{|\psi^{\mathrm{phs}}|}|\mathcal{D}|\mathtt{C}_{i,j}^{\mathrm{phs}}|}. \tag{24}$$

The main metrics used to evaluate the RIS-AR-LSTM (forecasting model) are the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). Let $\mathbf{e}(t) = \widehat{\mathbf{X}}_t^{\mathrm{RIS}} - \mathbf{X}_t^{\mathrm{RIS}} = \left[\widehat{X}_{t+1}^{\mathrm{RIS}} - X_{t+1}^{\mathrm{RIS}}, \ldots, \widehat{X}_{t+k}^{\mathrm{RIS}} - X_{t+k}^{\mathrm{RIS}}\right]$ denote the error vector for k predicted values of RIS in the future. Then, the metrics are defined as shown below

$$\mathrm{MAE}(\widehat{\mathbf{X}}_t^{\mathrm{RIS}}) = \frac{\sum_{i=1}^{k}\left|\widehat{X}_{t+i}^{\mathrm{RIS}} - X_{t+i}^{\mathrm{RIS}}\right|}{k} \tag{25}$$

$$\mathrm{RMSE}(\widehat{\mathbf{X}}_t^{\mathrm{RIS}}) = \sqrt{\frac{1}{k}\sum_{i=1}^{k}\left(\widehat{X}_{t+i}^{\mathrm{RIS}} - X_{t+i}^{\mathrm{RIS}}\right)^2} \tag{26}$$

III. THE PROPOSED IDCF FRAMEWORK

A. MOTIVATION OF THE PROPOSED IDCF ALGORITHM

ESM radar systems can be impacted by interference from various sources, including inband communication and jammers, which can decrease their effectiveness and compromise sa. Traditional methods for interference detection, classification, and forecasting, such as energy and hypothesis testing, may not be effective in complex and dynamic environments. To address this challenge, ESM radar systems convert received signals into TFD representations, and DL techniques can be used to extract hidden and meaningful features from these representations. This can lead to improved accuracy and advance the state of the art in ESM radar systems. Thus, a novel hybrid framework, IDCF, is proposed to deal with these problems.

B. THE PROPOSED IDCF ALGORITHM

The victim ESM radar utilizes cognitive techniques to establish its own DL framework as illustrated in Fig. 1. The principles of the IDCF algorithm are explained as follows:

- **Step 1**: We utilize two features of the received signals to classify and forecast the interference, namely CWD representations and past RIS values. It is noted that the neural network model is trained based on the values of pixels in CWD images in the first and second phases and the past RIS values in the third phase.
- **Step 2**: The radar first performs outlier detection optimally designed on the extracted CWD representations to predict whether there is interference in the detected radiation or not using a novel dual-threshold mechanism. If interference is detected, the victim radar applies a classifier to the CWD representation to verify the result of Phase I and identify the source of interference. The classifier is designed according to phase I design. Next, the radar selects the appropriate sliding window parameter for the identified class in Phase I and

Algorithm 1 Proposed IDCF Algorithm

Input: Dataset of received signals, reconstruction threshold, density threshold
for $i \leftarrow (0, \text{length(dataset)})$ **do**
  $\mathbf{X}_{\text{CWD}} \leftarrow \text{TFD(dataset}(i))$ // TFD of the ith signal using Alg 2
  $\widehat{\mathbf{X}}_{\text{CWD}} \leftarrow \mathbf{CWD-Res-AE}(\mathbf{X}_{\text{CWD}})$ // **Start Phase I**
  $\mathbf{Z}_{\text{CWD}} \leftarrow \text{Encoder}(\mathbf{X}_{\text{CWD}})$; score $\leftarrow \text{KDE}(\mathbf{Z}_{\text{CWD}})$ // Apply encoder and KDE
  $\text{error}(\mathbf{X}_{\text{CWD}}) = \text{mean}((\mathbf{X}_{\text{CWD}} - \widehat{\mathbf{X}}_{\text{CWD}})^2)$ // Compute the reconstruction error
  if $\text{error}(\mathbf{X}_{\text{CWD}}) \leq$ reconsthreshold *or* score $\leq$ KDEthreshold **then**
    testresult $-$ **CWD** $-$ **Res** $-$ **AE**.append(H_0') // There is no interference
  else
    testresult $-$ **CWD** $-$ **Res** $-$ **AE**.append(H_1) // There is interference
    $\widehat{\text{C}}_i \leftarrow$ **CWD-Res-CNN**. predict($\mathbf{X}_{\text{CWD}}$)// **Start Phase II**
    testresult $-$ **CWD** $-$ **Res** $-$ **CNN**.append($\widehat{\text{C}}_i$)
    $\mathbf{X}_t^{\text{RIS}} \leftarrow$ RISdata
    $\mathbf{X}_t^{\text{RIS}} \leftarrow \text{normalization}(\mathbf{X}_t^{\text{RIS}})$,// **Start Phase III**
    if testresult-CWD-Res-CNN(i) $\in \text{C}_1, \text{C}_3$ **then**
      for $i \leftarrow (0, k)$ **do**
        $\widehat{X}_{t+i}^{\text{RIS}} \leftarrow$ **RIS** $-$ **AR** $-$ **LSTM**. predict $(\mathbf{X}_{t+i}^{\text{RIS}}[i : i+35])$ // Predict RIS
        $\mathbf{X}_{t+i}^{\text{RIS}} \leftarrow \text{combine}(\mathbf{X}_{t+i}^{\text{RIS}}[i+1 : i+35], \widehat{X}_{t+i}^{\text{RIS}})$ // Update the input
    else
      for $i \leftarrow (0, k)$ **do**
        $\widehat{X}_{t+i}^{\text{RIS}} \leftarrow$ **RIS** $-$ **AR** $-$ **LSTM**. predict $(\mathbf{X}_{t+i}^{\text{RIS}}[i : i+23])$ // Predict RIS
        $\mathbf{X}_{t+i}^{\text{RIS}} \leftarrow \text{combine}(\mathbf{X}_{t+i}^{\text{RIS}}[i+1 : i+23], \widehat{X}_{t+i}^{\text{RIS}})$ // Update the input
output Phase I: H_1, H_0 decisions, Phase II: optimal $\widehat{\text{C}}$, Phase III: predicted level $\widehat{\mathbf{X}}_{t+k}^{\text{RIS}}$

forecasts the future values of the RIS using the past RIS values. Each phase is designed and trained according to the results of the past phase, and it is an integrated framework.

- **Step 3**: The trained proposed algorithm is tested with a new set of data, which includes varying parameters for the interference source. The evaluation process enables an assessment of the algorithm's ability to generalize beyond the training data to new and unseen scenarios.

In Step 2 of the framework, our integrated three-stage IDCF algorithm significantly enhances the capabilities of LPI/LPD radars through a systematic approach to interference management. The first phase uses CWD for precise feature extraction, aiding in the early detection of interference, which is critical for maintaining sa. The dual-threshold mechanism used in this phase increases detection sensitivity, enabling the radar to identify early threats and promptly react to them. Next, the classification phase categorizes the type of interference detected, thus facilitating targeted and effective responses to specific threats, and bolstering operational reliability. Finally, the forecasting phase uses RIS data to predict future interference patterns, allowing the radar system to make proactive adjustments. This forward-looking capability is essential for adaptation in dynamic and potentially hostile environments. Collectively, these phases ensure the radar system not only effectively responds to current interference, but also anticipates and prepares for future challenges, maintaining high performance in the face of evolving electronic warfare scenarios.

C. FEATURE ENGINEER AND EXTRACTION

Feature engineering was used to achieve various advantages, such as improved predictive results, reduced computational times, minimized unnecessary noise, and increased transparency. We used domain knowledge in radar systems to employ the CWD for feature selection due to its superior ability to capture essential radar signal features as compared to the Wigner-Ville distribution (WVD) and I/Q representations [24].

1) FEATURE GENERATION AND CWD REPRESENTATION ACQUISITION

In essence, the goal was to break down the received bandpass signal into features that could help the proposed DL algorithm understand it better. To accomplish this, we mapped the received raw signal into the time-frequency domain. The energy of a signal is distributed over the two-dimensional time-frequency space as a resultant TFD of the time-frequency analysis (TFA) of the received signal. Signal processing can then take advantage of the feature created by signal energy concentration in two dimensions (time and frequency), rather than just one (time or frequency) [24].

There are four main steps for TFD representation. First we compute the auto-correlation of the received signal $r_{k,\text{c}}^{\text{mix}}(t)$ as $R(t,\tau) = r_{k,\text{c}}^{\text{mix}}\left(t+\frac{\tau}{2}\right) r_{k,\text{c}}^{\text{mix}}\left(t-\frac{\tau}{2}\right)^*$, [24, Eq. (5)]. Next, we compute the continuous WVD using the Fourier transform (FT) of the auto-correlation function [24, Eq. (6)], which can be expressed as $\text{WVD}(t,f) = \int_{-\infty}^{+\infty} R(t,\tau)\exp(-jf\tau)\,d\tau_{k'}$.

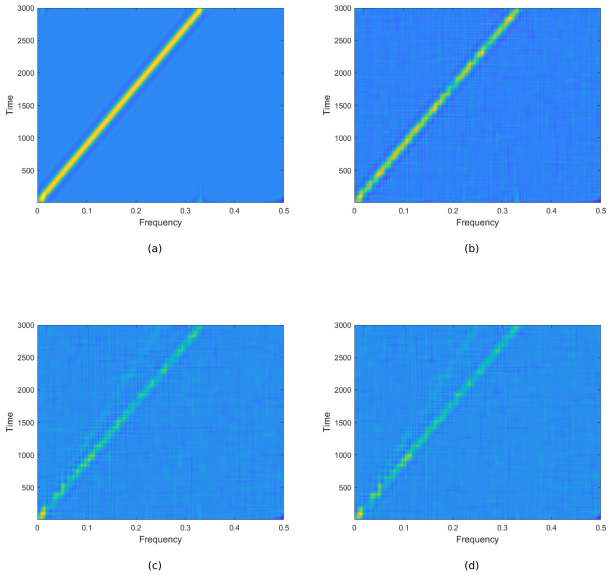


FIGURE 2. Illustrations of LFM reflected signal interfered by (a) No interference, (b) jammer, (c) inband interference, and (d) jammer plus the inband interference.

Algorithm 2 Feature Engineer

Input: Received signal
for $n \leftarrow (0, \text{length(Receivedsignal)})$ **do**
  for $m \leftarrow (0, N-1)$ **do**
    $K[2\,n, m] \leftarrow r^{\text{mix}}_{k,\text{c}}[n+m] r^{\text{mix}}_{k,\text{c}}[n-m]^*$;
    $K[2n+1, m] \leftarrow r^{\text{mix}}_{k,\text{c}}[n+m+1] r^{\text{mix}}_{k,c}[n-m]^*$
    $\text{WVD}[2\,n, k] = \text{WVD}[2\,n, k]$
    $+K[2\,n, m] \exp(-\text{j}2\pi mk/N)$
    $\text{WVD}[2n+1, k] = \text{WVD}[2n+1, k]$
    $+ \exp(-\text{j}\pi k/N)$
    $\times K[2n+1, m] \exp(-\text{j}2\pi mk/N)$
  $\text{CWD}[n, k] = \text{WVD}[n, k] \underset{n}{*} \underset{k}{*} G[n, k]$;
  $\text{CWD}[l, k] = \text{IDFT}_{n \to l}\{\text{CWD}[n, k]\}$
$\text{CWD}[l, \omega] = \text{DFT}_{k \to \omega}\{\text{CWD}[l, k]\}$
`xreal` $\leftarrow$ Receivedsignal`.real`;
`ximag` $\leftarrow$ Receivedsignal`.imag`
`IQdatareceived` $\leftarrow$ `np.stack((xreal, ximag), axis = 1)`
$\mathbf{X}^{\text{RIS}}_t \leftarrow \sqrt{|X_{\text{I}}(t)|^2}$
output: CWD dataset, RIS dataset

Taking the change of variable, $\tau/2$, we have $\text{WVD}(t,f) = 2\int_{-\infty}^{+\infty} R(t, 2\tau) \exp(-j2f\tau)\, d\tau$. The major drawback of the WVD is the presence of cross terms interference between the signal components in different locations in the time-frequency plane. The window function is used to eliminate the WVD cross-interference terms (also known as the exponential kernel function), which is given by $G(t,\tau) = \sqrt{\frac{a}{4\pi\tau^2}} \exp\left(-\frac{at^2}{4\tau^2}\right)$, where $(a > 0)$ is a scaling factor [24]. Thus, the TFD representation of the received signal can be expressed as

$$\text{CWD}(t,f) = \iint \sqrt{\frac{a}{4\pi\tau^2}} \exp\left(-\frac{at^2}{4\tau^2}\right) R(\mu, 2\tau) \times \exp(-j2f\tau)\, d\mu d\tau. \quad (27)$$

In Fig. 2, we illustrate the CWD representations of different received FMCW waveforms. We applied Min-Max normalization using the MinMaxScaler to the extracted CWD representation. This scaling method adjusts the values to a range between 0 and 1, thereby enhancing the model's ability to effectively process and learn from the data.

