Institut für Theoretische Elektrotechnik und Mikroelektronik

Universität Bremen

PhD proposal

Research project:
System-on-Chip architectures for real-time machine-learning algorithms in industrial Internet-of-Things applications

Candidate name:
Yarib Israel Nevárez Esparza

March 13, 2021

1 General information

1.1 Personal information

First name: Yarib Israel
Last name: Nevárez Esparza
Date of birth: May 26, 1986
Phone: +49 (1) 788 936794
E-mail: nevarez@item.uni-bremen.de

Yarib Israel Nevárez Esparza

1.2 Supervisor

Title: Prof. Dr.-Ing.
First name: Alberto
Last name: García-Ortiz
Department: Fachbereich 1 - Physik / Elektrotechnik, Institut für theoretische Elektrotechnik und Mikroelektronik
Office address: Building NW 1, Otto-Hahn Allee 1, 28359 Bremen
Phone: +49 (0) 421 218 62533
E-mail: agarcia@item.uni-bremen.de

Prof. Dr.-Ing. Alberto García-Ortiz

1.3 Specialization and working direction

Specialization: Electrical and computer engineering
Working direction: Integrated digital systems

1.4 Expected total duration

The planned duration is 3 years. Start date: May 27, 2019. End date: May 27, 2022.

Abstract

Machine Learning (ML) algorithms represent a promising solution for delivering intelligence to the emerging era of Internet-of-Things (IoT). As the volume of data collected increases, applications in this field become more intelligent and profitable, driving the evolution of many aspects of daily life, science, and industry. However, ML algorithms, particularly Spiking Neural Networks (SNNs) and deep Convolutional Neural Networks (CNNs), are highly compute and data intensive. Therefore, the substantial demand for power and hardware resources of these algorithms represents a restriction for IoT devices in the scope of embedded systems.

Considering the intrinsic error resilience of ML algorithms, paradigms such as approximate computing come to the rescue by offering promising efficiency gains, especially in terms of performance, power consumption, and resource utilization. Approximation techniques are widely used in ML algorithms at the model-structure as well as at the hardware processing level. However, state-of-the-art approximate computing methodologies do not sufficiently address accelerator designs for SNN and deep CNN as power- and resource-demanding algorithms.

This PhD proposal focuses on the investigation of approximate computing techniques to exploit the intrinsic error tolerance of ML algorithms to optimize computing embedded systems at the hardware architecture and circuit-level to achieve efficiency gains. The goal of this research is to contribute to state-of-the-art knowledge of this domain with methodologies to address accelerator designs for SNNs and deep CNNs. Furthermore, the expected outcome of this work is to develop high-efficiency accelerator architectures for SNN and deep CNN algorithms for computer vision applications (e.g., real-time multiple object detection and classification). Lastly, this concludes to enhance machine learning processing capabilities for the rise of the next generation of IoT devices.

2 Introduction

2.1 Problem description

2.2 Motivations

2.2.1 Scientific motivations

1. Efficient implementation of real-time feature extraction and machine learning algorithms in resource-constrained devices

2. Use of approximate techniques to reduce complexity of ML algorithms (i.e. trade-off between hardware complexity and accuracy)

3. Dedicated System-on-Chip (SoC) architectures for ML

2.2.2 Practical motivations

1. System-on-Chip architectures to extend the use of IIoT for scenarios not possible today for real-time machine learning algorithms (e.g. industrial computer vision, signal recognition, feature filtering, machine translation, material inspection, etc.)

2. System-on-Chip architectures to reducing the cost of applying IIoT solution in Industry 4.0

3 State of the art

3.1 Industrial Internet-of-Things

Internet of Things (IoT) is a computing concept describing ubiquitous connection to the Internet, turning common objects into connected devices. The key idea behind the IoT concept is to deploy billions or even trillions of smart objects capable of sensing the surrounding environment, transmit and process acquired data, and then feedback to the environment. Connecting unconventional objects to the Internet will improve the sustainability and safety of industries and society, and enable efcient interaction between the physical world and its digital counterpart, i.e., what is usually addressed as a cyber-physical system (CPS). IoT is usually depicted as the disruptive technology for solving most of present-day society issues such as smart cities, intelligent transportation, pollution monitoring, and connected healthcare, to name a few [1].

