Processing Raw Images Efficiently with the MAX78000 AI Neural Network Accelerator

Mehmet Gorkem Ulkar

Principal Engineer, Machine Learning

Analog Devices

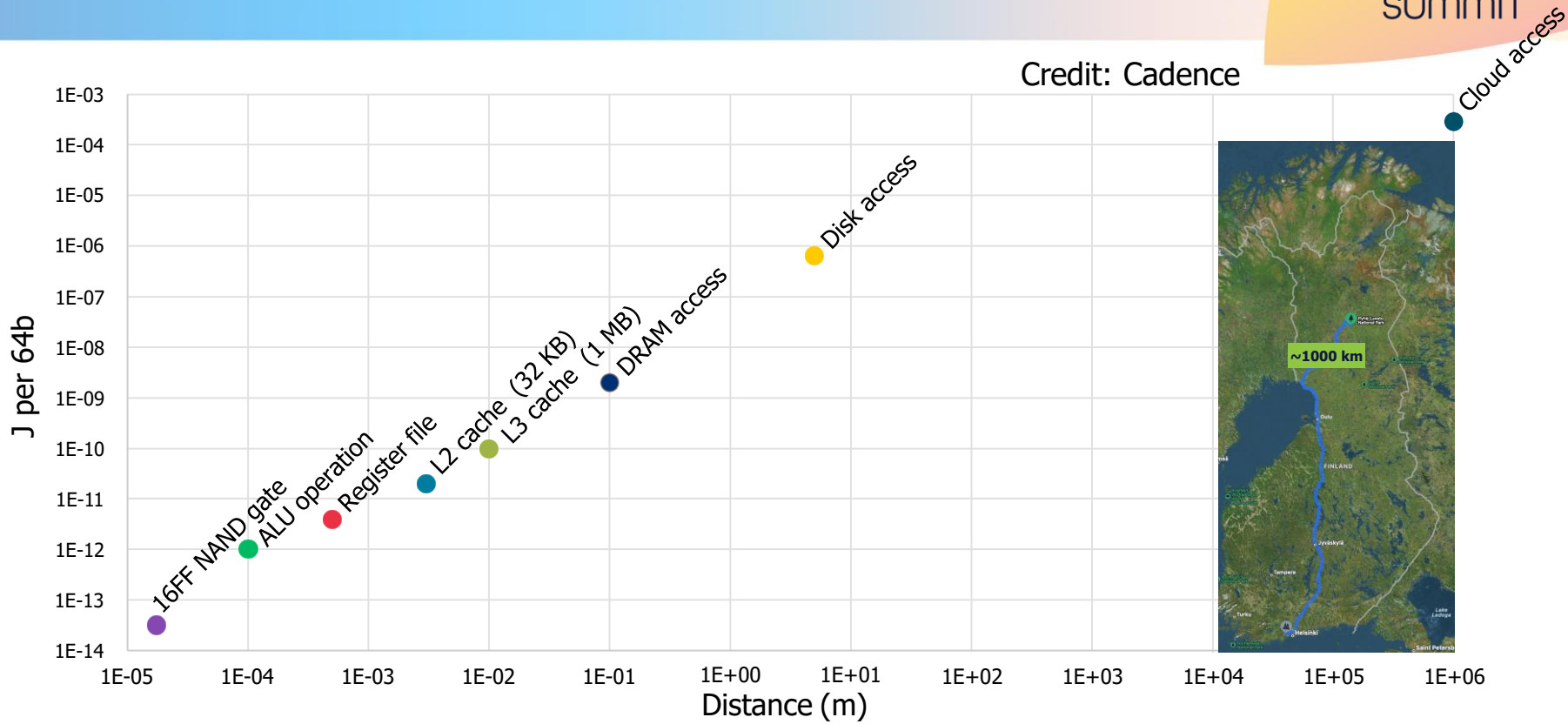
ANALOG DEVICES

Agenda

Mehmet Gorkem Ulkar, PhD
Dallas, TX
Principal ML Engineer

1. Challenges of AI at the edge
2. MAX78000 overview
3. MAX78000 sample applications
4. Energy requirements for data manipulation
5. Proposal: CNN based de-bayerization
6. Results
7. Q&A

Keep Your Data Close: The Physics of Data



Sources:
- Rick Zarr, TI, 2008, The True Cost of an Internet "Click" - estimate of transfer cost for 30KB page from server http://energyzarr.typepad.com/energyzarrnationalcom/2008/08/the-true-cost-o.html
- J Kunkel et al, University of Hamburg 2010,Collecting Energy Consumption of Scientific Data
- Horowitz ISSCC 2014, 1300-2600 pJ per 64b access
- Chris Rowen, Cadence Design Systems, January 2016, Get Real! – Neural Network Technology for Embedded Systems