2) FEATURE ENGINEERING FOR THE TIME SERIES DATA

Let X_{I} be the received interference at time t, obtained by subtracting the known echo value from the received signal: $X_{\text{I}}(t) = r^{\text{mix}}_{k,\text{c}}(t) - r^{\text{C}_0,\text{mix}}_{k,\text{c}}(t)$. The interference level at t is given by $X^{\text{RIS}}_t = \sqrt{|X_{\text{I}}(t)|^2}$ [25]. Furthermore, the The input vector at time t is denoted by $\mathbf{X}^{\text{RIS}}_{t-L} = \left[X^{\text{RIS}}_{t-(L-1)}, \ldots, X^{\text{RIS}}_{t-1}, X^{\text{RIS}}_t\right]$, which contains the RIS over the past L time steps. We perform data normalization using Standard Scaler (SS) [26]. This method removes the mean of each feature and scales the variance to one. For each observation X^{RIS}_t from a sample with mean $\mathbb{E}[\mathbf{X}^{\text{RIS}}_t]$ and standard deviation $\sigma_{X^{\text{RIS}}}$, its normalized version $X^{\text{RIS},*}_t$ is given by the following: $X^{\text{RIS},*}_t = \frac{X^{\text{RIS}}_t - \mathbb{E}[\mathbf{X}^{\text{RIS}}_t]}{\sigma_{X^{\text{RIS}}}}$.

D. PHASE I: CWD-RES-AE FOR INTERFERENCE DETECTION

To identify interference in ESM radar signals and ensure optimal performance and sa, we propose a detection approach that integrates residual autoencoders with KDE to enhance detection accuracy. The proposed AD technique helps to detect images containing abnormalities, i.e., the CWD representations of C_1, C_2, and C_3 compared to normal representations, i.e., C_0.

1) ARCHITECTURE OF THE PROPOSED CWD-RES-AE

The CWD-Res-AE consists of two main parts, i.e. encoder and decoder. The encoder takes the input TFD representation and maps it into an internal representation for the hidden layers. S_{e} denotes the encoding function [27]. The transition from the input layer to the latent layer is given by

$$S_{\text{e}} : \mathcal{X} \longmapsto \mathcal{Z}, \quad \mathbf{X}_{\text{CWD}} \to S_{\text{e}}(\mathbf{X}_{\text{CWD}}) = \mathbf{Z}_{\text{CWD}}, \quad (28)$$

where $\mathcal{X}$ and $\mathcal{Z}$ are the input and output sets of the encoder, respectively, $\mathbf{X}_{\text{CWD}}$ is the CWD image input, and $\mathbf{Z}_{\text{CWD}}$ is the output of the encoder in the latent space. The encoder block is composed of several sub-blocks, the first three blocks contain a `Conv2D` layer with an activation function, followed by a `MaxPooling2D` layer. The last block is a residual block. Let us denote the activation of the l-th convolution layer by $f_e^{\text{conv},l}$. The output of the `Conv2D` layer of the l-th block is

be given by

$$\mathbf{X}_{\mathrm{CWD}}^{\mathrm{conv},l}[m,n] = f_{\mathrm{e}}^{\mathrm{conv},l}\left(\sum_{i=0}^{\mathrm{H}_{\mathrm{e}}^{l}-1}\sum_{j=0}^{\mathrm{W}_{\mathrm{e}}^{l}-1}\mathbf{X}_{\mathrm{CWD}}^{\mathrm{pool},l-1}[m+i,n+j] \times K_{\mathrm{e}}^{l}[i,j]+\mathbf{b}_{\mathrm{e}}^{l}\right), \quad (29)$$

where $K_e[i,j]$ defines the element in the i-th row and j-th column of the $\mathrm{H}_{\mathrm{e}}^{l} \times \mathrm{W}_{\mathrm{e}}^{l}$ filter matrix and $\mathbf{b}_{\mathrm{e}}^{l}$ is the bias vector. Let us denote by P_{e}^{l} the pooling function of the `MaxPooling2D` layer and $P_{\mathtt{pool}}$ is the pooling size, thus the output is given by

$$\mathbf{X}_{\mathrm{CWD}}^{\mathrm{pool},l}[m,n] = P_{\mathrm{e}}^{l}(\mathbf{X}_{\mathrm{CWD}}^{\mathrm{conv},l}[m,n]) = \max_{i,j\in\{0,\cdots,P_{\mathtt{pool}}\}}\left\{\mathbf{X}_{\mathrm{CWD}}^{l}[m+i,n+j]\right\}. \quad (30)$$

The output of the first three blocks thus is given by

$$\mathbf{X}_{\mathrm{CWD}}^{\mathrm{out}-1} = p_{\mathrm{e}}^{3}\big(f_{\mathrm{e}}^{\mathrm{conv},3}\big(p_{\mathrm{e}}^{3}\big(f_{\mathrm{e}}^{\mathrm{conv},2}\big(p_{\mathrm{e}}^{2}\big(f_{e}^{\mathrm{conv},1}\big(\mathbf{X}_{\mathrm{CWD}}\big)\big)\big)\big)\big)\big), \quad (31)$$

where $f_{\mathrm{e}}^{\mathrm{conv},3}$, $f_{\mathrm{e}}^{\mathrm{conv},2}$, $f_{e}^{\mathrm{conv},1}$, p_{e}^{3}, p_{e}^{2}, and p_{e}^{1} are the convolution functions and the pooling functions for each block. Using just the proposed three blocks, we had a problem with gradient disappearance (the gradient reduces to zero). To deal with this problem, we used a residual block in the encoder and decoder. The CWD-Res-AE model contains two residual blocks, where there is a connection that omits some layers of the model. This connection is called the skip connection and it is the heart of the residual blocks. A residual block consists of several convolutional layers with the same feature map size and number of filters. Thus the output of this block is $f_{e}^{\mathrm{res}}(\mathbf{X}_{\mathrm{CWD}}^{\mathrm{out}-1})[m,n] = \mathsf{h}\big(\mathbf{X}_{\mathrm{CWD}}^{\mathrm{out}-1}\big)[m,n] + \mathcal{F}\big(\mathbf{X}_{\mathrm{CWD}}^{\mathrm{out}-1};\mathbf{W}_{\mathrm{out}}\big)[m,n]$, where $\mathcal{F}$ is the local residual mapping to be learned, $\mathbf{W}_{\mathrm{out}}$ is a collection of weights associated with the residual block, and $\mathsf{h}\big(\mathbf{X}_{\mathrm{CWD}}^{\mathrm{out}-1}[m,n]\big)$ is a skip connection unit, where $\mathsf{h}\big(\mathbf{X}_{\mathrm{CWD}}^{\mathrm{out}-1}[m,n]\big) = W_{\mathrm{out}}^{\mathtt{skip}} \times \mathbf{X}_{\mathrm{CWD}}^{\mathrm{out}-1}[m,n]$. The function h is fixed to an identity mapping, thus $\mathsf{h}\big(\mathbf{X}_{\mathrm{CWD}}^{\mathrm{out}-1}[m,n]\big) = \mathbf{X}_{\mathrm{CWD}}^{\mathrm{out}-1}[m,n]$. As a result, the encoder function is given by

$$\begin{aligned}\mathbf{Z}_{\mathrm{CWD}} &= S_{\mathrm{e}}(\mathbf{X}_{\mathrm{CWD}})\\ &= f_{\mathrm{e}}^{\mathrm{res}} \circ p_{e}^{3} \circ f_{\mathrm{e}}^{\mathrm{conv},3} \circ p_{\mathrm{e}}^{3} \circ f_{\mathrm{e}}^{\mathrm{conv},2}\\ &\quad \circ p_{\mathrm{e}}^{1} \circ f_{\mathrm{e}}^{\mathrm{conv},1}(\mathbf{X}_{\mathrm{CWD}}).\end{aligned} \quad (32)$$

Decoding is the process of mapping the hidden layer's output back into the input feature set, which can be mathematically described as [27]

$$S_{\mathrm{d}} : \mathcal{Z} \longmapsto \mathcal{X}, \quad \mathbf{Z}_{\mathrm{CWD}} \to S_{\mathrm{d}}(\mathbf{Z}_{\mathrm{CWD}}) = \widehat{\mathbf{X}}_{\mathrm{CWD}}, \quad (33)$$

where S_{d} is the decoding function. Similar to the encoder, the decoder consists of four blocks. Three blocks are composed of `Conv2D` layer followed by a `UpSampling2D` layer and one residual block. The input of the decoder is the $\mathbf{Z}_{\mathrm{CWD}}$ and the output of the decoder after the l-th `Conv2D` layer is given by

$$\mathbf{Z}_{\mathrm{CWD}}^{\mathrm{conv},l}[m,n] = f_{\mathrm{d}}^{\mathrm{conv},l}\left(\sum_{i=0}^{\mathrm{H}_{\mathrm{d}}^{l}-1}\sum_{j=0}^{\mathrm{W}_{\mathrm{e}}^{l}-1}\mathbf{Z}_{\mathrm{CWD}}^{\mathrm{up},l-1}[m+i,n+j]K_{\mathrm{d}}^{l}[i,j]+\mathbf{b}_{\mathrm{d}}^{l}\right). \quad (34)$$

The `UpSampling2D` layer is a simple layer that increases the dimension of the input by $U^{l} \times O^{l}$, thus the output is given by $U_{\mathrm{d}}^{l}(\mathbf{Z}_{\mathrm{CWD}}^{\mathrm{conv},l}) = \mathbf{Z}_{\mathrm{CWD}}^{\mathrm{up},l}[m : m+U^{l}, n : n+O^{l}] = \mathbf{Z}_{\mathrm{CWD}}^{\mathrm{conv},l}[m,n]$, where U_{d}^{l} is the upsampling function, $m \in \{0,\cdots,U^{l} \times M^{l}\}$ and $n \in \{0,\cdots,O^{l} \times N^{l}\}$, N^{l} and M^{l} are, respectively, the number of columns and rows of the image $\mathbf{Z}_{\mathrm{CWD}}^{\mathrm{conv}}$. Similar to the encoder, the last block is the residual block, as a result, the output of the decoder is given by

$$\begin{aligned}\widehat{\mathbf{X}}_{\mathrm{CWD}} &= S_{\mathrm{d}}(\mathbf{Z}_{\mathrm{CWD}})\\ &= f_{\mathrm{d}}^{\mathrm{res}} \circ U_{\mathrm{d}}^{3} \circ f_{\mathrm{d}}^{3} \circ U_{\mathrm{d}}^{3} \circ f_{\mathrm{d}}^{2} \circ U_{\mathrm{d}}^{1} \circ f_{\mathrm{d}}^{1}(\mathbf{Z}_{\mathrm{CWD}}),\end{aligned} \quad (35)$$

where $f_{\mathrm{d}}^{\mathrm{conv},3}$, $f_{d}^{\mathrm{conv},2}$, $f_{\mathrm{d}}^{\mathrm{conv},1}$, U_{d}^{3}, U_{d}^{2}, and U_{d}^{1} are the convolution functions and the upsampling functions for each block. $f_{\mathrm{d}}^{\mathrm{res}}$ denotes the function of the residual block used in the decoder. The training of CWD-Res-AE consists of minimizing the loss function, which is the MSE given by

$$\mathtt{MSE} = \frac{1}{|\mathbf{X}_{\mathrm{CWD}}|}\sum_{i,j}\left(\mathbf{X}_{\mathrm{CWD}}[i,j] - \widehat{\mathbf{X}}_{\mathrm{CWD}}[i,j]\right)^{2}. \quad (36)$$

2) OPTIMAL LATENT SPACE DIMENSION

The first part of our CWD-Res-AE maps high-dimensional input samples $\mathcal{X}$ to a compressed, low-dimensional latent space $\mathcal{Z}$ using a non-linear mapping S_e. This space is a low-rank representation of $\mathbf{X}_{\mathrm{CWD}}$ that captures the data's underlying structure and important features with fewer parameters. Choosing an appropriate latent space dimension is crucial for achieving good TPR and FPR in AD, thus finding the optimal CWD-Res-AE architecture. The encoder architecture determines the dimension of the latent space. The optimal latent space dimensionality balances the model's capacity to represent the input data with the risk of overfitting. Additionally, the rank of the encoder output can also affect the model's performance. Therefore, it is important to consider both the latent dimensionality and rank of the latent space when designing an encoder. $\mathbb{Z}$ defines the latent space containing a set of $\mathbf{Z}_{\mathrm{CWD}}$ generated by the encoder, as shown in eq (32). Let $x_{\mathtt{in}}^{(l)}$, $y_{\mathtt{in}}^{(l)}$, and $z_{\mathtt{in}}^{(l)}$ denote the dimensions of the 3D input of the l-th layer (it can be the input layer or hidden layer). The l-th convolutional layer $f_{\mathrm{e}}^{\mathrm{conv},l}$ is composed of $z_{\mathrm{phs}}^{(l)}$ filters with spatial size $\left(k_{\mathrm{h,e}}^{(l)}, k_{\mathrm{w,e}}^{(l)}\right)$. $\left(s_{\mathrm{h,e}}^{(l)}, s_{\mathrm{w,e}}^{(l)}\right)$ presents the size of the stride, and $P_{\mathrm{H1,e}}^{(l)}$, $P_{\mathrm{H2,e}}^{(l)}$, $P_{\mathrm{W1,e}}^{(l)}$, $P_{\mathrm{W2,e}}^{(l)}$ are the padding parameters. After the l-th Conv2D layer, the output dimension is given by $\left(x_{\mathrm{conv,e}}^{(l)}, y_{\mathrm{conv,e}}^{(l)}, z_{\mathrm{conv,e}}^{(l)}\right) = \left(x_{\mathrm{in,e}}^{(l)} - k_{\mathrm{h,e}}^{(l)} + P_{\mathrm{H1,e}}^{(l)} + P_{\mathrm{H2,e}}^{(l)}\right)/s_{\mathrm{h,e}}^{(l)} + 1, y_{\mathrm{in,e}}^{(l)} - k_{\mathrm{w,e}}^{(l)} + P_{\mathrm{W1,e}}^{(l)} +$

