Reconfigurable Fused and Branched CNN Accelerator

(Rizwan Tariq Syed, Marko Andjelkovic, Markus Ulbricht, Milos Krstic)

Rizwan Tariq Syed

Fast Machine Learning for Science 2023,
Imperial College London

September 27, 2023

IHP – Leibniz-Institut für innovative Mikroelektronik

Universität Potsdam

Leibniz Association

Outline

1. Challenges
2. Shared Layers Approach (Takeaway-1)
3. Implementation Results
4. Reconfigurable CNN Accelerator (Takeaway-2)
5. Summary and Future Work

Outline

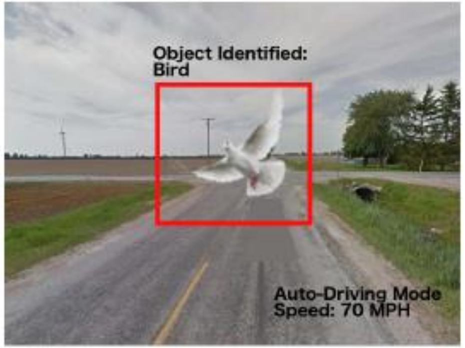
Challenges:

❖ **Varying AI Requirements**
- New data Collections
- Application requirements change
- Addition of new sensors (Camera(s), Radars, Lidars etc.)
- Changes in the AI model
- Change in the accuracy requirements

❖ **Change in AI Requirements directly impacts**
- Power Consumption
- Hardware resource utilization

❖ **Major Goal For Safety Critical Applications (Automotive, Space, etc.)**
- Fulfil AI application requirements
- Ensuring reliability against faults
  (i.e., Single Event Upsets, Single Event Transients, Aging, etc.)

Fault-Free Execution [1]



Erroneous Execution [1]

[1] G. Li, S. K. S. Hari, M. Sullivan, T. Tsai, K. Pattabiraman, J. Emer, and S. W. Keckler, “Understanding error propagation in deep learning neural network (dnn) accelerators and applications,” SC 17, 2017.

Outline

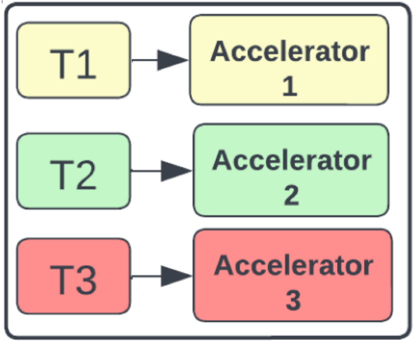
1. Challenges
2. Shared Layers Approach
3. Implementation Results
4. Reconfigurable CNN Accelerator
5. Summary and Future Work

Traditional way of Implementing Application-Specific Accelerators

- Considers one Dataset/Task

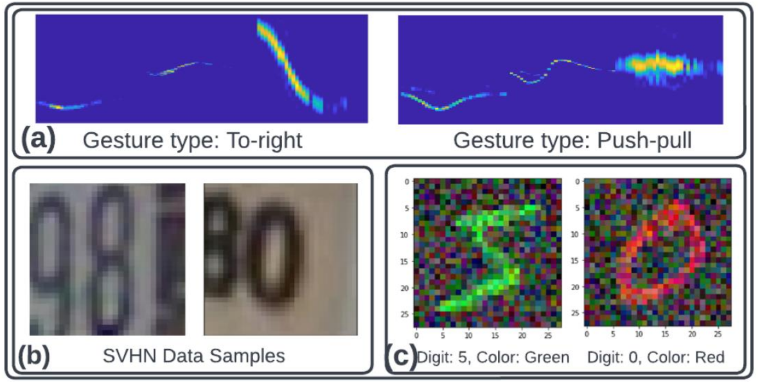
- Considers one sensor Modality (one type of input data)

