Hardware Acceleration Techniques for Automatic Modulation Classification: A Survey

Nandini R. Kendre[1], Dr. Jitesh R. Shinde[2], Mr. T. A. Mohije[3], Dr. Ganesh B. Dongare[4]
[1,2,3,4] *Department of Electronics and Communication, CHH. Shahu College of Engineering, Chh.Sambhajinagar, India*

***Abstract*—AMC is an integral part of present-day wireless communication systems that allow recognizing modulation schemes without any prior information about parameters of the transmitter. As a result of fast expansion of 5G/6G, cognitive radio, software-defined radio, and intelligent spectrum monitoring, the necessity of AMC algorithms with low latency and throughput has been greatly increased. Despite the high classification performance provided by traditional methods based on likelihood, features, and machine learning, the high computational complexity of these approaches has prevented their use in real time on common processing units. Hardware acceleration can be used as a method of overcoming these problems. The aim of this survey is to make a review of hardware acceleration methods for AMC, including CPU-, GPU-, ASIC-, and FPGA-based approaches. Recent developments in machine learning and deep learning-based AMC approaches and their hardware implementation are considered as well. FPGA architectures, AXI4-Stream approaches, hardware–software co-design techniques, and performance metrics are thoroughly analyzed.**

***Index-Terms*—Automatic Modulation Classification (AMC), Hardware Acceleration, Field-Programmable Gate Array (FPGA), Graphics Processing Unit (GPU), Application-Specific Integrated Circuit (ASIC), Machine Learning, Deep Learning, AXI4-Stream, Hardware–Software Co-Design, Wireless Communication, Cognitive Radio, Software-Defined Radio (SDR).**

I. INTRODUCTION

Wireless communication technologies have shown remarkable growth in the last two decades owing to the development and deployment of 5th generation (5G) networks, the evolution of 6th generation (6G) communication systems, Internet of Things (IoT), cognitive radio, Software Defined Radio (SDR), Unmanned Aerial Vehicles (UAVs), satellite communications, and intelligent sensing applications. The aforementioned technologies demand the ability to work efficiently in a dynamic and heterogeneous RF environment, where many users, different modulation schemes, diverse channel conditions, and limited spectrum are present. In such scenarios, it is necessary that communication receivers quickly classify the unknown signals and adapt the processing techniques according to the changing conditions. Automatic Modulation Classification (AMC) is among those technologies that have gained much importance to build the intelligent wireless communication systems. The basic aim of Automatic Modulation Classification is to automatically classify the

modulation scheme of a received signal without complete information about the transmitter. Successful identification of the modulation scheme facilitates adaptive demodulation, dynamic spectrum access, signal intelligence, electronic warfare, interference monitoring, cognitive radio, and spectrum surveillance, which makes AMC essential technology for contemporary communication systems. Traditional AMC schemes are broadly categorized into two categories: likelihood-based and feature-based classifiers. Likelihood-based classifiers use probabilistic decision theories However, this can be quite difficult because such methods require computationally intensive processes like probability density function estimation, likelihood calculation, and hypothesis testing. In feature-based modulation classification, the representative features of the signal are determined first, like higher-order cumulants, cyclostationary features, spectral features, or

statistical moments, after which decision algorithms are applied. This is comparatively easier computation-wise but highly depends on the extraction of good quality features, especially in low SNR conditions. With the fast development in Artificial Intelligence (AI) and Machine Learning (ML), new approaches to modulation classification have emerged. Deep learning methods based on Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) network, and Transformer have shown great classification performance through automatic extraction of signal representations from raw in-phase and quadrature (I/Q) samples. However, the computational complexity, memory needs, and inference delay inherent in the deep learning algorithm pose major problems in the deployment of such algorithms to resource-limited and real-time communication systems. In order to solve such challenges, hardware acceleration has been proposed as one of the major research directions for implementing AMC. Dedicated hardware devices can greatly minimize the computational delay and increase the throughput and energy efficiency. Some of the existing hardware acceleration technologies include CPUs, GPUs, ASICs, and FPGAs. Although CPUs have the advantage of programming flexibility, they have minimal levels of parallelism in the execution of computationally expensive signal processing operations. On the other hand, GPUs utilize thousands of parallel processing cores and have become popular in deep learning inference; however, high power consumption remains one of their main limitations in embedded communication systems. The ASIC is very fast and energy efficient; however, it takes a long time to develop an ASIC, it involves non-recurring engineering cost, and it cannot be reconfigured after fabrication. On the other hand, the FPGA presents an interesting trade-off between flexibility, hardware parallelism, deterministic behavior, and energy efficiency, making them very promising for real-time communication applications. Nowadays, FPGA devices are equipped with configurable logic blocks, memories, digital signal processing (DSP) blocks, high-speed transceivers, and embedded processing systems which allow building extremely parallel and pipelined hardware systems. Also, standard communication interfaces like AXI4 and AXI4-Stream can efficiently exchange data streams between different hardware components. All these factors have motivated many researchers to study FPGA acceleration of AMC algorithms. Despite considerable advancement made in the last few years, current literature lacks coherence, as it is spread over different algorithmic frameworks, hardware, data sets, evaluation techniques, and performance measures. Most of the survey papers concentrate on algorithms used for modulation classification and offer little information regarding the hardware aspect. Similarly, the hardware-based papers mainly concentrate on FPGA framework or deep learning acceleration without exploring other forms of hardware acceleration solutions. With advancements in communication systems toward 6G and edge intelligence, along with the demand for real-time spectrum sensing, a need has arisen for a survey paper examining the relation between AMC algorithms and hardware acceleration solutions.

This survey gives a thorough overview of hardware acceleration solutions for AMC, including conventional statistical methods, machine learning and deep learning methods, and their hardware implementations. In addition, this survey includes an assessment of CPUs, GPUs, ASICs, and FPGA-based hardware architectures, the methodology of hardware-software co-design, comparison of current FPGA implementations, the problems in current research, and the future directions for further research. Combining the state-of-the-art of both algorithms and hardware architectures, this survey is designed to help researchers and engineers get a clear vision of the efficient AMC implementation for intelligent future wireless communications systems.

The key contributions of this survey can be listed as follows:

- An exhaustive survey of conventional AMC and AMC through machine and deep learning approaches.
- An intensive study of various hardware accelerators such as CPUs, GPUs, ASICs, and FPGAs for efficient real-time AMC design.
- A systematic survey of state-of-the-art FPGA-based AMC designs, focusing on parallelism, pipeline architecture, AXI4-Stream interface, and hardware-software co-design.

- A comparative study of the existing implementations of AMC with respect to accuracy, latency, throughput, usage of FPGA resources, operating frequency, and power consumption.
- The identification of challenges and problems that exist in the researhe discussion of future research areas such as Edge AI, TinyML, adaptive hardware architectures, pach area of hardware-assisted modulation classification.
- Trtial reconfiguration, and 6G-enabled intelligent communication systems.

