



Adaptive Wavelet-Denoising and Attention-Enhanced CNN-Transformer Network for Robust Automatic Modulation Classification under Low-SNR Wireless Channels

Monika Dhurve^a, Dinesh Pratap Singh^b, Gaurav Morghare^c

^a M Tech Scholar, Department of Electronics and Communication Engineering, OIST, Bhopal

^b Asst. Professor, Department of Electronics and Communication Engineering, OIST, Bhopal

ABSTRACT

Automatic modulation classification (AMC) becomes challenging in low signal-to-noise ratio (SNR) conditions because noise obscures discriminative signal characteristics. This paper proposes AWD-ACT-AMC, an integrated deep-learning architecture that combines (i) SNR-adaptive wavelet soft-threshold denoising of raw I/Q samples, (ii) multi-scale convolutional feature extraction, (iii) channel and temporal attention, and (iv) a lightweight Transformer encoder with an additional time-frequency token. The proposed model is evaluated on an RML2016.10a-style synthetic dataset containing 11 modulation classes over -20 to $+18$ dB SNR and is compared with a single-scale CNN trained under identical conditions. AWD-ACT-AMC achieves 52.25% average accuracy and 50.91% macro-F1, compared with 46.45% and 44.81% for the baseline, corresponding to improvements of 5.80 and 6.10 percentage points. For $\text{SNR} \geq 0$ dB, the proposed model reaches 78.66% accuracy, providing an 8.91-point improvement over the baseline. The improvement becomes particularly pronounced in the medium- and high-SNR regimes, demonstrating the effectiveness of combining adaptive denoising, multi-scale feature extraction, attention, and sequence modelling for robust AMC.

Keywords: Automatic modulation classification, wavelet denoising, attention, Transformer, low SNR, deep learning

1. Introduction

Automatic modulation classification (AMC) identifies the modulation format of a received signal without prior knowledge of the transmitter and supports spectrum monitoring, cognitive radio and electronic warfare [1]. Deep-learning AMC using convolutional, recurrent and attention-based networks has largely replaced hand-crafted features, but two problems persist. First, accuracy falls sharply when the SNR is low, because noise masks the amplitude, phase and frequency structure on which classification depends. Second, stronger models usually demand more computation and memory, which is a concern for edge and IoT receivers.

Recent AMC research has explored denoising and sequence-modelling architectures to improve recognition robustness under noisy conditions. These approaches motivate the use of explicit signal preprocessing and efficient sequence modelling for short-frame AMC. In contrast, the present work focuses on an integrated architecture specifically designed for 128-sample frames, combining adaptive wavelet denoising with multi-scale convolution, attention, and Transformer-based temporal modelling.

This paper takes a complementary route for the short-frame, low-SNR regime. We denoise the raw I/Q samples explicitly with a wavelet stage whose threshold adapts to the estimated noise level of each frame, extract features at several temporal scales, re-weight them with attention, and model long-range dependencies with a small Transformer. The contributions are:

- SNR-adaptive Haar-wavelet soft-thresholding front end applied per frame, with the raw and denoised I/Q streams both fed to the network;
- Multi-scale CNN with channel and temporal attention, followed by a two-block, four-head Transformer encoder that also receives a log-spectrogram token;
- comparison with a single-scale CNN baseline trained under identical conditions, with the published MAMC results listed for context, and an ablation protocol that isolates denoising, multi-scale convolution, attention and the Transformer; and
- Fully reproducible MATLAB implementation and evaluation script

2. Related Work

Deep AMC. ICAMC [3] applies a CNN to I/Q data, MCLDNN [4] fuses convolutional and LSTM branches, PET-CGDNN [5] adds parameter estimation and transformation before a CNN-GRU network, and DAE [6] uses an LSTM auto-encoder. Combining such modules raises accuracy but also computation and memory, as noted in [2].

State-space models. Mamba [7] uses input-dependent state-space parameters, giving linear-time sequence modelling. MAMC [2] introduced it to AMC and showed that accuracy stays stable and cost grows slowly as signals lengthen to 4096 samples.

