A Power-Efficient Attention-Infused CNN Hardware Accelerator for RF Spectrum Monitoring

Zhifan Song, Abdelrahman Emad Abdelazim, Pirouz Bazargan-Sabet, Franck Wajsburt, Haralampos-G. Stratigopoulos, Hassan Aboushady

A Power-Efficient Attention-Infused CNN Hardware Accelerator for RF Spectrum Monitoring

Zhifan Song, Abdelrahman Emad Abdelazim, Pirouz Bazargan Sabet, Franck Wajsbürt, Haralampos-G. Stratigopoulos and Hassan Aboushady
Sorbonne Université, CNRS, LIP6, Paris, France

***Abstract*—In this paper, we propose a power-efficient attention-infused convolutional neural network (CNN) hardware accelerator for RF spectrum monitoring. The AI model achieves 73.3% average accuracy across all Signal-to-Noise Ratios (SNRs) ranging from -20dB to +30dB, and a 99% accuracy for SNRs higher than 4dB using the RadioML2018 dataset . The number of parameters of the proposed attention-infused CNN is reduced by 93% compared to the baseline CNN model. An efficient hardware implementation on FPGA achieves 61 GOPS and consumes only 1191 mW. Compared to the state of the art, it achieves the highest efficiency of 51 GOPS/W.**

***Index Terms*—AI Hardware Acceleration, RF Modulation Recognition, Deep Learning, CNN.**

I. Introduction

The rapid growth of 5G communication and the Internet of Things (IoT) has intensified demands on the radio spectrum, leading to uneven frequency band utilization [1]. Efficient spectrum monitoring is critical for optimal spectrum usage [2]. Cloud-based RF classification is unsuitable for real-time applications due to its high latency [3]. As communication systems grow more complex, traditional methods struggle with accuracy and computational efficiency [4] [5]. Deep learning offers a promising alternative, with RF signals, represented as I/Q components, being effectively processed as "image-like" inputs, making them suitable for AI-based models, as illustrated in Fig. 1.

While Long Short-Term Memory (LSTM) networks [6] have demonstrated success with time-series data, their sequential nature limits parallelization. Transformer models [7], though powerful, are memory-intensive, making them impractical for deployment on edge devices. Convolutional Neural Networks (CNNs), by contrast, are well-suited for parallelization, enabling faster inference with lower memory usage, making them a better fit for IoT devices.

Many deep learning-based spectrum monitoring approaches focus primarily on improving accuracy [8]–[14], yet they often neglect hardware implementation challenges. These models, with parameter counts ranging from 72k to over 1.26 million, rely heavily on GPUs for inference, rendering them impractical for low-power embedded devices. While the Denoising Auto-Encoder (DAE) [15] model is the most lightweight choise, its reliance on LSTMs poses parallelization difficulties and introduces additional pre-processing steps with L2-normalized amplitude and normalized phase. Existing hardware implementations either suffer from high power consumption or limited accuracy [17]–[21], highlighting the need for more efficient, hardware-optimized solutions. To tackle the challenges, in this work, we introduce the following contributions:

- A lightweight attention-infused CNN RF modulation recognition model which achieves high accuracy with a significantly lower number of parameters compared to the baseline CNN [17].
- A power-efficient hardware accelerator of this attention-infused CNN model. An FPGA implementation shows that the model can achieve 61 GOPS with a power consumption of 1191 mW.

This work was funded by the Chips JU project Resilient Trust of the EU's Horizon Europe research and innovation programme under Grant agreement N° 101112282.

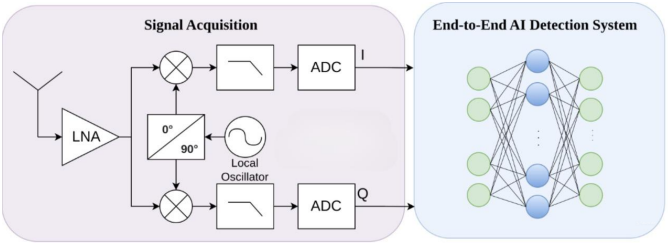

Fig. 1. End-to-end hardware implemented AI spectrum monitoring system.

