Evaluation of a Multilayer Perceptron Neural Network for Intrapulse Modulation Classification in RADAR Signals

Rodolfo Norio Toma, Felipe Streitenberger Ivo, Felipe Carvalho Ueta, Lucio Pinheiro Amaro, Renato Machado and Vitor V. Curtis
Instituto Tecnológico de Aeronáutica (ITA), São José dos Campos/SP – Brasil

***Abstract*— This work addresses the problem of RADAR signal analysis in dense electromagnetic environments with multiple active emitters. It focuses on the use of waveform-derived features for classification in pulsed RADAR systems, particularly considering intrapulse modulation parameters. In particular, Multilayer Perceptrons (MLP) are evaluated for the classification of intrapulse modulation patterns. The proposed approach leverages the capability of MLP models to learn complex relationships among signal features, aiming to improve discrimination between emitters with similar temporal characteristics. The proposed model reaches 80.15% overall accuracy on the 11-class subset, attaining the 90% threshold at 8 dB SNR. Results show competitive performance for structurally distinct waveforms, while polyphase codes remain the dominant source of confusion.**

***Keywords*— RADAR signal processing, de-interleaving, intrapulse modulation, multilayer perceptron (MLP), classification.**

I. Introduction

Airborne military operations involving RADAR signal monitoring in dense electromagnetic environments, characterized by the simultaneous presence of multiple active emitters, impose significant challenges on signal processing systems [1], [2]. In such scenarios, the reliable extraction and interpretation of received signals are critical for effective Electronic Support Measure (ESM) system performance [3].

Signal separation techniques, commonly referred to in the literature as de-interleaving, play a fundamental role in isolating and associating pulses from multiple RADAR emitters [3]. These techniques are directly related to the operational reliability of ESM receivers operating under high-density signal conditions [1], [3], [4].

Furthermore, the classification of RADAR emitters enhances situational awareness by enabling the identification and characterization of potential threats, supporting decision-making processes in complex and dynamic operational environments [4] .

Within this context, de-interleaving and classification techniques rely on waveform-derived features as input parameters. In the case of pulsed RADAR systems, several fundamental characteristics can be exploited: the Time of Arrival (TOA), the Pulse Repetition Interval (PRI), and the intrapulse modulation (modulation on pulse - MOP), which describes the frequency or phase variation within a single RADAR pulse [3], [4]. These parameters provide essential information for distinguishing between emitters operating simultaneously in dense environments [3].

Artificial Intelligence (AI) techniques have emerged as a promising approach for RADAR signal analysis [2], [4]. In particular, Multilayer Perceptrons (MLP) are evaluated in this work for the classification of intrapulse modulation patterns. The direct benchmark for this comparison is the work of Abissamra et al. [5], who evaluated LSTM and CNN architectures on an 11-class subset of the DeepRadar2022 dataset [6], reporting accuracy-latency trade-offs and per-modulation SNR sensitivity thresholds.

This capability to separate signals based on intrapulse modulation is especially relevant in dense electromagnetic environments, where intrapulse characteristics provide complementary information to traditional parameters such as TOA and PRI [4]. As a result, the evaluation of MLP-based approaches is assessed in terms of its ability to discriminate between emitters with similar temporal behaviors but distinct intrapulse modulation schemes.

This paper is organized as follows. Section II presents the proposed MLP-based approach for RADAR signal classification. Section III describes the experimental setup. Section IV reports and discusses the obtained results. Finally, Section V concludes the paper.

II. Methodology

This work applied a deep learning model, namely a Multi-Layer Perceptron (MLP) [7], in the Automatic Modulation Classification task. The models of Abissamra et al. [5] were adopted as the reference for performance comparison. The dataset and the feature extraction procedure are described in the following subsections.

A. DeepRadar2022 Dataset

The dataset used in this work is the DeepRadar2022 dataset, as proposed by Clerico et al. [6], [8]. DeepRadar2022 comprises 782,000 signals generated using modulation schemes commonly employed in RADAR systems. This database was developed by the Radar and Microwave Group at Universidad Politécnica de Madrid [6].

