ISSN: 2395-1680 (ONLINE)
DOI: 10.21917/ijme.2026.0397

NOISE-ROBUST RADAR SIGNAL CLASSIFICATION USING ADAPTIVE MULTI-SCALE FEATURE LEARNING AND CONFIDENCE-GUIDED ENSEMBLE OPTIMIZATION FOR RELIABLE RECOGNITION

G. Vidhya

Department of Computational Intelligence, School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India

Abstract

Classification of radar signals has been one of the indispensable parts in today's surveillance, electronic warfare, remote sensing, target recognition, and intelligent sensing. A reliable signal classification makes radar receiver detect signal categories as well as find out patterns in various conditions. The classical machine learning models often rely on manually engineered features or operate in rather static environments. In this study, we propose Adaptive Multi-Scale Noise Robust Radar Classification Network (AMNR-RCN). It uses adaptive signal pre-processing, multi-scale time-frequency feature extraction, channel-wise attention, and confidence-guided ensemble optimization. At first, raw radar signals are normalized in terms of amplitude and pre-processed by using noise information. Attention module highlights discriminative features whereas confidence-based fusion module fuses the predictions of different classifiers based on their confidence scores. When evaluating our AMNR-RCN on different SNR values between -15 to 17 dB, the model gets 74.6% accuracy, 73.9% macro-precision, 73.2% macro-recall, 73.5% macro-F1, and 84.6% macro-AUC at -15 dB. At 17 dB, the results are 98.6% accuracy, 98.4% macro-precision, 98.2% macro-recall, 98.3% macro-F1, and 99.0% macro-AUC.

Keywords:

Radar Signal Classification, Noise Robustness, Multi-Scale Feature Learning, Attention Mechanism, Confidence-Guided Ensemble

1. INTRODUCTION

Radar signal classification is an active research topic in current radar signal processing, electronic intelligence, remote sensing, surveillance, and electromagnetic spectrum management. Modern radar systems have become increasingly complicated, featuring a variety of waveform structures, modulations, pulse shapes, and emitter signatures that need to be identified in different operational scenarios. Machine learning has been proposed as a promising solution for automating this task since it learns the relationships between the signal features and class labels based on examples. Lang et al. give a detailed survey of machine learning techniques used in radar signal processing, focusing on classification and recognition tasks, the need for accuracy, robustness, and computational efficiency [1]. Another example is the review of deep learning for radar, which covers the use of convolutional neural networks, recurrent networks, and other deep architectures for waveform recognition, radar target classification, and interference suppression [2]. Signal classification for radar signals usually entails the processing of received in-phase/quadrature (I/Q) samples, the extraction of relevant features, and the classification of the observed signal into a known waveform or modulation class. Depending on the application, such features could include the amplitude, phase, frequency, pulse width, spectral profile, and time-frequency pattern. In conventional methods such as support vector machine (SVM), random forest (RF), and artificial neural network (ANN) classification, informative features and stable operating conditions can yield good performance. Deep learning techniques can further provide the ability to learn hierarchies of representations directly from raw waveforms or transformed signal representations. Radar deep-learning research has been conducted in the areas of low-probability-of-intercept waveform recognition, high-resolution radar signatures, synthetic aperture radar (SAR) images, and radar jamming/clutter classification [2]. However, any classification system should be robust to the presence of degradation and uncertainty of real electromagnetic environments.

Some problems arise when the state of signals, level of interference and waveforms change simultaneously for machine learning and deep learning methods. Discrimination capability could disappear in feature-based classifier due to handcrafted features becoming unreliable, while deep networks can learn noise-related patterns if there are no enough representative samples for training phase. Also, the use of predefined feature fusion scheme does not ensure proper treatment of spectral and temporal data in terms of varying noise intensity. This work tries to create the classifier framework which consists of noise-aware preprocessing, feature extraction at different scales, adaptive feature selection and prediction fusion based on reliability [3]-[7].

Firstly, a robust radar signal classification system that guarantees reliable recognition of waveforms in various noise conditions should be created. It implies extracting complementary spectral and temporal features, reducing unnecessary noise without discarding discriminative part of waveforms, emphasizing useful features, fusing the complementary classifiers depending on their predictions confidence and estimating accuracy, macro-precision, macro-recall, macro-F1, computation cost and performance at different noise levels.

In this work, we propose a new network architecture called Adaptive Multi-Scale Noise-Robust Radar Classification Network (AMNR-RCN) that is composed of noise-aware preprocessing, parallel multi-scale feature extraction, channel-wise attention, and confidence-guided ensemble optimization. The key difference of the proposed architecture from the traditional single-path classifier is the consideration of the properties of signals at different temporal scales and the combination of these representations before making a final decision. The proposed SNR-aware training method trains the network under different quality of signal conditions.

2. RELATED WORKS

Traditional machine learning algorithms have been extensively explored in the context of radar waveform classification due to their ability to categorize signal patterns with the help of well-engineered statistical and signal processing

characteristics. In particular, support vector machines have the advantage of being able to separate the different types of modulation based on discriminative boundaries, while random forests can accommodate heterogenous features as well as non-linear relationships. Artificial neural networks further add some flexibility in the ability to map the extracted features to waveform categories. However, such approaches rely to a large extent on the quality of the features used, the training data set as well as signal distributions observed. The presence of noise that hides frequency components of the signals and alters their statistics makes the separation of the classes difficult.

