FPGA-Based Machine Learning: Platforms, Applications, Design Considerations, Challenges, and Future Directions

Tianrun Zhao

Department of Statistics, University of Illinois at Urbana-Champaign, Illinois, US

tianrun4@illinois.edu

Abstract. Field-Programmable Gate Arrays (FPGAs) have emerged as a promising platform for accelerating machine learning tasks due to their high parallelism, low latency, and hardware customization ability. In this paper, the authors provide an overview of popular FPGA platforms for machine learning and compare the tradeoffs among FPGAs, GPUs, and CPUs for machine learning. The authors also present specific applications of machine learning based on FPGAs, including those in autonomous driving and healthcare. Additionally, the paper explores FPGA design considerations, such as architecture, resource utilization, and power consumption. Nonetheless, obstacles persist in the realm of FPGA-based machine learning that require attention. Identifying the ideal balance between adaptability and performance, considering factors such as space, energy usage, and latency, is still challenging. As the capabilities of FPGAs expand, there is a significant need for devices that have a smaller footprint, reduced power consumption, and minimized delays. The paper emphasizes the necessity of ongoing research in the field of FPGA-based machine learning to address these issues and continue enhancing the performance and effectiveness of machine learning systems.

Keywords: FPGA, machine learning, autonomous driving, healthcare, design considerations.

1. Introduction

The domain of machine learning, which falls under the umbrella of artificial intelligence (AI), enables machines to learn from data without the need for explicit programming. It leverages algorithms to discover patterns and draw predictions or decisions from input data.

Conversely, Field-Programmable Gate Arrays (FPGAs) represent a type of integrated circuit that can be reconfigured after production to carry out specific tasks. FPGAs are purpose-built to execute a wide spectrum of tasks concurrently, which makes them a compelling solution for computationally demanding applications, including machine learning. FPGAs are an attractive option for machine learning since they can be tailored to accelerate specific computations and process vast amounts of data in real-time. By programming an FPGA to execute a specific machine learning algorithm, the performance is significantly faster than a conventional CPU or GPU. Furthermore, FPGAs exhibit high energy efficiency and low latency, making them a desirable technology for numerous machine learning applications. Essentially, FPGAs represent a promising platform for enhancing machine learning operations due to their high parallelism, low latency, and the ability to customize hardware implementation for specific tasks.

Numerous hardware approaches have been proposed and executed to improve deep neural network performance, including specialized integrated circuits (ASICs), versatile graphics processing units (GPUs), and adaptable FPGA-based solutions. Although ASIC-based designs excel in speed and power efficiency, they fall short in the flexibility and adaptability needed in the constantly changing AI/ML landscape. In contrast, GPU and CPU designs offer great programmability and adaptability but lack the speed and power efficiency of ASICs. FPGA-based designs, however, strike a balance by delivering high parallelism, flexibility, easy configurability, and rapid, energy-efficient processing [1].

2. Popular FPGA Platforms for ML

It features a wide range of FPGA chips and can be programmed using popular machine learning frameworks like TensorFlow and PyTorch. Xilinx Alveo (Fig. 1), Achronix Speedster (Fig. 2), Microsoft Brainwave (Fig. 3), and Intel Arria (Fig. 4) are all popular FPGA platforms for machine learning.

Xilinx Alveo and Intel Arria FPGAs are well-established platforms, with a large ecosystem and extensive support for machine learning frameworks, such as TensorFlow and PyTorch. They offer high performance and energy efficiency, as well as flexible programming options. In contrast, Achronix Speedster and Microsoft Brainwave FPGAs are relatively new entrants to the market, but they offer unique features such as support for high precision arithmetic and real-time inference, respectively.

Another point of comparison is their architecture. Xilinx Alveo and Intel Arria FPGAs use traditional static routing architectures, whereas Achronix Speedster and Microsoft Brainwave FPGAs use more novel routing architectures, such as dynamic routing and mesh networks, respectively. This can affect their performance and resource utilization, depending on the specific application.

Overall, the choice of FPGA platform for machine learning depends on the specific needs and requirements of the application. Each platform offers unique strengths and tradeoffs in terms of performance, energy efficiency, architecture, and programming options.