A dataset consisting of 23 radar modulation classes was generated, including frequency, phase, and hybrid modulation schemes, as well as non-modulated and noise signals sampled at 100 MHz. Each sample is represented by a 1024×2 real-valued vector corresponding to the in-phase (I) and quadrature

R.N. Toma, noriornt@ita.br; F.S. Ivo, fivo@ita.br; F.C. Ueta, felipefcu@ita.br; L.P. Amaro, lucio@ita.br; R. Machado, rmachado@ita.br; V.V. Curtis, curtis@ita.br.

(Q) components, with additive noise. Signals were generated under Signal-to-Noise Ratio (SNR) conditions ranging from −12 dB to 20 dB, with a step size of 2 dB, summing up 782,000 different signals with a (1024 × 2) shape. They are already separated into 60% for training, 20% for test and 20% validation.

In order to enable a fair comparison with the work of [5], who evaluated LSTM and CNN architectures on an 11-class subset of the DeepRadar2022, a reduced dataset comprising 11 classes was employed in this study, including Linear Frequency Modulation, Frequency Modulation with Costas code, Binary Phase-Shift Keying, and Phase Modulation signals using Barker, Frank, P1, P2, P3, and P4 codes, Non-Modulation and Complex White Gaussian Noise (AWGN) signals. The total number of samples used for training was 224,400, while 74,800 samples were allocated for testing and validation.

The eleven selected classes can be grouped according to their intrapulse modulation mechanism. Linear Frequency Modulation (LFM) applies a linear instantaneous frequency sweep across the pulse, producing a quadratic phase law. Frequency Modulation with Costas codes employs a permutation of discrete frequency hops with near-ideal thumbtack ambiguity properties. Binary Phase-Shift Keying (2PSK) and Barker-coded signals are binary phase codes, in which the carrier phase switches between 0 and π according to a predefined sequence. The polyphase family (Frank, P1, P2, P3, and P4) comprises codes whose phase increments take multiple discrete values: Frank, P1, and P2 are obtained by sampling a stepped-frequency waveform, while P3 and P4 are derived from a discretized linear frequency chirp [3], [9]. The Non-Modulated (NM) class corresponds to a constant-frequency pulse, and the Noise class consists of complex additive white Gaussian noise.

This grouping is directly relevant to the results reported in Section IV. All polyphase codes share a constant envelope and comparable occupied bandwidth, and all approximate — to different degrees — the phase law of an LFM waveform. Their mutual discrimination therefore depends almost exclusively on the fine structure of the phase-transition sequence, which is the feature most strongly degraded by additive noise.

B. Multilayer Perceptron

The Multi-Layer Perceptron (MLP) is a neural network composed of fully connected (dense) layers. It is one of the simplest deep learning architectures and is commonly used when feature extraction has already been performed [7]. In the MLP, each layer performs a linear transformation followed by a nonlinear activation function. The hidden representation of each fully connected layer is computed by applying a linear transformation followed by a nonlinear activation function:

$$\mathbf{h}^{(l)} = f\left(\mathbf{W}^{(l)}\mathbf{h}^{(l-1)} + \mathbf{b}^{(l)}\right), \tag{1}$$

where $\mathbf{W}^{(l)}$ and $\mathbf{b}^{(l)}$ denote the learnable weight matrix and bias vector of the l-th layer, respectively. The term $\mathbf{W}^{(l)}\mathbf{h}^{(l-1)}$ represents the weighted combination of the inputs from the previous layer, while $\mathbf{b}^{(l)}$ provides a learnable offset that adjusts the activation response. The input representation is defined as $\mathbf{h}^{(0)} = \mathbf{x}$, where $\mathbf{x}$ corresponds to the Unified Feature Vector extracted from the intrapulse modulations of radar signal. The nonlinear activation function $f(\cdot)$ is implemented as the Rectified Linear Unit (ReLU) [10], defined as:

$$f(z) = \max(0,\, z). \tag{2}$$

The output layer consists of 11 neurons followed by a softmax activation function, which converts the network logits into a probability distribution over the predefined classes. The predicted modulation class is obtained by selecting the class with the highest posterior probability.

$$\hat{p}_k = \frac{\exp(z_k)}{\sum_{j=1}^{K}\exp(z_j)}, \quad k = 1, \ldots, K, \tag{3}$$

where z_k denotes the logit associated with class k, and $K = 11$ is the total number of modulation classes. The final predicted class is given by $\hat{y} = \arg\max_k \hat{p}_k$.