Software Inference: Slow and Power Hungry

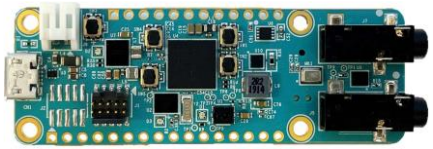
- In inference, computational effort is in **forward propagation**
  - On classic hardware, almost all spent in a triple nested matrix multiplication loop
  - $O(n^3)$ to $O(n^{2.8})$ *
- Very energy intensive even with fast matrix multiply using integer math on DSP or GPU
  – large number of memory accesses

```
// Main loop
for (l = 1; l < NLAYERS-1; l++) {
    // Compute z = w * a
    matrix_mul(&w[l-1], &a[l-1], &z[l]);

    // Add the bias values : z = w * a + b
    matrix_add(&b[l], &z[l]);

    // Compute a = g(z)
    nn_activate(z[l].elements, a[l].elements, lv[l]);
}

matrix_mul(matrix_f32_t *a, matrix_f32_t *b, matrix_f32_t *c)
{
    uint32_t m = a->nrows;
    uint32_t n = a->ncols;
    uint32_t p = b->ncols;

    c->nrows = m;
    c->ncols = p;

    int i, j, k;

    for (i = 0; i < m; i++) {
        for (j = 0; j < p; j++) {
            f_t sum = 0;
            for (k = 0; k < n; k++) {
                sum += a->elements[i * n + k] * b->elements[k * p + j];
            }

            c->elements[i * p + j] = sum;
        }
    }
```