II. FUNDAMENTALS OF AUTOMATIC MODULATION CLASSIFICATION

Automatic Modulation Classification (AMC) is a very basic signal processing method which facilitates the process of automatically identifying the modulation scheme used by a received communication signal without any need of prior information about the transmitter parameters. AMC works as an indispensable tool for intelligent wireless communication systems in which there is a necessity for instantaneous recognition of the characteristics of the signal for the purposes of adaptive communication, spectrum awareness, electronic warfare, cognitive radio and software defined radio (SDR). With the development of wireless networks towards 5G and 6G technology and beyond, it has become increasingly crucial to classify unknown modulation schemes [1]. The main aim of AMC lies in identifying the most likely modulation technique among the predetermined set of modulation techniques based on the features of the received signal. Based on the communication environment, the received signal can be disturbed by additive noises, multipath fading, interference, synchronization errors, Doppler effects, and so forth. Hence, the process of classifying the modulation technique becomes difficult and a complex pattern recognition problem. Therefore, a successful AMC should attain high classification accuracy with minimum complexity [2]. The processing in the AMC process involves several successive processing steps that can be visualized in Figure 1. At first, the analogue input signal gets transformed into the I and Q digital samples by way of analogue-to-digital conversion. Preprocessing of the samples includes processes of noise removal, synchronization, frequency offset correction, timing adjustment, and normalization. After the samples have been preprocessed, the classifier retrieves the characteristic properties from the signal or computes the statistical parameters necessary for classification. The last step in the process is taken by the decision module, which determines the modulation scheme of the received signal [3].

Based on the chosen methodology, the intermediate step in processing can consist of handcrafted feature extraction, computation of the probability, or learning of features by means of machine learning. The conventional feature-based methods are based on the calculation of some statistics like higher-order cumulants, spectrum features, cyclostationary properties, or moments of the signal. The methods based on the computation of the probability calculate the probability of the received signal belonging to the certain modulation hypothesis. The recent developments in machine learning and deep learning allow end-to-end learning of signal representations without reliance on handcrafted features [4],[5].

The AMC selection problem can be modeled as selecting the modulation scheme that has the maximum posterior probability conditioned on the received signal observations. If the received signal is represented as r, and Mi is the ith possible modulation class, then the modulation scheme is given by:

$$\widehat{M} = \underset{\text{Mi}}{arg\,max}\ P(\text{M}_\text{i} \mid \text{r})$$

where $P(M_i \mid r)$ is the posterior probability of the observed signal being modulated using the modulation type Mi. The classifier identifies the modulation type with the highest posterior probability out of all the modulation types.

Depending on the classification algorithm, AMC algorithms can be broadly divided into three categories: likelihood-based methods, feature-based methods, and machine learning methods. Likelihood-based classifiers use probabilistic models and Bayesian decision rules to attain accurate classification results; however, they involve a lot of computation power. On the other hand, feature-based classifiers make use of carefully selected signal features in order to minimize computation power;

yet, the classification result depends on how well the features are extracted. Machine learning and deep learning algorithms learn the discriminatory features in the communication signals and attain amazing classification results in complicated wireless environments [6]. The rise of computational challenges presented by the current AMC algorithms has resulted in numerous efforts towards developing hardware acceleration architectures suitable for real-time operation. One of such promising platforms is the Field-Programmable Gate Array, which benefits from the potential to perform simultaneous parallel computations, pipeline implementation, determinism, and low-power consumption. These features allow FPGA-based AMC architectures to efficiently compute several signal samples and hypotheses on the go, ensuring that the AMC algorithm meets the challenging constraints of future wireless communication systems.

Thus, the comprehension of AMC basics is critical for the evaluation of both traditional and current AMC methods. In what follows, the paper provides an overview of classical AMC algorithms, AMC based on machine learning, and hardware acceleration architectures used in AMC systems.

III. CONVENTIONAL AUTOMATIC MODULATION CLASSIFICATION TECHNIQUES

Prior to the development of machine learning (ML) and deep learning (DL) based methods, AMC was mainly tackled by means of classical signal processing and statistical decision approaches. Such approaches are based on the fundamental theory behind modulation recognition and have been studied widely owing to their mathematical soundness, interpretability, and moderate complexity of implementation. Classical AMC approaches normally depend on the analytical model of the communications signals and employ some of their statistical, spectral, or probabilistic properties for classification. In spite of the quick progress in the field of data-driven methods, many classical AMC approaches still remain benchmark algorithms for modern AMC systems [7].

Traditionally, the AMC techniques have two main classes: the likelihood approach and the feature approach. The likelihood approach calculates the probability for every possible modulation hypothesis to which the received signal might belong and then chooses the modulation scheme having the maximum likelihood value. The above approach relies on Bayesian decision theory and is close to optimality if accurate channel state information is known. Nevertheless, the calculation of probability density functions, likelihood values, and hypotheses testing involves high computational overheads, thus making it hard to implement this technique in real time, especially for numerous modulation hypotheses [8]. Contrarily, feature based modulation recognition techniques involve first extracting the relevant features from the input signal followed by the classification stage. Some commonly used features include high order cumulants, statistics, cyclostationary features, spectrum-related features, instantaneous amplitude, phase, and frequency domain features. The features that have been extracted are then compared with certain predefined thresholds or provided to conventional classifiers for modulation recognition purposes. In comparison to the likelihood based techniques, feature based techniques do not require many computational resources and are easy to implement on an embedded hardware platform. However, their classification efficiency is highly dependent upon the relevance and robustness of the features selected [9]. There have also been several statistical classification algorithms which can be considered an approach midway between the likelihood and the feature based approach. These classification algorithms take advantage of the statistical properties of signals, such as higher order moments or decision boundaries obtained via statistics. In general, the statistical classifiers make a good compromise between complexity and accuracy [10]. The traditional AMC algorithms have proven to work well over the past three decades in a variety of communications environments and have been used successfully in radar, satellite communications, spectrum sensing, cognitive radio, electronic warfare, and software defined radio. The fairly simple mathematical structure of the traditional AMC algorithms has enabled their implementation in DSPs, microprocessors, and hardware accelerators built using FPGAs. As the wireless communications systems keep progressing towards higher data rates and complex channels, the limitations of the

traditional AMC approaches have become clearer. The sequential processing, synchronization problems, dependence on accurate signal models, and inability to adapt to complex channel environments have resulted in the creation of intelligent classification methods [11].

In other words, recent studies have paid increased attention to using machine learning and deep learning methods in the design of AMC systems. The use of data-driven techniques does not require manually crafted decision rules but instead learns discriminative features from the communication signals automatically. On the other hand, the high computational costs associated with advanced learning methods have also led to the advancement of dedicated hardware implementations, such as FPGA-based accelerators, to meet the high latency and throughput constraints of future wireless communication systems. Considering their basic methodology, the most prominent classes of traditional AMC techniques presented in this survey include likelihood-based classification, feature-based classification, higher-order cumulant-based classification, and statistical decision-based classification, explained in subsequent subsections.