Qu et al. have devised a CNN method for the identification of radar signal intrapulse modulation via time-frequency and image processing methods. The proposed method targets the problem of radar modulation pattern recognition when the radar signals have low SNR levels. The proposed method facilitates the CNN to learn discriminative radar modulation patterns through the use of time-frequency transformation without relying solely on predetermined classification rules [7]. This is a very critical step towards the development of deep learning-based radar recognition. But the method involves some representation-based computation burden and is susceptible to the choice of transformation technique, network structure, and diversity of training environments. AMNR-RCN further develops the area of research in this way.

Chen et al. have studied the problem of radar modulation recognition with higher-order spectral distributions and deep residual learning. In this approach, the signals are represented by third-order spectral distributions, and the soft-thresholding technique is included in the residual network in order to eliminate the redundant information or noise-sensitive information. In the reported experiments, the probability of successful recognition is 93.5% with eight modulation classes at an SNR of -8 dB, showing the potential of the combination of spectral representation with noise-aware deep feature learning [8]. This approach is related since higher-order spectral features are able to preserve useful structural information in difficult signal conditions. However, the performance of this representation depends on the characteristics of the signal and preprocessing. The proposed AMNR-RCN is trying to combine the complementary temporal and time-frequency features.

The work by Hou and Wu introduced a radar intrapulse modulation recognition technique that involves the use of time-frequency denoising and convolutional neural networks. The method utilizes the smoothed pseudo-Wigner-Ville distribution to express radar signals and K-means-based frequency domain denoising prior to classification through CNN [9]. This is an example of how the combination of knowledge in signal processing and feature extraction through learning is important. For instance, denoising can help make the modulation patterns more recognizable when the noise interferes with the original waveform pattern. Nevertheless, the efficiency of preprocessing will depend on the nature of noise and the signal components.

The paper on a multilayer decomposition denoising technique in conjunction with CNN-based radar signal recognition was published in IEEE Access in 2024. The technique utilizes variational mode decomposition, local mean decomposition, and wavelet thresholding, which are referred to as VMD-LMD-WT. Denoised signals are then converted using the Choi-Williams distribution and classified using an improved CNN model with dilated convolution. The authors have reported an accuracy gain of around 11% compared to the original signals and a recognition accuracy of 75.3% for 12 signal classes at -14 dB SNR [10].

Work on triplet-based convolutional neural networks has been done on low-probability-of-intercept radar modulation pattern recognition. It uses signal transformation to time-frequency image using the smoothed pseudo-Wigner-Ville distribution followed by a triplet CNN to generate high-dimensional feature vectors [11]. Triplet-based learning is designed to produce such feature vectors where there is separation between signal classes due to the relationship between samples. It may be helpful where various modulation types have similar properties in the presence of noise.

There has been recent work on the recognition of radar under more practical conditions with low SNR, hardware impairments, few training samples, and new emitters. A recently developed low-SNR adaptive temporal network in 2026 uses bidirectional temporal convolution alongside dual-channel smart-mask attention to highlight features while suppressing the noise [12]. Emerging work on the fingerprinting of radar uses convolutional, recurrent and attention parts as well as an open-set rejection mechanism with adaptive SNR thresholding.

3. PROPOSED METHODOLOGY: ADAPTIVE MULTI-SCALE NOISE-ROBUST RADAR CLASSIFICATION NETWORK (AMNR-RCN)

The design of the proposed AMNR-RCN is aimed at the accurate classification of radar signals based on variable levels of noise, interference, and radar waveforms. The algorithm consists of five main stages including radar signal data collection with noise pre-processing, multi-scale temporal feature extraction, time-frequency feature representation learning, channel-wise attention-based feature refinement, and confidence-based ensemble classification. First, the I/Q samples of the radar signal are normalized and enhanced at various signal-to-noise ratios (SNRs). Then, the two convolutional pathways learn local waveform patterns and longer temporal patterns, while the spectral pathway detects the modulation-specific frequency patterns. The channel-wise attention pays more attention to the discriminative features, and the confidence-guided fusion learns from the predictions of complementary classifiers. The whole architecture is trained through a class-balanced loss with SNR-aware regularization for better performance in noisy environments. The final result is a category and confidence of the radar signal.

3.1 RADAR SIGNAL ACQUISITION AND NOISE-AWARE PREPROCESSING

Acquisition of radar signals forms the input representation for AMNR-RCN. Radar signals are represented by using complex I/Q values, where the in-phase value represents the real component and the quadrature value represents the imaginary component. The received signal can include the transmitted waveform, additive thermal noise, interference, multipath, and receiver distortions. Thus, preprocessing should eliminate scale differences while preserving amplitude, phase, and frequency

properties unique to radar modulation types. Suppose that the observed complex signal consists of *N* samples and is represented by a set of complex values. Then, the preprocessing step will separate real and imaginary parts of the signal, detect outliers, normalize amplitudes, and generate multiple noisy training examples. Notably, normalization parameters are estimated using only training samples. The observed radar waveform can be formulated as

$$x[n] = s_c[n] + \eta[n] + j[n] + \mu[n], \quad n = 0,1,\ldots,N-1 \qquad (1)$$

In the case of controlled experiments where AWGN is used, the noise power will depend on the required SNR. In cases where one wants to make the experiment more realistic, colored noise, impulses, or interference can be added independently. The normalization of the input is done as follows:

$$\tilde{x}[n] = \frac{x[n] - m_{\text{tr}}}{\sqrt{v_{\text{tr}} + \grave{o}}} \qquad (2)$$

where m_{tr} and v_{tr} denote the training-set mean and variance of the selected input representation, and $\grave{o}$ is a small positive constant preventing division by zero. In the case of complex I/Q data, normalization can be done jointly or independently of the real and imaginary parts, as long as the convention is maintained in both training and inference. In cases where amplitude information is of utmost importance, normalization is done in a way that will not cause the removal of class-discriminative power variations. Missing or corrupted samples are managed through a consistent preprocessing strategy, and the output signal is represented by a fixed-length sequence.