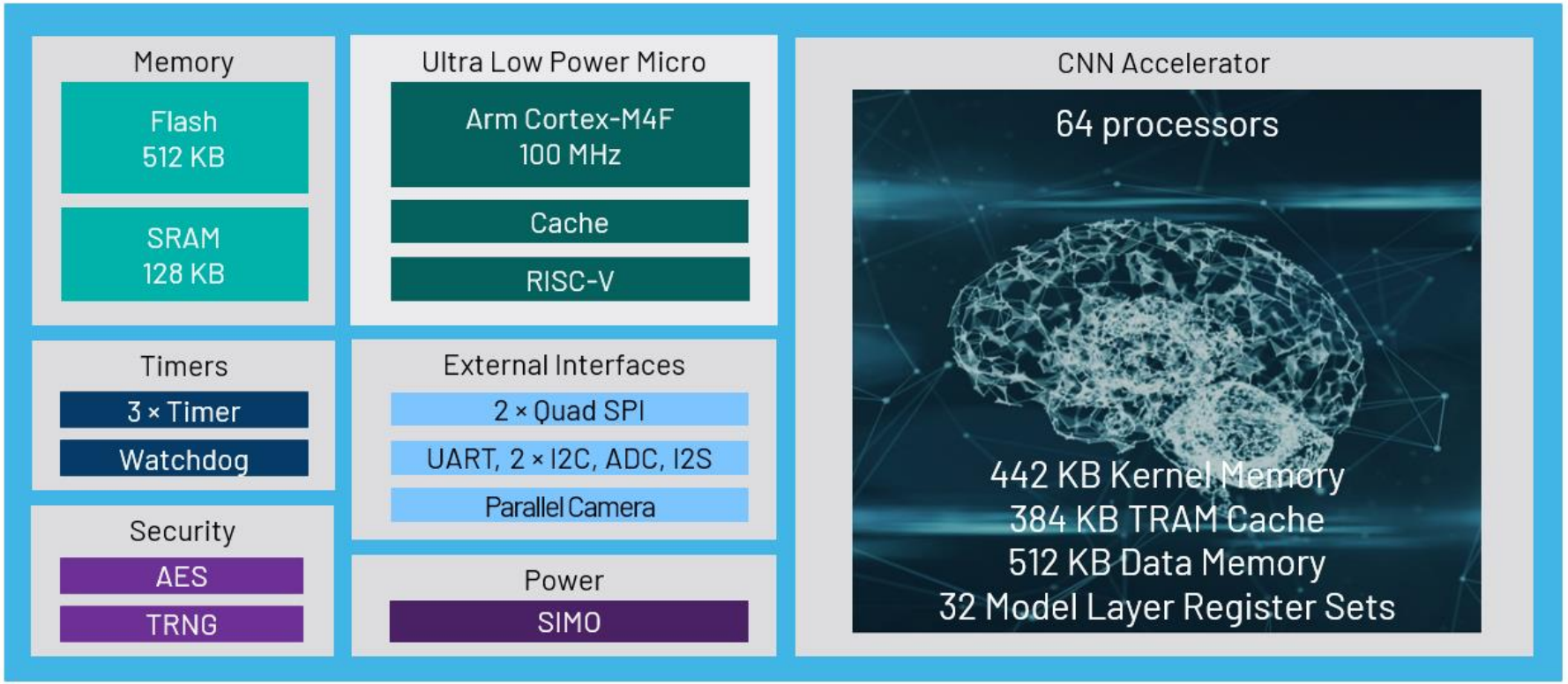
*Strassen's algorithm

CNN Accelerator: MAX78000/MAX78002

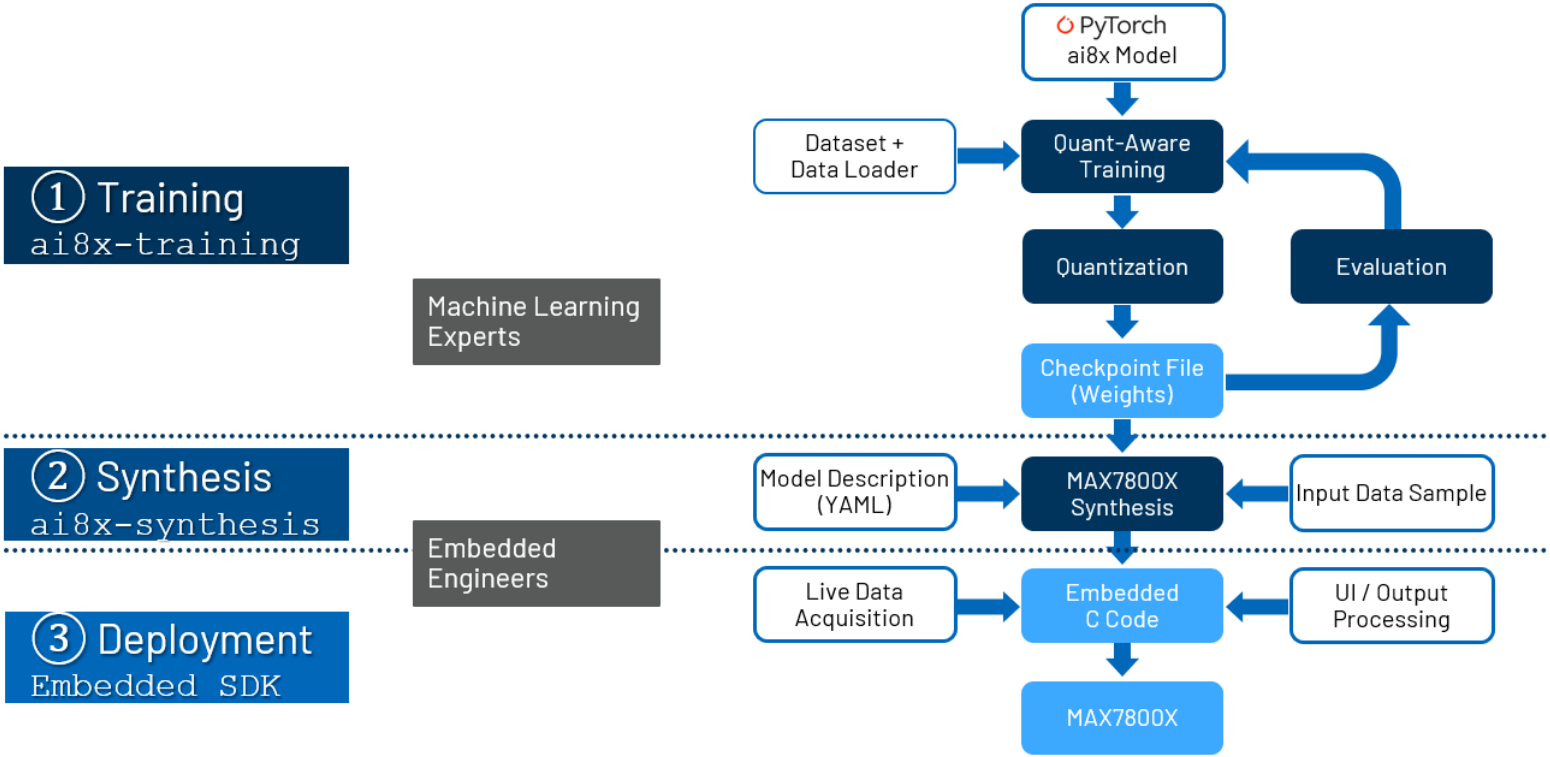
- The conv operation is parallelizable in the channel dimension.
  - 64 processors in total, more channels are processed in a multi-pass fashion
- Proper architecture that minimizes data movement provides energy efficiency
  - Each input channel is processed in parallel using different processors to minimize data movement
  - Each processor uses dedicated memory