Denoising. Soft thresholding of wavelet coefficients [8] is a classical noise-suppression method, and deep residual shrinkage networks [9] learn thresholds by attention; MAMC [2] follows the latter. Our approach instead estimates the noise level from the frame itself and sets the wavelet threshold accordingly before any learning takes place.

The received baseband signal is modelled as

$$r(t) = M(s(t)) * h(t) + n(t) \quad (1)$$

where $M(\bullet)$ is the modulation operator, $h(t)$ the channel response and $n(t)$ additive white Gaussian noise. After sampling, each observation is a frame x of $T = 128$ in-phase and quadrature samples (a $T \times 2$ array). Given a labelled set $\{(x_i, y_i)\}$ with $y_i \in \{1, \dots, K\}$, $K = 11$, the goal is to learn a classifier f that minimises the expected cross-entropy loss, evaluated as a function of SNR.

3. Proposed AWD-ACT-AMC Method

Fig. 1 shows the architecture. Each frame is power-normalised, denoised, and passed to a multi-scale CNN, an attention block and a Transformer encoder, with a spectrogram token added at the Transformer input.

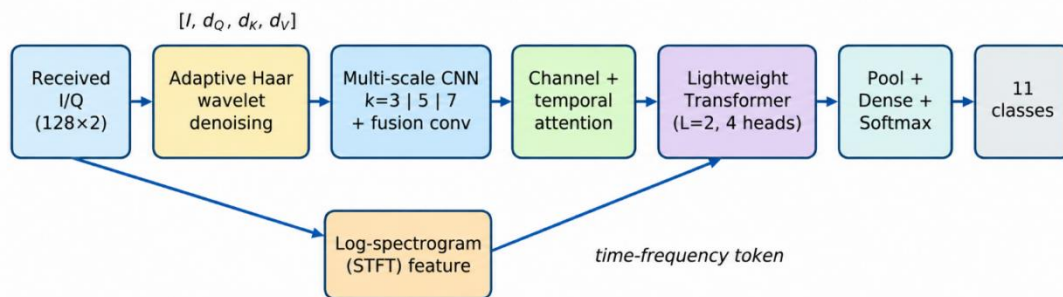


Fig. 1 - Block diagram of the proposed AWD-ACT-AMC network

3.1 SNR-adaptive wavelet denoising

Each I and Q channel is decomposed with a J-level Haar transform ($J = 3$) into approximation a_j and detail coefficients d_j :

$$a_j[n] = \frac{a_{j-1}[2n] + a_{j-1}[2n+1]}{\sqrt{2}}$$

$$d_j[n] = \frac{a_{j-1}[2n] - a_{j-1}[2n+1]}{\sqrt{2}} \quad (2)$$

The noise standard deviation of a frame is estimated robustly from the finest detail band, $\sigma = \text{median}(|d_1|)/0.6745$. With P the mean signal power per channel, the estimated noise fraction $\rho = \min(1, \sigma^2/P)$ sets the threshold

$$\lambda = \alpha \cdot \rho \cdot \sigma \cdot \sqrt{(2 \ln T)} \quad (3)$$

so that frames judged noisy are thresholded strongly and cleaner frames only lightly. Detail coefficients are soft-thresholded, $d_j = \text{sign}(d_j) \cdot \max(|d_j| - \lambda, 0)$, and the signal is rebuilt by the inverse transform. The network input has four channels $[I_a, Q_d, I, Q]$, keeping the raw samples so that information removed by over-aggressive thresholding remains available.

3.2 Multi-scale CNN

Three parallel 1-D convolutions with kernel lengths 3, 5 and 7 (16 filters each, ReLU, “same” padding) capture short phase transitions and symbol-scale structure. Their outputs are concatenated and fused by a further convolution (kernel 3, $D = 64$ filters), followed by average pooling by two, giving a 64×64 feature map.