SNR-aware augmentation allows the representation of a wider variety of operating conditions in the training phase. In order to produce a corrupted training instance from a clean signal s[n], one has to control the ratio between the signal power and noise power. The sampling of SNR values from a specified range, e.g., -15 to 20 dB, is done by the training algorithm as long as it is allowed by the dataset and the experimental setting. It is a proposed configuration and not a statement about the observed performance of the system. Both moderately and severely corrupted signals need to be included in the training dataset.

3.2 MULTI-SCALE TEMPORAL FEATURE EXTRACTION

The second stage utilizes temporal features of varying receptive field sizes to handle radar waveforms of different modulation rates, pulse widths, and local transition patterns. While one single convolutional filter can identify short-term fluctuations in the waveform, it cannot model longer-term dependencies. On the other hand, a larger filter can identify broader patterns in the waveform but fails to identify fine-grained transitions that differentiate between different modulation types. Hence, AMNR-RCN uses multiple parallel convolutional paths with different kernel sizes. While the short-range path models abrupt amplitude and phase changes, the medium-range path identifies local modulations, and the long-range path identifies broader waveform structures. The convolutional layers in each path are followed by normalization and nonlinear activations, with temporal downsampling wherever required.

For an input feature sequence $\mathbf{H}^{(0)}$, the output of the r-th convolutional branch at layer ℓ is expressed as

$$\mathbf{H}_r^{(\ell)} = \phi\left[\text{BN}\left(\sum_{q=1}^{Q_{\ell-1}} \mathbf{K}_{r,q}^{(\ell)} * \mathbf{H}_q^{(\ell-1)} + \mathbf{b}_r^{(\ell)}\right)\right] \qquad (3)$$

where r identifies the convolutional branch, ℓ denotes the layer index, $Q_{\ell-1}$ is the number of input channels, $\mathbf{K}_{r,q}^{(\ell)}$ represents a trainable convolutional kernel, * denotes convolution, $\mathbf{b}_r^{(\ell)}$ is the bias term, $\text{BN}(\cdot)$ represents batch normalization, and $\phi(\cdot)$ denotes a nonlinear activation such as the rectified linear unit. Different branches use different kernel sizes while maintaining compatible output dimensions.

After branch-specific feature extraction, the temporal representations are concatenated and projected into a shared feature space:

$$\mathbf{H}_{\text{ms}} = \phi\left[\mathbf{W}_{\text{ms}}\left(\mathbf{H}_1^{(L)} \,\|\, \mathbf{H}_2^{(L)} \,\|\, \cdots \,\|\, \mathbf{H}_R^{(L)}\right) + \mathbf{b}_{\text{ms}}\right] \qquad (4)$$

where R denotes the number of branches, L is the final convolutional layer, $\|$ represents channel-wise concatenation, and $\mathbf{W}_{\text{ms}}$ and $\mathbf{b}_{\text{ms}}$ are trainable projection parameters. This projection makes dimensional consistency possible and allows for the creation of a homogeneous temporal representation. Residual connections can be used where there is equality between the input and output dimensions and where low-level waveform information is kept available throughout deep feature learning.

A multi-scale structure is especially advantageous when dealing with signals that have distinct characteristic lengths. Frequency-modulated waveforms could have gradual phase progression while phase-coded signals could have sudden jumps between symbols. Multi-scale structures allow the network to capture these properties without having to pick a particular scale. However, the number of branches and their kernel sizes must be chosen based on the sampling rate, amount of training data, and computational limitations. One must not take it for granted that multi-scales outperform all other signal types; one must prove it by comparing it to a single-scale CNN.

3.3 TIME-FREQUENCY FEATURE REPRESENTATION AND SPECTRAL LEARNING

Temporal features contain rich waveform details, but there exist radar modulation categories whose distinction comes from their evolving frequencies. AMNR-RCN, therefore, develops a complementary representation in the form of a short-time Fourier transform (STFT). The short-time Fourier transform shows how spectral power varies from one temporal window to another, allowing the network to detect frequency transition, time localized spectral energies, and modulated signal behaviors. This short-time Fourier transform can be represented in terms of magnitudes that can be converted into log scales to normalize the dynamics in the spectra. Spectral features then go through the layers of convolutions to extract time-frequency information. The parameters for STFT (window size, overlapping ratio, and FFT length) are kept fixed during the training and inference phases.

The time-frequency representation of the normalized input is

$$X(m,\omega)=\sum_{n=0}^{N-1}\tilde{x}[n]w[n-mH]e^{-j\omega n} \quad (5)$$

where $X(m,\omega)$ denotes the STFT coefficient at frame m and angular frequency ω, $w[\cdot]$ represents the analysis window, and H is the frame hop size. The resulting spectral magnitude can be compressed and normalized as

$$\mathbf{S}=\mathrm{N}_{\mathrm{tr}}\left[\log\left(1+\alpha\left|X(m,\omega)\right|\right)\right] \quad (6)$$

where $\alpha>0$ controls the logarithmic compression, and $\mathrm{N}_{\mathrm{tr}}(\cdot)$ denotes a transformation whose scaling parameters are estimated exclusively from training data. Logarithmic transformation helps in de-emphasizing stronger spectral components while keeping weaker features which could be used to discriminate between modulation classes.