As a subset of IoT, Industrial IoT (IIoT) covers the domains of machine-to-machine (M2M) and industrial communication technologies with automation applications. IIoT paves the way for better understanding of the manufacturing process, thereby enabling efcient and sustainable production. IIoT applications typically require relatively small throughput per node and the capacity is not a main concern. Instead, the need for connecting a very large number of devices to the Internet at low cost, with limited hardware capabilities and energy resources (e.g., small batteries) makes latency, energy efciency, cost, reliability, and security/privacy more desired features [2].

The scientific community has explored ideas for IIoT deployment architectures, as an example the Cloud of Things [3, 4]. Where it considered direct communication between things and the Cloud, which is not real-time helpful. As an improved architecture, the Fog and Edge computing concepts provide computational resources closer to the industrial devices, mitigating some of the problems that cannot be addressed by the cloud [5, 6]. Nevertheless, still with real-time difficulties.

Since most of IIoT will deal with real-time and concurrent scenarios requiring time synchronization, bringing real-time issues to the core of the problem. Also, reliability, predictability, robustness, and fault tolerance are necessary when dealing with mission critical systems [7].

In order to enhance the real-time ML performance of IIoT devices, in this research,

it is proposed the development of SoC architectures equipped with dedicated hardware acceleration for real-time ML computations.

3.2 Machine learning accelerators

Recently, ML algorithms have been successfully used to learn in a wide variety of applications, but their heavy computation demands have considerably limited their practical applications. These demands have led to arise in specialized hardware acceleration for ML.

The following paragraphs describe state of the art publications in ML hardware accelerators. We can find accelerators based on algorithm, morphologic, and also based on available hardware resources. The state of the art implementations are taken as references, for further research.

3.2.1 DLAU: A scalable deep learning accelerator unit on FPGA

A suitable example of accelerator unit is presented in [8], which presents the design of deep learning accelerator unit (DLAU), which is a scalable accelerator architecture for large-scale deep learning networks using eld-programmable gate array (FPGA) as the hardware prototype. The DLAU accelerator employs three pipelined processing units to improve the throughput and utilizes tile techniques to explore locality for deep learning applications. Experimental results on Xilinx FPGA prototype show that DLAU can achieve 36.1 speedup with reasonable hardware cost and low power utilization.

3.2.2 Hybrid working set algorithm for SVM learning with a kernel coprocessor on FPGA

Another example of hardware acceleration is presented in [9], which implements the model of Support vector machines (SVM). The associated compute intensive learning algorithm limits their use in real-time applications. In [9] it is presented a fully scalable architecture of a coprocessor, which can compute multiple rows of the kernel matrix in parallel. Further, [9] proposes an extended variant of the popular decomposition technique, sequential minimal optimization, which is called hybrid working set (HWS) algorithm, to effectively utilize the benets of cached kernel columns and the parallel computational power of a coprocessor. The coprocessor is implemented on Xilinx Virtex 7 eld-programmable gate array based VC707 board and achieves a speedup of upto 25 for kernel computation over single threaded computation on Intel Core i5. An application speedup of up to 15 over software implementation of LIBSVM and speedup of up to 23 over SVMLight is achieved using the HWS algorithm in unison with the coprocessor.