Figure 1. Xilinx Alveo [2]



Figure 2. Achronix Speedster [3]



Figure 3. Microsoft Brainwave [4]



Figure 4. Intel Arria [5]

3. Applications of ML Based on FPGAs

3.1. ML on FPGAs in Vehicle Applications

Autonomous driving systems typically use DNN to process sensor data and make real-time decisions. Includes identifying and tracking objects, detecting lanes, etc. FPGAs provide a powerful platform for ML in autonomous driving because of their ability to perform complex computations in real-time with low latency. Unlike traditional CPUs and GPUs, FPGAs can be reconfigured to perform specific tasks, making them well-suited for the high-performance and real-time requirements of autonomous driving systems. Furthermore, FPGAs offer significant advantages in terms of power consumption compared to traditional computing platforms. This is an important consideration in autonomous driving, as these systems require significant processing power but must also be energy-efficient to ensure long battery life [6, 7].

Despite these benefits, there are also several challenges associated with using FPGAs for ML in autonomous driving. This involves creating and refining machine learning algorithms for FPGA, balancing the relationship between efficiency and energy use, and verifying and validating the system to guarantee its dependability and security [8].

3.2. ML on FPGAs in Healthcare

In healthcare, FPGAs can be used to perform real-time processing for wearable medical devices, telemedicine systems, clinical decision support systems, and predictive health monitoring. These applications require fast and accurate processing of large amounts of data, and FPGAs provide the performance and low latency needed to meet these requirements. One of the key applications of FPGA-based machine learning in healthcare is in real-time electrocardiogram (ECG) classification. ECG classification involves analyzing ECG signals to determine the presence of various cardiac conditions, such as arrhythmias and heart disease. FPGAs can be used to perform real-time ECG classification by implementing deep neural networks (DNNs) that can accurately classify ECG signals in real-time [9].

Another application of FPGA-based machine learning in healthcare is in real-time electroencephalogram (EEG) analysis. EEG analysis involves analyzing EEG signals to determine the presence of various neurological conditions, such as epilepsy and brain tumors. FPGAs can be used to perform real-time EEG analysis by implementing DNNs that can accurately analyze EEG signals in real-time. Despite these benefits, there are also several challenges associated with using FPGAs for machine learning in healthcare. This involves developing and refining machine learning algorithms for FPGA, balancing the relationship between efficiency and energy usage, and confirming and evaluating the system's dependability and security to guarantee its proper functioning [10].

4. FPGA Design Considerations

4.1. Key design considerations

When designing FPGA-based solutions for machine learning, it is crucial to consider a variety of factors to ensure the efficient implementation of algorithms and workloads. These factors include optimizing resource utilization to maximize performance while minimizing power consumption and area; exploiting parallelism by designing custom pipelines and processing elements that take advantage of FPGA architectures; carefully selecting numeric representation and bit-width for computations, considering trade-offs between accuracy and resource utilization; and designing efficient memory hierarchies that minimize latency and maximize throughput by effectively managing on-chip and off-chip memory access patterns.

Additionally, it is vital to optimize dataflow by designing efficient data movement and communication strategies between processing elements, considering pipelining, data reuse, and minimizing data dependencies; ensuring scalability by designing solutions that can accommodate larger models, increased data throughput, or additional processing resources; and leveraging the integration capabilities of modern FPGAs to incorporate other processing elements like CPUs, GPUs, or custom accelerators in a single system-on-chip (SoC) for a more comprehensive solution.

Moreover, it is essential to tackle power usage by employing methods like clock gating, dynamic voltage, and frequency scaling in order to lower energy consumption without compromising performance. Ease of deployment can be facilitated using high-level synthesis tools, libraries, and frameworks, while portability and reusability can be achieved by designing modular and parameterizable components that can be easily adapted or reused in different machine learning applications and across various FPGA platforms. By addressing these comprehensive design considerations, FPGA-based machine learning solutions can offer significant advantages in terms of performance, power efficiency, and flexibility, making them competitive with traditional CPU and GPU-based approaches.

4.2. Architecture Innovation

Arora A et al. [11] proposes a new FPGA architecture design for machine learning applications. The authors argue that existing FPGA architectures are not optimized for the specific requirements of machine learning, which involve large matrix multiplications. To address this problem, the authors introduce a new type of hard block, called a matrix multiplier block, that is specifically designed for matrix multiplication operations commonly used in machine learning.