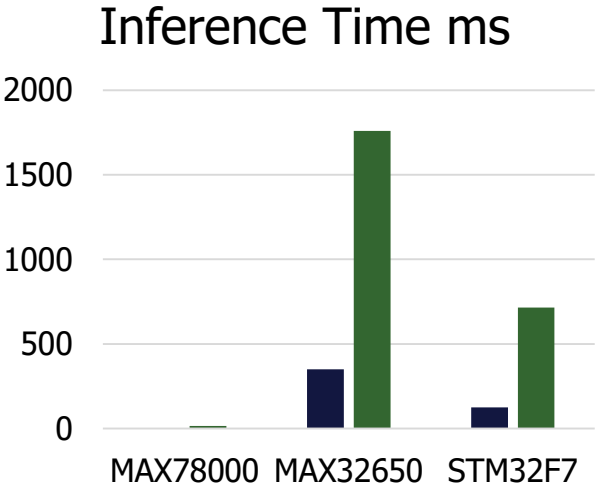
MAX78000 AI Micro - System-on-Chip

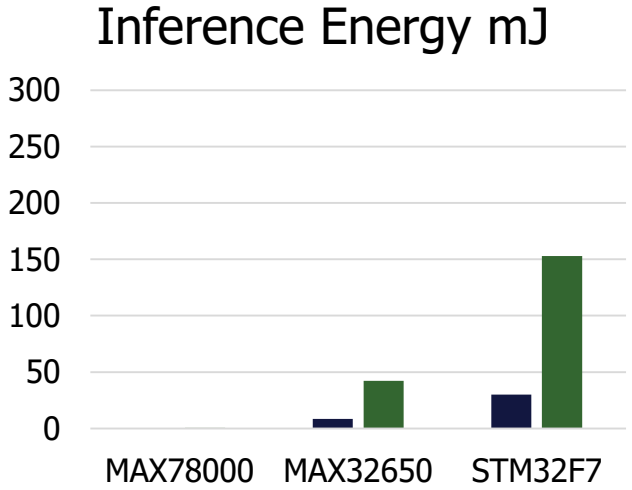


Model, Training, Deployment: Development Flow



MAX78000 Benchmarks





[1]128 billion operations/second,
[2]ARM DSP with CMSIS-NN, running exact same INT8 network as MAX78000, [3]STMF722ZE, internal memory,
[4]Includes time to load input,
[5]Does not include time to load input,
[6]STMF746NG + external 3.3V SDRAM IS42S32400F-6BL + SDRAM controller

Network	MACs	MAX78000 CNN at 50 MHz[1], 1.2V	MAX32650[2] Cortex-M4, 120 MHz, 1.2V	STM32F7[2] Cortex-M7, 216 MHz, 2.1V
KWS20	13,801,088	2.0 ms, 0.14 mJ	350 ms, 8.37 mJ	125 ms, 30.1 mJ[3]
FaceID	55,234,560	13.89 ms[4], 0.40 mJ	1760 ms[5], 42.1 mJ	714 ms[5], 153 mJ + 59 mJ[6]

Battery Life Leader in Independent Benchmarks

ETH zürich

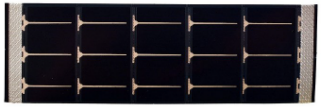
Proof-of-Concept standalone smart camera

Assumptions:

- Trigger time: once per minute.
- Battery capacity: 8.64J.
- Energy per camera image captured: 0.5 mJ.

Platform	Energy per inference (mJ)	Battery Lifetime
SAMD51	5.34	24h30’
Apollo3	1.31	80h00’
Spresense	3.80	46h00
GAP8	0.52	140h15’
VEGA	**0.14**	**225h00’**
xCORE.ai	1.26	81h50’
MAX78000	**0.09**	**244h00’**

Best



Perpetual work with only 100Lux with less than 1 second starting time

PROJECT BASED LEARNING

TINY ML

ANALOG DEVICES

A Battery-Free Long-Range Wireless Smart Camera for Face Detection: An accurate benchmark of novel Edge AI platforms and milliwatt microcontrollers Michele MAGNO, Head of the Project-based learning Center, ETH Zurich, D-ITET, EMEA TinyML Talks June 2021