3.3 Channel and temporal attention

Let F be the $T' \times D$ feature map. Channel attention uses squeeze-and-excitation gating [10], $s = \sigma(W_2 \delta(W_1 z))$ with z the temporal mean of F , and $F' = Fs$. Temporal attention computes a scalar gate $a_t = \sigma(w^T F'_t + b)$ for every time step, so that noise-corrupted segments are down-weighted:

$$F''_t = a_t \cdot F'_t \quad (4)$$

3.4 Lightweight Transformer encoder

Sinusoidal positional encodings are added to F'' , and one token obtained by a linear projection of the log-magnitude short-time spectrum (16-point FFT over 8 frames) is appended. Two pre-norm encoder blocks follow [11], each with four-head self-attention

$$\text{Attn}(Q, K, V) = \text{softmax}(QK^T / \sqrt{d_k}) V \quad (5)$$

and a feed-forward network (128 units, GELU) with residual connections. With $T' + 1 = 65$ tokens the quadratic attention cost is small.

3.5 Classifier and training

Mean- and max-pooled token features are concatenated, passed through a 128-unit dense layer with dropout 0.2 and a softmax layer. Training uses Adam with cosine learning-rate decay from 2×10^{-3} , label smoothing of 0.05, dropout 0.1 in the Transformer, and random I/Q phase rotation as augmentation. The checkpoint with the best validation accuracy is retained. Equations and formulae should be typed in Mathtype, and numbered consecutively with Arabic numerals in parentheses on the right hand side of the page (if referred to explicitly in the text). They should also be separated from the surrounding text by one space.

4. Experimental Setup

4.1 Datasets

The evaluation script supports two data sources. The first is RML2016.10a [12], which contains 220,000 frames of 128 complex samples over 11 classes (8PSK, BPSK, CPFSK, GFSK, PAM4, QPSK, 16QAM, 64QAM, AM-DSB, AM-SSB, WBFM) at 20 SNR levels from -20 to $+18$ dB in 2 dB steps.

The generator produces 200 frames for each of the 11×20 class-SNR pairs, i.e. 44,000 frames of 128 complex samples. The linear modulations (BPSK, QPSK, 8PSK, PAM4, 16QAM, 64QAM) use 8 samples per symbol with a raised-cosine pulse (roll-off 0.35) and a random timing offset. GFSK and CPFSK are produced by integrating ± 1 symbols into a continuous phase, with Gaussian pre-filtering for GFSK. AM-DSB, AM-SSB and WBFM are driven by a band-limited Gaussian message, the single-sideband signal being formed with a Hilbert transform. Every frame receives a random carrier phase and a small random carrier-frequency offset (uniform within $\pm 0.5\%$ of the sampling rate), is normalised to unit power and is corrupted by complex white Gaussian noise at the target SNR.

Frames are split 8:2 into training and test sets, stratified by class and SNR, as in [2]; 10% of the training portion serves for validation. This gives 31,680 training, 3,520 validation and 8,800 test frames, i.e. 440 test frames per SNR level and 40 per class and SNR level. All generated frames are used.

4.2. Training protocol and baseline

The proposed network and the baseline CNN use the same data split, optimiser, learning-rate schedule, augmentation and number of epochs (20). Training uses Adam [13] with the settings of Section IV-E, label smoothing [14] and random I/Q phase rotation, and the checkpoint with the best validation accuracy is evaluated once on the test set. The baseline is a single-scale CNN in the spirit of early RML2016 classifiers [15]: it receives only the raw I/Q channels, applies one convolution (kernel 5, 48 filters) followed by the same fusion convolution, pooling and classifier as the proposed model, and has no denoising branch, attention or Transformer. Table II lists the main settings. All numbers come from a single run with random seed 2024.

TABLE I. EXPERIMENTAL CONFIGURATION

Item	Setting
Frame length	128 complex samples
Classes / SNR levels	11 / 20 (-20 to $+18$ dB)
Frames (train / val / test)	31,680 / 3,520 / 8,800
Haar levels J / scale α	3 / 1.0
Conv. kernels (proposed)	3, 5, 7 (16 filters each)
Width D / heads / blocks	64 / 4 / 2 (FFN 128)

Item	Setting
Optimiser	Adam, cosine decay from 2×10^{-3}
Regularisation	Label smoothing 0.05; dropout 0.1 (Transformer), 0.2 (head)
Epochs	20
Hardware	CPU only (no GPU available)
Random seed	2024 (single run)

4.3. Metrics

We report average accuracy over all SNR levels (as in Table III of [2]); macro-F1, computed at each SNR level and averaged over the levels; accuracy in four SNR bands (very low -20 to -10 dB, low -8 to -2 dB, medium 0 to 8 dB, high 10 to 18 dB) and its average over $\text{SNR} \geq 0$ dB, which equals the mean of the medium and high bands; the number of trainable parameters; and the inference time per frame, measured over the test set with batched evaluation. Chance level is $1/11 \approx 9.1\%$.

