AITIA: embedded AI techniques for industrial applications

Brandalero, M.; Veleski, M.; Hernandez, H.G.M.; Ali, M.; Jeune, L. Le; Goedeme, T.; ... ; Hubner, M.

2021 31st International Conference on Field-Programmable Logic and Applications (FPL) | 978-1-6654-3759-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/FPL53798.2021.00071

AITIA: Embedded AI Techniques for Industrial Applications

Marcelo Brandalero[1], Mitko Veleski[1], Hector Gerardo Munoz Hernandez[1], Muhammad Ali[2], Laurens Le Jeune[3,4], Toon Goedemé[4], Nele Mentens[3,5], Jurgen Vandendriessche[6], Lancelot Lhoest[6], Bruno da Silva[6,7], Abdellah Touhafi[6,7], Diana Goehringer[2], Michael Hübner[1]

[1]*Chair of Computer Engineering, Brandenburg University of Technology Cottbus-Senftenberg, Germany*
[2]*Chair of Adaptive Dynamic Systems, Technische Universitaet Dresden, Germany*
[3]*ES&S - imec-COSIC, Department of Electrical Engineering (ESAT), KU Leuven, Belgium*
[4]*EAVISE - PSI, Department of Electrical Engineering (ESAT), KU Leuven, Belgium*
[5]*Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands*
[6]*Department of Industrial Engineering (INDI), Vrije Universiteit Brussel, Belgium*
[7]*Department of Electronics and Informatics (ETRO), Vrije Universiteit Brussel, Belgium*

***Abstract*—Motivated by an increasing interest from startups in embedded Artificial Intelligence (AI) and by their limited expertise, the AITIA Project targets the development of embedded AI techniques for industrial applications. This extended abstract presents the motivation and the solutions being developed towards four use cases: smart sensors, network intrusion detection, driver-assistance systems, and Industry 4.0.**

***Index Terms*—artificial intelligence, machine learning, embedded systems, smart sensors, network intrusion detection, driver-assistance systems, industry 4.0**

I. Motivation

Artificial Intelligence (AI) and Machine Learning (ML) techniques are pervading all devices and technologies, with intelligent processing being brought closer to the data sources to sustain low latency and security requirements. Germany and Belgium currently face an enormous growth of startups focusing on smart products that would have huge benefits in integrating into their platforms AI/ML algorithms that can efficiently be executed in embedded, resource-constrained devices. However, there is still a large gap between the academic achievements in AI/ML and their practical implementation at a level that is easily reuseable by small companies.

The AITIA project, a consortium of four German and Belgian universities, aims to address the needs of these companies and bridge the gap between the academic knowledge of embedded AI/ML and industry products. That will be achieved by developing easy-to-use algorithms and systems for embedded AI/ML targeting four use-cases that are aligned with the industrial user group: smart sensors, network intrusion detection, driver-assistance systems, and Industry 4.0.

II. AITIA Use-Cases

A. Smart Sensors

Near-sensor AI promises to bring power-efficient intelligence to sensors. Its adoption is especially promising when considering sensor arrays, where data fusion and AI techniques can be used. The computational needs for embedding these AI techniques make FPGAs power-efficient candidates. In the context of this project, multiple AI techniques are explored for three different applications:

- *Anomaly Detection*: The combination of multiple sensors using AI can provide higher accuracy when detecting anomalies. Our experiments evaluate the accuracy improvement of AI techniques when using sensor arrays.
- *Urban Sound Recognition*: Several AI techniques have been used for recognizing urban sounds in smart cities. This application, however, is often embedded in battery-powered devices with limited computational capacity [1]. Their implementation on a PYNQ-Z2 FPGA using tools such as HLS4ML [2] is evaluated here.
- Super-resolution Acoustic Imaging: Acoustic cameras are nowadays used to display the origin and intensity of sound over images from RGB cameras. Their computational complexity and susceptibility for rounding errors limit the resolution of these cameras when working in real-time. Super-resolution can be used to overcome these problems and reduce artifacts caused by noise [3]. We use tools like Xilinx Vitis AI [4] to embed these solutions into FPGAs, whose architectures are already well suited for acoustic cameras.