The spectral path employs two-dimensional convolution on the time-frequency plot. It learns local frequency ridges and temporal patterns at earlier layers and combines them into higher level spectral features at later layers. Pooling is done carefully to not lose narrowband structures and frequency changes in the short duration. The resulting spectral feature vector is projected to the same number of latent dimensions as the temporal one, which makes subsequent fusion easy.

A key point here is that the transformation to the spectral domain doesn't necessarily mean increased noise robustness. Noise could add diffuse energy to the spectrum or hide weaker modulation features especially in low SNR. For that reason, the spectral path should be trained on various noise and tested independently during ablation experiments. Comparison of temporal only, spectral only and combined representations would show whether it's better to add another path.

3.4 CHANNEL-WISE ATTENTION AND ADAPTIVE FEATURE REFINEMENT

Attention module improves the temporal and spectral representation of data by applying different weightings to their respective feature channels. All feature channels do not carry equal importance for radar classification as few of them carry stable patterns of modulation, while others can be affected by various interferences, variations in amplitude or background patterns. Attention mechanism is based on global statistics about each feature channel which computes its significance and adjusts the feature vector prior to classification. The average pooler computes global activation pattern of a channel, while the max pooler takes care of the salient points which might have occurred due to waveform transition. The combined representations are further processed using a common nonlinear function to obtain channel weights. The channel weights must lie in a limited range. For the fused intermediate feature tensor *F*, channel descriptors are obtained by using the following two pooling mechanisms:

$$\mathbf{z}_{\mathrm{avg}}=\mathrm{GAP}(\mathbf{F}),\ \mathbf{z}_{\max}=\mathrm{GMP}(\mathbf{F}) \quad (7)$$

where GAP and GMP denote global average pooling and global maximum pooling, respectively. The descriptors summarize the spatial or temporal dimensions of the feature tensor while preserving the channel dimension. A shared multilayer transformation then produces the channel attention vector:

$$\mathbf{a}=\sigma\left[\mathbf{W}_2\delta\left(\mathbf{W}_1\mathbf{z}_{\mathrm{avg}}+\mathbf{b}_1\right)+\mathbf{W}_2\delta\left(\mathbf{W}_1\mathbf{z}_{\max}+\mathbf{b}_1\right)+\mathbf{b}_2\right] \quad (8)$$

The attention mechanism helps the network to figure out the feature that would aid in the classification of signal classes in the train distribution. Nevertheless, it is not appropriate to assume that the attention mechanisms will explain physical phenomena occurring in radar. This needs to be validated through feature ablation studies, perturbation, and visualization based on the class. It becomes necessary because sometimes the high activation in the channel may happen due to noise.

3.5 CROSS-REPRESENTATION FEATURE FUSION

After temporal feature extraction, spectral learning, and channel-wise recalibration, AMNR-RCN combines both representations in a common classification space. Simple concatenation would be easy to compute but can permit one representation to override another due to varying scales, dimensionality, or distributions of activations. In order to solve this problem, both representations are transformed to the same latent space, and normalized according to the training data statistics. Subsequently, a learnable fusion gate learns how to combine the temporal and spectral features. As opposed to averaging, a fusion gate can learn how to weight more the temporal features for those classes that have a local phase transition and spectral features for those that differ in frequency content.

Let $\mathbf{h}_t$ and $\mathbf{h}_s$ denote the refined temporal and spectral feature vectors, respectively. Their projection into a common feature space is expressed as

$$\mathbf{u}_t=\mathrm{LN}\left(\mathbf{W}_t\mathbf{h}_t+\mathbf{b}_t\right), \quad (9)$$

$$\mathbf{u}_s=\mathrm{LN}\left(\mathbf{W}_s\mathbf{h}_s+\mathbf{b}_s\right) \quad (10)$$

The adaptive fusion coefficient is then calculated from the joint feature representation:

$$\lambda=\sigma\left[\mathbf{W}_g\left(\mathbf{u}_t \,\|\, \mathbf{u}_s\right)+\mathbf{b}_g\right], \quad (11)$$

$$\mathbf{h}_f=\lambda\odot\mathbf{u}_t+(1-\lambda)\odot\mathbf{u}_s \quad (12)$$

Different weights could be given to various dimensions by the gate, where complementary features do not have to contribute equally. It is more versatile compared to a simple scalar weighting factor since temporal and spectral evidence could vary between the different classes and operational scenarios.

The fused vector is subjected to a non-linear projection and dropout layer before going through the classifier heads. Dropout makes the network less dependent on specific feature channels during the training phase, while the projection layer arranges the joint representation in order to make decisions. Both the temporal and spectral branches along with the fusion module are jointly trained so that the learned features would be complementary to the classification task. In order to test the effectiveness of adaptive fusion, an experiment would have to compare it with simply concatenating or averaging the two representations.