IV. MACHINE LEARNING-BASED AUTOMATIC MODULATION CLASSIFICATION (AMC)

There have been immense advancements in Artificial Intelligence (AI) and Machine Learning (ML) that have brought about drastic changes in the field of Automatic Modulation Classification (AMC). Unlike traditional methods where the signal model was probabilistic or featured handcrafted features, machine learning algorithms allow automatic identification of the discriminative features within the communication signals. These properties make machine learning a good choice for future wireless communications [12].

In the contemporary communication networks including 5G, future generation 6G, cognitive radio, software defined radio (SDR), Internet of Things (IoT), and satellite communication, the received signals may be contaminated with noise, multipath fading, interference, frequency offset, and hardware issues. However, traditional AMC algorithms may face degradation in performance under such scenarios since they require high accuracy in signal model or feature extraction method used. Machine learning can resolve most of these issues since it learns the characteristics of the signal directly from the training data [13], [4].

The machine learning algorithms used for AMC can broadly be classified into conventional machine learning algorithms and deep learning algorithms. The conventional machine learning algorithms consist of ANN, SVM, DT, RF, and k-NN algorithms, which rely on manually extracted signal features such as higher order statistics cumulants, statistical moments, spectral properties, or cyclostationary features. The deep learning algorithms include CNN, RNN, LSTM, autoencoders, and transformer-based algorithms that get rid of the requirement of manual extraction of signal features by learning hierarchical features from raw I/Q signal samples [14].

Among various deep learning methods, there is a large number of studies that focused on CNN-based classifiers since they show excellent results in learning spatial patterns from communication signals. CNNs use several convolutional layers to find local features of the signal while decreasing the number of dimensions using pooling operations. Such models have achieved high levels of accuracy for different modulation types and Signal-to-Noise Ratio (SNR). Recurrent networks, such as RNN and LSTM, have been effective in finding temporal patterns in sequence-based communication signals. Moreover, recent transformer architectures with self-attention mechanism can help to achieve better AMC results by taking into account relations between the distant signal samples. However, they require significantly higher computational and memory resources [15].

Though they can perform better than traditional methods in terms of classification accuracy, there are some issues related to implementing ML-based AMC algorithms. Deep neural networks may comprise several millions of parameters that are used for learning. For this reason, when they are implemented in software, the inference procedure becomes more time-consuming and less efficient, thus, cannot be used in real-time communications systems. However, GPU is capable of accelerating computations, but it consumes more power than FPGA, thus, it is not a preferable solution for implementing wireless

embedded devices. These problems led to the emergence of hardware acceleration, in particular, FPGAs [16].

The implementation of AMC using machine learning, therefore, is successful not just because of classification accuracy but also because of efficient hardware implementation. Hardware accelerators such as FPGAs can enable the parallel processing of neural network calculations along with pipelining and optimal use of resources, thus making them perfect candidates for real-time modulation classification in communication systems. This is why there has been an increasing emphasis on utilizing machine learning algorithms with hardware accelerator methods [17].

Advantages of Machine Learning-Based AMC

- Automatically learns discriminative features from communication signals.
- Eliminates or significantly reduces manual feature engineering.
- Provides high classification accuracy across multiple modulation schemes.
- Adapts effectively to noisy and dynamic wireless environments.
- Supports classification of complex and high-order modulation formats.
- Enables end-to-end learning directly from raw I/Q samples.
- Improves scalability for future wireless communication systems.

Limitations of Machine Learning-Based AMC

- Requires large labeled datasets for effective training.
- Deep learning models involve high computational complexity.
- Memory requirements increase with network depth.
- Software implementation introduces significant inference latency.
- GPU implementations consume relatively high power.
- Hardware acceleration is often necessary for real-time deployment.
- Model optimization is required for resource-constrained embedded FPGA platforms [17],[3].

V. HARDWARE ACCELERATION PLATFORMS FOR AUTOMATIC MODULATION CLASSIFICATION

Advances in the increasing computational complexity of Automatic Modulation Classification (AMC) have necessitated the use of hardware acceleration platforms that can meet the rigorous demands of latency and throughput in the current wireless communication systems. Traditional implementations using software processors have been challenged by their inherent sequential approach of processing large amounts of communication data in real-time. As such, special-purpose hardware platforms like Central Processing Units (CPUs), Graphics Processing Units (GPUs), Field Programmable Gate Arrays (FPGAs), and Application Specific Integrated Circuits (ASICs) have been utilized to achieve faster AMC computation.

5.1 CPU-Based Acceleration

Central Processing Units (CPUs) are still the most commonly adopted hardware architecture to run AMC algorithms due to its programmability, well-developed software infrastructure, and ease of implementation. Most classical likelihood-based, feature-based and machine learning algorithms have been first tested through CPU implementation in frameworks like MATLAB, Python or C/C++. CPUs are quite flexible in terms of algorithm development and testing, which makes them appropriate for offline research purposes.

Nevertheless, CPUs are highly sequential machines without efficient parallelism capabilities, which is becoming a major problem in case of large datasets and complex deep learning models processing. Due to the growth of the number of modulation classes, samples, and complexity of models, CPU-based implementation experiences higher execution time and lower throughput. Consequently, CPUs are not suitable for real-time AMC systems due to non-deterministic processing delays [18].

5.2 GPU-Based Acceleration

Graphics Processing Units (GPUs) possess the ability to deliver huge data-level parallelism using thousands of small and light-weighted processing units that can carry out arithmetic computations concurrently. It is

due to this reason that GPUs have been highly successful in speeding up AMC models based on deep learning such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Transformer models.

However, despite being extremely computationally intensive, GPUs consume significantly higher energy as compared to CPUs and FPGAs. In addition to it, the performance of GPUs has limitations in the form of memory transfer overhead between host and device, causing increased latency in embedded communication systems. As a result, GPU-based AMC solutions find use in cloud computing and offline model training only [19].

5.3 FPGA-Based Acceleration

One such technology which is considered highly suitable for hardware accelerators for AMC in real-time scenarios is the Field Programmable Gate Array (FPGA). This is due to the fact that they possess hardware parallelism, re-configurability, and efficiency. Contrary to CPUs and GPUs, FPGA systems enable the implementation of algorithms in dedicated hardware blocks allowing for simultaneous execution of multiple computational modules. Pipeline architecture enhances this aspect allowing the processing of consecutive sample signals simultaneously [20].

Modern FPGA devices have configurable logic blocks (CLB), lookup tables (LUT), flip-flops (FF), Digital Signal Processing (DSP) slices, Block RAM (BRAM), and high-speed communication interfaces like AXI4 and AXI4-Stream. Such resources help implement signal pre-processing, feature extraction, likelihood calculation, neural network inference, and communication interface in one programmable device.