3.2.3 DeepX: Deep learning accelerator for restricted boltzmann machine artificial neural networks

In [10], Lok-Won Kim proposes a fully pipelined acceleration architecture to alleviate high computational demand of an articial neural network (ANN) which is restricted Boltzmann machine (RBM) ANNs. The implemented RBM ANN accelerator (integrating 1024 1024 network size, using 128 input cases per batch, and running at a 303-MHz clock frequency) integrated in a state-of-the art eld-programmable gate array (FPGA) (Xilinx Virtex 7 XC7V-2000T) provides a computational performance of 301-billion connection-updates-per-second and about 193 times higher performance than a software solution running on general purpose processors. Most importantly, the architecture enables over 4 times (12 times in batch learning) higher performance compared with a previous work when both are implemented in an FPGA device (XC2VP70).

3.2.4 A digital implementation of extreme learning machines for resource-constrained devices

A particular example for resource constrained embedded systems, is given in [11], which exhibit the implementation of single hidden-layer feed forward neural networks, based on hard-limit activation functions, on recongurable devices. The resulting design strategy relies on a novel learning procedure that inherits the approach adopted in the Extreme Learning Machine paradigm. The eventual training process balances accuracy and network complexity effectively, thus supporting a digital architecture that prioritizes area utilization over computational performance. Experimental verications proved that the proposed design strategy supports the realization of embedded classiers in both FPGA and low-end, inexpensive devices such as CPLDs.

3.2.5 Simplifying deep neural networks for FPGA-like neuromorphic systems

Brain-like hardware platforms for the brain-inspired computational models are being studied, but the maximum size of neural networks they can evaluate is often limited by the number of neurons and synapses equipped with the hardware. The [12] presents two techniques, factorization and pruning, that not only compress the models but also maintain the form of the models for the execution on neuromorphic architectures. The [12] also proposes a novel method to combine the two techniques. The proposed method shows signicant improvements in reducing the number of model parameters over standalone use of each method while maintaining the performance. Our experimental results show that the proposed method can achieve 30x reduction rate within 1% budget of accuracy for the largest layer of AlexNet.

4 Research goals

The main goal of this research is to develop dedicated system-on-chip architectures that allow resource constrained embedded devices to efficiently execute the real-time machine leaning algorithms required for industrial Internet-of-Things applications.

4.1 Outcomes

The main outcome of this research is to produce further knowledge and advance in the field of hardware acceleration for real-time machine learning algorithms for Industry 4.0 devices.

5 Project plan

The research project will be divided into three phases, the prospective milestones schedule is shown in **Tab.** 1. The total timeframe is expected to be 3 years.

5.1 Phase 1

In-depth research of existing work. Existing findings should be incorporated into the own work effectively. For this purpose, a trial and evaluation of existing analysis and design tools. The goal of Phase 1 is to develop first FPGA-based prototypes compatible with the requirements of Industry 4.0 devices.

Outcome: First, it will result in a library of IP-blocks and hardware architectures consisting on prior state of the art. Second, it will result in a quantitative evaluation of existing techniques as well as the identification of the most relevant bottlenecks in previous approaches. The library of IP-block is expected to be open-sourced and the quantitative analysis reported in a journal paper.

5.2 Phase 2

Selection of suitable design tools as well as a strategy for the further improvements in performance and real-time characteristics. In particular stochastic and approximation techniques in hardware acceleration will be considered. Investigations on feature extraction, ML approximations, and acceleration approaches in configurable logic and hardware. Development of a second prototype based on the automated design flow. The second prototype has a higher computing capabilities.

Outcome: Improved FPGA prototype that demonstrates in real Industry 4.0 applications the advantages of the proposed architecture.

5.3 Phase 3

Selection of final approach, strategies, and hardware architectures for the final project phase. Development of an optimized System-on-Chip architecture with resource constrained embedded devices that efficiently execute real-time machine leaning algorithms in IIoT applications. The research will be documented and published in written form.

Outcome: Development of a SoC in a nanometric technology using the architecture proposed in phase 2.