- Mainly considers correlated tasks



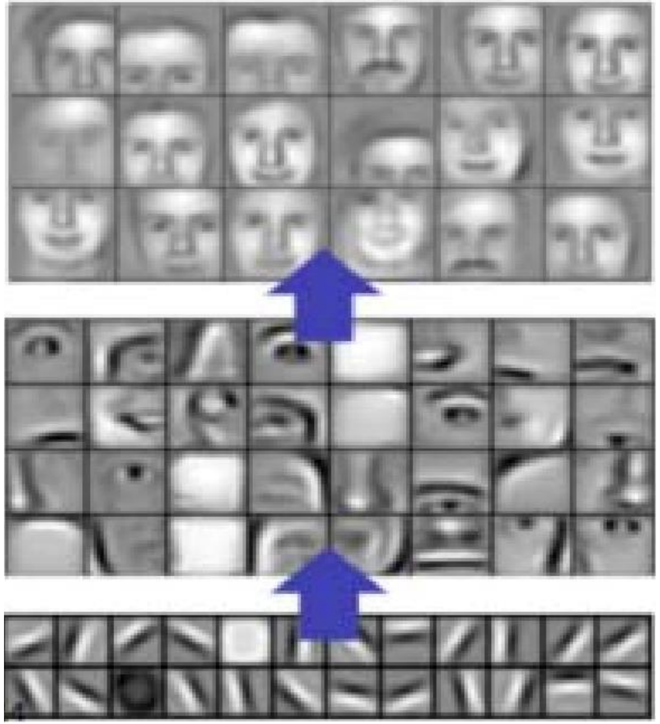
Traditional way of executing on application specific accelerators

- Multiple datasets are emulated as multiple tasks
  - T1: FMCW radar hand gesture samples
  - T2: SVHN samples
  - T3: Transformed MNIST dataset



(a) FMCW radar hand gesture samples (b) SVHN samples (c) Transformed MNIST dataset

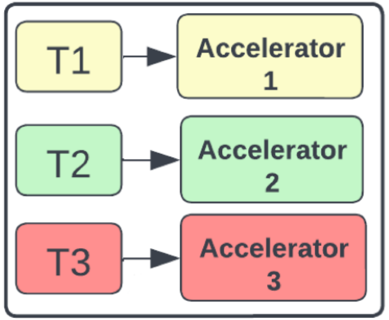
Shared Layers for CNNs Accelerator



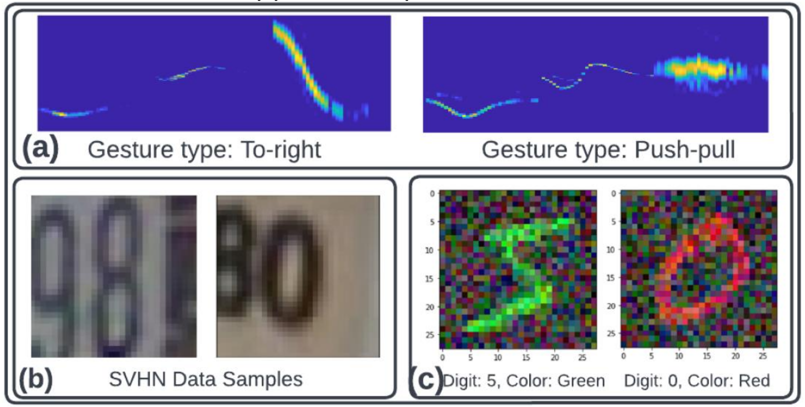
Deeper Layers learns high-level (or more abstract) features

Lower Layers learns low-level features (i.e., edges, curves, blobs, etc.).