The FPGA-based likelihood AMC allows evaluating multiple modulation hypotheses simultaneously, and that allows significantly decreasing classification latency comparing to sequential calculations on a computer. Moreover, standard streaming interfaces like AXI4-Stream enable transferring information continuously from one hardware block to another, which makes FPGA an optimal choice for low-latency communication systems. Thanks to the balance between flexibility, performance, and power efficiency, FPGAs have become the platform of choice for SDR, cognitive radio, radar, satellite communications, and upcoming 5G/6G developments [21].

5.4 ASIC-Based Acceleration

Application-Specific Integrated Circuits (ASICs) offer the best combination of performance and low power consumption amongst all hardware accelerator technologies since the hardware is customized for an application during the manufacturing process. Implementation of ASICs reduces the overhead due to programmable hardware, thus offering better operating frequency, lower silicon area, and better energy consumption.

While ASICs are ideal for communication products that can be produced in large volumes, the technology is plagued by the drawbacks of high design complexity, high manufacturing costs, and lengthy design times. In addition, since ASICs cannot be reprogrammed once manufactured, they are not suitable for changing communication standards and changing algorithms in research settings. Therefore, ASIC based AMC implementations can only be considered for established applications [22].

5.5 Comparison of Hardware Acceleration Platforms

All hardware platforms have different compromises concerning flexibility, computation power, power efficiency, and cost of implementation. The CPUs can be programmed to the highest level but with limited parallelism. On the other hand, the GPUs can handle deep learning computations very well but consume a lot of power. The ASICs have unparalleled efficiency and power but cannot be used flexibly. FPGAs come in handy since they bring together all these attributes.

Table 1. Comparison of Hardware Acceleration Platforms for Automatic Modulation Classification

Platform	Parallelism	Performance	Power Consumption	Reconfigurability	Development Cost	Suitability for Real-Time AMC
CPU [23]	Low	Moderate	Moderate	High	Low	Moderate
GPU [24]	Very High	Very High	High	High	Moderate	High (mainly AI models)
FPG	High	High	Low	High	Moderat	Very

Platform	Parallelism	Performance	Power Consumption	Reconfigurability	Development Cost	Suitability for Real-Time AMC
A [25]					e	High
ASIC [26]	Very High	Very High	Very Low	None	Very High	Very High (fixed)

5.6 Summary

Hardware acceleration has now been a crucial component for the implementation of computationally complex AMC algorithms in today's wireless communication systems. Though CPUs are the best for designing and developing the algorithms and GPUs are the best for deep learning training, FPGAs serve as the best platform when it comes to executing the algorithms on real-time basis due to their high speed, less delay time, deterministic behavior, and power efficiency. ASICs offer the highest performance, but lack in flexibility and need higher development expenses. In view of this, FPGA-based acceleration has come into the picture.

VI. FPGA-BASED AUTOMATIC MODULATION CLASSIFICATION ARCHITECTURES

The increasing need for real-time wireless communication systems has increased the studies on AMC architectures using FPGAs. In comparison with general-purpose processors executing software implementations of AMC, FPGA implementation takes advantage of hardware parallelism, pipelining, and special-purpose arithmetic circuits in order to provide a better latency, throughput, and energy consumption. Additionally, the reconfigurability feature of FPGAs allows designers to develop an architecture customized according to the specific AMC algorithm.

AMC architectures that have been implemented using FPGAs could generally be categorized into four main types: architectures based on features, architectures based on probabilities, architectures based on machine learning, and architectures based on deep learning.

6.1 Feature-Based FPGA Architectures

The feature-based implementation of FPGAs classifies the modulated signals by analyzing their representative characteristics prior to performing the decision-making process. Some of the commonly extracted characteristics are higher order cumulants, statistical moments, cyclostationary characteristics, wavelet coefficients, and spectrum properties. As only a few characteristics are analyzed, these kinds of architectures generally have lower computational complexity than the likelihood-based architectures.

Some researchers have implemented feature extraction modules on the FPGA platform. Hardware blocks perform the task of filtering and characteristic calculations using the threshold-based approach. These blocks consume moderate amounts of resources and have relatively lower power requirements, and therefore, they are suited for communication devices. However, such types of architectures depend greatly on the quality of the extracted characteristics, and hence their classification is affected at low SNRs or dynamically changing wireless environments [27].

6.2 Likelihood-Based FPGA Architectures

Probability-based FPGA designs use likelihood based algorithms to make decision in hardware directly on the basis of likelihood computation of various modulation hypotheses. The signal received is compared against the predetermined statistics and the modulation scheme that produces the highest likelihood is chosen as the output classification result.

Despite the fact that likelihood based approaches are more accurate than others in terms of classification, their use involves complex arithmetic operations like computing probabilities, calculating metrics, and comparing hypotheses. The problem can be solved through using parallel computation and pipelining to allow multiple computations of the probability to be performed at the same time. Several works have shown that hardware parallelism allows reducing the time needed for classification without compromising its throughput [28].

6.3 Machine Learning-Based FPGA Architectures

FPGA architecture using machine learning techniques is used for speeding up conventional classifiers like Artificial Neural Network (ANN),

Support Vector Machine (SVM), Decision Tree, Random Forest, and k-Nearest Neighbour (k-NN). The hardware implementation is mostly used for feature extraction, matrix multiplication, activation function and classification operations.

Implementation of machine learning using FPGA technology shows that these systems offer much faster inference time than traditional software implementations and use less energy compared to GPU-based machines. But most conventional machine learning techniques need manual feature extraction and therefore are unable to adapt to different communication settings. Also, the hardware gets more complicated with larger datasets and high dimensional feature vectors [29].

6.4 Deep Learning-Based FPGA Architectures

Recent studies have paid more attention to enhancing the performance of AMC based on deep learning models using FPGA technology. It has been shown that CNN, RNN, LSTM networks, and Transformer models exhibit excellent classification accuracy by learning discriminative representations of signals from the input IQ samples or constellation images.

To deploy deep learning models using FPGA technology, the use of various optimization methods, including network pruning, weight quantization, pipelining, loop unrolling, and parallel processing, is necessary to minimize computation complexity and memory requirements. Dedicated hardware modules for convolution, activation functions, pooling, and fully-connected layers have been designed. Although the utilization of FPGAs in terms of performance improvement compared with GPUs is very effective, the deployment of large-scale deep learning models is quite difficult due to the limited memory and hardware resources [30].

6.5 Streaming FPGA Architectures Using AXI4-Stream

For more efficient processing, the majority of new-generation AMC solutions implemented using FPGAs apply streaming architecture, which is designed according to the AXI4-Stream protocol. In contrast to traditional methods where the intermediate results are stored in external memory, the signal samples are streamed through hardware blocks via special standardized streaming interfaces. The AXI4-Stream protocol makes use of such handshaking signals as TVALID, TREADY, TDATA, and TLAST for reliable and fast data transfer.