In radar signal classification, this architecture can be employed to address the automatic modulation classification (AMC) problem. As described in Subsection II-A, the dataset consists of raw in-phase and quadrature (I/Q) radar signal samples, each represented as a 1024 × 2 matrix.

In this work, a feature extraction stage is applied to the raw I/Q signals, generating a set of handcrafted descriptors. The proposed architecture is designed to learn both temporal and spectral characteristics of radar intrapulse modulation signals. In this context, four groups of features are extracted: amplitude variations, phase differences, spectral signatures, and correlation properties of the analyzed radar waveforms. These features are concatenated into a single feature vector and used as input to the MLP classifier. The output layer performs classification into one of 11 predefined modulation classes.

C. Feature Extraction Stage

1) Radar Signal Amplitude: The signal amplitude (i.e., the signal envelope) can be defined in terms of the in-phase (I) and quadrature (Q) components provided in the dataset as:

$$A(t) = |s(t)| = \sqrt{I(t)^2 + Q(t)^2}, \quad t = 0, \ldots, 1023. \tag{4}$$

This feature generates a 1024-dimensional vector representing the signal amplitude. It is worth noting that the amplitude remains nearly constant for phase-coded waveforms, whereas Linear Frequency Modulation (LFM) and Frequency Shift Keying (FSK) signals exhibit slight fluctuations, providing a coarse first-level discriminator [3].

2) Radar Signal Phase Difference: The instantaneous phase is given by $\phi(t) = \arg\{s(t)\}$. After unwrapping to remove 2π discontinuities, the first-order phase difference, is defined as:

$$\Delta\phi(t) = \phi_{\text{unwrap}}(t) - \phi_{\text{unwrap}}(t-1), \quad t = 1, \ldots, 1023. \tag{5}$$

This feature generates a 1023-dimensional vector representing the signal phase difference. It is worth noting that LFM signals exhibit a linearly varying $\Delta\phi$, Costas codes show a permuted constant pattern, and phase-coded signals (BPSK, Barker, Frank, P1–P4) exhibit characteristic step patterns whose exact sequence encodes the modulation type [3], [9].

3) Radar Signal Spectral Signature: The spectral signature can be defined in terms of the Normalized Power Spectrum, which is given by:

$$\hat{S}(k) = \frac{|\mathcal{F}\{s\}(k)|}{\max_k |\mathcal{F}\{s\}(k)| + \epsilon}, \quad k = 0, \ldots, 1023, \tag{6}$$

where $\mathcal{F}$ denotes the Discrete Fourier Transform (DFT) of the input signal and $\epsilon = 10^{-8}$ prevents division by zero. It is worth noting that the spectral shape is a classical discriminator for Linear Frequency Modulation (LFM), Frequency Shift Keying (FSK), and Phase Shift Keying (PSK) signals [3], [9], providing complementary information to the phase difference feature.

4) Radar Signal Correlation: The correlation feature is derived from the autocorrelation function of the complex baseband signal $s[n] = I[n] + jQ[n]$, $n = 0, \ldots, N-1$, with $N = 1024$, which characterizes the self-similarity structure of the waveform. Linear (aperiodic) autocorrelation is defined as

$$R[k] = \sum_{n=k}^{N-1} s[n]\, s^*[n-k], \quad k = 0, \ldots, L_{\text{ac}}, \tag{7}$$

where $s^*[n]$ denotes the complex conjugate of $s[n]$ and $L_{\text{ac}} = 64$. In practice, (7) is evaluated through the Wiener–Khinchin relation, $R = \mathcal{F}^{-1}\{|\mathcal{F}\{s\}|^2\}$, with the input zero-padded to $2N$ samples so that the circular correlation inherent to the DFT reduces to the linear one.