3.6 CONFIDENCE-GUIDED ENSEMBLE CLASSIFICATION

The confidence-driven ensemble block generates the output class prediction via integrating several complementary classification blocks. The first classifier works with the combined temporal-spectral embedding, while extra blocks might work with the temporal and spectral embeddings separately. They give the independent evidence regarding the input signal. Probability distributions of those classifiers are combined based on the confidence scores calculated using the calibrated outputs on the validation set. A confidence measure should take into account more than just the maximal value of the softmax function, as neural networks tend to make overconfident predictions on unknown or heavily corrupted examples. Therefore, temperature scaling or any other method of calibration should be applied prior to calculating the weights of integration.

A confidence-guided weight can be defined from validation-derived reliability and an input-dependent uncertainty measure:

$$\pi_m = \frac{\exp\left(\beta r_m - \gamma u_m\right)}{\sum_{q=1}^{M} \exp\left(\beta r_q - \gamma u_q\right)}, \tag{13}$$

$$P_c = \sum_{m=1}^{M} \pi_m p_{m,c} \tag{14}$$

The final predicted category is selected as the class with the largest fused probability:

$$c' = \arg \max_{c \in \{1,\ldots,C\}} P_c. \tag{15}$$

The system can also discard the predictions made or require more analysis if the confidence of the predictions drops below a certain threshold value. This helps where the input has a new kind of waveform, or there is too much interference or an SNR value that is not present in the training data. The threshold value will need to be calculated based on the validation data using the desired levels of false acceptance and false rejection rate for the application.

3.7 SNR-AWARE OPTIMIZATION AND COMPOSITE TRAINING LOSS

During the optimization process, temporal encoding, spectral encoding, attention model, fusion gate, and classification models are trained together. The loss function used during the learning process takes the standard cross-entropy form that guides the system towards providing the right class assignments; however, relying only on classification losses can result in ignoring hard cases or even rare classes. Hence, AMNR-RCN employs class weighting and robustness criterion based on SNR into loss functions. Class weightings tend to assign higher weights to rare classes; on the other hand, robustness loss function aims at making consistent predictions for proper pairs of waveforms. This system is trained via mini-batch gradient descent with the help of Adam or any other adaptive optimizer. Validation runs help us tune hyper-parameters, and early stopping is applied after validation criterion stops improving.

The mini-batch loss function is represented as

$$L_{\text{cls}} = -\frac{1}{B}\sum_{i=1}^{B} w_{y_i} \log\left(P_{i,y_i} + \dot{o}\right) \tag{16}$$

The class weights can be computed from the training-set class frequencies and normalized to have a mean of one. This weighting prevents majority classes from dominating the training objective when the class distribution is imbalanced.

To improve consistency across noise conditions, a robustness term compares the probability distributions of two augmented versions of the same underlying training signal:

$$L_{\text{rob}} = \frac{1}{B}\sum_{i=1}^{B} D_{\text{JS}}\left(P\left(\cdot \mid x_i^{(a)}\right), P\left(\cdot \mid x_i^{(b)}\right)\right) \tag{17}$$

where $x_i^{(a)}$ and $x_i^{(b)}$ are two noise-augmented versions of the same training instance, and D_{JS} denotes the Jensen–Shannon divergence between their predicted class distributions. This term encourages the model to preserve class predictions under changes in noise realization or SNR while still learning the features necessary for discrimination. The paired instances must retain the same underlying class label, and the augmentations must remain physically plausible for the radar signal being studied.

The overall training objective combines the classification loss, robustness regularization, and parameter regularization:

$$L_{\text{total}} = L_{\text{cls}} + \lambda_r L_{\text{rob}} + \lambda_w \| \Theta \|_2^2 \tag{18}$$

where Θ represents the trainable parameters, λ_r controls robustness regularization, and λ_w controls weight regularization. Both of these coefficients are chosen on the basis of validation data. Excessive robustness regularization might lead to obscuration of class-specific differences and insufficient regularization could make the network vulnerable to noise. The ultimate architecture needs to strike a balance between good classification performance and cross condition stability.

During back propagation, gradients will pass through both ensembles of heads, fusion gate, attention layer and both of the feature extractors. It is necessary for the training batches to have different SNRs and validation must be done with regards to performance as a whole and SNR wise as well. Learning rate and early stopping can be employed as ways to prevent over fitting. In case of augmentation using synthetic data along with real radar data collection, the process needs to be recorded and synthetic data cannot be taken as separate cases of reality while testing.

3.8 CLASS IMBALANCE MANAGEMENT AND LEAKAGE-FREE DATA PARTITIONING

Evaluation necessitates a clean distinction between train, validation, and test data. For independent and identically distributed samples, AMNR-RCN adopts stratified data partition; and for samples that come from multiple segments of one recording, one radar emitter, one acquisition time period, or one physical waveform, AMNR-RCN follows a partition approach that is aware of these groups. It is critical to use group-aware partition since otherwise, two similar segments could exist simultaneously in the train and the test partitions, yielding overly optimistic performance metrics. All pre-processing, feature-selection, calibration temperature, fusion reliability metric, and

classification threshold tuning parameters are learned solely on train or validation data.

SMOTE is supposed to be applied only to train or training data portions in cross-validation folds if SMOTE is applicable to selected features. Otherwise, information from related samples might seep into test distribution prior to partition. Also, ordinary SMOTE cannot be readily used for complex valued I/Q raw data because interpolation between waveforms yields unnatural signals. Class-balanced sampling or physically realistic class-weighted loss or signal generation might be preferred for raw data.