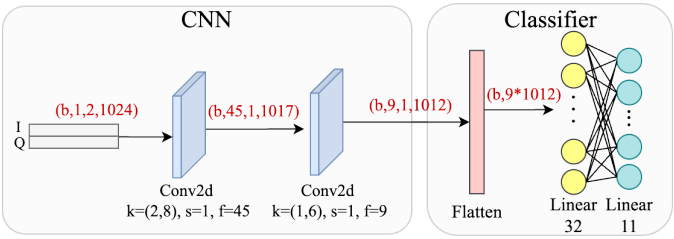

Fig. 2. The baseline CNN model, used in [17] for modulation recognition, has more than 295k parameters. The model contains two Convolutional Layers and two Dense Layers.

II. The Proposed AI Model

A. Datasets

The RadioML2018 [22] dataset was employed for modulation recognition. As in [15], [21], we use the 11 normal classes: FM, GMSK, OQPSK, BPSK, 8PSK, AM-SSB-SC, 4ASK, AM-DSB-SC, QPSK, OOK, and 16-QAM. Each frame size is 2×1024, with a Signal-to-Noise Ratio (SNR) ranging from -20 dB to +30 dB. Following the repartition as in [15],

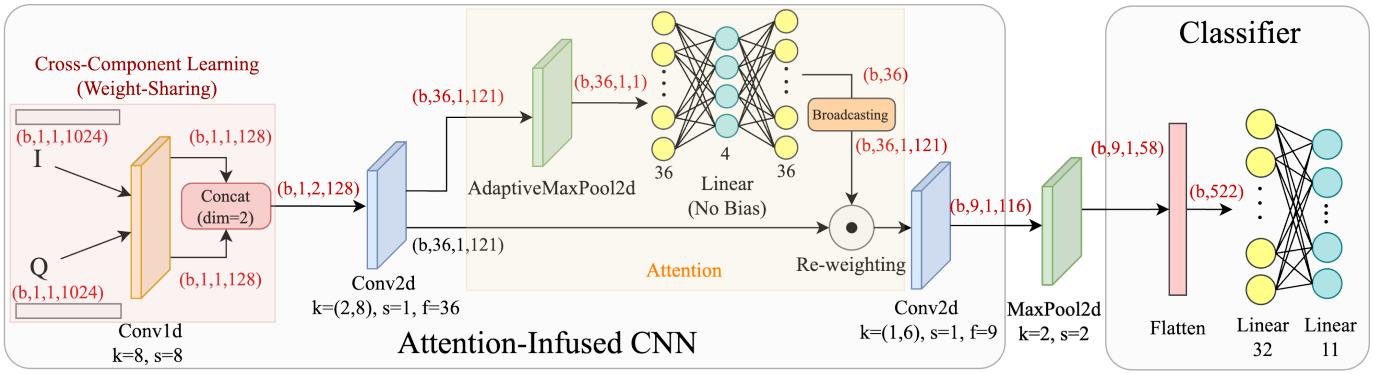

Fig. 3. Architecture of the proposed attention-infused CNN model. The number of parameters is reduced to 19k. The kernel size, stride, and number of filters are denoted by k, s, and f, respectively.

the RadioML2018 dataset, which contains 2,555,904 frames, was split into 50%-25%-25% for training, validation, and testing, respectively. Unlike the RadioML2016 [23] dataset, the RadioML2018 dataset provides a more realistic over-the-air data captures.

B. Proposed Model Architecture

Fig. 2 shows the baseline CNN model proposed for modulation recognition [17]. The model features 2 Convolutional Layers (CL) and 2 Dense Layers (DL), with 45 filters in the first CL and 9 in the second. This model achieves good accuracy for the modulation recognition RadioML2016 dataset with a much lower number of parameters compared to other models [14], [16].