Since $R[k]$ is in general complex-valued while the MLP operates on real-valued inputs, the retained descriptor is the absolute value of the real part of $R[k]$, normalized by the zero-lag term:

$$\tilde{R}[k] = \frac{|\operatorname{Re}\{R[k]\}|}{R[0]}, \quad k = 1, \ldots, L_{\text{ac}}, \tag{8}$$

where $R[0] = \sum_n |s[n]|^2$ is the total signal energy, which is real and positive. The zero-lag term is excluded from the feature vector: the normalization in (8) forces $\tilde{R}[0] = 1$ for every sample, so this component would be constant across the entire dataset and provide no discriminative information. The resulting descriptor is a 64-dimensional real-valued vector.

The value $L_{\text{ac}} = 64$ was adopted to keep the correlation block compact in relation to the remaining feature groups, restricting it to approximately 2% of the total dimensionality of the feature vector so that short-lag correlation values do not dominate the input representation. Two limitations of this descriptor should be noted. First, the truncation restricts the analysis to the first 6.25% of the available lags, which may be insufficient to resolve the longer periodicity associated with the sub-pulse structure of polyphase codes. Second, retaining only the real part discards the phase of $R[k]$, which is itself part of the analytical signature distinguishing the polyphase families. Both aspects are plausible contributing factors to the confusion observed within this group and are left as future work.

5) Unified Feature Vector: The four groups of extracted features (amplitude variations, phase differences, spectral signatures, and correlation) are concatenated into a single feature vector as follows

$$\mathbf{x} = \left[A;\ \Delta\phi;\ \hat{S};\ \tilde{R}\right], \tag{9}$$

This Unified Feature Vector has a total dimensionality of 3,135, comprising 1,024 amplitude features, 1,023 phase difference features, 1,024 spectral features, and 64 autocorrelation features.

It is worth noting that, prior to training, each feature dimension is standardized using z-score normalization (mean and standard deviation computed exclusively on the training set), thereby preventing data leakage.

III. Experimental Setup

A. MLP Network Architecture

The proposed MLP architecture is designed to map the 3135-dimensional Unified Feature Vector into a probability distribution over 11 radar intrapulse modulation classes. The network processes the input as a fixed-length feature representation, where the discriminative information is embedded in the extracted amplitude, phase difference, spectral signature, and autocorrelation features.

Based on this design, Table I summarizes the proposed MLP architecture in terms of layer-wise output dimensions, parameter counts, and overall model complexity. The network comprises three hidden layers with 512, 256, and 128 neurons, containing 1,606,656, 131,840, and 33,152 trainable parameters, respectively. The output layer consists of 11 neurons with 1,419 trainable parameters, corresponding to the final modulation classification stage. Each hidden layer is composed of a fully connected block (FC) followed by a ReLU activation, Batch Normalization (BN), and Dropout with a rate of 0.3. The complete model contains approximately 1.77 million trainable parameters.

TABLE I
Proposed MLP Architecture for Intrapulse Modulation Classification in RADAR Signals

Layer	Output Dim.	Parameters
Input (Unified Feature Vector)	3135	—
FC_1 + ReLU + BN + Dropout(0.3)	512	1,606,656
FC_2 + ReLU + BN + Dropout(0.3)	256	131,840
FC_3 + ReLU + BN + Dropout(0.3)	128	33,152
Output (Softmax, 11 classes)	11	1,419
Total trainable	—	**1,773,067**

B. MLP Regularization

Each hidden layer is followed by Batch Normalization [11] and Dropout [12] to improve training stability and generalization performance. Batch Normalization standardizes the pre-activation values within each mini-batch, reducing internal covariate shift, improving numerical stability, and accelerating convergence during optimization, while also reducing sensitivity to weight initialization [11]. Dropout randomly deactivates neurons with probability $p_{\text{drop}} = 0.3$ during training, acting as an implicit ensemble and preventing co-adaptation of hidden units [12].