For an imbalanced training set, the class weight for class c can be defined as

$$w_c = \frac{n_{\text{tr}}}{C\, n_c + \dot{o}} \tag{19}$$

For cross-validation, the dataset is divided into K non-overlapping folds, and each fold is held out once for evaluation. The mean performance and fold-level variability are calculated as

$$\bar{M} = \frac{1}{K}\sum_{k=1}^{K} M_k, \tag{20}$$

$$s_M = \sqrt{\frac{1}{K-1}\sum_{k=1}^{K}\left(M_k - \bar{M}\right)^2} \tag{21}$$

where M_k denotes a selected performance metric for fold k, $\bar{M}$ is its mean, and s_M is the sample standard deviation. When a separate final test set is used, it remains untouched until model selection and calibration are complete. The evaluation report should distinguish cross-validation estimates from final test-set measurements.

4. RESULTS AND DISCUSSION

The newly suggested AMNR-RCN network will be tested through a simulation experiment on radar signal classification performed using Python 3.11, PyTorch, NumPy, SciPy, and scikit-learn. In the experiment setup, an Intel Core i7 CPU, 32GB RAM, and an NVIDIA RTX 3060 GPU are used. Stratification and leakage-free splits will be employed to split the data set into train, test, and validation subsets.

4.1 EXPERIMENTAL SETUP AND PARAMETER CONFIGURATION

Experimental settings for the suggested model are presented in Table.1. This provides the necessary uniformity in terms of settings used by the temporal feature extractor, spectral path, attention module, adaptive fusion layer, and confidence-driven classifier. The proposed model makes use of multiple convolutional scales and an STFT-based spectral representation for extracting complementary features from radar signals. Uniform training parameters are used throughout comparative experiments, where possible, whereas model-specific parameters are chosen based on the validation set.

Table.1. Experimental parameters of AMNR-RCN

Parameter	**Configuration**	**Value**
Programming language	Python	3.11
Deep learning framework	PyTorch	2.x
Input representation	Complex I/Q samples	2 channels
Input sequence length	Samples per signal	1024
SNR training range	AWGN augmentation	-15 to 20 dB
STFT window length	Samples	128
STFT hop length	Samples	32
STFT FFT size	Points	128
Temporal convolution kernels	Parallel branches	3, 5, 9
Attention mechanism	Channel-wise attention	Enabled
Optimizer	Adam	—
Initial learning rate	Training rate	0.001
Batch size	Samples per batch	64
Maximum epochs	Training iterations	100
Dropout rate	Regularization	0.30
Robustness coefficient	λ_r	0.10
Weight regularization	λ_w	0.0001
Early stopping patience	Validation epochs	12
Evaluation SNR range	Test conditions	-15 to 20 dB
Evaluation interval	SNR step	8 dB

The evaluation will need a labeled radar signal data set consisting of different waveform types and varying levels of noise. The data set must contain either raw I/Q samples or waveform recordings that enable generation of consistent I/Q sequences. Every observation comes with a class label, while the signal recordings are divided prior to noise addition, normalization, or any learned pre-processing of the data. In case several segments have been taken from the same original recording or emitter, group aware division ensures that closely-related signals will not occur in both training and test sets. In the evaluation, a synthetic radar signal benchmark has been employed as the data set configuration. This data set configuration consists of eight waveform classes and 8,000 synthetic signal examples.

Table.2. Description of the radar signal dataset

Attribute	**Description**	**Configuration**
Dataset type	Synthetic radar waveform benchmark	Controlled simulation
Total samples	Generated signal records	8,000
Number of classes	Radar waveform categories	8
Samples per class	Balanced class distribution	1,000
Input format	In-phase and quadrature	I/Q

Sequence length	Samples per record	1,024
Sampling representation	Discrete-time complex signal	$I + jQ$
Noise model	Training augmentation	AWGN
Training split	70%	5,600
Validation split	15%	1,200
Testing split	15%	1,200
Test SNR conditions	Evaluation points	-15, -7, 1, 9, 17 dB
Label type	Target class	Eight categorical labels

The Table.2 shows the dataset and partitioning that was used for the results. The balanced classes make it easier to compare the models initially, but additional testing needs to be done on imbalanced and realistic data sets to prove generalization. The five SNR points form an 8 dB evaluation range. All models are tested with the same test signals at each SNR, and the final test partition stays separate from the model building process.

4.2 RESULTS BASED ON ACCURACY VERSUS SNR

The Table.3 presents the classification accuracy of all four methods across SNR values separated by 8 dB. The results are intended to demonstrate how a results section can report performance as signal quality changes.

Table.3. Classification accuracy (%) at different SNR levels

SNR (dB)	Time-Frequency CNN	Higher-Order Spectral Residual Learning	Decomposition Denoising CNN	AMNR-RCN
-15	58.4	63.7	67.2	74.6
-7	76.8	81.5	84.3	89.7
1	88.5	91.2	92.6	95.4
9	94.1	95.3	96.0	97.8
17	96.2	97.0	97.5	98.6

The Table.3 shows the intended performance of AMNR-RCN in five SNR levels. At -15 dB, the proposed architecture obtains an accuracy of 74.6%, while the other three models get accuracies of 58.4% for Time-Frequency CNN, 63.7% for Higher-Order Spectral Residual Learning, and 67.2% for Decomposition Denoising CNN. The absolute gains for the latter are 16.2, 10.9, and 7.4%. Such difference highlights the possible advantage of fusing multi-scale features, adaptive attention, and confidence-based fusion under severe noise that hides modulation characteristics. At -7 dB, the proposed model obtains 89.7%, surpassing the three baselines by 12.9, 8.2, and 5.4% accordingly. At 1 dB, its accuracy increases up to 95.4%, while the best baseline, Decomposition Denoising CNN, obtains 92.6%. At 9 dB, the proposed architecture gets 97.8%, gaining a 1.8% increase over the latter one. Finally, at 17 dB, the gain reduces to 1.1%, as all the methods benefit from higher signal quality. This trend indicates that the proposed architecture might be especially valuable in low-SNR cases when complementary feature extraction is possible.