$P_{\mathrm{W2,e}})/s_{\mathrm{w,e}}^{(l)}+1, z_{\mathrm{in,e}}^{(l)})$. Let us denote $(k_{\mathrm{ph,e}}^{\mathrm{pool},(l)}, k_{\mathrm{pw,e}}^{\mathrm{pool},(l)})$ the kernel size of the Pooling layer and $(s_{\mathrm{ph,e}}^{\mathrm{pool}\,(l)}, s_{\mathrm{pw,e}}^{\mathrm{pool},})$ is the stride. The output dimension of the l-th Pooling is given by $(x_{\mathrm{pool,e}}^{(l)}, y_{\mathrm{pool,e}}^{(l)}, z_{\mathrm{pool,e}}^{(l)}) = ((x_{\mathrm{in,e}}^{(l)} - k_{\mathrm{ph,e}}^{\mathrm{pool},(l)})/s_{\mathrm{ph,e}}^{\mathrm{pool},(l)} + 1, (y_{\mathrm{in,e}}^{(l)} - k_{\mathrm{pw,e}}^{\mathrm{pool},(l)})/s_{\mathrm{pw,e}}^{\mathrm{pool},(l)} + 1, z_{\mathrm{in,e}}^{(l)})$. The output dimension after a residual block $f_{\mathrm{e}}^{\mathrm{res}}$ is the same as applying two consecutive Conv2D layers. In the proposed encoder, we have three convolution layers each followed by a pooling layer, and the last layer is a residual block that contains two convolution layers. Hence, we implemented eight layers, and the dimension of $\mathbf{Z}_{\mathrm{CWD}}$ is related to all the dimensions of the encoder, and is given by $(x_{\mathrm{e}}^{\mathrm{out}}, y_{\mathrm{e}}^{\mathrm{out}}, z_{\mathrm{e}}^{\mathrm{out}}) = (x_{\mathrm{in,e}}^{(7)} - k_{\mathrm{h,e}}^{(7)} + P_{\mathrm{H1,e}}^{(7)} + P_{H2,\mathrm{e}}^{(7)})/s_{\mathrm{h,e}}^{(7)} + 1, y_{\mathrm{in,e}}^{(7)} - k_{w,\mathrm{e}}^{(7)} + P_{W1,\mathrm{e}}^{(7)} + P_{W2,\mathrm{e}}^{(7)})/s_{\mathrm{w,e}}^{(7)} + 1, z_{\mathrm{in,e}}^{(l)}{}^{(7)})$. Thus, the latent space dimension $\mathbb{Z}$ is given by $\dim = |\mathbb{Z}| = x_{\mathrm{e}}^{\mathrm{out}} \times y_{\mathrm{e}}^{\mathrm{out}} \times z_{\mathrm{e}}^{\mathrm{out}}$. To find the function between the TPR and the latent space dimension dim, we first create a dataset $\mathcal{D}_{\mathrm{Lat}}$ containing different values of the latent space dimension. Next, we train the CWD-Res-AE and find the minimum TPR for each latent dimension. Specifically, we use the least squares regression approach to fit a function to the set of data between the TPR and latent dimension

$$\mathrm{TPR}_{\mathrm{i}} = f(\dim_{\mathrm{i}}, \boldsymbol{\beta}) + \epsilon, \tag{37}$$

where $\boldsymbol{\beta}$ is a vector of the unknown parameters of the function f, and ϵ is the error term. In our case, f is a quadratic function to the TPR data, so our function f would be $f(\dim_{\mathrm{i}}, \boldsymbol{\beta}) = \beta_0 + \beta_1 \dim_{\mathrm{i}} + \beta_2 \dim_{\mathrm{i}}^2$, where $\boldsymbol{\beta} = [\beta_0, \beta_1, \beta_2]$ are the unknown coefficients that we estimated using least squares regression. Particularly, our goal here is to find the optimal $\boldsymbol{\beta}^*$ using this optimization problem

$$\min_{\beta} \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \left(\mathrm{TPR}_i - \left(\beta_0 + \beta_1 \dim_{\mathrm{i}} + \beta_2 \dim_{\mathrm{i}}^2\right)\right)^2. \tag{38}$$

Let us denote $\mathrm{err} = \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \left(\mathrm{TPR}_i - \left(\beta_0 + \beta_1 \dim_{\mathrm{i}} + \beta_2 \dim_{\mathrm{i}}^2\right)\right)^2$. To find $\boldsymbol{\beta}$, we use the derivative. We know that the derivative will be 0 when the error is at a minimum. Also, we have three unknowns, β_0, β_1, and β_2. Thus we take the derivative of the function with each unknown separately

$$\frac{\partial \mathrm{err}}{\partial \beta_j} = -2 \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \left[\left(\mathrm{TPR}_i - \beta_0 - \beta_1 \dim_{\mathrm{i}} - \beta_2 \dim_{\mathrm{i}}^2\right) \dim_{\mathrm{i}}^{\mathrm{j}}\right] = 0 \quad \forall \quad j = 0, 1, 2. \tag{39}$$

Thus, we have

$$\sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \beta_0 \dim_{\mathrm{i}}^{\mathrm{j}} + \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \dim_{\mathrm{i}}^{\mathrm{j}+1} \beta_1 + \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \dim_{\mathrm{i}}^{\mathrm{j}+2} \beta_2 = \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \mathrm{TPR}_i \dim_{\mathrm{i}}^{\mathrm{j}} \quad \forall \quad j = 0, 1, 2. \tag{40}$$

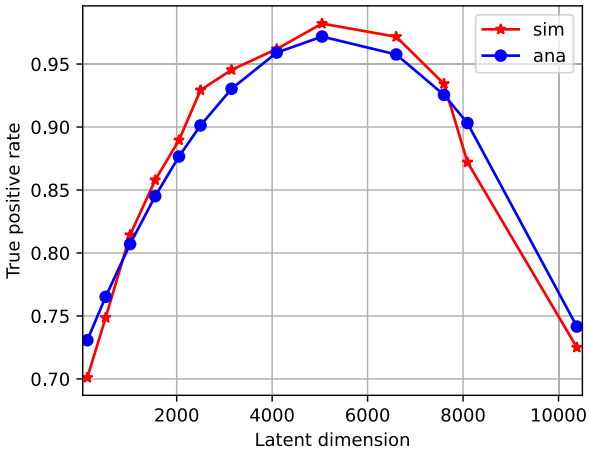


FIGURE 3. Comparison of simulated and analytical TPR as a function of latent dimension.

These equations can be re-arranged into matrix form

$$\begin{bmatrix} |\mathcal{D}_{\mathrm{Lat}}| & \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \dim_{\mathrm{i}} & \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \dim_{\mathrm{i}}^2 \\ \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \dim_{\mathrm{i}} & \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \dim_{\mathrm{i}}^2 & \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \dim_{\mathrm{i}}^3 \\ \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \dim_{\mathrm{i}}^2 & \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \dim_{\mathrm{i}}^3 & \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \dim_{\mathrm{i}}^4 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \mathrm{TPR}_i \\ \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \mathrm{TPR}_i \dim_{\mathrm{i}} \\ \sum_{i=1}^{|\mathcal{D}_{\mathrm{Lat}}|} \mathrm{TPR}_i \dim_{\mathrm{i}}^2 \end{bmatrix}. \tag{41}$$

Solving equation (41) by employing the inverse of a matrix, we successfully obtained the solution for the system of equations, affirming the effectiveness of our methodology as demonstrated in Fig. 3. The obtained optimal parameters are given by

$$[\beta_0, \beta_1, \beta_2] = [-9.04078229e-9, 9.62805686e-5, 0.716977218]. \tag{42}$$

We employed the LSR method to demonstrate an analytical function and simulate the relation between TPR and latent space dimension. The resulting curve was concave, indicating an optimal point for achieving the best TPR. Using a quadratic function and LSR approach, we obtained a stable prediction model and captured the relationship between TPR and latent space dimension. This provides guidance for selecting an optimal latent space dimension in the CWD-Res-AE, improving performance and interoperability.

3) DUAL-THRESHOLD MECHANISM

In the decision-making phase, we use a binary cross-entropy (BCE) loss to calculate the reconstruction error. The reconstruction error is calculated as the reduced mean of the BCE, which yields the difference between the input and the reconstructed CWD representation. The reconstruction of

sample $\mathbf{X}_{\mathtt{CWD}}$ is given by

$$\mathtt{Bce}(\mathbf{X}_{\mathtt{CWD}}, \widehat{\mathbf{X}}_{\mathtt{CWD}}) = \frac{1}{|\mathcal{D}^{\mathtt{phsI}}|} \sum_{i} \big(\mathbf{X}^{i}_{\mathtt{CWD}} \log(\widehat{\mathbf{X}}^{i}_{\mathtt{CWD}}) + (1 - \mathbf{X}^{i}_{\mathtt{CWD}}) \log(1 - \widehat{\mathbf{X}}^{i}_{\mathtt{CWD}})\big), \quad (43)$$

where $\mathbf{X}^{i}_{\mathtt{CWD}}$ is the i-th element of the dataset of the image $\mathbf{X}_{\mathtt{CWD}}$. Relying solely on the BCE causes a degradation in the probability of detection. Specifically, when the model tries to recreate the $\mathtt{C}_1$, $\mathtt{C}_2$, and $\mathtt{C}_3$ images, it generates a layout that is similar to $\mathtt{C}_0$ images. Thus, we use the likelihood of an image in the latent space (the most compressed layer of the autoencoder) using KDE. The proposed CWD-Res-AE is combined with KDE. KDE is a non-parametric method for estimating the distribution of latent space samples ($\mathbf{Z}_{\mathtt{CWD}}$). Given $|\mathcal{D}^{\mathtt{phsI}}|$ data samples $\mathbf{Z}_{\mathtt{CWD}}$, drawn from an unknown distribution with density function $q(\mathbf{Z}_{\mathtt{CWD}})$, the probability density function at test point $\mathbf{Z}_{\mathtt{CWD}}$ can be estimated as [28, Eq. (1)]

$$\tilde{q}(\mathbf{Z}_{\mathtt{CWD}}) = \frac{1}{|\mathcal{D}^{\mathtt{phsI}}|} \sum_{i=1}^{|\mathcal{D}^{\mathtt{phsI}}|} \frac{1}{\beta\big(\mathbf{Z}^{i}_{\mathtt{CWD}}\big)^{\mathtt{dim}}} \times \mathcal{K}\Big(\frac{\mathbf{Z}_{\mathtt{CWD}} - \mathbf{Z}^{i}_{\mathtt{CWD}}}{\beta\big(\mathbf{Z}^{i}_{\mathtt{CWD}}\big)}\Big), \quad (44)$$

where $\beta\big(\mathbf{Z}^{i}_{\mathtt{CWD}}\big)$ is the bandwidth, *dim* is the number of dimensions in $\mathbf{Z}_{\mathtt{CWD}}$, and $\mathcal{K}$ is a multivariate Gaussian function with zero mean and unit standard deviation, formulated as $\mathcal{K}(\mathbf{Z}_{\mathtt{CWD}}) = \frac{1}{(2\pi)^{|\mathbf{Z}_{\mathtt{CWD}}|}} \exp\big(-\frac{\|\mathbf{Z}_{\mathtt{CWD}}\|^2}{2}\big)$ [28, Eq. (2)]. Based on the estimated PDF, we compute the KDE threshold, $\tilde{q}(\mathbf{Z}^{\mathtt{th}}_{\mathtt{CWD}})$ values for normal and abnormal data points. Any data point with a probability density below the threshold for normal points is considered abnormal.

In ESM radars, signals can be highly variable and may exhibit correlations among multiple dimensions, such as time delay, frequency, and angle of arrival. The multivariate Gaussian kernel function is a flexible and computationally efficient way to model these complex distributions [28]. Overall, the input $\mathbf{X}_{\mathtt{CWD}}$ is classified as abnormal or normal using this dual threshold,

$$H^* = \begin{cases} \mathrm{H}_0, \text{ if } \tilde{q}(\mathbf{X}_{\mathtt{CWD}}) > \lambda_{\mathtt{ds}} \text{ OR} \\ \quad \mathtt{loss}(\mathbf{X}_{\mathtt{CWD}}, \widehat{\mathbf{X}}_{\mathtt{CWD}}) < \lambda_{\mathtt{th}}, \\ \mathrm{H}_1, \text{ if } \tilde{q}(\mathbf{Z}_{\mathtt{CWD}}) < \lambda_{\mathtt{ds}} \text{ AND} \\ \quad \mathtt{loss}(\mathbf{X}_{\mathtt{CWD}}, \widehat{\mathbf{X}}_{\mathtt{CWD}}) > \lambda_{\mathtt{th}}, \end{cases} \quad (45)$$

where $\lambda_{\mathtt{ds}}$ is a probability threshold for AD and $\lambda_{\mathtt{th}}$ is the reconstruction error threshold. We determine the thresholds that maximize the TPR and the accuracy of classification performance by optimizing these metrics over a suitable validation set containing normal and anomalous data.

E. PHASE II: INTERFERENCE CLASSIFICATION BASED ON CWD-RES-CNN

The classification phase differentiates between types of interference, which is crucial for choosing effective countermeasures, selecting the best sliding window to forecast the interference, and maintaining radar integrity. To address the challenges of 2D classification and time efficiency, we opted to work with CNNs and ResNet blocks. After obtaining the results of Phase I, we classify abnormal images into $\mathtt{C}_1$, $\mathtt{C}_2$, and $\mathtt{C}_3$ to accurately predict the source of interference.