Understanding of a Convolutional Neural Network [1]



Traditional way of executing on application specific accelerators



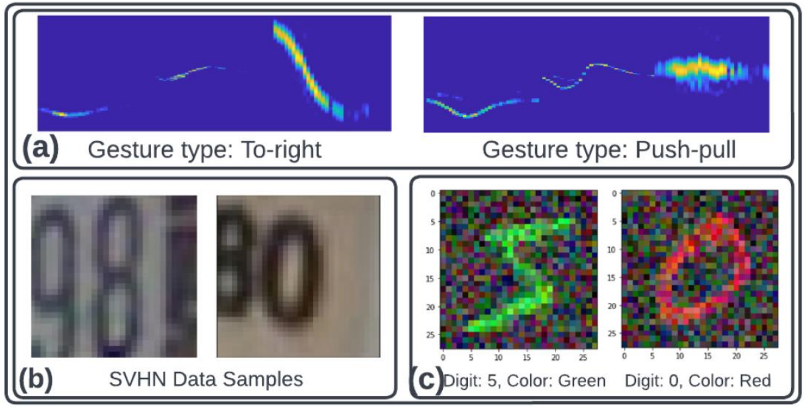
(a) FMCW radar hand gesture samples (b) SVHN samples (c) Transformed MNIST dataset

[1] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 2017, pp. 1-6, doi: 10.1109/ICEngTechnol.2017.8308186.

Shared Layers for CNNs Accelerator

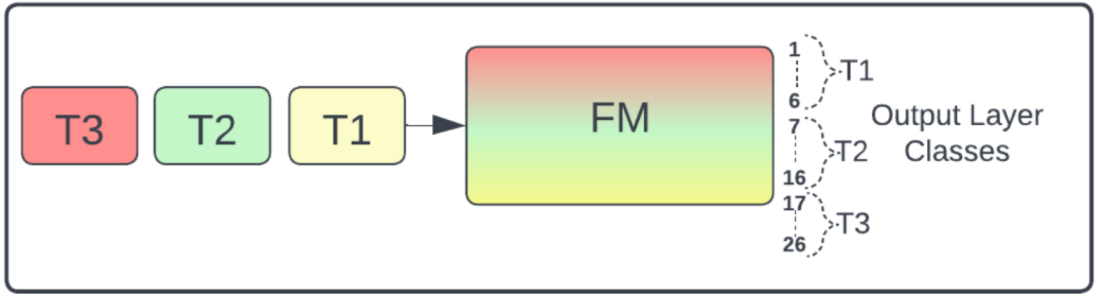
Our Approach:

- Considers multiple datasets/Tasks from different modalities
- Hardware efficient and power efficient
  - One accelerator instead of three
  - Reuse of the weights
- Complements the previously proposed model compression methods (i.e., quantization and pruning considering multiple tasks/datasets)
- Considers un-correlated tasks



(a) FMCW radar hand gesture samples (b) SVHN samples
(c) Transformed MNIST dataset

T1: FMCW radar hand gesture samples
T2: SVHN samples
T3: Transformed MNIST dataset (added noise)

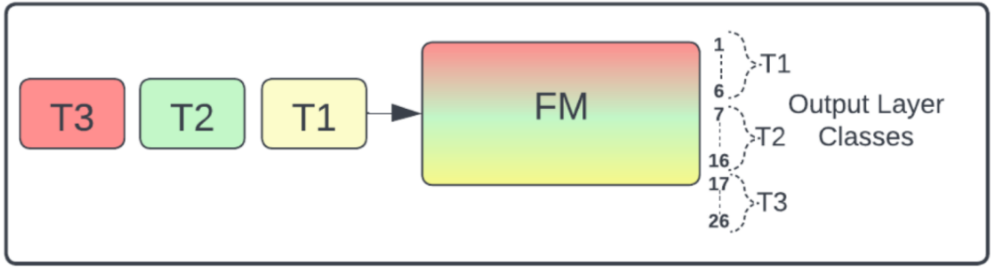


Fused CNN Architecture

Fused and Branched Architectures

Fused model:

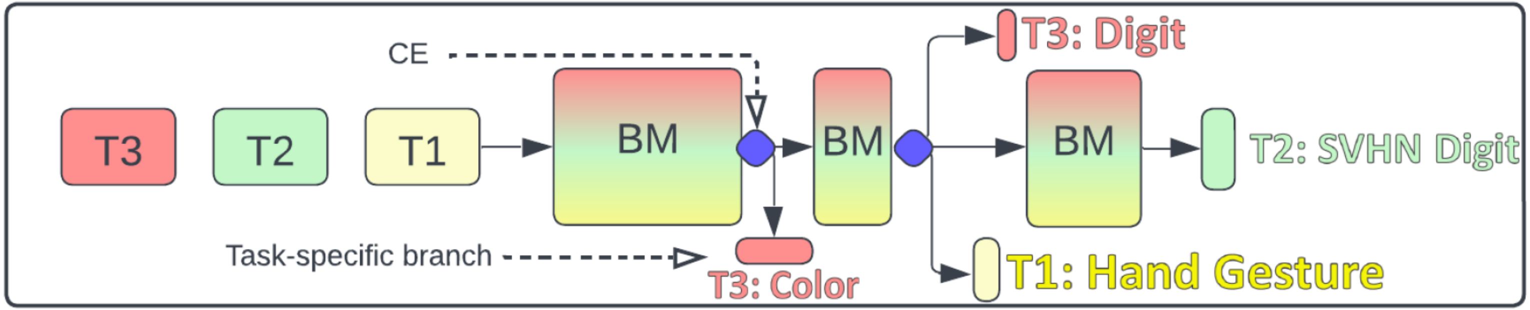
This is an un-branched model, where all the tasks share all the layers of the neural network



Fused Architecture

Branched Model:

It consists of tasks-specific branches and shares only particular layers



Branched Architecture

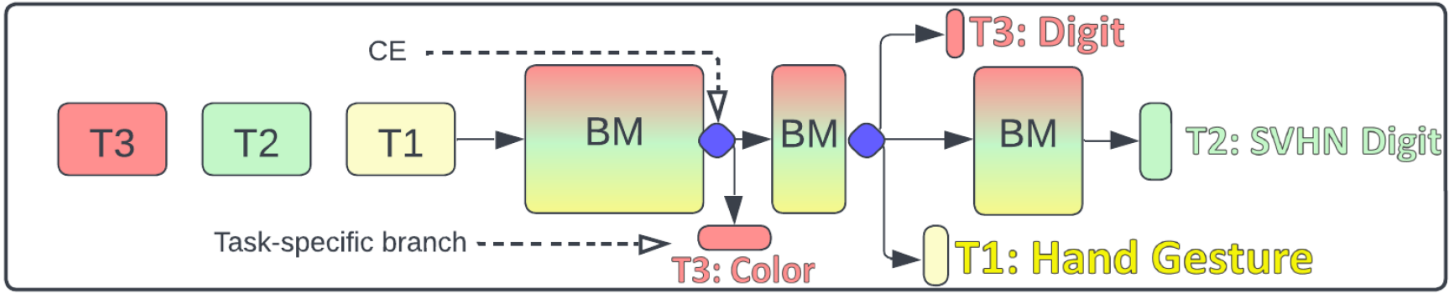
Fused and Branched Architectures

Branched Model:

It consists of tasks-specific branches and shares only particular layers

Advantages:

1) Task isolation in case of faults (faults will not affect entire network)
2) Task-specific bit-stream reconfiguration in FPGAs (no need to reconfigure entire network)
3) Selective replication of only specific layers (e.g., more vulnerable layers or tasks-specific layers)
4) Addition of sub-task (i.e. T3:Color)
5) Adding extra layers to achieve more accuracy for specific tasks



Branched Architecture [2]

[2] R. T. Syed, M. Andjelkovic, M. Ulbricht and M. Krstic, "Towards Reconfigurable CNN Accelerator for FPGA Implementation," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 70, no. 3, pp. 1249-1253, March 2023, doi: 10.1109/TCSII.2023.3241154.