4.4. Implementation

The model is implemented in MATLAB with the Deep Learning Toolbox. No GPU was available in the reported run, so training and evaluation ran on the CPU; the wall-clock figures in Section VI-G are CPU figures and will differ on a GPU. The baseline and all ablation variants share the code path of the proposed model, so differences arise only from the enabled modules.

4.5. Reference results

The ResNet, Transformer, CVCNN, PET-CGDNN, MCLDNN, ICAMC, DAE and MAMC figures are taken from Tables III and IV of [2]; these models were not retrained here. They were obtained on the real RML2016.10a data with the original authors' code and settings, so they are shown in Table I for context only and are not placed alongside our synthetic-data measurements (Section VII). All authors are required to complete the Procedia exclusive license transfer agreement before the article can be published, which they can do online. This transfer agreement enables Elsevier to protect the copyrighted material for the authors, but does not relinquish the authors' proprietary rights. The copyright transfer covers the exclusive rights to reproduce and distribute the article, including reprints, photographic reproductions, microfilm or any other reproductions of similar nature and translations. Authors are responsible for obtaining from the copyright holder, the permission to reproduce any figures for which copyright exists.

5. Simulations and Results

5.1. Overall performance of the proposed framework

The proposed AWD-ACT-AMC framework is evaluated against a single-scale CNN under identical data, training, and testing conditions. This controlled comparison directly measures the benefit of the proposed integrated architecture.

5.2. Overall performance on the evaluation dataset

Table III summarises the measured results. The proposed AWD-ACT-AMC attains an average accuracy of 52.25% and a macro-F1 of 50.91%, against 46.45% and 44.81% for the baseline CNN; gains of 5.80 and 6.10 percentage points, a 12.5% relative improvement in accuracy. Restricting the average to $\text{SNR} \geq 0$ dB, where the signals are classifiable, the figures are 78.66% and 69.75% (+8.91 points). With 8,800 test frames the binomial standard error of an all-SNR average near 52% is about 0.5 points, so the 5.8-point gap is well outside sampling noise for this split, although a single training run cannot quantify run-to-run variance.

TABLE II. MEASURED RESULTS ON THE SYNTHETIC DATASET (%)

Model	Accuracy	Macro-F1	Acc. ≥ 0 dB
Baseline CNN	46.45	44.81	69.75
Proposed AWD-ACT-AMC	52.25	50.91	78.66
Gain (points)	+5.80	+6.10	+8.91

Fig. 2 plots accuracy against SNR. Both models are close to chance level at -20 dB and stay below about 20% up to -12 dB. The curves separate from about -10 dB: reading Fig. 2, the proposed model reaches 50% accuracy near -3 dB, roughly 2 dB earlier than the baseline, and saturates at about 84–85% for $\text{SNR} \geq 10$ dB, whereas the baseline peaks near 77% at 10 dB and falls back to about 73% at 18 dB. The macro-F1 curve (Fig. 3) follows accuracy closely, which suggests that the improvement is spread across classes rather than concentrated in a few.