The authors evaluate their proposed design using several benchmarks and show that it outperforms existing FPGA designs and even some GPU-based implementations in terms of throughput and energy efficiency. The paper concludes that their design could be useful in a range of machine learning applications, including deep learning, convolutional neural networks, and matrix factorization. The authors suggest that their proposed architecture could be used to build specialized ML accelerators that are more energy-efficient and provide higher throughput than existing solutions. Overall, the paper presents a novel approach to designing FPGA architectures for ML applications and shows promising results in terms of performance and energy efficiency.

Table 1 displays the outcomes achieved when creating more extensive matrix multiplier configurations. It illustrates the variations in frequency and area for an FPGA incorporating DSP slices compared to an FPGA equipped with 4x4x4 Matrix Multipliers. For the 64x64x64 design, there is a 7.3 times reduction in total area, highlighted in red, and a 3.6 times increase in clock frequency speedup, highlighted in green, when 4x4x4 Matrix Multipliers are employed, compared to the baseline FPGA.

Fig. 5 demonstrates the clock frequency attained when running a 16x16x16 matrix multiplier on different hardware configurations. Moving from left to right, which represents a shift from an FPGA with DSP slices to one with 4x4x4 matrix multipliers, and ultimately to an ASIC, a clear increase in

clock frequency and reduction in overall area can be observed. Although the design consideration cannot surpass the ASIC, it brings us much closer to it.

Table 1. Comparison between design size [11]

Design size	FPGA with matmuls		FPGA with DSP slices	
	Clk freq (MHz)	Area (MWTA)	Clk freq (MHz)	Area (MWTA)
4x4	1932.05	4.29E+06	1143.01	1.55E+07
8x8	1927.28	1.19E+07	1072.25	6.58E+07
16x16	1685.18	3.84E+07	788.78	2.47E+08
24x24	1689.45	7.70E+07	658.71	5.35E+08
32x32	1613.75	1.34E+08	462.90	9.72E+08
64x64	1213.72	4.92E+08	333.56	3.62E+09

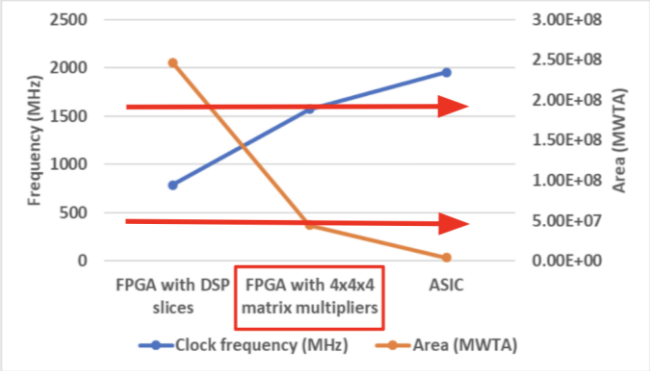


Figure 5. Comparison between design size [11]

5. Challenges & Future Directions

As the capacity of FPGAs continues to increase, there is a growing demand for FPGAs that have a smaller area, lower power consumption, and less delay. Future studies on FPGAs must take into account the obstacles emphasized in Fig. 6, demonstrating the interconnectedness of different quality factors such as size, energy dissipation, efficiency, and expense. For example, minimizing the quantity of interconnect switches can enhance the size, energy usage, and production cost, but it may also compromise interconnect adaptability and impact overall performance. As a result, the primary difficulty in developing FPGAs lies in achieving the right equilibrium between adaptability and efficiency with respect to size, energy usage, and delay [12].