Outline

Implementation Results

Model	Accuracy (%)	HLS4ML Accuracy (%)	Quantization Type	Pruning (%)	Latency (us)	Total Latency(us)	Power (W)	BRAM18K	DSP48E	FF	LUT
FM	T1 = 94.67 T2 = 86.33 T3 = 93.78	T1= 93.33 T2 = 86.4 T3 = 93.67	PTQ (BW= 20,10)	0	T1= 5.21 T2= 5.21 T3= 5.21	15.63	1.724	59	5701	67137	299416
FMP	T1= 100 T2= 90.53 T3= 95.46	T1= 100 T2= 90.38 T3= 95.51	PTQ (BW= 20,10)	50	T1= 5.21 T2= 5.21 T3= 5.21	15.63	1.518	59	4940	53880	181248
FMQ	T1= 94.00 T2= 88.40 T3= 93.55	T1= 92.00 T2= 87.97 T3= 91.77	QAT (Varying BW)	0	T1= 5.21 T2= 5.21 T3= 5.21	15.63	0.925	42.5	2320	39349	215727
FMQP	T1= 97.33 T2= 89.22 T3= 94.36	T1= 96.67 T2= 89.140 T3= 93.91	QAT (Varying BW)	CNN=53 Dense=75	T1= 5.21 T2= 5.21 T3= 5.21	15.63	0.588	43	955	33015	120202
BMQP	T1 = 98 T2 = 89.31 T3 = 95.33 T3c=96.24	T1 = 97.33 T2 = 89.31 T3 = 95.39 T3C= 96.22	QAT (Varying BW)	50-85(Vary for different branches)	T1= 5.13 T2= 5.20 T3= 5.14 T3c= 5.09	15.47	0.624 0.001[1]	46 0.5[1]	1256 0[1]	43171 1801[1]	141008 2589[1]

❖ **FMQP (most optimized FM)**
- Achieves very good accuracy
- Quantized using QAT, Pruned (magnitude based pruning)
- Consume fewer hardware resources as compared to FM, FMP, FMQ

❖ **BMQP (most optimized branched model)**
- Slightly higher accuracy compared to FMQP
- Slightly lower latency compared to FMQP
- Higher power consumption and hardware utilization compared to FMQP
- Offers reliability advantages (discussed before)

Complete Results Analysis: R. T. Syed, M. Andjelkovic, M. Ulbricht and M. Krstic, "Towards Reconfigurable CNN Accelerator for FPGA Implementation," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 70, no. 3, pp. 1249-1253, March 2023, doi: 10.1109/TCSII.2023.3241154.

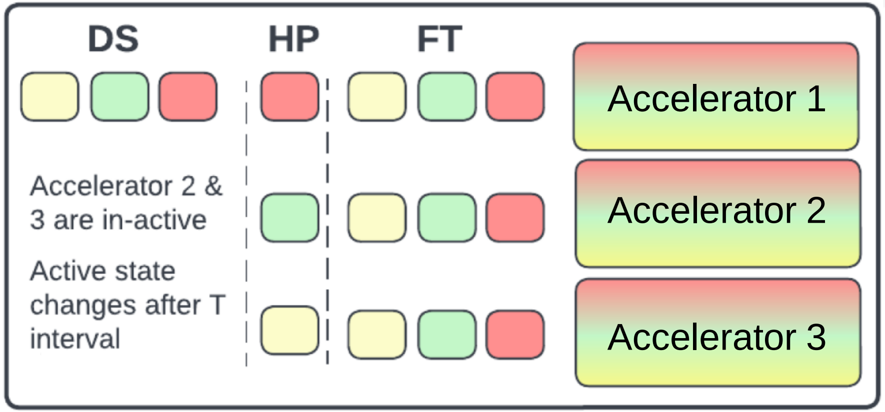
[1] Values marked with a superscript '1' in Table are additional resource utilization when T3c is added

Outline

Reconfigurable CNN Accelerators

- **Multiple hardware copies with multiple operating modes**
  - Trade-off between reliability, power consumption and high performance
  - Typically used for multi-core processors, not common for AI accelerators

- **Operating modes for 3 accelerators**
  - **Fault-tolerant (FT) mode**
    - N-modular redundancy (DMR, TMR)
    - All accelerators execute all tasks
    - SET, SEU, SEU in CRAM
  - **High-performance (HP) mode**
    - Parallel execution of tasks
  - **De-stress mode (DS) mode (Aging aware)**
    - One accelerator is active at a given time
    - Reduces aging and power consumption