Thinking About Edge AI Use Cases…

If my [application] { sees / hears / senses } _____

object/sound/event/situation/…

then do _____

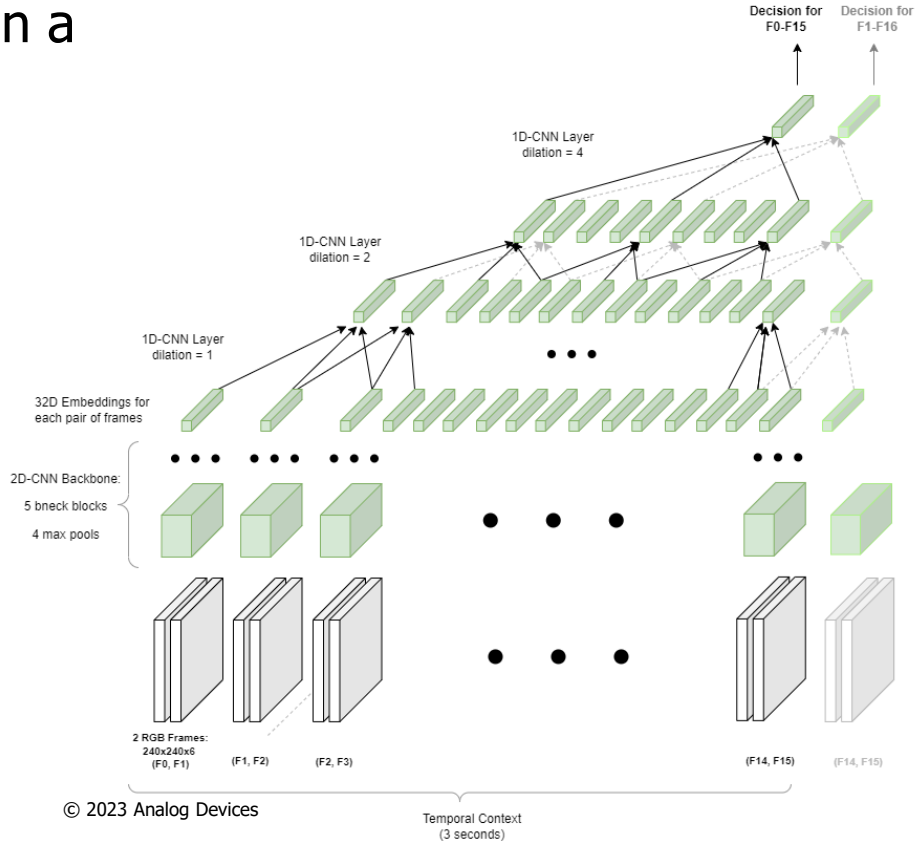
action

If my [camera] sees a bear, then take a high-resolution picture and send over cell network
If my [thermostat] hears glass break, then send a text message to the owner
If my [factory robot] sees a person nearby, then shutdown until they leave
If my [pet door] sees a cat with a mouse in its mouth, then lock the pet door and send me a text message