1) ARCHITECTURE OF THE PROPOSED CWD-RES-CNN

The proposed CWD-Res-CNN architecture is inspired by the encoder architecture designed in Phase I. The encoder architecture was carefully designed based on the fitting process and input $\mathbf{X}_{\mathtt{CWD}}$. The CWD-Res-CNN architecture consists of two blocks: the encoder block and a classification block. The classification block includes a `Flatten` layer followed by several fully connected layers. The encoder block takes the input data and compresses it into a lower-dimensional representation, which is then passed to the classification block for prediction. By using the encoder part of the CWD-Res-AE, we can effectively learn a feature representation of the input data and use it to train the classifier. The input of CWD-Res-CNN is the same as the input for the first phase, $\mathbf{X}_{\mathtt{CWD}}$, the output after the first block, the encoder is given by

$$\mathbf{X}^{\mathtt{e}}_{\mathtt{CWD}} = f_e^{\mathtt{res}} \circ p_e^3 \circ f_e^{\mathtt{conv},3} \circ p_e^3 \circ f_e^{\mathtt{conv},2} \circ p_e^1 \circ f_e^{\mathtt{conv},1}(\mathbf{X}_{\mathtt{CWD}}). \quad (46)$$

In the classification block, the encoder block output $\mathbf{X}^{\mathtt{e}}_{\mathtt{CWD}}$ is converted into a 1D format using a `Flatten` layer. Then, we applied a fully connected layer on the incoming data, the `Fully Connected` layer's equation is given by $\mathbf{X}^{\mathtt{Relu}}_{\mathtt{CWD}} = \mathrm{ReLU}\left(W_l^{\top} \times \mathbf{X}^{\mathtt{flatt}}_{\mathtt{CWD}} + \mathbf{b}_r\right)$, where $\mathbf{X}^{\mathtt{flatt}}_{\mathtt{CWD}} = \mathrm{Flatten}(\mathbf{X}^{\mathtt{e}}_{\mathtt{CWD}})$ is the output of the `Flatten` layer, $W_l^{\top}$ is the weight matrix, and $\mathbf{b_r}$ is the bias vector of the `Fully Connected` layer. The last layer is a `Fully Connected` layer using a `softmax` activation function, which can be expressed as $\widehat{\mathbf{X}}^{\mathtt{out}} = \mathtt{softmax}\left(\mathbf{X}^{\mathtt{Relu}}_{\mathtt{CWD}}\right)$. The `softmax` formula is given by

$$\sigma_i = \frac{e^{\left(W_l^{\top} \times \mathbf{X}^{l-1} + b_l\right)(i)}}{\sum_{j=1}^{k} e^{\left(W_l^{\top} \times \mathbf{X}^{l-1} + b_l\right)(j)}}, \quad i = 1, \cdots, |\mathcal{C}^{\mathtt{phsII}}|, \quad (47)$$

where k is the number of targets which is set to 3 in our model. Moreover, $\sigma_i = \Pr(y_i = 1 \mid \mathbf{X}_{\mathtt{CWD}})$ indicates the predicted probability of the i-th class, and y_i is the i-th label in the truth vector $\mathbf{y}$, which can be $[1, 0, 0]$ (for $\mathtt{C}_1$), $[0, 1, 0]$ (for $\mathtt{C}_2$), or $[0, 0, 1]$ (for $\mathtt{C}_3$). The output vector is given by $\widehat{\mathbf{X}}^{\mathtt{out}} = [\sigma_1, \sigma_2, \sigma_3]$, and the output with the highest probability is the result of the classification of the current waveform in the network.

2) LOSS FUNCTION

The objective function for a multi-class classification task like interference classification is frequently the `categorical cross − entropy`, which is expressed as

$$L\big(\mathbf{X}_{\mathtt{CWD}}\big) = -\sum_{i=1}^{K} y_i \log_2\big(\sigma_i\big). \quad (48)$$

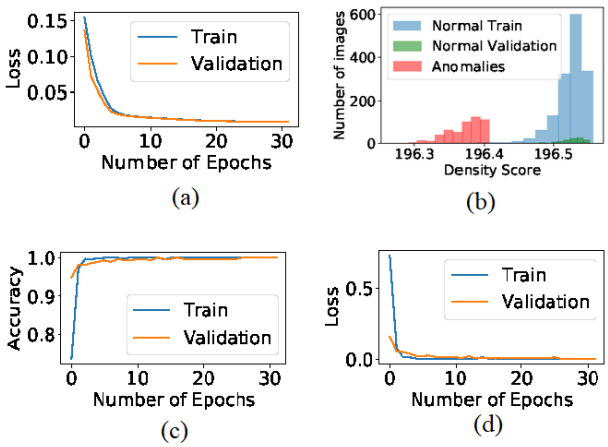


FIGURE 4. Evaluation of the training using cross-validation in terms of (a) loss of the CWD-Res-AE, (b) KDE of the latent space of CWD-Res-AE, (c) loss of CWD-Res-CNN, and (d) accuracy of the CWD-Res-CNN.

To choose the optimal hyper-parameters, such as the learning rate, number of layers, dropout rate, number of epochs, and batch size that control the learning process, we tried the cross-validation technique with different parameters.

F. PHASE III: FORECASTING INTERFERENCE

The forecasting phase of the ESM radar aims to predict future interference, enabling proactive system adjustments to improve reliability and sa. This phase integrates an AR model with a LSTM network, thus effectively capturing complex time series patterns and dynamically adapting to identified interference types. The RIS dataset for interference prediction contains the RIS values at each time step.

1) AUTOCORRELATION LINKING TIME t OBSERVATIONS WITH HISTORICAL DATA

We analyze the time series data obtained from the identified source of interference to identify its level, trend, seasonality, and noise. The autocorrelation between the time series data points is computed to determine the presence of these components [29]. Interference correlation is measured in terms of Pearson's correlation coefficient [30] which is given by

$$\rho(\tau) = \rho\left[X^{\text{RIS}}_{t_{i'}}, X^{\text{RIS}}_{t_{j'}}\right] = \frac{\text{cov}\left[X^{\text{RIS}}_{t_{i'}}, X^{\text{RIS}}_{t_{j'}}\right]}{\sqrt{\text{var}\left[X^{\text{RIS}}_{t_{i'}}\right]\text{var}\left[X^{\text{RIS}}_{t_{j'}}\right]}}, \quad (49)$$

where $\text{cov}\left[X^{\text{RIS}}_{t_{i'}}, X^{\text{RIS}}_{t_{j'}}\right] = \mathbb{E}\left[X^{\text{RIS}}_{t_{i'}} X^{\text{RIS}}_{t_{j'}}\right] - \mathbb{E}\left[X^{\text{RIS}}_{t_{i'}}\right] \mathbb{E}\left[X^{\text{RIS}}_{t_{i'}}\right]$ is the interference covariance and $\text{var}\left[X^{\text{RIS}}_{t_{i'}}\right]$, its variance. The time lag from $t_{i'}$ to $t_{j'}$ is $\tau_{i'j'} = t_{j'} - t_{i'}$. We consider two sources of correlation of interference, node locations, and wireless channel [30]. The interfering nodes' positions introduce a correlation that can be explained as follows, radars that have nearby interferes are more likely to be disrupted during target detection. In the interference correlation expression the effect of the channel is covered by the term $\mathbb{E}\left[h_{\text{I}\to\text{R}_k}(t_{i'})^2 h_{\text{I}\to\text{R}_k}(t_{j'})^2\right]$. For Nakagami fading, this term is given by $\mathbb{E}\left[h_{\text{I}\to\text{R}_k}(t_{i'})^2 h_{\text{I}\to\text{R}_k}(t_{j'})^2\right] = \frac{m+1}{m} - \frac{\tau_{i'j'}}{m}$.

The autocorrelation function (ACF) is a way to measure the linear relationship between an observation at time $t_{i'}$ and the observations at previous times $t_{j'}$. In addition, to the linear components, the relationship between $X^{\text{RIS}}_{t_{i'}}$ and $X^{\text{RIS}}_{t_{j'}}$ has non-linear components, thus we use a regression model, i.e., LSTM, to compute the non-linear components. In our work, the time series data is restructured as a supervised learning problem using a sliding window technique.

2) SLIDING WINDOW

A sliding window, $\mathbf{X}^{\text{RIS}}t$, over $\mathbf{X}^{\text{RIS}}$ is a subset with a fixed number of elements, L, which updates at each iteration. It follows the rule: $\mathbf{X}^{\text{RIS}}t = \left[X^{\text{RIS}}t, \ldots, X^{\text{RIS}}t - 2 - L, X^{\text{RIS}}_{t-1-L}\right]$. This process uses walk-forward validation [29] and is defined by the following two parameters: L (window size) and k (number of steps to forecast). The characteristics of interference from jammers differ from interference from inband communication. Specifically, jammer interference is random, requiring a larger sliding window memory to capture how the interference is changing. In contrast, interference from inband communication changes in a less random manner. To find the optimal sliding window parameters, we trained the proposed RIS-AR-LSTM with different parameters on the interference identified in Phase II. For each interference class, i.e., C_1, C_2, C_3, we found optimal sliding window parameters that achieve the minimum MAE. Thus, the optimal sliding window parameters are given by

$$(L^*, k^*) = \begin{cases} (36, 24), & \arg\max_{\text{C}_i, \forall i} \left(\hat{\mathbf{X}^{\text{out}}}\right) = \text{C}_1, \\ (24, 24), & \arg\max_{\text{C}_i, \forall i} \left(\hat{\mathbf{X}^{\text{out}}}\right) = \text{C}_2, \\ (30, 24), & \arg\max_{\text{C}_i, \forall i} \left(\hat{\mathbf{X}^{\text{out}}}\right) = \text{C}_3. \end{cases} \quad (50)$$

3) ARCHITECTURE OF THE PROPOSED RIS-AR-LSTM MODEL

With time series data, the input vector and the output variable represent the same quantity at different time instants. The forecasting model is given by

$$\widehat{X}^{\text{RIS}}_{t+1} = f\left(X^{\text{RIS}}_{t-(L^*-1)}, \ldots, X^{\text{RIS}}_{t-1}, X^{\text{RIS}}_{t}\right), \quad (51)$$

where the state at time t is represented as a function of the L immediate past values. The process is defined by a function f and a lag order L^*. The AR model decomposes the prediction into individual time steps. Then, each output can be fed back into itself at each, step, and the next predictions can be made based on the previously predicted values of RIS. The function f can be any learning model such as a neural network model. In our proposed solution, f is an LSTM model. The first output of our model is given by $\widehat{X}^{\text{RIS}}_{t+1} = \text{LSTM}[X^{\text{RIS}}_{t-(L^*-1)}, \ldots, X^{\text{RIS}}_{t-1}, X^{\text{RIS}}_{t}]$. Next, we add the predicted output to the input, we repeat this for k steps, and the last input is $\mathbf{X}^{\text{RIS}}_{t+k^*} = (X^{\text{RIS}}_{t}, \widehat{X}^{\text{RIS}}_{t+1}, \ldots, \widehat{X}^{\text{RIS}}_{t-2+k^*}, \widehat{X}^{\text{RIS}}_{t-1+k^*})$. The last output is denoted as $\widehat{X}^{\text{RIS}}_{t+k^*} = \text{LSTM}[X^{\text{RIS}}_{t}, \widehat{X}^{\text{RIS}}_{t+1}, \ldots, \widehat{X}^{\text{RIS}}_{t-2+k^*}, \widehat{X}^{\text{RIS}}_{t-1+k^*}]$. In general, an LSTM model consists of three gates:

forget, input, and output gates, denoted by i, f, and o, respectively. To find which information from previous states to be kept for further computations, we use the f_t in the forget gate, computed using $f_t = \sigma\left(\left[W_{\mathrm{fI}}, W_{\mathrm{fh}}\right] \times \left[X_t^{\mathrm{RIS}}, \mathrm{h}_{t-1}^s\right]^\top + b_{\mathrm{f}}\right)$, where σ denotes the sigmoid activation function, h_{t-1}^s represents the hidden state at time step $t-1$, W_{fI} and W_{fh} are weight matrices, and b_{f} is a constant bias. Next, the input gates make the decision of whether or not the new information will be added to the LSTM memory, using two layers. The outputs are given by $i_t = \sigma\left(\left[W_{\mathrm{iI}}, W_{\mathrm{ih}}\right] \times \left[X_t^{\mathrm{RIS}}, \mathrm{h}_{t-1}^s\right]^\top + b_i\right)$ and $c_t = \tanh\left(\left[W_{\mathrm{cI}}, W_{\mathrm{ch}}\right] \times \left[X_t^{\mathrm{RIS}}, \mathrm{h}_{t-1}^s\right]^\top + b_{\mathrm{c}}\right)$, where $W_{\mathrm{iI}}, W_{\mathrm{ih}}, W_{\mathrm{cI}}, W_{\mathrm{ch}}, b_i$ and b_{c} are the weights and biases used of the different layers, respectively. These two layers work together to update the LSTM memory by multiplying the old value c_t and adding the new value before utilizing the forget gate layer to erase the existing value. Its mathematical equation is given by $\tilde{c}_t = f_t \tilde{c}_{t-1} + i_t c_t$. The output gate then decides which area of the LSTM memory will contribute to the output using a sigmoid layer. A non-linear tanh function is then used to map the values between -1 and 1. The output of a sigmoid layer is then multiplied by the outcome. The formulas used to calculate the output are $o_t = \sigma\left(\left[W_{\mathrm{oI}}, W_{\mathrm{oh}}\right] \times \left[X_t^{\mathrm{RIS}}, \mathrm{h}_{t-1}^s\right]^\top + b_{\mathrm{o}}\right)$ and $\tilde{\mathrm{h}}_t^s = o_t \times \tanh\left(\tilde{c}_t\right)$.