The introduction of the streaming architecture allows avoiding redundant memory accesses, decreases the overhead of communication, and boosts the system throughput. The modular AXI4-Stream IP cores can be easily integrated in the system and used in hardware-software co-design based on embedded processors like AMD-Xilinx Zynq MPSoCs [31].

6.6 Research Trends and Challenges

However, despite the progress that has already been made in FPA based AMC, there are still some issues to consider. Firstly, many approaches concentrate mainly on the accuracy improvement rather than on latency, throughput, and resource efficiency. While deep learning accelerators are generally resource-intensive and consume much memory, likelihood approaches may require complex probability computations. Thus, efficient usage of LUTs, DSP slices, BRAMs, and flip-flops remains an important goal for further investigation.

Another promising area for future research is development of parallel streaming approaches for evaluation of several modulation schemes at once and processing of continuously received signal samples [32].

6.7 Summary

AMC architectures based on FPGAs have developed from basic feature-based classifiers to complex architectures based on likelihood classification, machine learning, and deep learning. Feature-based architectures have the advantage of being computationally efficient, while likelihood-based architectures have the benefit of high accuracy of classification. Machine learning and deep learning have the advantages of improved recognition performance but higher hardware and memory complexity. The modern trend of recent studies is to use parallelization, pipelining, and data flow based on AXI4-Stream interface. This proves that FPGA technology can be used as an appropriate platform for AMC systems.

VII. HARDWARE–SOFTWARE CO-DESIGN FOR AUTOMATIC MODULATION CLASSIFICATION

The rising complexity of contemporary wireless communications has led to hardware-software co-design being employed as a design methodology for the AMC implementation. Contrary to the pure execution of all operations in software or hardware only, hardware-software co-design is based on dividing the system into software and hardware parts based on the computational demands of the operation. The computationally heavy signal processing functions are carried out in the programmable logic, whereas the system control and configuration, data handling, and human-machine interface are performed through embedded software. Such approach utilizes the benefits of both approaches – flexibility of software and performance & parallelism capabilities of hardware accelerators.

Hardware-software co-design is usually implemented in the SoC-based FPGA communication system architecture. One of such implementations is, for example, the AMD-Xilinx Zynq UltraScale+ MPSoC combining ARM Cortex-A53 processors and FPGA fabric on the same chip. The processing system (PS) performs software applications, while the programmable logic (PL) runs hardware accelerators implementing the computationally heavy operations like signal pre-processing, feature extraction, computation of likelihoods, and classification. The communication between PS and PL can be performed using the standard interfaces, such as AXI, AXI4, AXI4-Lite, AXI4-Stream [33].

In the case of AMC systems, hardware/software co-design allows continuous acquisition of communication signal samples by software that is executed on the processing platform and subsequent efficient transferring of this data to the hardware accelerator via AXI interfaces. Parallel signal processing and modulation classification are performed by the FPGA accelerator, followed by returning the classification results back to the processing platform, where they can be visualized, stored, and used for higher-level communication tasks. Offloading of computationally heavy processes into hardware accelerators provides significant speed-up and reduces latency of the whole system compared to purely software solutions.

There were several works using hardware/software co-design for AMC systems based on FPGAs. Earlier works mostly focused on acceleration of feature extraction and mathematical calculations but still kept classification in software. More recent studies expanded the scope of acceleration to include the whole pipeline – from feature extraction to the inference process in the neural network and making decisions. Development of heterogeneous systems became easier due to the existence of design environments such as AMD-Xilinx Vivado Design Suite and Vitis Unified Software Platform.

An important benefit of hardware-software co-design is design flexibility. Accelerator circuits can be designed separately from application software, which enables new classification or communication protocols to be added without having to redesign the whole system. Moreover, the use of standard AXI interfaces enables modular system design, providing interconnectivity between custom IPs and vendor hardware blocks. It is also important that stream interface, for example, AXI4-Stream, is very useful for AMC, since it allows streaming of communication signal samples with low protocol overhead.

However, there are some disadvantages of hardware-software co-design that need to be considered. Proper functionality partitioning between hardware and software is critical to ensure maximum performance of the system and minimum usage of FPGA resources. Another problem is high communication overhead, which can become a bottleneck if it is not optimized. In addition, synchronization, memory management, and software drivers development are additional sources of design complexity. Thus, achieving the optimal trade-off between computational performance, resource usage, power consumption, and software flexibility is an important research goal. [34].

In conclusion, it can be said that the combination of hardware and software has proved to be very useful in designing real-time Automatic Modulation Classification systems. The use of programmable hardware together with software for managing the entire system results in high computational efficiency, reduced latency, efficient resource utilization, and also future-proof algorithms. Hardware-software combination is thus a very

important enabling technology for FPGAs in intelligent communications.

VIII. COMPARATIVE ANALYSIS OF EXISTING WORKS

The evolution of Automatic Modulation Classification (AMC) has shifted from the traditional use of statistical and likelihood methods to the use of more sophisticated machine learning algorithms and hardware implementations. There is a wide disparity among existing works regarding the approach, computational complexity, implementation platform, parallelization level, streaming capabilities, and applicability to real-time applications. A comparative study is necessary to identify architectural issues that have inspired the research in AMC in hardware.

Traditional AMC research relied on likelihood and statistical approaches. Dobre et al. [1] provided an extensive review of traditional AMC techniques and emphasized the high efficiency of likelihood-based techniques and their high computational complexity. Panagiotou et al. [11], [35] explored the likelihood ratio method and showed its capacity of achieving nearly optimal classification under certain channel models. Swami and Sadler [8] investigated the cumulants-based technique while Abdelmutalab et al. [33] used higher-order cumulants along with hierarchical polynomial classifiers. Such approaches provide a structured solution and require relatively low memory resources but can rely on assumptions about the signal/channel model and become complex in case of multiple hypotheses/measurements. The advent of machine learning methods greatly increased the design space of AMC. Hassan et al. [9] presented wavelet-based feature extraction with neural network classifier, showing the applicability of decision-making based on learning algorithms. Subbarao and Punniakodi [10] studied cumulants-based features and used ensemble classifiers, showing how statistical signal descriptors could be combined with supervised learning. Modern methods have mostly relied on the use of deep neural networks. For example, Zhang et al. [2] suggested a CNN-LSTM architecture, and Abd-Elaziz et al. [5] studied the use of a robust CNN architecture for cognitive radio applications. Attention-based and Transformer-inspired methods have become popular as well to learn complicated representations [13], [15]. Despite the fact that such methods are able to achieve high classification performance, they can become computationally expensive in case of real-time processing on embedded devices. It has thus become one area of research that needs to be considered. An automatic modulation recognition system for satellite communications has been implemented on the FPGA by Digdarsini et al. [17]. Cardoso et al. [23] proposed an IP core on the FPGA for AMC which proved the possibility of mapping the modulation classification functions in the dedicated hardware. Tridgell et al. [20], [24] studied the architecture of AMC systems operating in real time, including implementations for FPGA and RFSoC platforms.