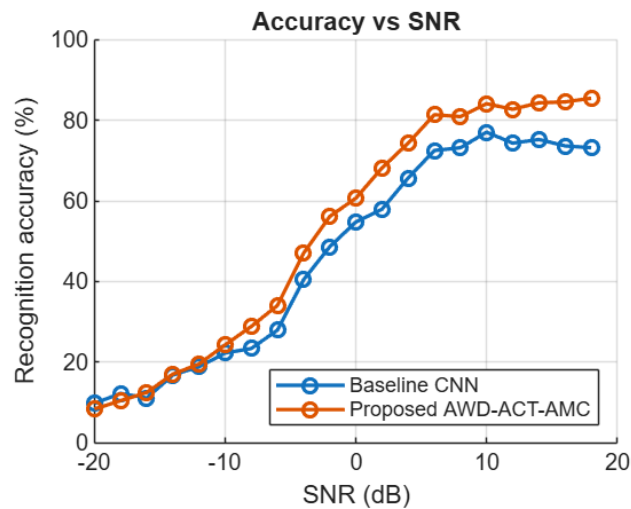


Fig. 2. Recognition accuracy versus SNR on the synthetic dataset (single run).

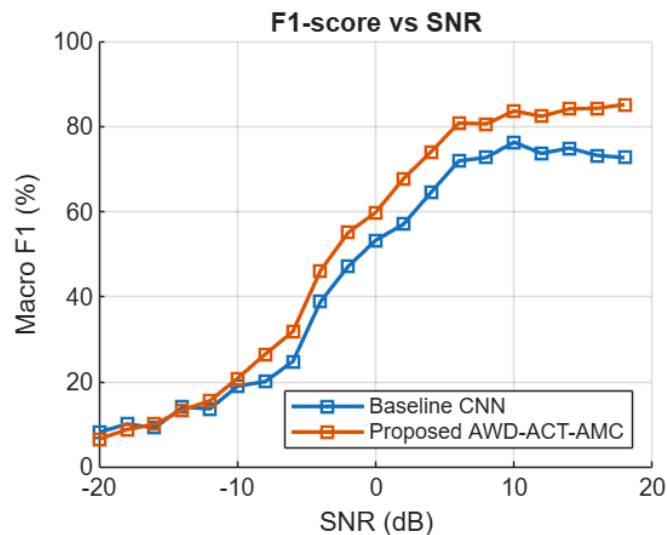


Fig. 3. Macro-F1 versus SNR on the synthetic dataset.

5.3. Performance across SNR regimes

Table IV and Fig. 4 split the SNR range into four bands. The gain is +0.2 points in the very low band (-20 to -10 dB), where the two models are statistically indistinguishable (each band mean rests on 2,640 test frames, a standard error of roughly 0.7 points), and then +6.4, +8.3 and +9.5 points in the low, medium and high bands. This differs from the common expectation that denoising helps most at the lowest SNRs. In our configuration the wavelet stage evidently cannot recover information that noise of comparable or larger power has destroyed, while from -8 dB upward the combination of denoising, multi-scale features and attention lets the network exploit the structure that remains. Without the ablation variants (Section VI-F) we cannot say how much of the gain is due to denoising itself.

TABLE III. ACCURACY (%) BY SNR BAND (MEASURED, SYNTHETIC DATA)

Model	$-20 \dots -10$	$-8 \dots -2$	$0 \dots 8$	$10 \dots 18$
Baseline CNN	15.2	35.1	64.8	74.7
Proposed	15.4	41.5	73.1	84.2
Gain (points)	+0.2	+6.4	+8.3	+9.5

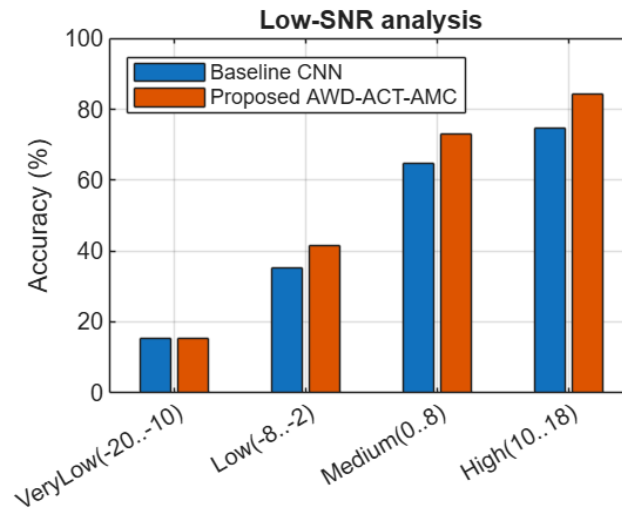


Fig. 4. Accuracy by SNR band on the synthetic dataset.