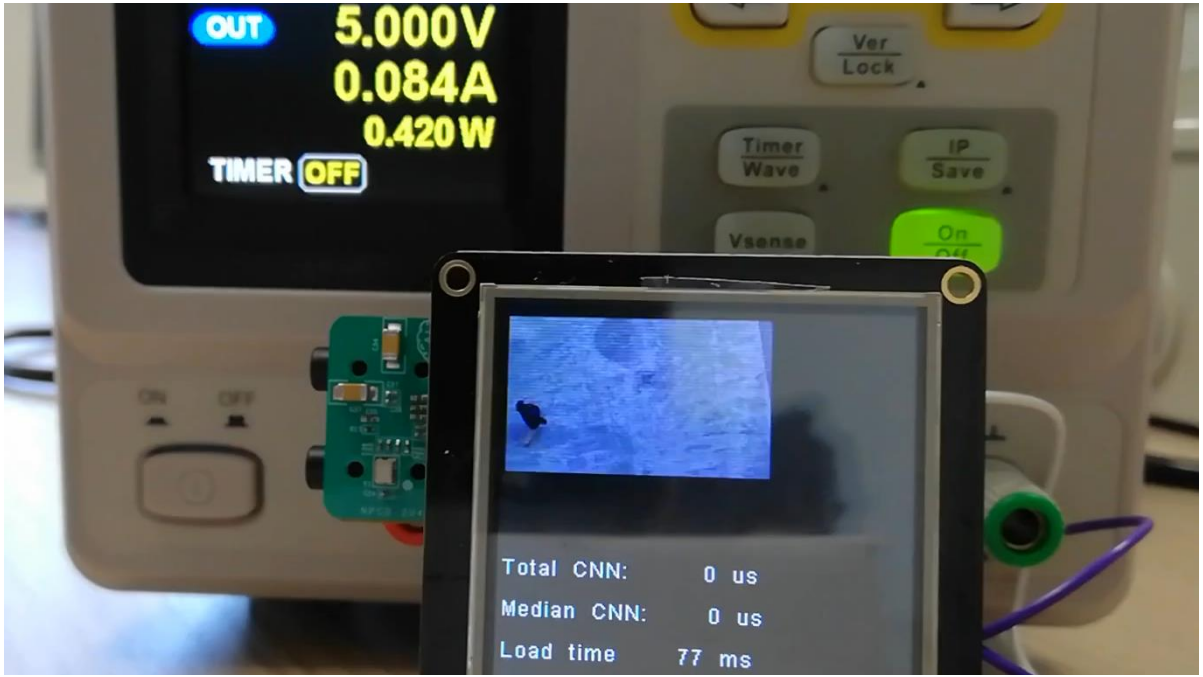
Action Recognition

- Embeddings saved in memory on a rolling basis
  - No redundant calculations

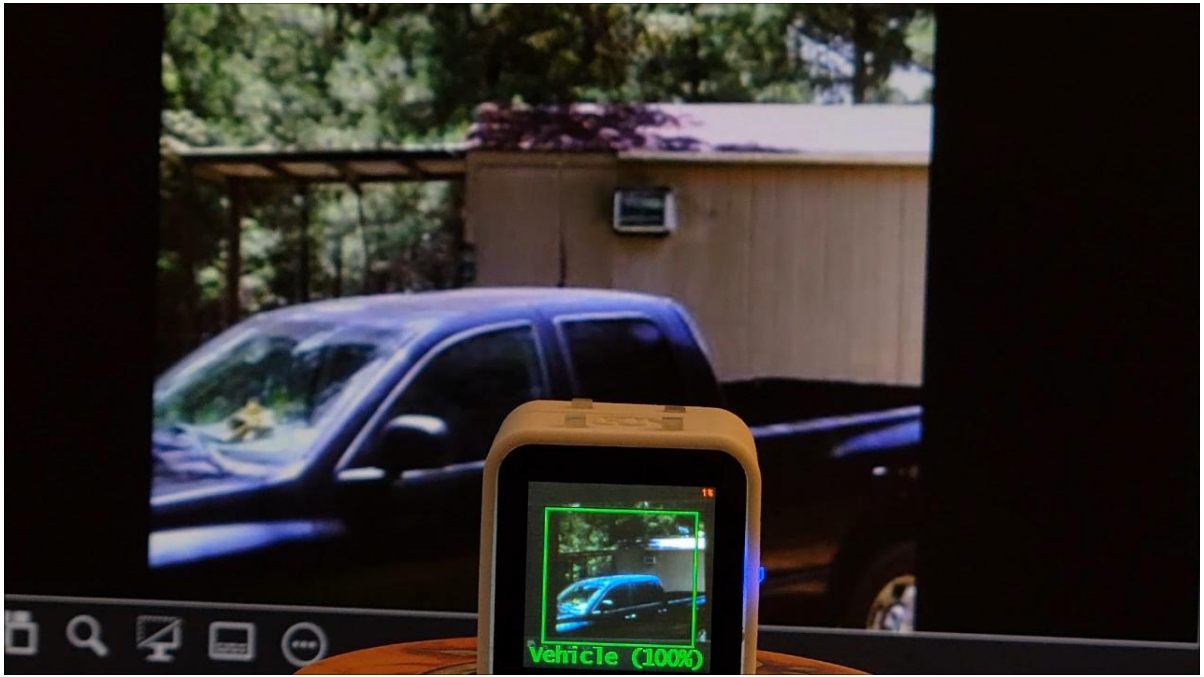
Dataset	Validation Acc.	Parameters
Kinetics-400 (4 classes + other)	79.8%	379k



People Tracking



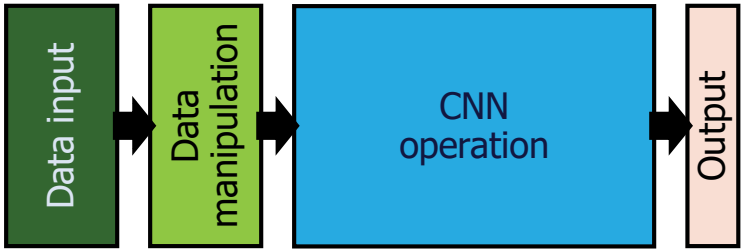
Trail Camera

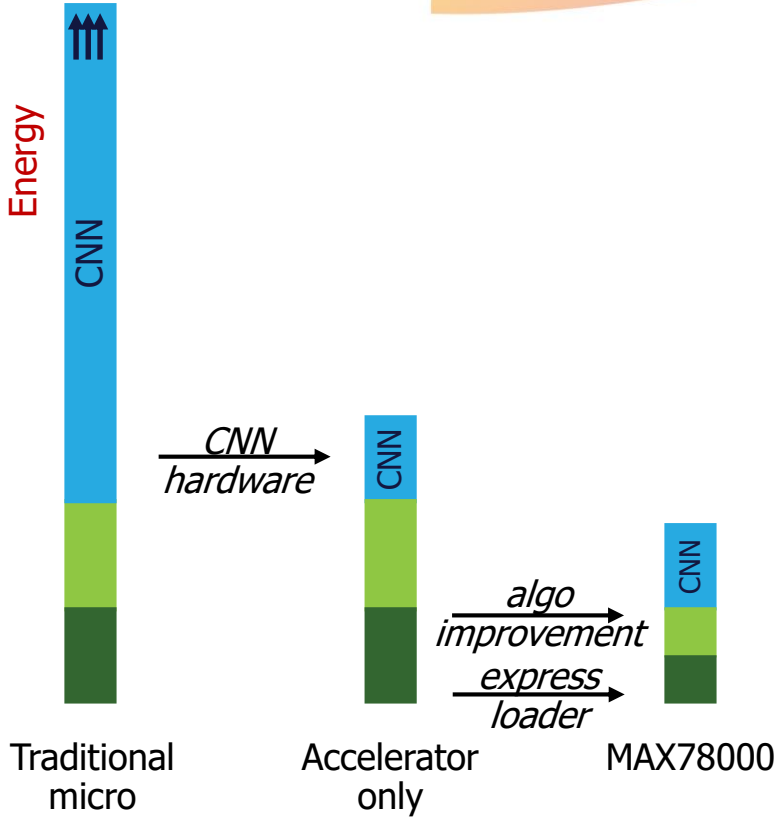


https://github.com/MaximIntegratedAI/refdes

System Energy: From Traditional Systems to MAX78000

- Accelerator drastically lowers CNN energy
- Input and **data manipulation** become much larger *relative* contributors to energy
- MAX78000 improves data loading, better algorithms can help with data manipulation: e.g. **better ways of handling raw images**





14

Data Manipulation: Debayerization

In order to obtain an RGB format, the raw image must be debayerized. There are several debayerization methods*:

- Bilinear Interpolation
- Sequential Demosaicing
- Iterative Demosaicing
- Machine Learning Methods
- Adaptive Color Plane Interpolation