The proposed lightweight model optimized for RF spectrum monitoring is shown in Fig. 3. The I and Q signals, which are the outputs of the Analog-to-Digital Converters (ADCs) in Fig. 1, are first passed through an 1D CL for key features extraction. Instead of treating the I and Q components as independent channels, we introduce a Cross-Component Learning (CCL) strategy, highlighted in red in Fig. 3. This approach allows the model to learn shared features between the I and Q components by using shared weights across both dimensions. By encouraging the model to capture common characteristics, we not only enhance feature learning but also reduce the kernel parameter count by 50%.

Attention mechanisms [7] were initially developed for language models to focus on specific input sequence parts, enhancing contextual relationship capture. This concept has been adapted to CNNs for improved feature extraction. Squeeze and Excitation [24] is a typical format that allows models to re-calibrate feature maps by emphasizing relevant channels and suppressing less important ones, enabling more efficient learning from input data.

To optimize the performance further, we propose integrating our customized attention variant between the two CLs. This includes replacing average pooling with adaptive max-pooling, changing the reduction size from 16 to 9 (resulting in 4 neurons in the hidden layer), and using the ReLU activation function instead of sigmoid for faster, quantization-friendly inference. This customization was selected due to its superior performance in the modulation recognition task after extensive hyperparameter tuning. Similar to Fig. 2, Fig. 3 illustrates the data flow in PyTorch format (batch, channel, height, width) to ensure easy reproducibility. The refined architecture includes:

- Cross-component learning for improved feature extraction and parameter reduction.
- A customized attention mechanism to enhance feature map calibration.
- Reduction of the first CL filters from 45 to 36, achieving further parameter reduction, and an additional max pooling layer after the second CL, halving the data length and further reducing parameters with minimal accuracy loss.

TABLE I
CNN TRAINING ENVIRONMENT

Category	Details
GPU	NVIDIA A100 80GB PCIe
CPU	Intel® Xeon® Gold 6300 @ 2 GHz
RAM	2 TB
Operating System	Ubuntu 22.04
Python	3.9.19
PyTorch	2.0.1
CUDA	11.8

C. Model Training and Implementation

Table I details the hardware and software configuration used for AI training. The models were trained for 300 epochs using the Adam optimizer with a learning rate of 0.001 and a batch size of 512 to handle the large dataset.

III. The AI Hardware Accelerator

A key issue with existing modulation recognition approaches is their reliance on shifting kernels, which are common in image processing tasks. However, unlike static images, I/Q data is sequentially generated by the ADCs, and buffering entire frames before processing introduces delays, often leading to frame loss. To overcome this, we propose a parallelized architecture utilizing stationary weights, as shown in Fig. 4. In this design, a FIFO buffers one I/Q sample per clock cycle, and once the kernel is full, computation begins immediately. Multiply and accumulate (MAC) operations are parallelized both within each kernel and across all 36 kernels simultaneously,

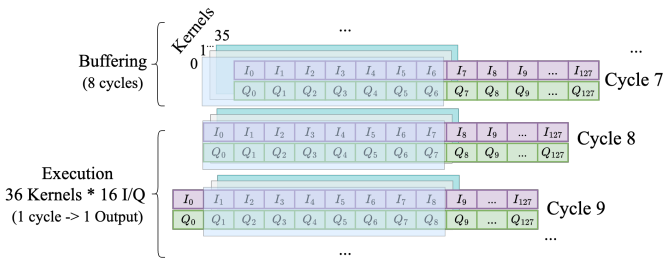

Fig. 4. Stationary weights kernel: the weights are fixed while the I/Q data is fed sequentially from the ADC in a FIFO mode, one sample per clock cycle. MAC operations are executed in parallel component-wise and kernel-wise.