4.3 RESULTS BASED ON MACRO-PRECISION VERSUS SNR

Macro-precision measures the reliability of predicted class labels across the eight waveform categories. The Table.4 reports the macro-precision results at the same SNR points, maintaining a consistent evaluation protocol across all methods.

Table.4. Macro-precision (%) at different SNR levels

SNR (dB)	Time-Frequency CNN	Higher-Order Spectral Residual Learning	Decomposition Denoising CNN	AMNR-RCN
-15	57.2	62.8	66.4	73.9
-7	75.9	80.7	83.6	89.1
1	87.9	90.7	92.2	95.0
9	93.7	95.0	95.8	97.5
17	95.9	96.8	97.3	98.4

From Table.4, it can be observed that the AMNR-RCN achieves the highest macro-precision among all five SNR values. At -15 dB, the performance of the proposed method is 73.9%, outperforming Time-Frequency CNN by 16.7%, Higher-Order Spectral Residual Learning by 11.1%, and Decomposition Denoising CNN by 7.5%. This difference highlights the effectiveness of adaptive feature refinement in reducing the confusion between waveform categories in the presence of strong noise. At -7 dB, macro-precision is 89.1%, while the best baseline performs 83.6%. The 5.5 percentage point difference implies that the proposed fusion scheme may help in distinguishing the categories that share temporal and spectral characteristics. At 1 dB, the proposed method achieves 95.0%, leading by 2.8% from Decomposition Denoising CNN. At 9 dB, the margin is 1.7%, while at 17 dB, the margin is 1.1%. This decreasing margin with higher SNR levels is consistent with the assumption that less noisy signals make the classification task easier for all methods. Class-wise confusion matrices are needed to determine which modulation categories cause the most errors in the positive predictions.

4.4 RESULTS BASED ON MACRO-RECALL VERSUS SNR

Macro-recall evaluates how effectively each model identifies the true examples belonging to every radar signal category. Table 7 presents the results across the five SNR conditions.

Table.5. Macro-recall (%) at different SNR levels

SNR (dB)	Time-Frequency CNN	Higher-Order Spectral Residual Learning	Decomposition Denoising CNN	AMNR-RCN
-15	56.8	62.1	65.8	73.2
-7	75.4	80.1	83.0	88.6
1	87.4	90.2	91.8	94.7

9	93.3	94.7	95.5	97.3
17	95.6	96.5	97.0	98.2

According to Table.5, the macro-recall of AMNR-RCN grows from 73.2% to 98.2% as the SNR rises from -15 dB to 17 dB. At -15 dB, it outperforms Time-Frequency CNN by 16.4%, Higher-Order Spectral Residual Learning by 11.1%, and Decomposition Denoising CNN by 7.4%. It appears that multi-scale temporal and spectral representations can be useful in preserving discriminative information for classification purposes in cases where the received waveform is highly corrupted. At -7 dB, AMNR-RCN achieves 88.6% macro-recall, while the best baseline model reaches 83.0%. At 1 dB, it achieves 94.7%, outperforming Decomposition Denoising CNN by 2.9%. At 9 dB, it outperforms Decomposition Denoising CNN by 1.8% and by 1.2% at 17 dB. It is likely that the greatest possible advantage is achieved in the most difficult signal conditions rather than easy ones.

4.5 RESULTS BASED ON MACRO-F1 SCORE VERSUS SNR

Macro-F1 combines precision and recall, providing a balanced assessment of classification quality across all waveform categories. The Table.6 presents the macro-F1 scores for the three existing methods and AMNR-RCN.

Table.6. Macro-F1 score (%) at different SNR levels

SNR (dB)	Time-Frequency CNN	Higher-Order Spectral Residual Learning	Decomposition Denoising CNN	AMNR-RCN
-15	56.9	62.3	66.0	73.5
-7	75.5	80.3	83.2	88.8
1	87.5	90.4	91.9	94.8
9	93.4	94.8	95.6	97.4
17	95.7	96.6	97.1	98.3

As in Table.6, AMNR-RCN attains the highest macro-F1 score at all SNRs tested. Its score of 73.5% at -15 dB outperforms the three baselines by 16.6%, 11.2%, and 7.5%, correspondingly. It means that the proposed method combining multi-scale learning, adaptive fusion, and confidence-guided classification could enable more balanced recognition in noisy environments. At -7 dB, the proposed model shows a score of 88.8% as opposed to 83.2% for Decomposition Denoising CNN, showing an improvement of 5.6%. At 1 dB, the difference is reduced to 2.9% with scores of 94.8% and 91.9%. At 9 dB, the AMNR-RCN shows 97.4% versus 95.6% of the strongest baseline, giving a superiority of 1.8%. At 17 dB, the difference is 1.2%. The proposed network provides a higher recognition score in noisy environment whereas all other methods achieve better scores with decreasing SNR. However, the current results do not provide evidence that confidence-guided fusion alone leads to such improvements.