Later studies have concentrated on the realization of deep learning based AMC on FPGAs. In this regard, Tang et al. [16] studied FPGA realization of deep learning-based modulation classifier on an SDR platform. Jentzsch et al. [19] showcased streaming FPGA architectures for RF applications by means of FINN where they pointed out the importance of data flow in efficient neural network inference. Maclellan et al. [18] studied streaming CNN architectures for RFSoC data converters. Moreover, Kumar et al. [21] proposed an AMC solution on FPGAs whereas Sar et al. [22] proposed a deep learning AMC solution on FPGA for Internet-of-Vehicles applications. Latter researches have concentrated on the optimization of CNNs using FPGA-oriented quantization and pruning [31], [32] and FPGA-oriented CNN architectures [25].

Other research efforts include work done on adaptive and dataflow-oriented FPGA architectures. Rubiano et al. [27], [30] studied adaptive inference for FPGA-based 5G AMC, which is in line with the necessity of implementing efficient inference within realistic constraints. Wahllöf [28] considered dataflow hardware acceleration for AMC on FPGA, highlighting the importance of stream processing. Finally, Zhang et al. [29] studied real-time automatic modulation recognition based on FPGA architecture. It can be seen that there is a trend towards focusing not only on inference but also on real-time processing and data movement.

From the existing literature, there are some key findings. Firstly, the conventional likelihood based approaches have theoretical soundness and potential to deliver high classification accuracy while being

computationally heavy where multiple hypotheses are involved [1], [11], [35]. Secondly, the machine learning and deep learning approaches are capable of learning complex signal representation models, thus achieving high classification accuracy while increasing their computation and memory requirements with higher model complexity [2], [5], [15], [26]. Thirdly, the FPGAs are good tools for exploiting the advantages of parallelism, pipelining, and specialized hardware resources to mitigate the latency issues of software-based AMC [16]-[24]. Fourthly, recent developments on dataflow and adaptive architecture show that efficient data handling and hardware-software cooperation becomes critical for real-world AMC [18], [19], [27]-[30].

However, it appears that there is still a trade-off between the accuracy of classification, complexity of computing, latency of decisions, speed, resource efficiency, and implementation flexibility. Quite a few works have concentrated recently on FPGA acceleration of deep learning classification approaches, while the likelihood-based AMC approach is appealing due to solid statistical background and good interpretability, but it provides great room for hardware-level parallelism.

Specifically, it is possible to perform evaluation of several hypothesis regarding modulation concurrently, which could decrease the latency of likelihood computation.

Hence, the comparison suggests the possibility for research on FPGA implementations that will capitalize on the benefits of likelihood-based AMC in conjunction with hardware-level parallelism and streaming data flow. The FPGA implementation is required to test multiple modulation schemes simultaneously, transfer continuous signal data efficiently, and strike a good balance between accuracy, latency, throughput, FPGA resource usage, and power consumption. This will form the foundation for investigating parallel likelihood computations together with a custom hardware implementation using AXI4-Stream protocol.

Table 1. Comparative Analysis of Existing Automatic Modulation Classification Approaches

Ref.	AMC Approach	Platform/Implementation	Main Contribution	Hardware Orientation	Key Limitation
[1]	Classical AMC	—	Comprehensive survey of classical and emerging AMC	—	High complexity of likelihood methods
[8]	Cumulant-based	—	Hierarchical digital modulation classification	—	Sensitive under difficult signal conditions
[11], [35]	Likelihood-ratio	Software/theoretical	Near-optimal likelihood-based classification	Limited	High computational complexity
[16]	Deep Learning	FPGA/SDR	DL-based AMC implementation	FPGA	DL computational requirements
[17]	Conventional AMC	FPGA	FPGA-based modulation recognition	FPGA	Limited scalability
[19]	Neural-network/dataflow	FPGA	Streaming RF application architecture	FPGA	Model-dependent resource requirements
[20]	AMC	RFSoC	Real-time AMC	FPGA/RFSoC	Hardware complexity
[21]	AMC	FPGA	FPGA implementation of	FPGA	Hardware-resource/performance

Ref.	AMC Approach	Platform/Implementation	Main Contribution	Hardware Orientation	Key Limitation
			modulation recognition		trade-off
[22]	Deep Learning	FPGA	DL-based AMC for IoV	FPGA	DL resource requirements
[27], [30]	Adaptive DL	FPGA	Adaptive inference for 5G AMC	FPGA	Adaptation/complexity trade-off
[28]	Dataflow AMC	FPGA	Dataflow hardware acceleration	FPGA	Requires optimized hardware mapping
[29]	AMC	FPGA	Real-time modulation recognition	FPGA	Limited modulation/architecture scope
[31], [32]	Optimized CNN	FPGA	Quantization/pruning and resource optimization	FPGA	Accuracy–complexity trade-off
Proposed direction	Parallel likelihood-based AMC	FPGA	Concurrent hypothesis evaluation + streaming architecture	FPGA/AXI4-Stream	Targets the identified hardware parallelism and dataflow gap

IX. RESEARCH CHALLENGES AND FUTURE DIRECTIONS

Although substantial progress has been made in Automatic Modulation Classification (AMC), there are still some challenges that need to be overcome in order to make efficient real-time classification systems. This is evident from existing literature where it has been observed that there is always a continuous trade-off between classification accuracy, complexity, latency, utilization of resources, and flexibility of implementation.

A. Computational Complexity of Likelihood-Based AMC

Statistical likelihood approaches have a robust foundation in statistics and may lead to good classification results, but the need for computation increases with the number of samples and modulation hypotheses [1], [11], [35]. Computation delay might be incurred by repeating probability and likelihood computations in series. Hardware designs that take advantage of parallel hypothesis computation, pipelining, and arithmetic operations must be developed in order to decrease computational cost.

B. Accuracy–Complexity Trade-Off in Machine Learning-Based AMC

Several machine-learning and deep-learning algorithms have shown good classification performance under different types of signals [2], [5], [13], [15], [26]. But with an increase in depth and complexity of models, there is a need for greater memory, computation, and hardware capabilities. Quantization, pruning, and model compression are some of the techniques that can be used to overcome these needs [31], [32].

C. Low-Latency Real-Time Processing

Applications in real-time communication systems necessitate that classification be carried out with stringent time constraints. Both the FPGA and RFSoC approaches have shown the capability of hardware acceleration in lowering AMC processing latency [16], [17], [20], [21], [24]. Nonetheless, there is more work that needs to be done in terms of

minimizing computation and data transfer time. Deep pipelining, parallelization and streaming can be effective techniques for this purpose.