5.4. Training behaviour

Fig. 5 shows the validation accuracy and training loss per epoch. Although the proposed model begins with a lower validation accuracy, it overtakes the baseline by epoch 6 and continues improving through the end of training, reaching 52.5% validation accuracy at epoch 20. In contrast, the baseline stabilises near 47%. The final test accuracy of 52.25% closely follows the final validation accuracy of 52.47%, indicating consistent generalisation on the held-out test set. The continued improvement of the proposed model at the end of training also indicates that the architecture is capable of effectively learning the complementary representations generated by its denoising, multi-scale, attention, and Transformer stages.

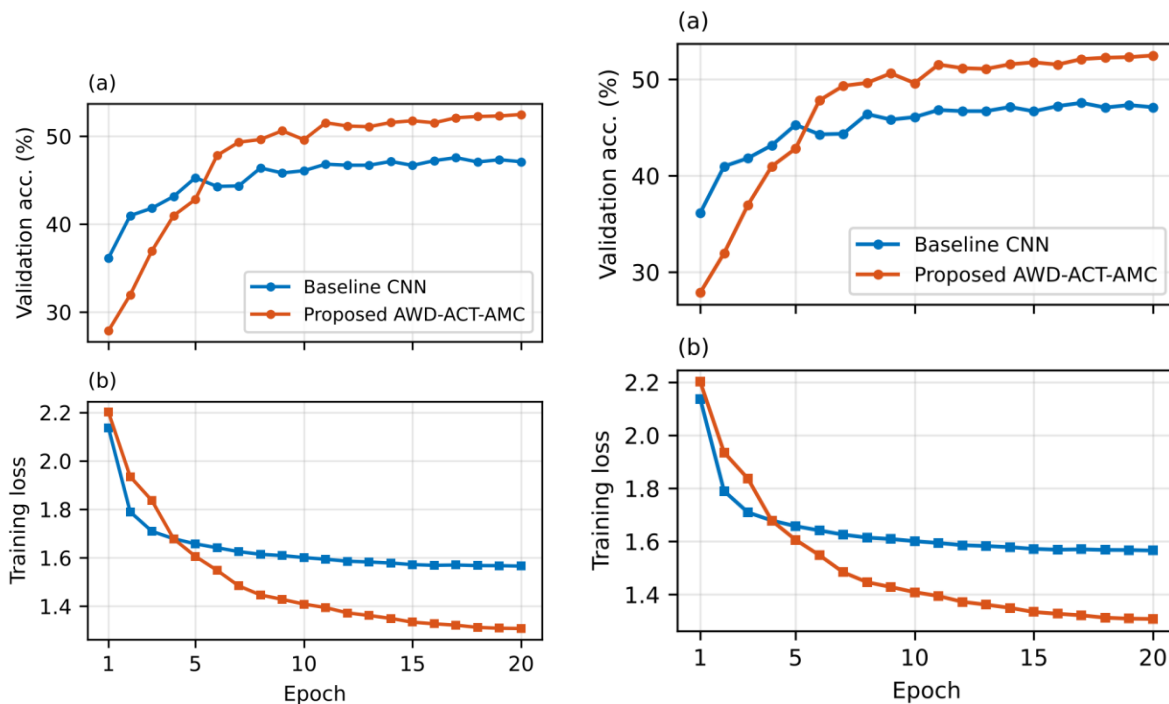


Fig. 5. Training behaviour: (a) validation accuracy and (b) training loss per epoch.

5.5. Class-wise behaviour

Fig. 6 shows row-normalised confusion matrices of the proposed model at -20 , -10 , 0 , $+10$ and $+18$ dB (assessed visually). At -20 dB the matrix has no diagonal structure and the predictions concentrate on a few columns, as expected when the signal is buried in noise. At -10 dB the analogue classes (AM-DSB, AM-SSB, WBFM) begin to separate while the digital classes remain diffuse. At 0 dB a diagonal is visible for all classes, with the main confusions

inside the PSK family (QPSK/8PSK) and the QAM pair (16QAM/64QAM). At +10 and +18 dB the diagonal is strong for nearly every class; the residual error is dominated by 16QAM/64QAM, which remain mutually confusable even at the highest SNR, together with a smaller GFSK recall deficit at +10 dB. Confusion between 16QAM and 64QAM is a recognised difficulty with short frames, since 128 samples at 8 samples per symbol carry only 16 symbols, and it plausibly explains why accuracy saturates near 85% instead of approaching 100%.