G. MODEL COMPLEXITY ANALYSIS AND PRACTICAL FEASABILITY

To measure the complexity of our proposed models, we present the number of parameters (NParam) and the number of floating-point operations (FLOPs) executed. Let us denote by $x_{\mathrm{in}}^{(l)}$, $y_{\mathrm{in}}^{(l)}$, $z_{\mathrm{in}}^{(l)}$ the dimension of 3D input of the l-th layer (it can the input layer or hidden layer). The l-th convolutional layer is composed of $z_{\mathrm{phs}}^{(l)}$ filters with spatial size $(k_{\mathrm{h,phs}}^{(l)}, k_{\mathrm{w,phs}}^{(l)})$. $(s_{\mathrm{h,phs}}^{(l)}, s_{\mathrm{wphs}}^{(l)})$ presents the size of stride, and $P_{h1,\mathrm{phs}}^{(l)}, P_{\mathrm{h2,phs}}^{(l)}, P_{\mathrm{w1,phs}}^{(l)}, P_{\mathrm{w2,phs}}^{(l)}$ are the padding parameters. $(k_{\mathrm{ph,phs}}^{\mathrm{pool},(l)}, k_{\mathrm{pw,phs}}^{\mathrm{pool},(l)})$ present the kernel size of the `Pooling` layer and $(s_{\mathrm{ph,phs}}^{\mathrm{pool},(l)}, s_{\mathrm{pw,phs}}^{\mathrm{pool},(l)})$ is the stride. $c_{Fully}^{(l)}$ gives the number of filters in a `Fully` connected layer.

1) COMPLEXITY OF PHASE I

Let us denote by $L_{\mathrm{conv}}^{\mathrm{phsI}}$ and $L_{\mathrm{pool}}^{\mathrm{phsI}}$ the number of convolutional layers and pooling layers, respectively in CWD-Res-AE. For CWD-Res-AE the number of parameters is given by

$$\mathrm{NParam}^{\mathrm{phsI}} = \sum_{l=1}^{L_{\mathrm{conv}}^{\mathrm{phsI}}} ((k_{\mathrm{h,phsI}}^{(l)} \times k_{\mathrm{w,phsI}}^{(l)}) \times z_{\mathrm{in}}^{(l)} + 1) \times z_{\mathrm{out}}^{(l)}. \tag{52}$$

The FLOPs is given by (53), as shown at the bottom of the next page.

2) COMPLEXITY OF PHASE II

Let us denote by $L_{\mathrm{conv}}^{\mathrm{phsII}}$, $L_{\mathrm{pool}}^{\mathrm{phsII}}$, and $L_{\mathrm{fc}}^{\mathrm{phsII}}$ the number of convolutional layers, pooling layers, and fully connected layers, respectively in CWD-Res-CNN. Thus, $\mathrm{NParam}^{\mathrm{phsII}} = \sum_{l=1}^{L_{\mathrm{conv}}^{\mathrm{phsII}}} \left((k_{\mathrm{h,phsII}}^{(l)} \times k_{\mathrm{w,phsII}}^{(l)}) z_{\mathrm{in}}^{(l)} + 1\right) \times z_{\mathrm{out}}^{(l)} + \sum_{l=1}^{L_{\mathrm{fc}}^{\mathrm{phsII}}} z_{\mathrm{out,Fully}}^{(l)} \times c_{\mathrm{Fully}}^{(l)} + c_{\mathrm{Fully}}^{(l)}$ and $\mathrm{FLOPs}^{\mathrm{phsII}} = \sum_{l=1}^{L_{\mathrm{conv}}^{\mathrm{phsII}}} 2 \times \left(z_{in,\mathrm{phsII}}^{(l)} \times k_{\mathrm{w,phsII}}^{(l)} \times k_{\mathrm{h,phsII}}^{(l)}\right) \times z_{\mathrm{conv,phsII}}^{(l)} \times y_{\mathrm{conv,phsII}}^{(l)} \times x_{\mathrm{conv,phsI}}^{(l)} + \sum_{l=1}^{L_{\mathrm{pool}}^{\mathrm{phsII}}} (y_{\mathrm{pool,phsII}}^{(l)} / s_{\mathrm{ph,phsI}}^{(l)}) \times z_{\mathrm{pool,phsII}}^{(l)} \times (x_{\mathrm{pool,phsII}}^{(l)} / s_{\mathrm{pw,phsII}}^{(l)}) + \sum_{l=1}^{L_{fc}^{\mathrm{phsII}}} 2 \times z_{\mathrm{out,Flatt}}^{(l)} \times c$.

3) COMPLEXITY OF PHASE III

RIS-AR-LSTM consists of an input layer and L^{phsIII} LSTM cell. The inputs to an LSTM cell are $\mathbf{h}^l$ hidden state values, $\mathbf{c}^l$ state values, and $\mathbf{X}^l$ input values. In addition, there are 4 gates in the LSTM unit, which have exactly the same dense layer architecture, thus the number of trainable parameters is given by $\mathrm{NParam} = 4 \times (|\mathbf{h}^l| + |\mathbf{X}^l|) \times |\mathbf{h}^l| + 4 \times |\mathbf{h}^l|$. Thus the number of trainable parameters for RIS-AR-LSTM model is given by $\mathrm{NParam}^{\mathrm{phsIII}} = \sum_{l=1}^{L^{\mathrm{phsIII}}} 4 \times (|\mathbf{h}^l| + |\mathbf{X}^l|) \times |\mathbf{h}^l| + 4 \times |\mathbf{h}^l|$, and $\mathrm{FLOPs}^{\mathrm{phsIII}} = \sum_{l=1}^{L^{\mathrm{phsIII}}} 4 \times 2 \times (|\mathbf{h}^l| + |\mathbf{X}^l|) \times |\mathbf{h}^l|$.

4) PRACTICAL FEASIBILITY

The IDCF algorithm was designed for seamless integration into existing radar systems, prioritizing compatibility with standard CPU architectures to avoid the need for specialized hardware like GPUs. This approach ensures broad applicability of the algorithm across various operational platforms, making it a cost-effective solution to enhance radar capabilities. Despite its moderate complexity, the IDCF algorithm efficiently handles large datasets and high-dimensional inputs in real-time, demonstrating robust performance even under limited computational resources. This makes it particularly suitable for applications requiring quick decision-making, such as in air traffic control and military radar systems. Additionally, the algorithm's low retraining requirements further support its operational feasibility, ensuring reliable and continuous performance in real-world environments.

IV. NUMERICAL RESULTS AND DISCUSSIONS

In this section, we present the simulated data, the training process, and the IDCF algorithm evaluation. Extensive simulations are carried out to assess the performance of the proposed three-stage method. We compare the proposed algorithm to recent methods based on DL, i.e., the Fully connected deep neural network (FCDNN) [4], AE [10], ResNet18 [11], LSTM [12], AE-LSTM [14], CNN [16], and LSTM-based forecasting [13] to demonstrate its superior performance.

A. DATASET GENERATION

To investigate our problem, we created a dataset with synthetically generated signals using the Texas Instruments design guide [31] and MATLAB's Communications Toolbox, ensuring a realistic simulation of radar systems. The dataset

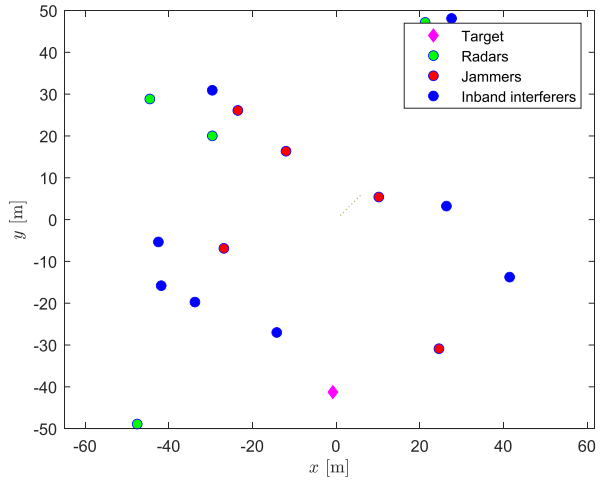


FIGURE 5. Node distribution inside two-dimensional reference elements with $N_R = 3$, $N_S = 4$, and $N_J = 5$.

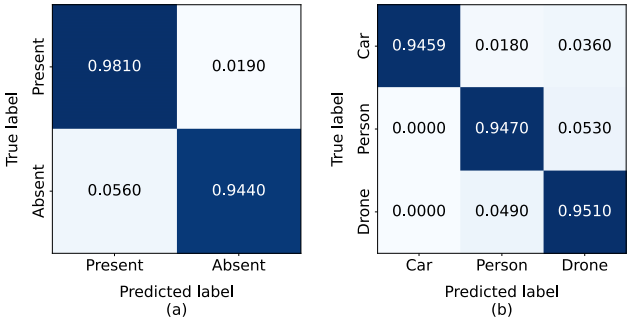


FIGURE 6. Confusion matrix of (a) Phase I a using RAD-DAR dataset and (b) Phase II using RAD-DAR dataset.

included multiple sources of interference and jammers, deployed via a uniform Poisson point process [32] to mimic the random spatial distribution typical in operational environments. Our setup included 5 radars, 4 in-band communications, and 5 jammers, with transmit power for in-band users and jammers varying from 0 to 30 dBm and 0 to 100 dBm, respectively. The network area was set to $1000 \times 1000\ \text{m}^2$, with a noise power density of -174 dBm/Hz and a noise figure of 10 dB, thus simulating realistic thermal noise and hardware-induced noise. A carrier frequency of 2.8 GHz with a 100 MHz bandwidth was used, so as to align with common radar operational frequencies. Additionally, we assumed a fixed operational radar detection range of 15 km to evaluate the algorithm's performance across different conditions. These parameters ensured that our simulations closely mirrored real-world radar deployments, as our goal was to test the algorithm's effectiveness and robustness under practical conditions.

The CWD dataset contains 12000 signals covering 4 classes with SINR levels ranging from -20 to 20 dB. The initial CWD dataset consists of two classes, normal signals, i.e., $\mathtt{C}_0$, and abnormal signals, i.e., $\mathtt{C}_1$, $\mathtt{C}_2$, and $\mathtt{C}_3$. To train the CWD-Res-AE in Phase I, we used 2500 samples of normal images, and tested the autoencoder on 500 and 9000 samples of normal and abnormal images, respectively. After the OD, we remove the samples classified as normal, in particular, 3008 images in our experiment, thus the remaining dataset contains 8992 images that belong to abnormal signals. In this dataset, there are 8 CWD images that actually belong to the normal image class that have been classified as normal image in Phase I. Then, we find the original class, i.e., $\mathtt{C}_1$, $\mathtt{C}_2$, and $\mathtt{C}_3$ of the 8992 remaining CWD images to construct another dataset for Phase II of the algorithm. After identifying the interference source, we measure its RIS for 1000 time steps and convert the data into time series using a sliding window approach based on the identified source in Phase II. We trained the IDCF framework using K-fold cross-validation and applied early stopping mechanisms in three phases to prevent overfitting.

B. HYPERPARAMETER DETERMINATION AND FINE-TUNING

To determine the hyperparameters of the proposed IDCF algorithm, we employed grid search techniques along with K-fold cross-validation to establish optimal hyperparameters that balance convergence speed with model accuracy and stability. The initial determination of hyperparameters was conducted by evaluating their effects on model performance (see Table 2). We tested various learning rates: 0.01, 0.001, and 0.0001. For 0.01, we observed a TPR of 0.82, an FPR of 0.108, and an MAE of 0.059, indicating potential instability due to high FPR and MAE. For 0.001, the TPR was significantly improved to 0.97, with a minimal FPR of 0.0012 and the lowest MAE of 0.004, marking it the optimal setting. For 0.0001, the TPR was 0.925 and FPR was 0.054; however, the higher MAE of 0.015 and slower convergence suggested that this rate might be too conservative. Consequently, we selected the learning rate of 0.001 for its superior performance, achieving high TPR, low FPR, and minimal MAE, ensuring robust and accurate interference detection

$$\begin{aligned}\text{FLOPs}^{\mathtt{phsI}} = &\sum_{l=1}^{L_{\text{conv}}^{\mathtt{phsI}}} \left(2\, z_{in,\mathtt{phsI}}^{(l)} \times k_{\mathtt{w},\mathtt{phsI}}^{(l)} \times k_{\mathtt{h},\mathtt{phsI}}^{(l)}\right) z_{\mathtt{conv},\mathtt{phsI}}^{(l)} \times y_{\mathtt{conv},\mathtt{phsI}}^{(l)} \times x_{\mathtt{conv},\mathtt{phsI}}^{(l)} \\ &+ \sum_{l=1}^{L_{\text{pool}}^{\mathtt{phsI}}} \left(y_{\mathtt{pool},\mathtt{phsI}}^{(l)} / s_{\mathtt{ph},\mathtt{phsI}}^{\mathtt{pool},(l)}\right) \times z_{\mathtt{pool},\mathtt{phsI}}^{(l)} \times \left(x_{\mathtt{pool},\mathtt{phsI}}^{(l)} / s_{\mathtt{pw},\mathtt{phsI}}^{\mathtt{pool},(l)}\right). \end{aligned} \tag{53}$$

TABLE 1. Simulation settings of the propsed IDCF algorithm.