The network is trained by minimizing the categorical cross-entropy loss [10], which quantifies the divergence between the predicted probability distribution and the ground-truth class labels. This objective encourages the model to maximize the

likelihood of the correct modulation class while penalizing incorrect predictions across all training samples.

C. Training Procedure

The proposed model is implemented in Python using TensorFlow 2.12. The optimization process employs the Adam optimizer combined with a Cyclical Learning Rate (CLR) schedule, following a triangular pattern between $\eta_{\min} = 10^{-7}$ and $\eta_{\max} = 10^{-3}$ with a half-cycle step size of 10 epochs.

To prevent overfitting, early stopping is applied with a patience of 15 epochs, monitoring validation accuracy and restoring the best-performing model weights.

Balanced class weights, computed as inversely proportional to class frequency in the training set, were applied to the loss function. Training was performed with a batch size of 256 for a maximum of 100 epochs, with a fixed random seed. Accuracy is defined as the fraction of correctly classified samples in the test set, and the reported results correspond to a single training run.

IV. Results

Fig. 1 presents the classification performance of the proposed MLP as a function of the signal-to-noise ratio (SNR). The results indicate that the model achieves 90% accuracy at 8 dB.

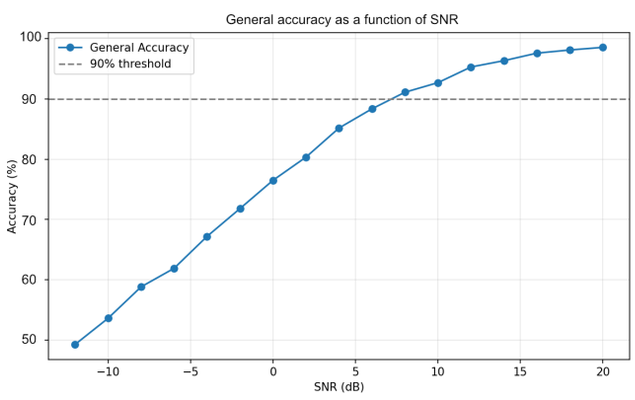

Fig. 1. General accuracy as a function of SNR for the proposed MLP on the 11-class DeepRadar2022 test set.

Fig. 2 shows the accuracy-versus-SNR profiles for all eleven modulation classes. The results reveal two clearly separated groups: a fast-rising converging group (NM, Noise, PM Barker, FM Costas, LFM, and 2PSK), which achieves saturation above 90% accuracy near 0 dB, and a slow-rising converging group composed of polyphase codes (PM P3, PM Frank, PM P4, PM P1, and PM P2), whose performance remains limited until approximately 6–16 dB.

Fig. 3 presents these results as a confusion matrix. The upper-left block (NM, PM Barker, FM Costas, Noise, LFM, 2PSK) is strongly diagonal, confirming near-perfect discrimination among structurally distinct waveforms. The lower-right block reveals dense off-diagonal structure among the polyphase codes.

Table II presents the per-class precision, recall, and F1-score for all modulation classes in the test set. The model achieves an overall accuracy of 80.15%, with macro-average precision, recall, and F1-score of 0.804, 0.802, and 0.801,

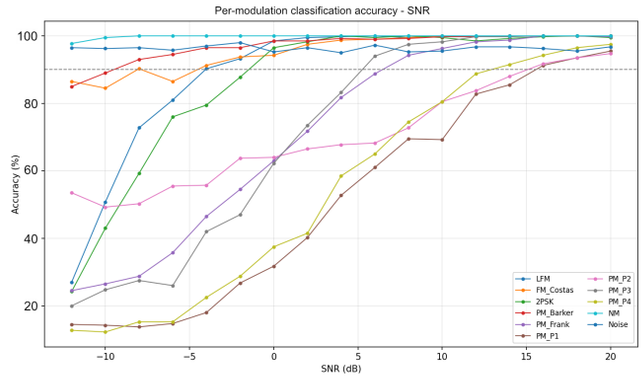

Fig. 2. Per-modulation classification accuracy as a function of SNR.