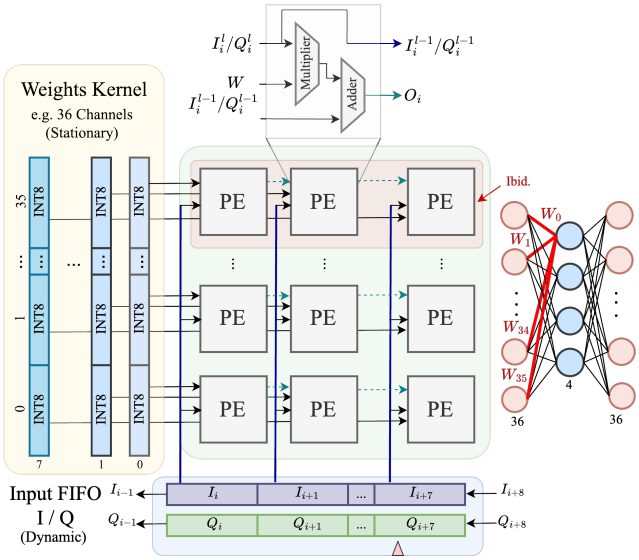

Fig. 5. Accelerator architecture (e.g. first CL): this architecture employs a 2x8 FIFO (same as baseline CNN) for the I/Q signals and 36 weights assigned to each input component. The PEs execute parallel MAC operations along each row, producing one output per kernel. Dense Layers are also parallelized similarly, with neuron connections highlighted in red.

ensuring maximum throughput and minimal latency. Dense layers follow a similar parallel execution strategy, with MACs across neuron connections processed concurrently, as depicted by the red neuron connections and red-shaded Processing Elements (PEs) shown in Fig. 5. A kernel counter is utilized to keep track of the components. The computations at CL1 begin as soon as the cross-component learning layer has produced 8 values, eliminating the need to wait for the entire output, resulting in a significantly faster pipeline. This design delivers optimal computational efficiency and substantial speed gains, making it ideally suited for real-time, edge-based modulation recognition with low power consumption.

IV. Results and Model Comparison

A. Model Comparison and Ablation Study

We conducted a comparative analysis of our model against other modulation recognition models. The DAE model [15] is the most lightweight with ours having a similar parameter scale (<20k), whereas other models have a significantly higher number of parameters ranging from 71k in [9] to 3894k in [13]. The proposed model achieved an average accuracy

TABLE II
Model Comparison and Ablation Study on the RadioML2018 Dataset with 11 classes.

Model	Accuracy	#Param.	Memory Size
ResNet [25]	66.0%	257,009	1000 KB
PET-CGDNN [9]	74.1%	71,614	290 KB
CGDNet [13]	70.6%	3,894,133	15580 KB
SNN [21]	64.3%	83,000	166 KB
DAE [15]	67.3%	14,989	60 KB
Ablation Study			
CNN [17]	70.1%	295,055	1155 KB
CCL+CNN	71.9%	37,016	150 KB
CNN*	69.9%	148,688	590 KB
CCL+CNN*	71.7%	19,673	80 KB
ATT+CNN*	71.5%	148,976	600 KB
CCL+ATT+CNN* (This Work)	**73.3%**	**19,961**	**80 KB**
Quantized Model (This Work)	**71.5%**	**19,961**	**20 KB**

CCL: Cross-Component Learning, ATT: Attention,
CNN*: refers to filter reduction and maxpooling.

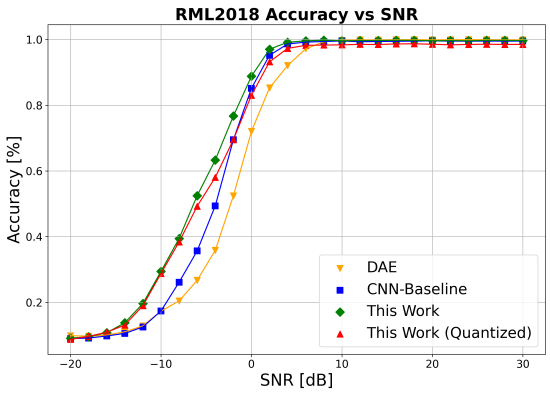

Fig. 6. Accuracy vs. SNR curve on the RML2018 dataset for normal classes. Quantization reduces accuracy, but when the SNR is relatively high, the difference is minimal.