Phase I CWD-Res-AE		Phase II CWD-Res-CNN	
Optimizer	Adam	Optimizer	Adam
Loss function	MSE	Loss function	Categorical crossentropy
Conv2D	filters = 16, kernel size = (4, 4), activation = ReLU	Input shape	96, 96, 3
MaxPooling2D	pool size = (2, 2)	Conv2D	filters=32, kernel size =3x3 , activation=ReLU
Conv2D	filters = 8, kernel size = (3, 3), activation = ReLU	MaxPooling2D	pool size = (2, 2)
MaxPooling2D	pool size = (2, 2)	Conv2D	filters = 16, kernel size = (4, 4), activation = ReLU
Conv2D	filters = 3, kernel size = (3, 3), activation = ReLU	MaxPooling2D	pool size = (2, 2)
MaxPooling2D	pool size = (2, 2)	Flatten	Flatten()
Residual Block	filters = 3, kernel size = 3	Dense	Output units = 3, activation=softmax
UpSampling2D	size = (2, 2)	**Phase III RIS-AR-LSTM**	
Conv2D	filters = 8, kernel size = (3, 3), activation = ReLU	Window size	(36, 24), (24, 24) or (30, 24)
UpSampling2D	size = (2, 2)	Loss function	MAE
Conv2D	filters = 16, kernel size = (3, 3), activation = ReLU	optimizer	Adam
UpSampling2D	size = (2, 2)	LSTM units	512
Residual Block	filters = 3, kernel size = 3, activation = ReLU	Outpput steps	24

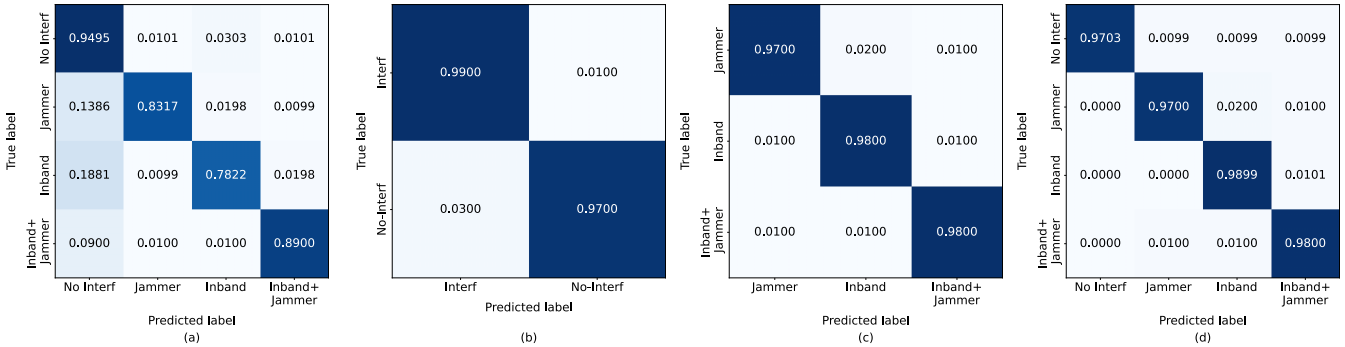


FIGURE 7. Confusion matrices of (a) CWD-Res-CNN standalone without Phase I, (b) the CWD-Res-AE used in Phase I to detect interference, (c) CWD-Res-AE used in Phase I to detect interference, and (d) the combined result of Phases I and II.

TABLE 2. Impact of tuned hyperparameters on performance metrics.

Parameter	Value	TPR	FPR	MAE
Learning Rate	0.01	0.82	0.108	0.059
Learning Rate	0.001	0.97	0.0012	0.004
Learning Rate	0.0001	0.925	0.054	0.015
Batch Size	32	0.87	0.13	0.042
Batch Size	128	0.978	0.0010	0.038
Batch Size	256	0.89	0.11	0.039
Epochs	50	0.86	0.14	0.043
Epochs	100	0.98	0.09	0.037
Epochs	200	0.89	0.11	0.036

and forecasting. The optimal hyperparameters, including the number of epochs, learning rate, optimizer specifics, and network topology, are detailed in Table 1.

C. FINDING THE OPTIMAL AE ARCHITECTURE, EVALUATION OF THE TRAINING PROCESS, AND CONVERGENCE

During the model training phase, we utilized cross-validation techniques to evaluate and fine-tune the hyperparameters for each model. The convergence of the validation loss for the proposed CWD-Res-AE model is depicted in Fig. 4a, which demonstrates its stability and proximity to the training loss. To identify the optimal architecture design for the autoencoder and determine the most appropriate latent space dimension that yields the highest TPR, we conducted a simulation curve fitting and LSR analysis. The relationship between the latent dimension and TPR is shown above.

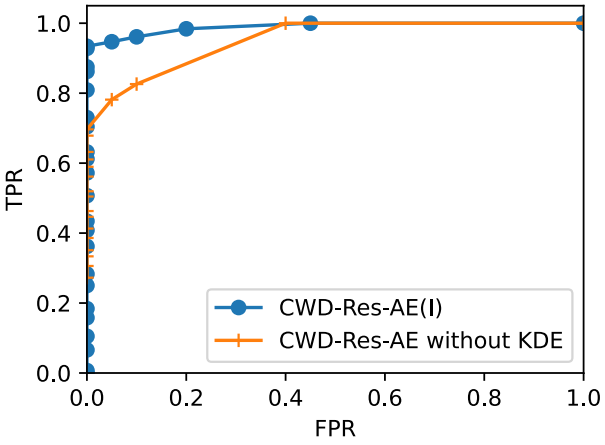


FIGURE 8. Comparison is made between the performance of CWD-Res-AE with and without KDE.

We found that the best latent dimension is 4420. Moreover, Fig. 4b shows that each point in the latent space KDE contributes a small bump to the overall density, which is controlled by the kernel and bandwidth. In areas where the map is sparsely populated, unusual and novel images are noticeable and stand out from the crowd. Figs. 4c-4d show that the loss and accuracy of CWD-Res-CNN converge to a stable value and stay consistently close to the training loss. This means that the model is not over-fitting and has good-fitting learning curves.

TABLE 3. Computational complexity comparisons between different models.

Model	FLOPs	Trainable Params	Precision	Recall	FPR	TPR	MAE
FCDNN [4]	37093952	1845116	0.8945	0.76100	0.0350	0.78345	–
AE [10]	185678090	356611	0.89978	0.8124	0.0410	0.7511	–
ResNet18 [11]	7818304	526916	0.9078	0.8424	0.0280	0.9002	–
LSTM [12]	6918304	206916	0.91018	0.8424	0.0310	0.90502	–
CNN [16]	2654082	213254	–	–	–	–	0.614
AE-LSTM [14]	2757888	260481	–	–	–	–	0.3014
LSTM based forecasting [13]	6812305	441233	–	–	–	–	0.584
Phase I: CWD-Res-AE	196870	4326	0.9978	0.9415	0.0004	0.9987	–
Phase II: CWD-Res-CNN	418930	69587	0.9784	0.7847	0.0020	0.9615	–
Phase III: RIS-AR-LSTM	1969920	105185	–	–	–	–	0.2100
Proposed IDCF	3809017	205708	**0.9874**	**0.97915**	**0.00025**	**0.9894**	**0.2100**

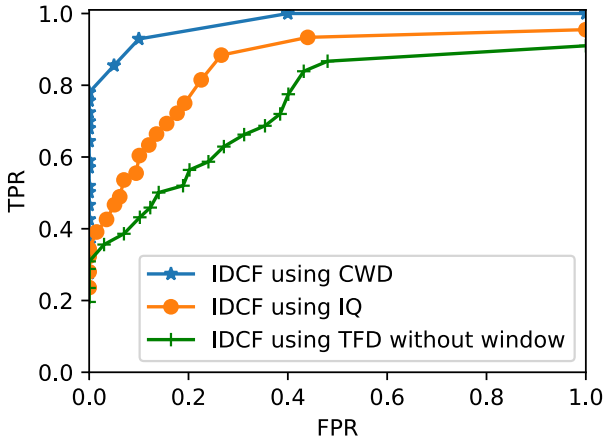


FIGURE 9. Comparison is made between the performance of CWD-IDCF, IQ-IDCF, and TFD-IDCF without window function.

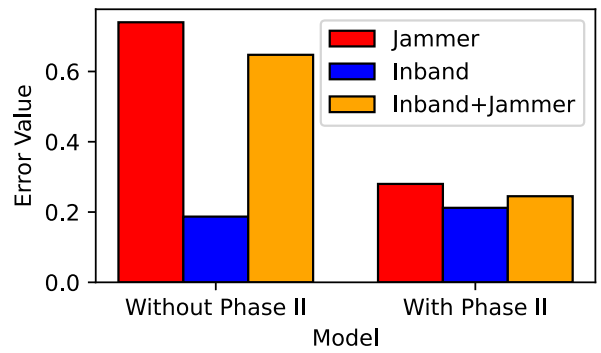


FIGURE 10. Comparison is made between the performance of RIS-AR-LSTM with and without Phase II at SINR = 5 dB.

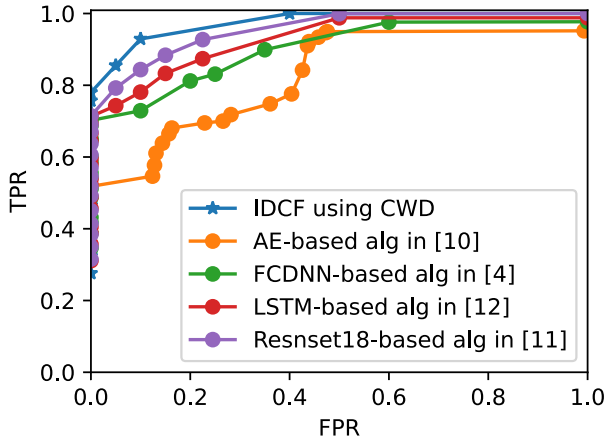


FIGURE 11. Comparison between the proposed algorithm versus existing DL-based algorithms in terms of ROC curve.

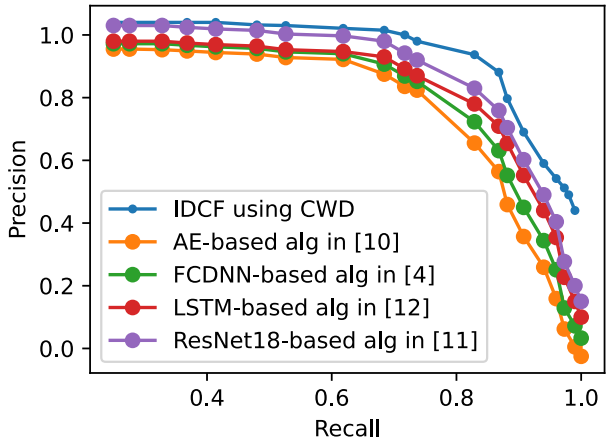


FIGURE 12. Comparison between the proposed algorithm versus existing DL-based algorithms in terms of precision-recall curve.

D. APPLY IDCF ON PRACTICAL AND SYNTHETIC DATASETS

The proposed algorithm was evaluated on the Doppler RAD-DAR database (DRRD) [33], which contains range-Doppler real data of drones, people, and cars under different situations. This dataset was used to train and test Phase I and Phase II of the algorithm. The RAD-DAR was created through a controlled trial test campaign and contains reliable labeled data collected in a real scenario. Fig. 6a and 6b show the confusion matrices of IDCF tested on a RAD-DAR database. The clear diagonal indicates that IDCF can detect and classify targets presented in the dataset.

E. IMPACT OF FEATURE EXTRACTION: THE PERFORMANCE OF CWD-IDCF AGAINST IQ-IDCF, AND WVD-IDCF

In Fig. 9, we demonstrate the effect of feature extraction on the performance of the proposed IDCF algorithm. We compare the TPR of the CWD-IDCF algorithm, the TFD-IDCF without window function, i.e., WVD representation, and the IQ-IDCF algorithm at low and high FPR. We observe that the curve of the CWD-IDCF has the highest TPR at both low and high FPR values, demonstrating that the proposed feature

extraction method improves the performance and enhances the security of radar communications. The I/Q representation of different classes is not robust to environmental conditions such as channel fading, path loss, and interference. Furthermore, using the TFD-IDCF without a window function did not achieve good performance. Overall, the CWD-IDCF algorithm achieves a higher TPR than the other approaches considered, indicating its superior performance and potential to improve radar communications' SA. Robustness of the CWD-IDCF in high interference environments particularly highlights its applicability in fields that require maintaining high sa, such as in military and aerospace applications where precise and reliable radar functionality is essential for both strategic operations and safety.

F. COMPLEXITY ANALYSIS

Table 3 provides both the number of parameters and the FLOPs to compare the proposed model complexity with baselines. As shown, the proposed IDCF framework has a relatively average number of trainable parameters of 205708, and its FLOP is not so large, i.e. equal to 3809017. It can be observed from the results that FCDNN and AE have a high FLOPs and a high number of parameters with low TPR and high FPR. However, the ResNet18 and IDCF have moderate complexity. Thus, the proposed algorithm can maintain a good trade-off between computational complexity/time and TPR. Considering its moderate complexity and favorable performance metrics, the IDCF framework's design ensures it is not only computationally efficient, but also effective in operational environments, making it particularly suitable for real-time applications that require both processing speed and accuracy for system performance.

G. THE PERFORMANCE OF THE DESIGNED CWD-RES-AE WITH KDE IN THE LATENT SPACE

As shown in Fig. 8, the ROC curve indicates that at low FPR, the TPR for CWD-Res-AE with KDE is significantly higher, at around 0.897, while for CWD-Res-AE without KDE, it is lower, at around 0.71. The proposed combination of CWD-Res-AE and KDE at the latent space has significantly improved the performance of the AD system, which in turn increases the security and SA of radar communications. With this improvement, the system can effectively detect and identify interference from various sources, including hostile attacks and inband communication, even in challenging scenarios with low SINR.

H. THE PERFORMANCE OF CWD-RES-CNN USING THE RESULT AND ARCHITECTURE OF PHASE I AND WITHOUT PHASE I

In Fig. 7, we assess the performance of the CWD-Res-CNN standalone (without Phase I). The confusion matrices present misclassifications between C_0 and other classes. Using CWD-Res-CNN directly on the dataset leads to low accuracy of around 86.5%. This is because the echo radiation from the target is present in all classes. Therefore, in our proposed solution, we designed the first phase to detect radiation corresponding to just echo targets and radiation corresponding to echo targets plus interferences. The second phase is to identify the source of interference. Moreover, the architecture of Phase II is based on the encoder of Phase I, which is carefully designed and verified with practical stimulation. In Fig. 7, we demonstrate the performance of the proposed design of Phase II and how relying on the Phase I result to train, test, and identify the source of interference increased the accuracy of Phase I to 98%. Using two phases to detect and identify the source of interference greatly enhances the accuracy and increases the SA and robustness of radar communications.