D. Efficient Data Movement and Streaming Architecture

It is not just the computing speed, but the efficiency of data transfer between different compute nodes that determines the performance of accelerators. The streaming architecture has thus received significant attention in FPGAs for signal processing [18], [19], [28]. Good streaming interfaces may help to process input signal samples continuously without any unnecessary buffering and communication overhead. In future AMC architectures, there should be joint optimization of computation and data transfer as separate entities.

E. FPGA Resource and Power Constraints

Although FPGAs offer significant parallel processing capabilities, the number of logic cells, flip-flops, DSP slices, BRAMs, and interconnects is always limited. More parallelism can be beneficial in terms of better latency and throughput but at the same time can lead to higher hardware usage and increased energy consumption. Therefore, the challenge for future designs is to find the right trade-off between parallelism and hardware efficiency.

F. Robustness under Realistic Wireless Conditions

Most AMC research relies on artificial or simulated signal environments for the testing of classification performance. In real-world applications, however, wireless communication systems are subjected to factors such as noise, fading, interference, frequency offsets, timing errors, multipath, and various channel conditions [1], [7], [12]. Future research on this subject needs to focus on more realistic data sets.

G. Scalability to Multiple Modulation Formats

The AMC architecture is supposed to accommodate more and more modulation hypotheses without incurring additional latency costs proportional to the growth of the number of hypotheses considered. The evaluation of hypotheses sequentially becomes inefficient with the increase in the number of the modulation hypotheses to be tested. Scalable AMC architectures exploiting parallel and configurable processing are well suited for this task.

H. Hardware–Software Co-Design

Today's FPGA devices are integrating programmable logic with embedded processors. This allows for AMC computations that are computationally intensive to be carried out using hardware, whereas software will take care of control and configuration operations. The next generation systems may use the hardware/software co-design technique in order to ensure flexibility with minimal latency.

I. Hardware-Aware Machine Learning

Future work could be to devise machine learning models which are optimized for FPGA implementation. Future works can combine the model architecture, numerical precision, memory mapping, parallelism, and data flow in the design phase rather than designing the classification model without considering the hardware architecture. Hardware-aware training, quantization-aware training, pruning, and neural architecture optimization can be utilized to decrease the inference cost while keeping the classification performance intact [25], [31], [32].

J. Integration with Emerging Communication Systems

The future wireless systems, such as advanced 5G technology and forthcoming 6G networks, will be deployed in diverse heterogeneous spectral domains. Adaptive Modulation and Coding could be helpful for smart receivers, cognitive radio, spectrum sensing, adaptive communications, and autonomous signal processing. Thus, future hardware implementation on FPGAs should include configurability, scalability, and adaptability to different modulations and signal environments.

K. Future Research Directions

On the basis of the above-mentioned problems, there are several areas in which research may take place. First, AMC based on likelihood can be optimized using the concept of parallel evaluation of hypotheses and pipelined hardware paths. Second, streaming architectures may be combined more closely with the computational core for minimizing data transfer costs. Third, machine learning methods that are aware of the hardware characteristics may be studied for the minimization of resource and power needs in the context of AMC based on deep learning. Fourth,

adaptive architecture that can dynamically change the complexity of processing depending on the signal may provide more efficient energy consumption.

In general, the future of AMC is expected to evolve towards being hardware-friendly, highly parallel, streaming, and adaptive architectures. Computation and data flow optimization would become very important in order to meet the demands of being accurate in classification, having low latency, and having high throughput.

X. CONCLUSION

AMC has been developed from conventional statistical and likelihood methods to machine-learning, deep-learning, and hardware-acceleration methods. In this survey, we have explored these different methods of AMC with focus on computational complexity, implementation constraints, and real-time communication applications.

Traditional AMC methods have a solid mathematical background and can perform classification with a high level of reliability. However, the computational complexity increases considerably in case of multiple hypotheses and many received signals. Machine learning and deep learning methods are more flexible and can discover the complex nature of a signal; however, with the increase in complexity of models, there is a rise in computational complexity and hardware and memory requirements.

The use of hardware accelerators, such as FPGAs, is a powerful method to address these issues. The parallel processing capability, pipelining support, customized datapaths, and stream communication make the use of FPGA a promising solution for real-time AMC application.

Moreover, the survey highlights some of the open challenges, which include the need to minimize computational complexity, low latency and high throughput, control over FPGA resource and power utilization, increase robustness in practical wireless environments, and develop scalable architectures for various modulation schemes. Efficient data transfer between various stages of computation is also crucial for preventing communication issues.

The future AMC solutions will rely more on hardware-aware approaches, including the usage of parallel and pipelined architecture, optimal machine-learning models, adaptive techniques, and hybrid hardware-software platforms. The use of intelligent classification algorithms along with FPGA architecture can be seen as a promising approach for designing AMC solutions.

REFERENCES

[1] Dobre, Octavia & Abdi, Ali & Su, Wei. (2007). Survey of automatic modulation classification techniques: Classical approaches and new trends. Communications, IET. 1. 137 - 156. 10.1049/iet-com:20050176.

[2] Zhang, Zufan & Luo, Hao & Wang, Chun & Gan, Chenquan & Xiang, Yong. (2020). Automatic Modulation Classification Using CNN-LSTM Based Dual-Stream Structure. IEEE Transactions on Vehicular Technology. 69. 13521-13531. 10.1109/TVT.2020.3030018.

[3] Zheng, Qinghe & Zhao, Penghui & Li, Yang & Wang, Hongjun & Yang, Yang. (2021). Spectrum interference-based two-level data augmentation method in deep learning for automatic modulation classification. Neural Computing and Applications. 33. 1-23. 10.1007/s00521-020-05514-1.

[4] Li, Tao & Li, Yongzhao & Dobre, Octavia. (2021). Modulation Classification Based on Fourth-Order Cumulants of Superposed Signal in NOMA Systems. IEEE Transactions on Information Forensics and Security. PP. 1-1. 10.1109/TIFS.2021.3068006.

[5] Abd-Elaziz, Ola & Abdalla, Mahmoud & Elsayed, Rania. (2023). Deep Learning-Based Automatic Modulation Classification Using Robust CNN Architecture for Cognitive Radio Networks. Sensors. 23. 9467. 10.3390/s23239467.

[6] Lawrence R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," Proceedings of the IEEE, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[7] Zhu, Z., & Nandi, A. K. (2015). Automatic modulation classification: principles, algorithms and applications. John Wiley & Sons.

[8] Swami, Ananthram & Sadler, B.M. (2000). Hierarchical Digital Modulation Classification Using Cumulants. Communications, IEEE

Transactions on. 48. 416 - 429. 10.1109/26.837045.