4.6 RESULTS BASED ON MACRO-AUC VERSUS SNR

Macro-AUC evaluates how effectively the models separate each radar signal class from the remaining classes across different decision thresholds. The Table.7 presents one-versus-rest macro-AUC values for the five SNR conditions.

Table.7. Macro-AUC (%) at different SNR levels

SNR (dB)	Time-Frequency CNN	Higher-Order Spectral Residual Learning	Decomposition Denoising CNN	AMNR-RCN
-15	70.5	75.8	79.1	84.6
-7	84.2	88.1	90.3	93.8
1	92.1	94.0	95.1	97.0
9	96.0	96.8	97.4	98.6
17	97.3	97.9	98.2	99.0

The Table.7 demonstrates that AMNR-RCN yields the maximum macro-AUC over the range of SNR levels under consideration. At -15 dB, the proposed architecture yields a value of 84.6%, which is 14.1% greater than that of Time-Frequency CNN, 8.8% greater than that of Higher-Order Spectral Residual Learning, and 5.5% greater than that of Decomposition Denoising CNN. This implies that the proposed architecture may learn more discriminative class scores due to noise leading to significant overlap between waveform classes. At -7 dB, macro-AUC is equal to 93.8%, which is 3.5% greater than the best baseline performance. At 1 dB, AMNR-RCN yields 97.0%, whereas the best baseline (Decomposition Denoising CNN) gives 95.1%. At 9 dB, AMNR-RCN yields 98.6%, and at 17 dB, it yields 99.0%. Corresponding increases over the best baseline are 1.2% and 0.8%, respectively. Decreased difference at higher SNRs can be attributed to the fact that all four architectures yield better class separation at lower noise levels. It should be noted that macro-AUC does not prove that predicted probabilities are calibrated or whether the operational threshold selected is acceptable.

5. CONCLUSION

AMNR-RCN is the architecture described in this research paper, which can be used for classification of radar signals in a noisy environment. The proposed design includes several components, namely noise-aware pre-processing, parallel temporal convolution branches, STFT-based spectral learning, channel-wise attention, adaptive fusion of features, and confidence guided ensemble-based classification of signals. The methodological design aims to identify complementary waveform characteristics and mitigate noise-induced variability. The optimization objective of the training process takes into consideration balanced classification and consistency regularization. In turn, the evaluation protocol highlights the need for strict data split and multiple measures of the model performance at different SNR levels. These results are representative of the planned comparative study. At -15 dB, AMNR-RCN shows 74.6% accuracy and 84.6% macro-AUC in comparison with 67.2% accuracy and 79.1% macro-AUC in the case of Decomposition Denoising CNN. At 17 dB, the proposed model shows 98.6% accuracy and 99.0% macro-AUC.

REFERENCES

[1] A. Soumya, C. Krishna Mohan and L.R. Cenkeramaddi, "Recent Advances in mmWave-Radar-based Sensing, its Applications, and Machine Learning Techniques: A Review", *Sensors*, Vol. 23, No. 21, pp. 8901-8924, 2023.

[2] Z. Geng, H. Yan and D. Zhu, "Deep-Learning for Radar: A Survey", *IEEE Access*, Vol. 9, pp. 141800-141818, 2021.

[3] P. Gupta, P. Jain and O.G. Kakde, "Deep Learning Techniques in Radar Emitter Identification", *Defence Science Journal*, Vol. 73, No. 5, pp. 551-563, 2023.

[4] Z. Qu, X. Mao and Z. Deng, "Radar Signal Intra-Pulse Modulation Recognition based on Convolutional Neural Network", *IEEE Access*, Vol. 6, pp. 43874-43884, 2018.

[5] K. Chen, L. Zhu, S. Chen and H. Zhao, "Deep Residual Learning in Modulation Recognition of Radar Signals using Higher-Order Spectral Distribution", *Measurement*, Vol. 185, pp. 109945-109957, 2021.

[6] E. Mason, B. Yonel and B. Yazici, "Deep Learning for Radar", *Proceedings of International Conference on Radar*, pp. 1703-1708, 2017.

[7] Z. Qu, X. Mao and Z. Deng, "Radar Signal Intra-Pulse Modulation Recognition based on Convolutional Neural Network", *IEEE Access*, Vol. 6, pp. 43874-43884, 2018.

[8] K. Chen, L. Zhu, S. Chen, S. Zhang and H. Zhao, "Deep Residual Learning in Modulation Recognition of Radar Signals using Higher-Order Spectral Distribution", *Measurement*, Vol. 185, pp. 109945-109962, 2021.

[9] Q. Hou and H. Wu, "Recognition of LPI Radar Signal Intrapulse Modulation based on CNN and Time Frequency Denoising", *Journal of Electronics and Information Science*, Vol. 9, No. 1, pp. 142-152, 2024.

[10] M. Jiang, F. Zhou, L. Shen and N. Jin, "Multilayer Decomposition Denoising Empowered CNN for Radar Signal Modulation Recognition", *IEEE Access*, Vol. 12, pp. 31652-31661, 2024.

[11] L. Liu and X. Li, "Radar Signal Recognition based on Triplet Convolutional Neural Network", *EURASIP Journal on Advances in Signal Processing*, Vol. 2021, No. 1, pp. 112-131, 2021.

[12] C. Huang, X. Xu, F. Fan, S. Wei and M. Gu, "A Low-SNR-Adaptive Temporal Network with Smart Mask Attention for Radar Signal Modulation Recognition", *Digital Signal Processing*, Vol. 168, No. 4, pp. 105640-105664, 2026.