I. THE PERFORMANCE OF RIS-AR-LSTM USING THE RESULT OF PHASE II AND WITHOUT USING THE RESULT OF PHASE II

Fig. 10 demonstrates the impact of incorporating interference source classification on the performance of RIS-AR-LSTM for forecasting interference. When forecasting interference without classifying its source, high prediction errors are observed. Specifically, for C_1, C_2, and C_3, we found an error around 0.742, 0.202, and 0.62, respectively. The error for C_1 and C_3 is higher compared to C_2, mainly because the window size is fixed since we do not know the type of interference. In case radar communications detect radiation from a jammer, we use the same window size as in other classes. However, interference from jammers can be intermittent and may have varying strengths over time, making it difficult to accurately predict its behavior. Using a longer sliding window for C_1 allows the system to capture more data and analyze the signal over a longer period, enabling it to detect patterns and trends that may not be visible in a shorter window. By using specific parameters for each class, the error of C_1 reduced from 0.742 to 0.146. Identifying the source of interference helps to choose the appropriate sliding window parameter for forecasting.

J. THE PERFORMANCE OF PHASE I, PHASE II, PHASE III AND OVERALL IDCF ALGORITHM

Fig. 7 shows the confusion matrix of Phase I, i.e., interference detector, Phase II, i.e., interference classifier and the combined result the two phases. Fig. 7b shows a high TPR for class H_1 (there is interference) and H_0 (there is no interference) around 0.971, indicating that our model is effective at correctly identifying the interfered received radiation by the radar communication. Fig. 7c demonstrates the performance of Phase II, the confusion matrices present a clean diagonal with low miss-classification between classes and high TPR, around 0.981. Overall, the confusion matrix in Fig. 7d demonstrates that our model has good performance with high TPR and low FPR for all classes. The proposed solution consists of three stages that are performed jointly. Each phase requires the result of the previous one, and each phase is designed and trained using the results of the previous phase. This merge between phases and the use

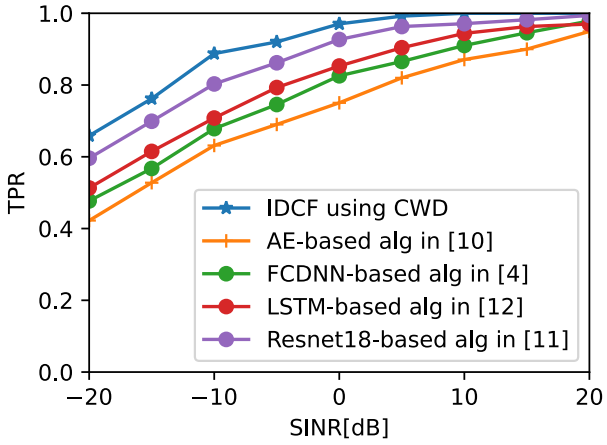


FIGURE 13. Comparison between the proposed algorithm versus existing DL-based algorithms in terms of TPR as a function of SINR.

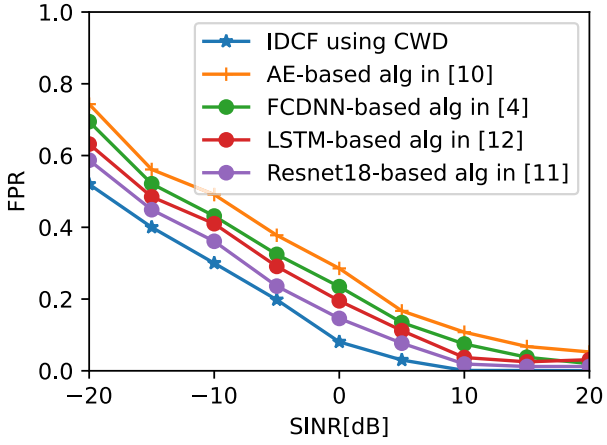


FIGURE 14. Comparison between the proposed algorithm versus existing DL-based algorithms in terms of FPR as a function of SINR.

of three cognitive techniques, detection, classification, and forecasting, make our proposed solution novel and robust under a hostile and dynamic environment. The high TPR and low FPR of the IDCF algorithm underscore its robustness and reliability, making it an optimal choice for radar systems operating in hostile and dynamic environments demanding accurate and quick response to interference so as to maintain operational integrity and safety.

1) THE SENSITIVITY ANALYSIS TO ASSESS THE ROBUSTNESS OF THE IDCF ALGORITHM

To evaluate robustness and practicality of the IDCF algorithm, we conducted a comprehensive sensitivity analysis, with a particular focus on the impact of varying input data characteristics, jammer locations, and model hyperparameters. Initially, the IDCF algorithm was trained on data generated at a SINR of 5 dB. During the inference phase, to evaluate performance of the algorithm under different signal quality conditions, we tested it on unseen data with SINR levels ranging from −20 dB to 20 dB. The results revealed the algorithm's stability and effectiveness across varying

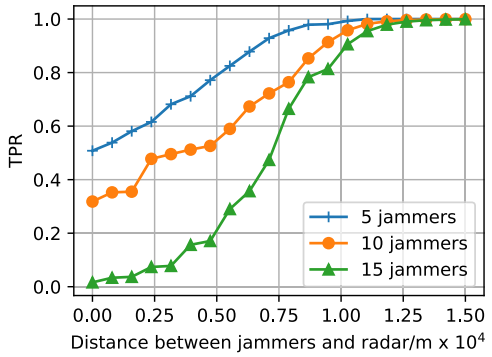


FIGURE 15. Measuring the sensitivity of the IDCF across various jammer locations.

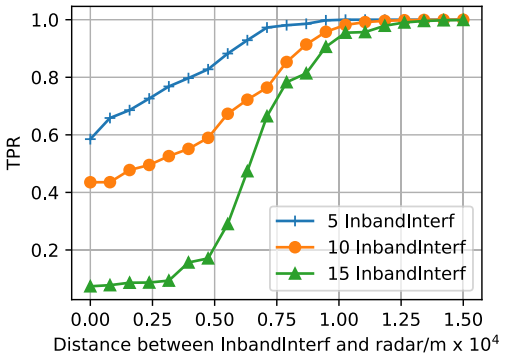


FIGURE 16. Evaluating the performance of the IDCF algorithm in terms of interference detection with varying inband interference sources and their mobility.

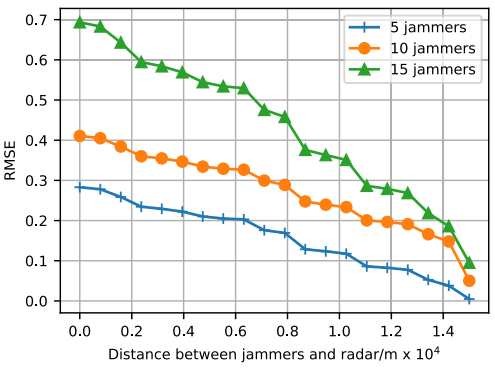


FIGURE 17. Measuring the IDCF performance for interference forecasting across various interference source locations.

SINR levels (see Fig. 13). Furthermore, we investigated how changes in jammer locations—tranging from 0.1 km to 14 km—affected the algorithm's performance. The results indicated that the algorithm effectively adapted to variations in both proximity and the number of jammers, maintaining robust performance even in densely interfered environments (see Fig. 15). Additionally, as shown in Fig. 16, the TPR for interference detection significantly improved with an increase of the distance between interference sources and the radar. With 5 sources, the TPR quickly reached nearly 1.0, while with 10 sources, it approached 0.9. For 15 sources, the

TPR rose more gradually, reaching about 0.85 at maximum distance, indicating the algorithm's robust performance even with high interference levels at closer distances. Furthermore, as can be seen in Fig. 17, with an increase of the distance between jammers and the radar, RMSE decreased. For 5 jammers, RMSE remained low throughout, dropping to nearly 0.1. With 10 and 15 jammers, RMSE started higher, but declined with increasing distance, demonstrating effective error reduction by the IDCF algorithm.

We also performed an extensive sensitivity analysis on hyperparameters, including learning rate, batch size, and number of epochs, to evaluate their impact on performance metrics. The optimal settings identified, which provided the best balance between training speed and model accuracy, were a learning rate of 0.001 and a batch size of 128.

Table 4 provides a detailed summary of the IDCF algorithm's performance across various scenarios involving different antenna types, noise levels, and environments. In rural areas, the algorithm demonstrated strong results with an omni-directional antenna, achieving a TPR of 0.98, an FPR of 0.01, and an RMSE of 0.10. Furthermore, in outdoor urban environments using a directional antenna, the TPR reached 0.94, with an FPR of 0.04 and an RMSE of 0.18. For indoor urban settings with high noise levels, the TPR was 0.85, FPR was 0.10, and RMSE was 0.30 with an omni-directional antenna, improving to a TPR of 0.90, FPR of 0.07, and RMSE of 0.22 with a phased array antenna. Collectively, These results underscore the IDCF algorithm's robustness and practicality across varying conditions and scenarios.

2) COMPARISON BETWEEN IDCF AND EXISTING SOLUTIONS

In Fig. 11, we plot the ROC curves of the IDCF, AE, FDCNN, ResNet18, and LSTM algorithms. The figure shows that for a given FPR, the TPR of the proposed algorithm is very high. For example, when FPR $= 0.2$, we have a *TPR* of 0.84. For the AE and FCDNN algorithms, when FPR varies between 0.14 and 1, the TPR varies between 0.54 and 0.944. For LSTM and ResNet18, when the FPR varies between 0.1 and 1, the TPR varies between 0.6 and 0.987. Moreover, the *TPR* of the proposed solution is significantly higher than that of the other methods. Thus, the IDCF algorithm yields higher accuracy than other algorithms. Fig. 12 shows the precision recall curve (PRC) of the proposed algorithm with the AE, FDCNN, ResNet18, and LSTM algorithms. We can observe that, with an increase in the recall rate, there is a gradual decrease in the precision. When the recall is approximately 0.8, the precision of AE, LSTM, ResNet18, and FCDNN algorithms declines significantly, reaching approximately 0.85; meanwhile, the precision of our proposed algorithm remains constant at approximately 0.93. In other words, for a fixed recall the proposed model is more advantageous in terms of the achieved precision rate, which indicates that using an outlier detector interference and relying on the result of that detector to design a classifier can improve the radar communication detection and classification performance.

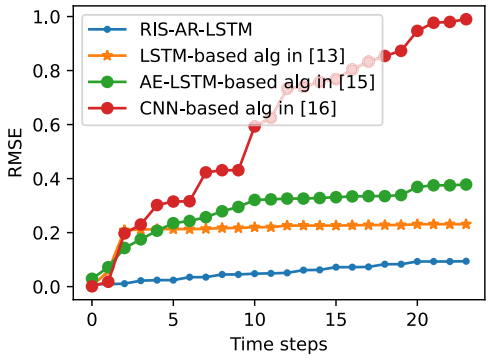


FIGURE 18. Comparison RMSE of IDCF and three forecasting methods.

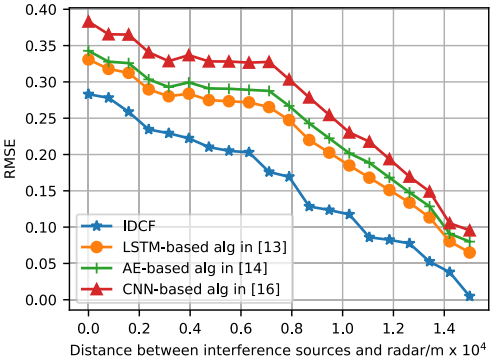


FIGURE 19. Comparison of IDCF with existing techniques for forecasting under dynamic jammers and inband Communications.

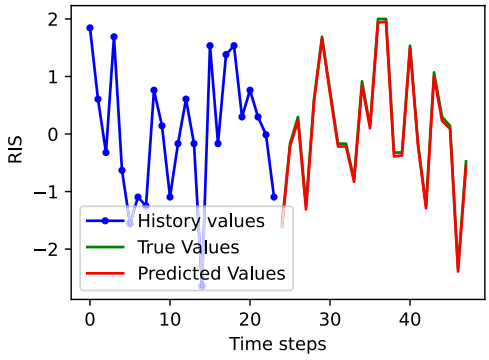


FIGURE 20. Comparison of real and predicted interference values.

In Figs. 13 and 14, we present the TPR and FPR, respectively, as functions of the SINR. At first glance, the DNN represents the worst TPR. The LSTM and ResNet18 are marginally better than the DNN and FCDNN algorithms. Compared to the DNN, the FCDNN algorithm is able to find more suitable features. It is demonstrated that the value of TPR is low for DNN, FDCNN, ResNet18, and LSTM. The problem is that applying DNN, FDCNN, ResNet18, and LSTM to identify the source of interference directly without outlier detector introduce a high misclassification between $\mathtt{C}_0$ and other classes ($\mathtt{C}_1$, $\mathtt{C}_2$, and $\mathtt{C}_3$). Therefore, in our proposed solution we start by applying the CWD-Res-AE to detect samples corresponding to $\mathtt{C}_0$ and next we

TABLE 4. Performance evaluation of the IDCF algorithm under different conditions.