  - ▶ Outside the CNN accelerator
  - ▶ Increased system energy consumption

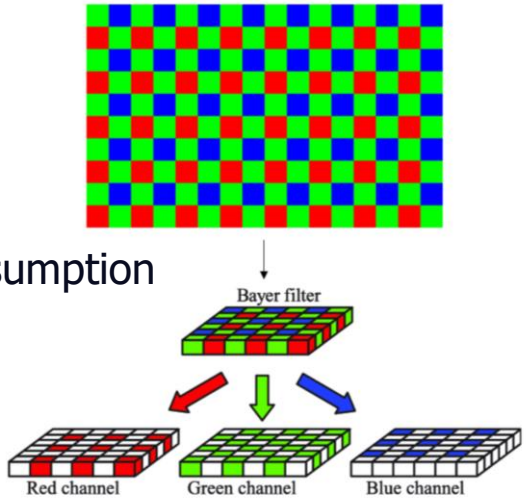


Figure 1. Bayer Filter (Nkansah et. al., 2022)

* Dammer, K., Grosz R., (2017). Demosaising using a Convolutional Neural Network approach. Lund University, Lund, Sweden.

CNN based Debayerization

- Approach 1: Learning the manipulation & interpolation by a CNN model and embedding this network into an accelerator ⟶ Efficient way of debayerization

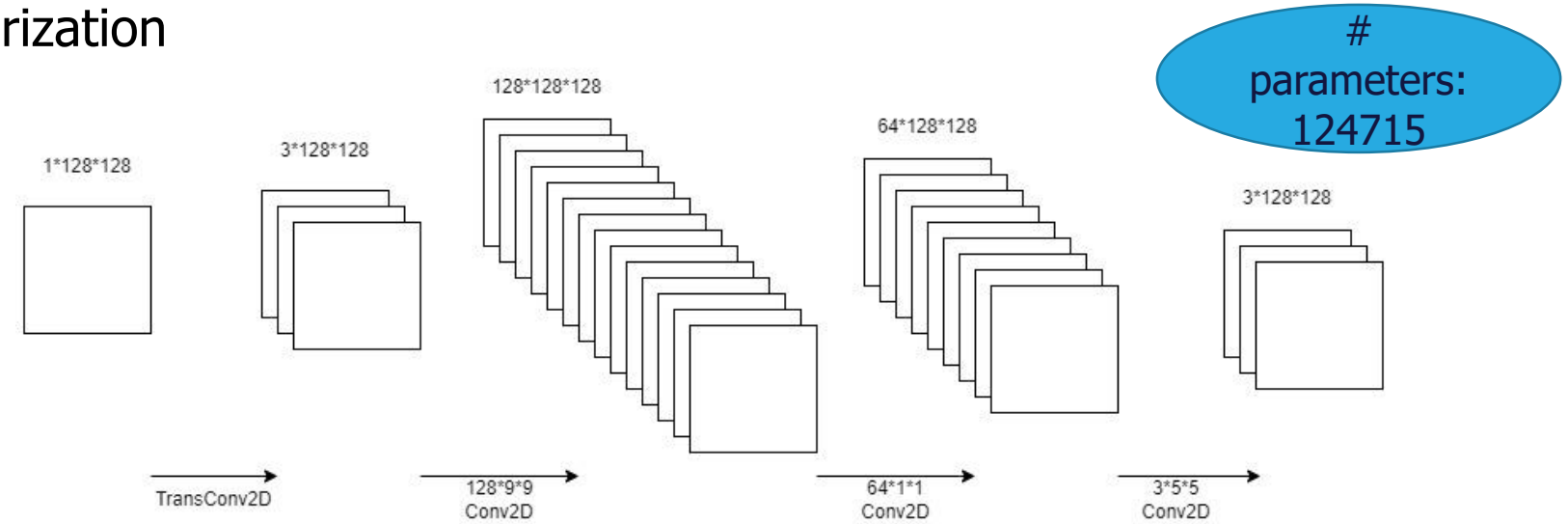
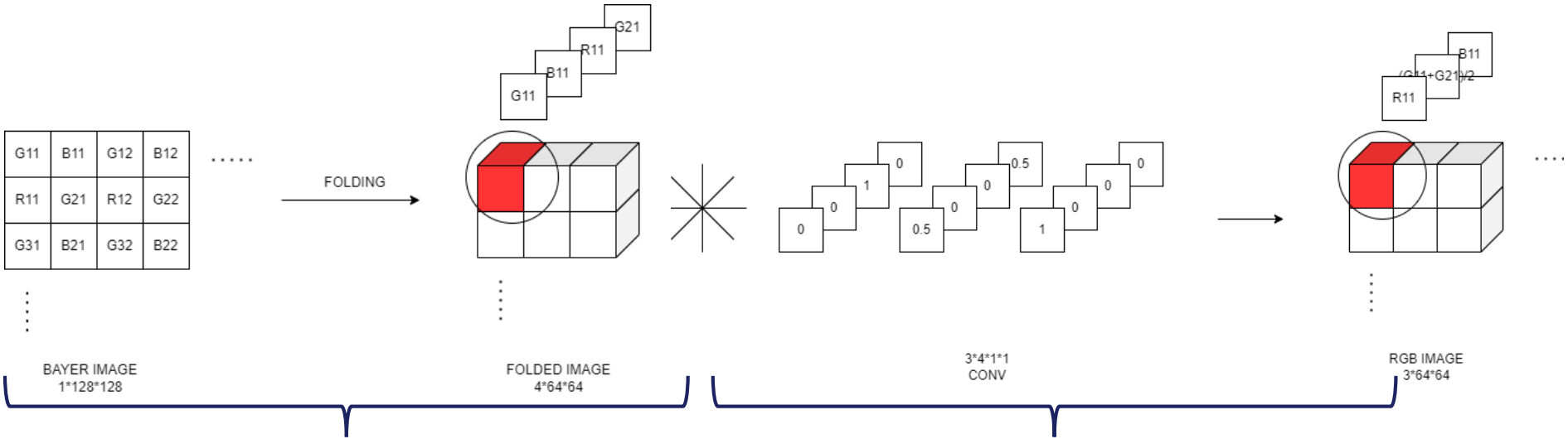