Milestone	Date	Description
M1	September, 2019	Completion of literature search
M2	January, 2020	Understand the key design considerations for efficient ML processing; understand trade-offs between various hardware architectures and platforms; learn about micro-architectural knobs such as precision, data reuse, and parallelism to architect ML accelerators given target area-power-performance metrics; evaluate the utility of various ML dataflow techniques for efficient processing; and understand future trends and opportunities from ML algorithms on Industry 4.0
M3	April, 2020	Outcome: Development of a library of IP-blocks and hardware architectures consisting of prior state of the art
M4	July, 2020	Outcome: Quantitative evaluation of existing techniques as well as the identification of the most relevant bottlenecks in previous approaches, report in a journal paper
M5	October, 2020	Selection of tools, strategies, techniques for further improvements in performance and real-time characteristics. Stochastic and approximation techniques in hardware acceleration
M6	January, 2021	Investigations on feature extraction, ML approximations, and acceleration approaches in configurable logic and hardware
M7	April, 2021	Outcome: Improved FPGA prototype that demonstrates in real Industry 4.0 applications the advantages of the proposed architecture
M8	July, 2021	Development of SoC architecture efficiently executing real-time machine leaning algorithms in IIoT is completed

M9	October, 2021	Development of a SoC in a nanometric technology is completed
M10	Jun, 2022	Written elaboration is completed

Table 1: Milestone schedule.

References

[1] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial internet of things: Challenges, opportunities, and directions," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 11, pp. 4724–4734, 2018.

[2] J. Åkerberg, M. Gidlund, and M. Björkman, "Future research challenges in wireless sensor and actuator networks targeting industrial automation," in *2011 9th IEEE International Conference on Industrial Informatics*. IEEE, 2011, pp. 410–415.

[3] S.-W. Lin, B. Miller, J. Durand, R. Joshi, P. Didier, A. Chigani, R. Torenbeek, D. Duggal, R. Martin, G. Bleakley *et al.*, "Industrial internet reference architecture," *Industrial Internet Consortium (IIC), Tech. Rep*, 2015.

[4] M. Aazam, I. Khan, A. A. Alsaffar, and E.-N. Huh, "Cloud of things: Integrating internet of things and cloud computing and the issues involved," in *Proceedings of 2014 11th International Bhurban Conference on Applied Sciences & Technology (IBCAST) Islamabad, Pakistan, 14th-18th January, 2014*. IEEE, 2014, pp. 414–419.

[5] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, 2012, pp. 13–16.

[6] . Cisco, "Fog computing and the internet of things: extend the cloud to where the things are," *]. URL: https://www. cisco. com/c/dam/en_us/solutions/trends/iot/docs/computing-overview. pdf.(: 10.03. 2019)*, 2015.

[7] M. S. de Brito, S. Hoque, R. Steinke, A. Willner, and T. Magedanz, "Application of the fog computing paradigm to smart factories and cyber-physical systems," *Transactions on Emerging Telecommunications Technologies*, vol. 29, no. 4, p. e3184, 2018.

[8] C. Wang, L. Gong, Q. Yu, X. Li, Y. Xie, and X. Zhou, "Dlau: A scalable deep learning accelerator unit on fpga," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 36, no. 3, pp. 513–517, 2016.

[9] S. Venkateshan, A. Patel, and K. Varghese, "Hybrid working set algorithm for svm learning with a kernel coprocessor on fpga," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 10, pp. 2221–2232, 2014.

[10] L.-W. Kim, "Deepx: Deep learning accelerator for restricted boltzmann machine artificial neural networks," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 5, pp. 1441–1453, 2017.

[11] E. Ragusa, C. Gianoglio, P. Gastaldo, and R. Zunino, "A digital implementation of extreme learning machines for resource-constrained devices," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 8, pp. 1104–1108, 2018.

[12] J. Chung, T. Shin, and J.-S. Yang, "Simplifying deep neural networks for fpga-like neuromorphic systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 38, no. 11, pp. 2032–2042, 2018.