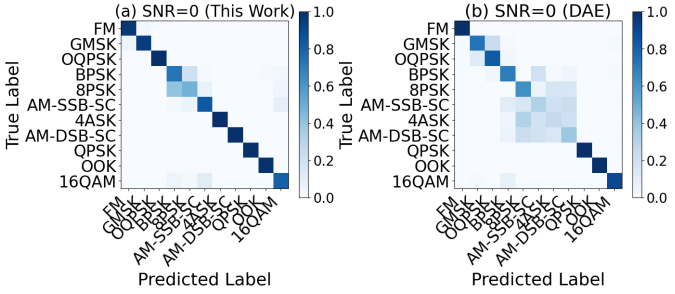

Fig. 7. The confusion matrix at SNR = 0 dB. (a) This work, (b) DAE [15].

across all SNR values of 73.3% on the RML2018 dataset, surpassing the CNN baseline by 3.2% and outperforming the DAE model by 6%, as shown in Table II. Moreover, the model achieves a remarkable 93.2% reduction in parameters compared to the CNN baseline [17]. Fig. 6 demonstrates its very good performance for low SNRs, reaching 99% of accuracy for an SNR of 4 dB. In contrast, the DAE model only achieves similar performance above 8 dB, underscoring the robustness of the proposed model in challenging noise conditions. Fig. 7 further illustrates the model's superior accuracy at 0 dB.

An ablation study was conducted to evaluate the impact of

TABLE III
COMPARISON OF STATE-OF-THE-ART FPGA IMPLEMENTATIONS FOR RF MODULATION RECOGNITION.

	FPGA	Clock (MHz)	NN	Weights # of Bits	LUT	FF	DSP	Power (mW)	Performance GOPS	Efficiency (GOPS/W)
This Work[1]	**ZCU104**	**115**	**CNN**	**8**	**75365**	**88623**	**1728**	**1191**	**61**	**51**
[17][2]	ZCU104	70	CNN	16	74680	57726	1116	847	33	39
[21][1]	PYNQ	137	SNN	16	31735	50,934	0	2167	79	35
[20][2]	XCZU5EG	200	CNN	8	67779	-	131	858	23	27
[18][2]	ZCU102	250	CNN	16	97900	139200	578	10500	179	17
[16][3]	XCZU9EG	-	ANN	16	158435	16222	210	1152	15	13

[1] RadioML2018 Dataset [22] with 11 classes, [2] RadioML2016 Dataset [23] with 11 classes, [3] Dedicated Dataset [16] with 6 classes.

TABLE IV
INFERENCE TIME PER INPUT FRAME ACROSS HARDWARE PLATFORMS FOR THE PROPOSED AI MODEL, COMPARED TO THE STATE OF THE ART.

Device	FPGA	GPU	ARM CPU
This Work	**10.1** μs	**13.4** μs (A100)	**331.5** μs (A76)
[17]	26.8 μs	36.6 μs (P100)	-
[20]	26.5 μs	7694 μs (Jetson)	14774 μs (A9)
[15]	-	753.3 μs (1080Ti)	4149.4 μs (A72)

the CCL layer, the customized attention (ATT) mechanism, along with filter reduction and the insertion of a max-pooling layer (CNN*). The results highlight the clear advantages of the proposed model. As shown in Table II, the introduction of CCL significantly reduced the model's parameter count while improving accuracy from 70.1% to 71.9%. Further optimization involved reducing the first CL's filters from 45 to 36 and adding a max-pooling layer after the second convolutional layer. This adjustment nearly halved the number of parameters, with only a minimal accuracy decrease of 0.2%.

ATT was applied to dynamically focus on important features learned in the earlier layers. When used independently, both CCL and ATT improved accuracy, CCL raised it from 69.9% to 71.7%, and ATT increased it to 71.5%. However, when ATT was combined with CCL, accuracy rose significantly to 73.3%, marking a 3.2% improvement over the baseline. Remarkably, these enhancements also reduced the parameter count by 93.2%, from 295,055 to just 19,961. These findings underscore the lightweight nature and high accuracy of our proposed design, highlighting its effectiveness in resource-constrained environments.