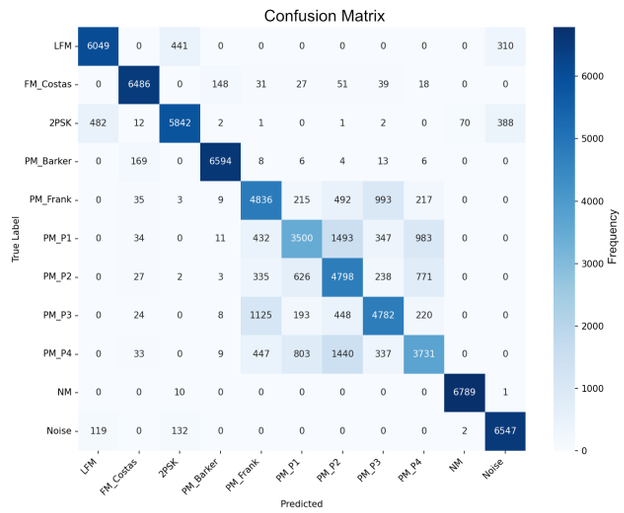

Fig. 3. Confusion matrix showing the misclassification errors of the proposed architecture.

respectively. Although the macro-average metrics and overall accuracy of 80.15% indicate stable global performance, the results suggest that class-level variability may still affect the robustness of the classifier.

TABLE II
PER-CLASS CLASSIFICATION REPORT (TEST SET, OVERALL ACC. = 80.15%)

Class	Precision	Recall	F1-score
NM	0.990	0.998	0.994
PM_Barker	0.972	0.970	0.971
FM_Costas	0.951	0.954	0.952
Noise (CWGN)	0.904	0.963	0.932
LFM	0.910	0.890	0.900
2PSK	0.909	0.859	0.883
PM_P3	0.708	0.703	0.706
PM_Frank	0.670	0.711	0.690
PM_P2	0.550	0.706	0.618
PM_P4	0.628	0.549	0.585
PM_P1	0.652	0.515	0.575
Macro avg	0.804	0.802	0.801

A. Discussion

The proposed model reaches the 90% performance threshold at approximately 8 dB, which is around 6 dB higher than the CNN-based models of Abissamra et al. (1.50 and 2.30 dB) and about 10.7 dB higher than their LSTM-based models (−2.58 and −2.66 dB). This performance gap highlights the reduced robustness of the proposed architecture in low-SNR regimes when compared to sequence-aware models. In particular, below −4 dB SNR, the accuracy drops sharply, mainly due to increased confusion among polyphase codes, which are more sensitive to noise and exhibit higher intra-class similarity.

The full accuracy-versus-SNR profile across the eleven modulation classes reveals sigmoid-shaped learning curves, anchored at the SNR thresholds. The results clearly indicate two distinct performance regimes. The first group, comprised by NM, Noise, PM-Barker, FM-Costas, LFM, and 2PSK, exhibits fast-rising curves that rapidly saturate above 90% accuracy by approximately 0 dB, indicating strong robustness to noise and effective feature separability. In contrast, the second group, formed by the polyphase codes (PM-P3, PM-Frank, PM-P4, PM-P1, and PM-P2), shows significantly slower convergence, with performance remaining suppressed until approximately 6–16 dB, reflecting higher structural similarity and increased susceptibility to noise-induced confusion.

The confusion matrix highlights a clear block structure in the classification performance. The five polyphase codes (PM P1, PM P2, PM P3, PM P4, and PM Frank) constitute the most challenging group, with F1-scores ranging from 0.575 to 0.706. A plausible explanation is that these signals share nearly identical instantaneous amplitude statistics, since all are phase-coded with constant envelope, as well as comparable occupied bandwidths. Under this hypothesis, their discrimination would rest almost exclusively on the phase-difference and correlation feature groups. This interpretation is consistent with the observed confusion pattern, but it was not verified experimentally: an ablation study isolating the contribution of each feature group would be required to confirm it.