Scenario	Antenna Type	Noise Level	Environment	TPR	FPR	RMSE
Indoor	Omni-directional	High	Indoor Urban	0.85	0.10	0.30
	Directional	Medium	Indoor Urban	0.88	0.08	0.25
	Phased Array	Low	Indoor Urban	0.90	0.07	0.22
Outdoor	Omni-directional	Medium	Outdoor Urban	0.92	0.05	0.20
	Directional	Medium	Outdoor Urban	0.94	0.04	0.18
	Phased Array	Low	Outdoor Urban	0.96	0.03	0.15
Rural	Omni-directional	Low	Rural	0.98	0.01	0.10
	Directional	Low	Rural	0.97	0.02	0.12
	Phased Array	Low	Rural	0.96	0.03	0.14

rely on the result of CWD-Res-AE to design the CWD-Res-CNN classifier and identify the source of interference. We demonstrated in Figs. 13 and 14 that the proposed solution perform better than existing advanced DL techniques. Fig. 18 show the comparative analysis of different time-series forecasting models. The best three models, with the lowest RMSE, are LSTM, AE-LSTM, and RIS-AR-LSTM. The performance of LSTM and AE-LSTM is poor due to their non-linear nature, whereas the relationship between RIS values contains both linear and non-linear components. As a non-linear and linear model, RIS-AR-LSTM can smoothly describe both the non-linear and linear relationship between inputs and outputs. The linear component is introduced by the AR model and the non-linear component by the LSTM model. Fig. 19 compares RMSE performance between the IDCF algorithm and existing methods in dynamic environments. As can be seen in the figure, the IDCF consistently achieved lower RMSE, particularly at closer distances, demonstrating superior accuracy and robustness in forecasting interference under varying conditions. Fig. 20 demonstrates the algorithm's performance over time steps, comparing the predicted RIS values against true RIS observed data. As illustrated, the algorithm adeptly captures the trends and fluctuations inherent to RIS data, with the blue line representing historical values used as input, while the red line indicates the algorithm's predictions. Green markers denote the true values.

V. CONCLUSION

In this study, we proposed the IDCF algorithm using the CWD dataset, a novel hybrid framework for interference detection, classification, and forecasting to enhance SA in ESM radar systems. The IDCF algorithm consists of the following three integrated phases: detecting abnormal signals using CWD-Res-AE, predicting the source of interference with CWD-Res-CNN, and forecasting future RIS values using RIS-AR-LSTM. The results of our experiments demonstrated the algorithm's effectiveness on practical datasets (CWD, WVD, and I/Q). Specifically, for interference detection, the method achieved a precision of 0.9978, a recall of 0.9415, and a FPR of 0.0004. For interference classification, it achieved a precision of 0.9784 and a recall of 0.7847. For interference forecasting, the MAE was 0.2100. The results of our with other models showed that our approach significantly improves detection, classification, and prediction accuracy. These results confirm that the IDCF algorithm using the CWD dataset is superior to existing methods, thus enhancing ESM radar systems' ability to effectively operate amidst interference and jammers. The IDCF algorithm is highly adaptable to existing radar systems, requiring no specialized hardware, making it a cost-effective and efficient solution. The algorithm's low complexity and operational feasibility ensure it can be easily deployed, enhancing radar system performance amidst interference and jamming threats.

ACKNOWLEDGMENT

An earlier version of this paper was presented in part at the Military Communications Conference (MILCOM), National Capital Region, USA, in November 2022 [DOI: 10.1109/MILCOM55135.2022.10017589].

REFERENCES

[1] Y. Wu and J. Linnartz, "Detection performance improvement of FMCW radar using frequency shift," in *Proc. 32nd Symp. Inf. Theory Benelux*, Brussels, Belgium, 2011, pp. 1–8.

[2] D. C. Schleher, "Introduction to electronic warfare," Dedham, MA, USA, Tech. Rep., 1986.

[3] Y. Chen, J. Zhang, W. Feng, and M.-S. Alouini, "Radio sensing using 5G signals: Concepts, state of the art, and challenges," *IEEE Internet Things J.*, vol. 9, no. 2, pp. 1037–1052, Jan. 2022.

[4] C. Liu, Y. Chen, and S.-H. Yang, "Deep learning based detection for communications systems with radar interference," *IEEE Trans. Veh. Technol.*, vol. 71, no. 6, pp. 6245–6254, Jun. 2022.

[5] G. Thomas, "Situation awareness issues in tactical cognitive radio," in *Proc. IEEE Int. Multi-Disciplinary Conf. Cognit. Methods Situation Awareness Decis. Support*, Mar. 2012, pp. 287–293.

[6] H. Lu, M. Jiang, and J. Cheng, "Deep learning aided robust joint channel classification, channel estimation, and signal detection for underwater optical communication," *IEEE Trans. Commun.*, vol. 69, no. 4, pp. 2290–2303, Apr. 2021.

[7] L. M. Hoang, M. Kim, and S.-H. Kong, "Automatic recognition of general LPI radar waveform using SSD and supplementary classifier," *IEEE Trans. Signal Process.*, vol. 67, no. 13, pp. 3516–3530, Jul. 2019.

[8] G. M. Brooker, "Mutual interference of millimeter-wave radar systems," *IEEE Trans. Electromagn. Compat.*, vol. 49, no. 1, pp. 170–181, Feb. 2007.

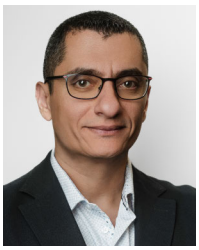
[9] J.-H. Choi, H.-B. Lee, J.-W. Choi, and S.-C. Kim, "Mutual interference suppression using clipping and weighted-envelope normalization for automotive FMCW radar systems," *IEICE Trans. Commun.*, vol. E99.B, no. 1, pp. 280–287, 2016.

[10] S. Chen, J. Taghia, U. Kühnau, N. Pohl, and R. Martin, "A two-stage DNN model with mask-gated convolution for automotive radar interference detection and mitigation," *IEEE Sensors J.*, vol. 22, no. 12, pp. 12017–12027, Jun. 2022.

[11] F. A. Bhatti, M. J. Khan, A. Selim, and F. Paisana, "Shared spectrum monitoring using deep learning," *IEEE Trans. Cognit. Commun. Netw.*, vol. 7, no. 4, pp. 1171–1185, Dec. 2021.

[12] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Trans. Cognit. Commun. Netw.*, vol. 4, no. 3, pp. 433–445, Sep. 2018.
[13] Y. Yang, D. B. Smith, and S. Seneviratne, "Deep learning channel prediction for transmit power control in wireless body area networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–6.
[14] A. Irankhah, S. Rezazadeh, M. H. Y. Moghaddam, and S. Ershadi-Nasab, "Hybrid deep learning method based on LSTM-autoencoder network for household short-term load forecasting," in *Proc. 7th Int. Conf. Signal Process. Intell. Syst. (ICSPIS)*, Dec. 2021, pp. 1–6.
[15] H. Bouzabia, G. Kaddoum, and T. N. Do, "Deep learning-based interference detection and classification for LPI/LPD radar systems," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Nov. 2022, pp. 655–660.
[16] N. Simmons, S. B. F. Gomes, M. D. Yacoub, O. Simeone, S. L. Cotton, and D. E. Simmons, "AI-based channel prediction in D2D links: An empirical validation," *IEEE Access*, vol. 10, pp. 65459–65472, 2022.
[17] Z. Lyu, L. Zhang, H. Zhang, Z. Yang, H. Yang, N. Li, L. Li, V. Bobrovs, O. Ozolins, X. Pang, and X. Yu, "Radar-centric photonic terahertz integrated sensing and communication system based on LFM-PSK waveform," *IEEE Trans. Microw. Theory Techn.*, vol. 71, no. 11, pp. 5019–5027, Nov. 2023.
[18] N. Levanon and E. Mozeson, *Radar Signals*. Hoboken, NJ, USA: Wiley, 2004.
[19] H. Zhang, L. Li, and K. Wu, "24 GHz software-defined radar system for automotive applications," in *Proc. Eur. Conf. Wireless Technol.*, Oct. 2007, pp. 138–141.
[20] E. Belyaev, P. Molchanov, A. Vinel, and Y. Koucheryavy, "The use of automotive radars in video-based overtaking assistance applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1035–1042, Sep. 2013.
[21] T. N. Do, G. Kaddoum, T. L. Nguyen, D. B. da Costa, and Z. J. Haas, "Aerial reconfigurable intelligent surface-aided wireless communication systems," in *Proc. IEEE 32nd Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2021, pp. 525–530.
[22] *Electronic Warfare and Radar Systems Engineering Handbook*, NAW Center, Electron. Warfare Division, Pont Mugu, CA, USA, 1997.
[23] S. D. Roy and A. Agrawal, "Digital low-pass filtering using the discrete Hilbert transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-26, no. 5, pp. 465–467, Oct. 1978.
[24] Y. Li, W. Song, F. Wu, E. Zio, and Y. Zhang, "Spectral kurtosis of Choi–Williams distribution and hidden Markov model for gearbox fault diagnosis," *Symmetry*, vol. 12, no. 2, p. 285, Feb. 2020.
[25] K. N. R. S. V. Prasad and V. K. Bhargava, "RSS localization under Gaussian distributed path loss exponent model," *IEEE Wireless Commun. Lett.*, vol. 10, no. 1, pp. 111–115, Jan. 2021.
[26] P. Ferreira, D. C. Le, and N. Zincir-Heywood, "Exploring feature normalization and temporal information for machine learning based insider threat detection," in *Proc. 15th Int. Conf. Netw. Service Manage. (CNSM)*, Oct. 2019, pp. 1–7.
[27] K. Zeng, J. Yu, R. Wang, C. Li, and D. Tao, "Coupled deep autoencoder for single image super-resolution," *IEEE Trans. Cybern.*, vol. 47, no. 1, pp. 27–37, Jan. 2017.
[28] L. J. Latecki, A. Lazarevic, and D. Pokrajac, "Outlier detection with kernel density functions," in *Proc. Mach. Learn. Data Mining Pattern Recognit.*, Jun. 2007, pp. 61–75.
[29] F. R. D. S. Moreira, F. A. N. Verri, and T. Yoneyama, "Maximum visibility: A novel approach for time series forecasting based on complex network theory," *IEEE Access*, vol. 10, pp. 8960–8973, 2022.
[30] U. Schilcher, J. F. Schmidt, M. K. Atiq, and C. Bettstetter, "Autocorrelation and coherence time of interference in Poisson networks," *IEEE Trans. Mobile Comput.*, vol. 19, no. 7, pp. 1506–1518, Jul. 2020.
[31] *RF Sampling S-Band Radar Receiver*, Texas Instrum., Dallas, TX, USA, vol. 21, 2016.
[32] A. Al-Hourani, R. J. Evans, S. Kandeepan, B. Moran, and H. Eltom, "Stochastic geometry methods for modeling automotive radar interference," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 333–344, Feb. 2018.
[33] H. Le, V.-S. Doan, D. P. Le, H.-H. Nguyen, T. Huynh-The, K. Le-Ha, and V.-P. Hoang, "Micro-Doppler-radar-based UAV detection using inception-residual neural network," in *Proc. Int. Conf. Adv. Technol. Commun. (ATC)*, Oct. 2020, pp. 177–181.

HAMDA BOUZABIA received the B.S. degree in telecommunication engineering from the Higher School of Communication of Tunis, Ariana, Tunisia, in 2018. He is currently a Ph.D. Scholar with the 'Ecole de Technologie Sup'erieure ('ETS), Universit'e du Qu'ebec, Montreal, Canada. His research interests include tactical wireless networks, device-to-device communication, wireless sensor networks, integrated sensing and communication systems, generative AI, machine learning, jamming detection, jamming mitigation, interference classification and forecasting, preserving data privacy, and security in communication environments.



GEORGES KADDOUM (Senior Member, IEEE) received the bachelor's degree in electrical engineering from the École Nationale Supérieure de Techniques Avancées (ENSTA Bretagne), Brest, France, the M.S. degree in telecommunications and signal processing (circuits, systems, and signal processing) from the Université de Bretagne Occidentale and Telecom Bretagne (ENSTB), Brest, in 2005, and the Ph.D. degree (Hons.) in signal processing and telecommunications from the National Institute of Applied Sciences (INSA), University of Toulouse, Toulouse, France, in 2009. He is currently a Professor and the Research Director of the Resilient Machine Learning Institute (ReMI); and the Tier 2 Canada Research Chair with the École de Technologie Supérieure (ÉTS), Université du Québec, Montréal, Canada. He has published over more than 300 journals, conference papers, and two chapters in books; and has eight pending patents. His current research interests include wireless communication networks, tactical communications, resource allocations, and network security. He received the Best Papers Awards at the 2014 IEEE International Conference on Wireless and Mobile Computing, Networking, Communications (WIMOB); at the 2017 IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC); and at the 2023 IEEE International Wireless Communications and Mobile Computing Conference (IWCMC). He also received the IEEE Transactions on Communications Exemplary Reviewer Award, in 2015, 2017, and 2019; the Research Excellence Award of the Université du Québec, in 2018; and the Research Excellence Award from ÉTS, in 2019, in recognition of his outstanding research outcomes. He also won the 2022 IEEE Technical Committee on Scalable Computing (TCSC) Award for Excellence (Middle Career Researcher). Lastly, he has received the prestigious 2023 MITACS Award for Exceptional Leadership. He served as an Associate Editor for IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY and IEEE COMMUNICATIONS LETTERS. He is serving as an Area Editor for IEEE TRANSACTIONS ON MACHINE LEARNING IN COMMUNICATIONS AND NETWORKING and an Editor for IEEE TRANSACTIONS ON COMMUNICATIONS.

TRI NHU DO (Member, IEEE) was born in Da Nang, Vietnam. He received the bachelor's degree in electronics and telecommunications engineering from the Posts and Telecommunications Institute of Technology, Vietnam, in 2012, and the master's and Ph.D. degree in electronics and computer engineering from Hongik University, South Korea, in 2015 and 2018, respectively. From 2017 to 2018, he was a Teaching Associate with the Department of Software and Communications Engineering, Hongik University. In 2019, he was a Research Associate with the Department of Computer Science, The University of Texas at Dallas, TX, USA. From 2020 to September 2023, he was a Postdoctoral Research Fellow with the Resilient Machine Learning Institute (ReMI), École de Technologie Supérieure (ÉTS), Université du Québec, Montréal, Canada. He is currently an Assistant Professor of telecommunications with the Department of Electrical Engineering, Polytechnique Montréal, QC, Canada.