[9] Hassan, K., Dayoub, I., Hamouda, W. et al. Automatic Modulation Recognition Using Wavelet Transform and Neural Networks in Wireless Systems. EURASIP J. Adv. Signal Process. 2010, 532898 (2010).

[10] Subbarao, M. & Punniakodi, Samundiswary. (2021). Automatic Modulation Classification Using Cumulants and Ensemble Classifiers. 10.1007/978-981-16-0443-0_9.

[11] Panagiotou, Prokopios & Anastasopoulos, Achilleas & Polydoros, Andreas. (2000). Likelihood Ratio Tests for Modulation Classification. 2. 670 - 674 vol.2. 10.1109/MILCOM.2000.904013.

[12] Al-Nuaimi, D. H., Hashim, I. A., Zainal Abidin, I. S., Salman, L. B., & Mat Isa, N. A. (2019). Performance of Feature-Based Techniques for Automatic Digital Modulation Recognition and Classification—A Review. Electronics, 8(12), 1407.

[13] Zhang, R., Yin, Z., Wu, Z., & Zhou, S. (2021). A Novel Automatic Modulation Classification Method Using Attention Mechanism and Hybrid Parallel Neural Network. Applied Sciences, 11(3), 1327.

[14] O. Ozdemir, T. Wimalajeewa, B. Dulek, P. K. Varshney, and W. Su, "Asynchronous linear modulation classification with multiple sensors via generalized EM algorithm," 2014.

[15] Zhang, Shiqi. (2025). Overview of Automatic Modulation Recognition Methods based on Deep Learning. Applied and Computational Engineering. 121. 139-146. 10.54254/2755-2721/2025.19842

[16] Zhiling, Tang & Li, Si-Min & Yu, Li-Juan. (2018). Implementation of Deep Learning-Based Automatic Modulation Classifier on FPGA SDR Platform. Electronics.7. 122. 10.3390/electronics7070122.

[17] Digdarsini, D., Kumar, M.S., Khot, G.R., Ram, T.V., & Tank, V.K. (2014). FPGA implementation of automatic modulation recognition system for advanced SATCOM system. 2014 International Conference on Signal Processing and Integrated Networks (SPIN), 464-46.

[18] Maclellan, Andrew & Crockett, Louise & Stewart, Robert. (2023). Streaming Convolutional Neural Network FPGA Architecture for RFSoC Data Converters. 1-5. 10.1109/NEWCAS57931.2023.10198198.

[19] Jentzsch, Felix & Umuroglu, Yaman & Pappalardo, Alessandro & Blott, Michaela & Platzner, Marco. (2022). RadioML Meets FINN: Enabling Future RF Applications with FPGA Streaming Architectures. IEEE Micro. 42. 125-133. 10.1109/MM.2022.3202091.

[20] Tridgell, Stephen & Boland, David & Leong, Philip & Khodamoradi, Alireza & Siddhartha, (2020). Real-time Automatic Modulation Classification using RFSoC. 82-89. 10.1109/IPDPSW50202.2020.00021.

[21] Kumar, Satish & Mahapatra, Rajarshi & Singh, Anurag. (2022). Automatic Modulation Recognition: An FPGA Implementation. IEEE Communications Letters. 26. 1-1. 10.1109/LCOMM.2022.3184771.

[22] Sar, Dipanjan & Das, Debanjan & Mahapatra, Rajarshi & Udutalapally, Venkanna. (2024). fAMC: A FPGA Implemented Automatic Modulation Classification via Deep Learning for IoV Applications. 1-6. 10.1109/INDICON63790.2024.10958390.

[23] Cardoso, C., Castro, A. R., & Klautau, A. (2013). An efficient FPGA IP core for automatic modulation classification. IEEE Embedded Systems Letters, 5(3), 42-45.

[24] Tridgell, S., Boland, D., Leong, P. H., & Siddhartha, S. (2019, December). Real-time automatic modulation classification. In 2019 International Conference on Field-Programmable Technology (ICFPT) (pp. 299-302). IEEE.

[25] Mohamed, K. A., Abdullah, M. B., Kolta, S. S., Hassan, S. E. A., Sayed, R., Draz, H. H., & Azmi, H. (2025, December). FPGA-Optimized CNN Architecture for Automatic Modulation Classification. In 2025 37th International Conference on Microelectronics (ICM) (pp. 1-6). IEEE.

[26] Hussein, H. S., Ali, M. H. E., Ismeil, M., Shaaban, M. N., Mohamed, M. L., & Atallah, H. A. (2023). Automatic modulation classification: Convolutional deep learning neural networks approaches. IEEE Access, 11, 98695-98705.

[27] de Oliveira Rubiano, D., Korol, G., & Beck, A. C. S. (2023, January). Adaptive inference for FPGA-Based 5G automatic modulation classification. In International Workshop on Design and Architecture for Signal and Image Processing (pp. 95-106). Cham: Springer Nature Switzerland.

[28] Wahllöf, O. (2025). Automatic Modulation Classification: Dataflow Hardware Acceleration on FPGA.

[29] Zhang, K., Gao, B., Yang, B., Ji, Y., Gao, F., & Li, Y. (2023, December). Real-time automatic modulation recognition based on FPGA. In 2023 9th International Conference on Computer and Communications (ICCC) (pp. 1440-1444). IEEE.

[30] Rubiano, D. D. O., Korol, G., & Schneider Beck, A. C. (2023). Adaptive Inference for FPGA-Based 5G Automatic Modulation Classification. DESIGN AND ARCHITECTURE FOR SIGNAL AND IMAGE PROCESSING, DASIP 2023, 13879, 12.

[31] Rothe, J. A. (2024). Quantization and Pruning of Convolutional Neural Networks for Efficient FPGA Implementation of Digital Modulation Detection Firmware (Doctoral dissertation, Johns Hopkins University).

[32] Rothe, J. A., & Shajaiah, H. (2025, March). Resource and Performance Improvements of Optimized Convolutional Neural Networks for FPGA Implementations of Automatic Modulation Recognition. In 2025 59th Annual Conference on Information Sciences and Systems (CISS) (pp. 1-6). IEEE.

[33] Ameen Abdelmutalab, Khaled Assaleh, and Mohamed El-Tarhuni. Automatic modulation classification based on high order cumulants and hierarchical polynomial classifiers. Phys. Commun., 21(C):10–18, December 2016.

[34] Gihan J Mendis, Jin Wei, and Arjuna Madanayake. Deep learning-based automated modulation classification for cognitive radio. In 2016 IEEE International Conference on Communication Systems (ICCS), pages 1–6. IEEE, 2016

[35] Prokopios Panagiotou, Achilleas Anastasopoulos, and A Polydoros. Likelihood ratio tests for modulation classification. In MILCOM 2000 Proceedings. 21st Century Military Communications. Architectures and Technologies for Information Superiority (Cat. No. 00CH37155), volume 2, pages 670–674. IEEE, 2000.