V. Conclusion

This work evaluated a Multilayer Perceptron neural network for intrapulse modulation classification in radar signals operating in dense electromagnetic environments with multiple active emitters. It focuses on waveform-derived feature extraction for classification in pulsed radar systems, particularly considering intrapulse modulation characteristics.

The dataset used in this work was the DeepRadar2022 dataset, from which a reduced subset comprising 11 classes was employed. A feature extraction stage was applied to the raw I/Q signals, generating a set of handcrafted descriptors. The proposed MLP architecture is designed to learn both temporal and spectral characteristics of radar intrapulse modulation signals. The MLP architecture consists of three hidden layers with 512, 256, and 128 neurons.

Although the proposed MLP architecture achieves lower performance than the LSTM and CNN models of Abissamra et al. [5], in terms of both overall accuracy and per-modulation classification, it relies on a structurally simpler feedforward topology, with no recurrent or convolutional operations and a total of approximately 1.77 million trainable parameters. It should be noted, however, that no computational cost measurements were carried out in this work: the comparison is restricted to architectural structure, and no claim of computational advantage is made. A quantitative assessment of MACs, inference latency, memory footprint, and training time, under identical hardware and implementation conditions and accounting for the cost of the feature extraction stage, remains as future work.

References

[1] W. Wei, R. Guo, and Z. Chen, "Radar measurement perception against dense deceptive jamming: A unified transformer for classification, association, and tracking," *IEEE Transactions on Radar Systems*, vol. 4, pp. 1000–1015, 2026.

[2] K. K. Deshmukh and H. D. Gadade, "Real-time radar target tracking and classification using machine learning," in *Proc. Int. Conf. Electron. Comput. Commun. Netw. Autom. Technol. (ICEC2NT)*. Pune, India: IEEE, Sep. 2025.

[3] J. Lundén and V. Koivunen, "Automatic radar waveform recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 1, pp. 124–136, Jun. 2007.

[4] P. Sharma, K. K. Sarma, and N. E. Mastorakis, "Artificial intelligence aided electronic warfare systems—Recent trends and evolving applications," *IEEE Access*, vol. 8, pp. 224 761–224 780, 2020.

[5] P. D. F. Abissamra, S. Leite, R. Machado, and D. I. Alves, "Deep learning models applied in automatic modulation classification of radar signals," in *Anais do XLIII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais*, ser. SBrT2025. Sociedade Brasileira de Telecomunicações, 2025. [Online]. Available: http://dx.doi.org/10.14209/sbrt.2025.1571144222

[6] V. Clerico and J. Grajal, "DeepRadar2022," 2023. [Online]. Available: https://www.kaggle.com/dsv/4930249

[7] O. Nazarevych, R. Peleshchak, V. Lytvyn, I. Peleshchak, and V. Motyka, "Recognition of low probability of intercept radar signals in a polar coordinate system based on a hybrid CNN and MLP," in *Proc. 13th IEEE Int. Conf. Intell. Data Acquisition Adv. Comput. Syst. (IDAACS)*. Gliwice, Poland: IEEE, Sep. 2025, pp. 339–345.

[8] V. Clerico, J. González-López, G. Agam, and J. Grajal, "LSTM framework for classification of radar and communications signals," in *2023 IEEE Radar Conference (RadarConf23)*. IEEE, May 2023, pp. 1–6. [Online]. Available: http://dx.doi.org/10.1109/RadarConf2351548.2023.10149618

[9] S.-H. Kong, M. Kim, L. M. Hoang, and E. Kim, "Automatic LPI radar waveform recognition using CNN," *IEEE Access*, vol. 6, pp. 4207–4219, 2018.

[10] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org

[11] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn. (ICML)*. Lille, France: PMLR, Jul. 2015, pp. 448–456.

[12] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.