B. Network Intrusion Detection

In an era where connectivity is becoming ever more important, network attacks are becoming increasingly relevant. One of the means to contribute to network protection is a Network Intrusion Detection System (NIDS), a system that detects malicious behavior in network traffic. Although traditionally NIDSs are mostly rule-based, research investigating the application of ML is slowly maturing: While the state-of-the-art is promising [5], most approaches are limited to software-based implementations using predefined features. Such systems are not ready for real-time and high-speed network environments,

This project is organized through the COllective Research NETworking (CORNET) platform. Belgian partners are funded by VLAIO under grant number HBC.2018.0491 and German partners by BMWi (Federal Ministry for Economic Affairs and Energy) under IGF-Project Number 249 EBG.

978-1-6654-3759-2/21/$31.00 ©2021 IEEE
DOI 10.1109/FPL53798.2021.00071

as their throughput is limited and their feature extraction is complex.

In this project, we investigate the application of raw traffic-based features for deep learning models, as well as the acceleration of those models on FPGA hardware platforms. Raw traffic-based features use bytes from incoming network traffic as input features rather than using the traditional dataset-specific features. This not only simplifies feature extraction [6], but also allows for algorithms that can be used accross datasets. Moreover, when compared against the state-of-the-art, these raw traffic-based features obtain promising results [7]. Using Brevitas [8] and FINN [9] to quantize and translate the model to hardware, we were able to accelerate a NIDS on a PYNQ-Z2 FPGA.

C. Driver-Assistance Systems

AI and ML are common tools for Advanced Driver-Assistance Systems (ADAS). ADAS includes many features such as adaptive cruise control and automatic parking, requiring a fast implementation of complex algorithms such as object detection and image segmentation to be realized. Artificial Neural Networks (ANNs) are gaining momentum in solving these computations due to higher accuracy and performance. However, for embedded platforms, ANNs have high computational and memory requirements [10] [11].

To overcome these requirements for different ANN algorithms, we develop a multi-core heterogeneous architecture consisting of different types of Processing Elements: RISC-V-based processors, Application-Specific Instruction-set Processors (ASIPs), and hardware accelerators. Each of these PEs is connected over a Network-on-Chip (NoC)- based architecture with a distributed memory system and features a) multi-core RISC-V processors developed using the RI5CY/CV32E40P cores [12] [13]; b) an ASIP implemented by developing a V-extension of RISC-V for a SIMD-based architecture that exploits data-level parallelism; c) hardware accelerators realized with an HLS based library that implements different convolutional neural network (CNN) functions [14]. These different PEs provide different computation and memory choices for the algorithms and can be configured at run-time by using dynamic partial reconfiguration (DPR).

D. Industry 4.0

The new paradigm of Industry 4.0 makes use of extensive use of smart data to automatize the manufacturing processes and to reduce the necessity of human intervention to a minimum [15]. The most recent Industry 4.0 trends are moving towards the implementation of next-generation sensors and actuators with extended functionalities like self-calibration, predictive maintenance, self-organization, and autonomous control [16]. The execution of embedded AI/ML algorithms on reconfigurable hardware could largely contribute towards the implementation of these extended functionalities, in particular, if the underlying hardware is cheap and affordable.

To efficiently run embedded AI/ML algorithms for this use case, we utilize FGPU [17], an overlay architecture that implements a GPU-like processor on FPGA. The parallelism offered by the FGPU allows for the efficient execution Convolutional Neural Networks (CNNs), which are extensively utilized in modern industrial environments. The FGPU can be programmed using a library of OpenCL kernels that implement the standard layers of a CNN architecture. To run and verify the models, we used the Zynq 7000 board, but other affordable boards like the ZedBoard or the Pynq board can also be used. Furthermore, to simplify the entire workflow as much as possible, we also developed in-house scripts that automatize some of the key procedures, such as the implementation of the FGPU on an FPGA board and the extraction of the trained model parameters for the inference.