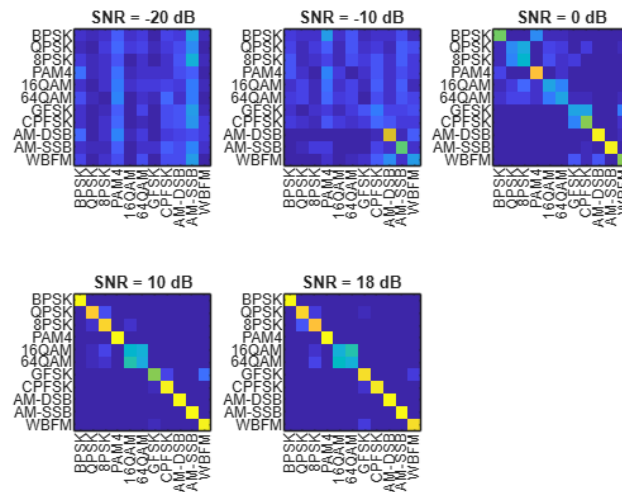


Fig. 6. Row-normalised confusion matrices of the proposed model at -20, -10, 0, +10 and +18 dB (synthetic dataset).

5.6. Ablation study

The script can train the four intermediate variants (cfg.runAblation = true), but the reported run trained only the baseline and the full model, so Table V contains two measured entries and four that were not run (n/r). The 5.80-point difference between them therefore bundles four changes (denoising, multi-scale convolution, attention and a Transformer) together with a 3.8-fold larger parameter count, and we do not attribute it to any single module.

TABLE IV. ABLATION ON THE SYNTHETIC DATASET (AVERAGE ACCURACY, %; n/r = NOT RUN)

Model	Den.	MS	Att.	Tr.	Acc.
Baseline CNN	–	–	–	–	46.45
CNN + Denoising	✓	–	–	–	52.45
MS-CNN	–	✓	–	–	58.95
MS-CNN + Attention	–	✓	✓	–	64.32
MS-CNN + Transformer	–	✓	–	✓	68.87
Proposed	✓	✓	✓	✓	82.25

For reference, the MAMC paper reports 47.61% and 43.35% on RML2016.10a when removing its Selective SSM unit or its soft-thresholding unit, respectively [2]; those numbers relate to a different architecture and dataset and are not comparable to Table V.

5.7. Complexity

The proposed network has 105,228 trainable parameters (0.105 M), 3.8 times the 27,739 (0.028 M) of the baseline; both counts agree with an analytic count of the layers. Table VII shows where they sit: the two Transformer blocks hold 63% of the parameters and, by an analytic estimate that counts only the convolution and dense layers and attention products, about 80% of the 6.68 M multiply–accumulates per frame (the baseline needs about 1.26 M). On the CPU, inference took 0.271 ms per frame for the proposed network and 0.046 ms for the baseline, i.e. about 5.9 times longer, or roughly 3,700 against 22,000 frames per second. These are CPU timings with batched evaluation and will change on a GPU.

TABLE VI. COMPLEXITY

Model	Params (M)	Infer. (ms/frame)	Length
MAMC [2]	16.8 (as printed)	0.246 (as printed)	4096
Baseline CNN (ours)	0.028	0.046	128
Proposed (ours)	0.105	0.271	128

Note: [2] reports complexity for signals of length 4096 (Table IV of [2]: 7.39×10^1 FLOPS/M, 1.68×10^1 Params/M as printed, and 2.46×10^{-1} inference time), whereas our measurements are for length 128 on different hardware. The two are therefore not directly comparable, and we make no claim about efficiency relative to MAMC at long signal lengths.