B. Quantization

Post-training quantization was applied to quantize weights in 8 bits and biases in 16 bits to exploit the faster integer computations compared to floating-point. The test data were quantized to 12 bits, simulating a 12-bit ADC. After the MAC operations, data were re-quantized to 16 bits via right-shifting. The quantized model achieved 71.5% accuracy, reducing the memory footprint to just 20 KB, while surpassing the CNN baseline accuracy by 1.4% and the DAE model accuracy by 4.2%, as shown in Table II.

V. HARDWARE RESULTS

The proposed architecture has been implemented in VHDL and tested it on a Zynq ZCU104 board with Xilinx Vivado 2020.2 version. Table III compares this work with several hardware implementations using similar datasets. The Spiking Neural Network (SNN) implementation in [21] achieves 79 GOPS with 0 DSP block but since it utilizes high-level synthesis, the power consumption is relatively high. The implementation proposed in [18] has a significantly higher throughput of 179 GOPS but at the cost of a much higher power consumption. The implementation in [17] achieved the lowest power consumption of 847 mW but with a limited performance of 33 GOPS. The proposed design can realize 61 GOPS while consuming only 1191 mW, thus achieving the highest efficiency of 51 GOPS/W.

In Table IV, we list the measured inference time per input frame of the proposed model on different hardware platforms: FPGA, GPU and CPU. The proposed FPGA implementation achieved an inference speed of 10.1 μs per frame, even outperforming the original PyTorch model tested on a premium grade A100 GPU (13.4 μs). Leveraging SIMD processing, we tested the model on an ARM Cortex-A76 processor, a high-performance embedded chip optimized for AI applications. The FPGA implementation is 33x faster than the ARM processor and 1.3x faster than the GPU. It is also significantly more power-efficient consuming 1191 mW, while the ARM processor consumes 750mW/Core and the GPU has a consumption in the range of 250-400W.

CONCLUSION

In this paper, we presented a lightweight CNN model dedicated to RF spectrum monitoring and modulation recognition in edge devices. Compared to prior art, the model demonstrates superior classification accuracy while significantly reducing the number of parameters. This is achieved by integrating into the CNN model a customized attention mechanism and a cross-component learning strategy. The FPGA implementation achieves real-time processing with low power consumption. It is much faster than a CPU implementation and consumes significantly less power than a GPU. Compared to other FPGA implementations, the proposed model achieves the best efficiency of 51 GOPS/W. In terms of future work, we plan to train the same CNN model for other cognitive tasks in RF communication, such as the detection of covert communication channels [26].

REFERENCES

[1] A. Jagannath, J. Jagannath and T. Melodia, "Redefining Wireless Communication for 6G: Signal Processing Meets Deep Learning With Deep Unfolding", IEEE Transactions on Artificial Intelligence, vol. 2, no. 6, Dec. 2021.

[2] J. M. de la Rosa, "AI-Managed Cognitive Radio Digitizers," IEEE Circuits and Systems Magazine, vol. 22, no. 1, pp. 10-39, Firstquarter 2022.

[3] P. D. Choudhury, A. Bora, and K. K. Sarma, "Big spectrum data and deep learning techniques for cognitive wireless networks," Deep Learning and Neural Networks: Concepts, Methodologies, Tools, and Applications, I. R. Management Association Ed. Hershey, PA, USA: IGI Global, 2020.

[4] A. Ali and F. Yangyu, "Automatic modulation classification using principle composition analysis based features selection," 2017 Computing Conference, London, UK, 2017, pp. 294-296.

[5] S. Kumar, V. A. Bohara and S. J. Darak, "Automatic modulation classification by exploiting cyclostationary features in wavelet domain," 2017 Twenty-third National Conference on Communications (NCC), Chennai, India, 2017, pp. 1-6.

[6] S. Hochreiter and J. Schmidhuber. "Long Short-Term Memory", Neural Computation, Nov. 1997.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in Advances in Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, Dec. 2017.

[8] H. Yang, L. Zhao, G. Yue, B. Ma and W. Li, "IRLNet: A Short-Time and Robust Architecture for Automatic Modulation Recognition," IEEE Access, vol. 9, 2021.