REFERENCES

[1] B. da Silva, A. W Happi, A. Braeken, and A. Touhafi, "Evaluation of classical machine learning techniques towards urban sound recognition on embedded systems," *Applied Sciences*, vol. 9, no. 18, p. 3885, 2019.

[2] J. Duarte, S. Han, P. Harris, S. Jindariani *et al.*, "Fast inference of deep neural networks in fpgas for particle physics," *Journal of Instrumentation*, vol. 13, no. 07, p. P07027, 2018.

[3] F. Almasri, J. Vandendriessche, L. Segers, B. da Silva *et al.*, "Xcycles backprojection acoustic super-resolution," *Accepted for Sensors MDPI*, 2021.

[4] V. Kathail, "Xilinx Vitis unified software platform," in *Proceedings of the 2020 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, 2020, pp. 173–174.

[5] R. Boutaba, M. A. Salahuddin, N. Limam, S. Ayoubi *et al.*, "A comprehensive survey on machine learning for networking: evolution, applications and research opportunities," *JOURNAL OF INTERNET SERVICES AND APPLICATIONS*, vol. 9, JUN 21 2018.

[6] L. Le Jeune, T. Goedemé, and N. Mentens, "Towards real-time deep learning-based network intrusion detection on FPGA," in *Accepted for AIHWS workshop in the ACNS conference*, 2021, pp. 1–18.

[7] L. Le Jeune, T. Goedemé, and N. Mentens, "Machine learning for misuse-based network intrusion detection: Overview, unified evaluation and feature choice comparison framework," *IEEE Access*, vol. 9, pp. 63 995–64 015, 2021.

[8] A. Pappalardo, "Xilinx/brevitas." [Online]. Available: https://doi.org/10.5281/zenodo.3333552

[9] Y. Umuroglu, N. J. Fraser, G. Gambardella, M. Blott *et al.*, "FINN: A framework for fast, scalable binarized neural network inference," in *FPGA'17*. ACM, 2017, pp. 65–74.

[10] Y. Ma, Y. Cao, S. Vrudhula, and J.-s. Seo, "Optimizing the convolution operation to accelerate deep neural networks on fpga," *IEEE Transactions on VLSI Systems*, vol. 26, no. 7, pp. 1354–1367, 2018.

[11] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "Xnor-net: Imagenet classification using binary convolutional neural networks," 2016.

[12] M. Gautschi, P. D. Schiavone, A. Traber, I. Loi *et al.*, "Near-threshold risc-v core with dsp extensions for scalable iot endpoint devices," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 10, pp. 2700–2713, 2017.

[13] M. Ali, P. Amini Rad, and D. Göhringer, "Risc-v based mpsoc design exploration for fpgas: Area, power and performance," in *Applied Reconfigurable Computing. Architectures, Tools, and Applications*. Cham: Springer International Publishing, 2020, pp. 193–207.

[14] L. Kalms, P. A. Rad, M. Ali, A. Iskander, and D. Göhringer, "A parametrizable high-level synthesis library for accelerating neural networks on fpgas," *Journal of Signal Processing Systems*, Mar 2021.

[15] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld, and M. Hoffmann, "Industry 4.0," *Business & information systems engineering*, vol. 6, no. 4, pp. 239–242, 2014.

[16] M. Pech, J. Vrchota, and J. Bednář, "Predictive maintenance and intelligent sensors in smart factory," *Sensors*, vol. 21, no. 4, p. 1470, 2021.

[17] M. A. Kadi, B. Janssen, J. Yudi, and M. Huebner, "General-purpose computing with soft gpus on fpgas," *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, vol. 11, no. 1, pp. 1–22, 2018.