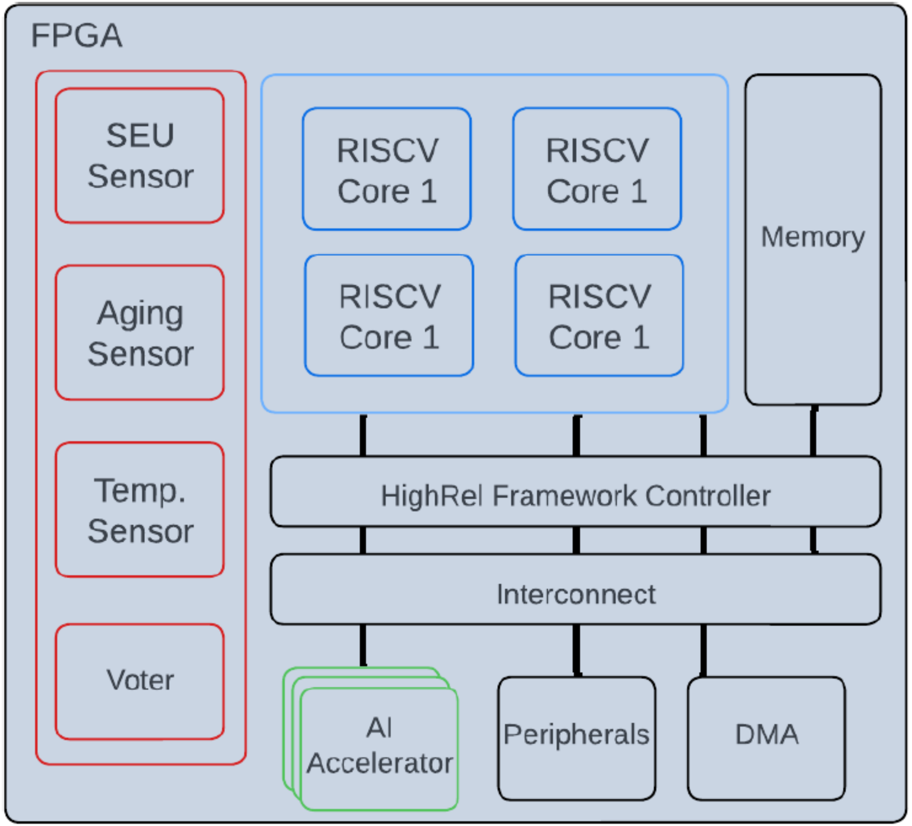
Traditional approach with single-task accelerators would require 9 accelerators for TMR with 3 tasks

Outline

Towards a Fully Reconfigurable/Adaptable AI Processing System

❖ Four Building Blocks

1. On-Chip Sensors
2. Reconfigurable RISC-V Cores
3. Reconfigurable AI accelerators
4. Reconfigurable hardware (i.e., FPGAs)



Summary

1. Fused and Branched models
   - Shared-layers approach for multiple tasks on application-specific CNN accelerators.
   - Experimental results
2. Reconfigurable CNN accelerators
   - FT, HP, and DS modes
   - Implementation results (will be published soon)
3. Future work
   - Towards a Fully Reconfigurable/Adaptable AI processing system consisting of on-chip sensors, quad-core RISCV processors, Reconfigurable AI Accelerators, and Reconfigurable hardware (i.e., FPGAs)

Thank you for your attention!

Rizwan Tariq Syed

IHP – Innovations for High Performance Microelectronics
Im Technologiepark 25
15236 Frankfurt (Oder)
Germany
Phone: +49 (0) 335 5625 264
Fax: +49 (0) 335 5625 671
Email: syed@ihp-microelectronics.com
Linkedin: https://www.linkedin.com/in/syedrizwantariq/

www.ihp-microelectronics.com

innovations
for high
performance
microelectronics

Leibniz
Association

A10