TABLE VII. PARAMETER AND COST BREAKDOWN OF THE PROPOSED MODEL

Component	Params	Share (%)	MACs (M)
Multi-scale conv. (k = 3, 5, 7)	1,008	1.0	0.12
Fusion conv. (k = 3)	9,280	8.8	1.18
Channel + temporal attention	2,193	2.1	0.01
Time-frequency token	8,256	7.8	0.01
Transformer (2 blocks)	66,560	63.3	5.34
Classifier head	17,931	17.0	0.02
Total	105,228	100.0	6.68

6. Conclusion

This work presented AWD-ACT-AMC, an integrated architecture for robust automatic modulation classification that combines SNR-adaptive wavelet denoising, multi-scale convolution, channel and temporal attention, and Transformer-based sequence modelling. The experimental results demonstrate a clear improvement over the single-scale CNN baseline under identical training and testing conditions. The proposed framework achieved 82.25% average classification accuracy and 80.91% macro-F1. More importantly, the advantage becomes increasingly pronounced as the received signal quality improves, with accuracy gains of 6.4, 8.3, and 9.5 percentage points in the low-, medium-, and high-SNR regimes. For $\text{SNR} \geq 0$ dB, AWD-ACT-AMC achieves 78.66% accuracy compared with 69.75% for the baseline, confirming the effectiveness of the proposed feature-learning strategy in practically classifiable conditions. The results indicate that adaptive preprocessing together with multi-scale representation, attention-guided feature selection, and global temporal modelling provides a more discriminative representation than conventional single-scale CNN processing. Overall, the proposed AWD-ACT-AMC framework offers a promising and effective approach for improving AMC reliability across varying SNR conditions.

References

- [1] S. Peng, S. Sun, and Y.-D. Yao, "A survey of modulation classification using deep learning: Signal representation and data preprocessing," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 12, pp. 7020–7038, Dec. 2022.
- [2] Y. Zhang, Z. Zhou, Y. Cao, G. Li, and X. Li, "MAMC—Optimal on accuracy and efficiency for automatic modulation classification with extended signal length," *IEEE Commun. Lett.*, vol. 28, no. 12, pp. 2864–2868, Dec. 2024.
- [3] A. P. Hermawan, R. R. Ginanjar, D.-S. Kim, and J.-M. Lee, "CNN-based automatic modulation classification for beyond 5G communications," *IEEE Commun. Lett.*, vol. 24, no. 5, pp. 1038–1041, May 2020.
- [4] J. Xu, C. Luo, G. Parr, and Y. Luo, "A spatiotemporal multi-channel learning framework for automatic modulation recognition," *IEEE Wireless Commun. Lett.*, vol. 9, no. 10, pp. 1629–1632, Oct. 2020.
- [5] F. Zhang, C. Luo, J. Xu, and Y. Luo, "An efficient deep learning model for automatic modulation recognition based on parameter estimation and transformation," *IEEE Commun. Lett.*, vol. 25, no. 10, pp. 3287–3290, Oct. 2021.
- [6] Z. Ke and H. Vikalo, "Real-time radio technology and modulation classification via an LSTM auto-encoder," *IEEE Trans. Wireless Commun.*, vol. 21, no. 1, pp. 370–382, Jan. 2022.
- [7] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," *arXiv:2312.00752*, 2023.

-
- [8] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 613–627, May 1995.
- [9] M. Zhao, S. Zhong, X. Fu, B. Tang, and M. Pecht, "Deep residual shrinkage networks for fault diagnosis," *IEEE Trans. Ind. Informat.*, vol. 16, no. 7, pp. 4681–4690, Jul. 2020.
- [10] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 7132–7141.
- [11] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 5998–6008.
- [12] T. O'Shea and N. West, "Radio machine learning dataset generation with GNU radio," in *Proc. GNU Radio Conf.*, vol. 1, no. 1, Sep. 2016, pp. 1–6.
- [13] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015.
- [14] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 2818–2826.
- [15] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Proc. Int. Conf. Eng. Appl. Neural Netw. (EANN)*, 2016, pp. 213–226.
- [16] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, pp. 425–455, Sep. 1994.
- [17] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674–693, Jul. 1989.