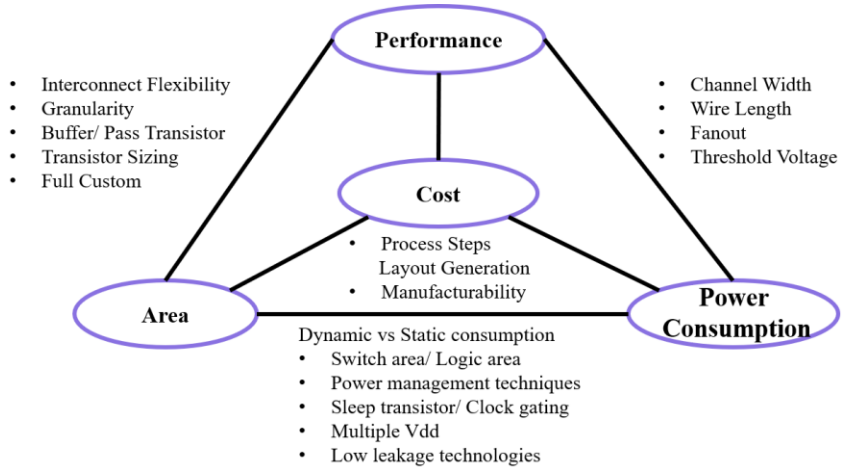


Figure 6. Tradeoff of different quality factors [12]

6. Conclusion

To summarize, the discussion began with the introduction of popular FPGA platforms for machine learning, which include Xilinx Alveo from AMD and Arria from Intel. The comparison between GPUs and CPUs revealed that FPGAs offer high parallelism and customizability, but also require a higher level of design complexity. The advantages of machine learning on FPGAs for applications such as autopilot and healthcare were also highlighted.

Furthermore, design considerations for implementing machine learning on FPGAs were addressed, including optimization of architecture and resource utilization, among others. Despite the challenges faced in implementing machine learning on FPGAs, future research and development can concentrate on enhancing design tools, as well as striking a balance between area, performance, cost, and power consumption. It is believed that future research and development can overcome these challenges and continue to expand the range of tasks that can be performed by FPGAs.

References

[1] Liu X. A hybrid GPU-FPGA-based computing platform for Machine Learning. Procedia Computer Science, 2018, 00: 1-9.

[2] Xilinx. Alveo U50 Data Center Accelerator Card. 2023-03-17. https://www.xilinx.com/products/boards-and-kits/alveo/u50.html.

[3] Achronix Semiconductor Corporation. Speedster7t FPGAs. 2023-03-17. https://www.achronix.com/product/speedster7t-fpgas.

[4] Running J. Microsoft unveils Project Brainwave for real-time AI. Microsoft Research, 2023-03-17. https://www.microsoft.com/en-us/research/blog/microsoft-unveils-project-brainwave/.

[5] Intel. Intel® Arria® 10 GX FPGA Development Kit. 2023-03-17. https://www.intel.com/content/www/us/en/products/details/fpga/development-kits/arria/10-gx.html.

[6] Sun Y, Li J, Xu X, Shi Y. Adaptive Multi-Lane Detection Based on Robust Instance Segmentation for Intelligent Vehicles. IEEE Transactions on Intelligent Vehicles, 2023, 8(1): 888-899.

[7] Harada K, Kanazawa K, Yasunaga M. FPGA-Based Object Detection for Autonomous Driving System. In: Proceedings of the 2019 International Conference on Field-Programmable Technology (ICFPT). Tianjin, China, 2019: 465-468.

[8] Sciangula G, Restuccia F, Biondi A, Buttazzo G. Hardware Acceleration of Deep Neural Networks for Autonomous Driving on FPGA-based SoC. In: Proceedings of the 2022 25th Euromicro Conference on Digital System Design (DSD). Maspalomas, Spain, 2022: 406-414.

[9] Azghadi MR, Amiri P, Soltani S, et al. Hardware Implementation of Deep Network Accelerators Towards Healthcare and Biomedical Applications. IEEE Transactions on Biomedical Circuits and Systems, 2020, 14(6): 1138-1159.

[10] Jameil AK, Al-Raweshidy H. Efficient CNN Architecture on FPGA Using High Level Module for Healthcare Devices. IEEE Access, 2022, 10: 60486-60495.

[11] Arora A, Wei Z, John LK. Hamamu: Specializing FPGAs for ML Applications by Adding Hard Matrix Multiplier Blocks. In: Proceedings of the 2020 IEEE 31st International Conference on Application-specific Systems, Architectures and Processors (ASAP). Manchester, UK, 2020: 53-60.

[12] Hutton M. The design of modern FPGA Architectures. In: Proceedings of the International Symposium the Future of Configurable Hardware, Gent, Belgium, 2004: 1-10.