[9] 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 Communications Letters, vol. 25, no. 10, Oct. 2021.

[10] T. Huynh-The, C. -H. Hua, Q. -V. Pham and D. -S. Kim, "MCNet: An Efficient CNN Architecture for Robust Automatic Modulation Classification," IEEE Communications Letters, vol. 24, no. 4, April 2020.

[11] J. Xu, C. Luo, G. Parr and Y. Luo, "A Spatiotemporal Multi-Channel Learning Framework for Automatic Modulation Recognition," IEEE Wireless Communications Letters, vol. 9, no. 10, Oct. 2020.

[12] A. P. Hermawan, R. R. Ginanjar, D. -S. Kim and J. -M. Lee, "CNN-Based Automatic Modulation Classification for Beyond 5G Communications," IEEE Communications Letters, vol. 24, no. 5, May 2020.

[13] J. N. Njoku, M. E. Morocho-Cayamcela and W. Lim, "CGDNet: Efficient Hybrid Deep Learning Model for Robust Automatic Modulation Recognition," IEEE Networking Letters, vol. 3, no. 2, Jun. 2021.

[14] M. Kulin, T. Kazaz, I. Moerman and E. De Poorter, "End-to-End Learning From Spectrum Data: A Deep Learning Approach for Wireless Signal Identification in Spectrum Monitoring Applications," IEEE Access, vol. 6, Jun. 2018.

[15] Z. Ke and H. Vikalo, "Real-Time Radio Technology and Modulation Classification via an LSTM Auto-Encoder," IEEE Transactions on Wireless Communications, vol. 21, no. 1, Jan. 2022.

[16] S. Soltani, Y. E. Sagduyu, R. Hasan, K. Davaslioglu, H. Deng and T. Erpek, "Real-Time and Embedded Deep Learning on FPGA for RF Signal Classification," IEEE Military Communications Conference, MILCOM, Norfolk, VA, USA, 2019.

[17] A. Emad, H. Mohamed, A. Farid, M. Hassan, R. Sayed, H. Aboushady, H. Mostafa, "Deep Learning Modulation Recognition for RF Spectrum Monitoring," IEEE International Symposium on Circuits and Systems, ISCAS, Daegu, Korea, May 2021.

[18] K. Jung, J. Woo and S. Mukhopadhyay, "An On-chip Accelerator with Hybrid Machine Learning for Modulation Classification of Radio Frequency Signals," IEEE/MTT-S International Microwave Symposium - IMS 2022, Denver, CO, USA, 2022.

[19] K. Jung, J. Woo and S. Mukhopadhyay, "On-Chip Acceleration of RF Signal Modulation Classification With Short-Time Fourier Transform and Convolutional Neural Network," IEEE Access, vol. 11, 2023.

[20] K. Zhang, B. Gao, B. Yang, Y. Ji, F. Gao and Y. Li, "Real-Time Automatic Modulation Recognition Based on FPGA," 2023 9th International Conference on Computer and Communications (ICCC), Chengdu, China, 2023, pp. 1440-1444.

[21] W. Guo, K. Yang, H. -G. Stratigopoulos, H. Aboushady and K. N. Salama, "An End-To-End Neuromorphic Radio Classification System With an Efficient Sigma-Delta-Based Spike Encoding Scheme," IEEE Transactions on Artificial Intelligence, vol. 5, no. 4, Apr. 2024.

[22] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," IEEE Journal of Selected Topics in Signal Processing, vol. 12, no. 1, pp. 168-179, 2018.

[23] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," Springer Engineering Applications of Neural Networks, Aberdeen, UK, Sep. 2016.

[24] J. Hu, L. Shen and G. Sun, "Squeeze-and-Excitation Networks," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018.

[25] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778.

[26] A. R. Díaz-Rizo, A. Abdelazim, H. Aboushady and H.-G. Stratigopoulos, "Covert Communication Channels Based On Hardware Trojans: Open-Source Dataset and AI-based Detection," IEEE International Symposium on Hardware Oriented Security and Trust (HOST), Tysons Corner, VA, USA, 2024, pp. 101-106.