Figure 3. The Network of B2RGBNet (Syu et. al., 2018)

CNN based Debayerization

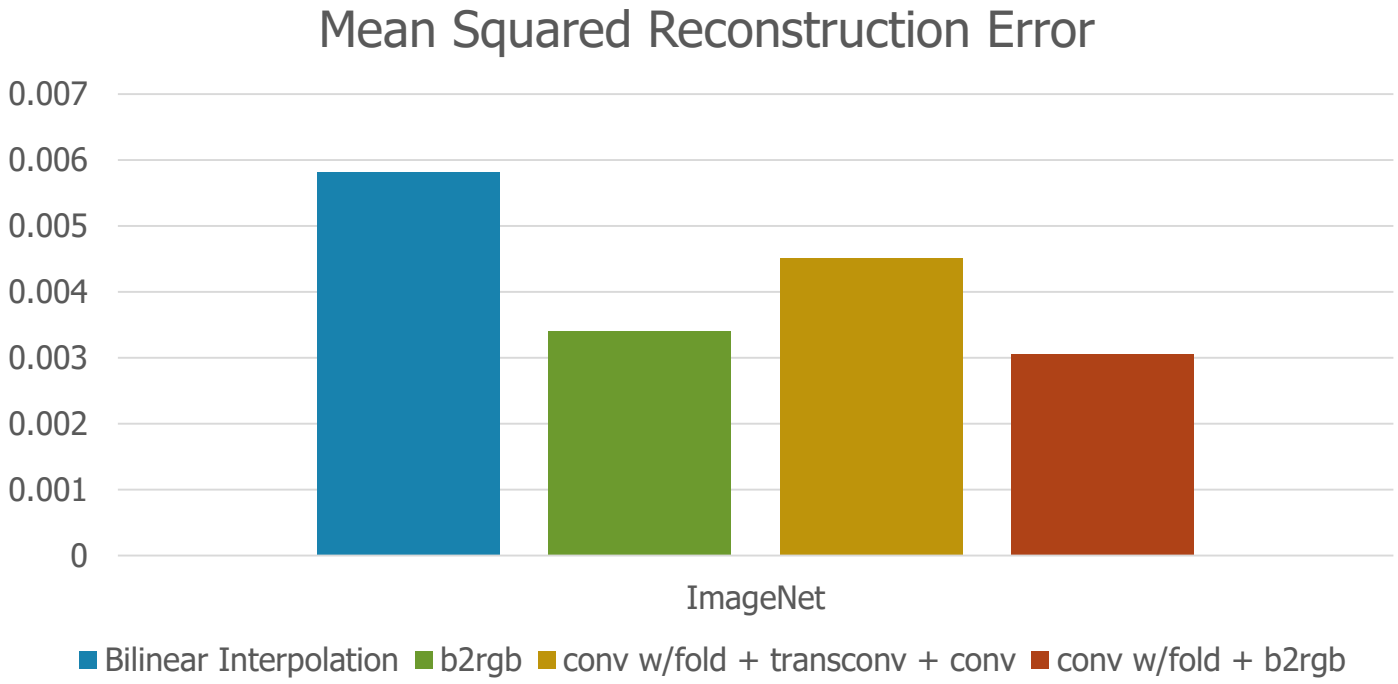
- Approach 2: Using folding and fixed 1x1 kernels



Step 1: Folding the pixels into channels

Step 2: Convolution with the fixed kernel to obtain RGB

Accuracy Results



Conclusion

- MAX78000 enables battery-powered smart applications at the edge
- Effective data manipulation and preprocessing are much more important when using highly-efficient NN inference engines
- Two methods proposed to perform interpolation inside CNN accelerator, MAX78000
- Results show better accuracies compared to simple conventional interpolation; the work is ongoing

Resources

- We are waiting for you at the ADI booth!
- Upper-level AI repo: https://github.com/MaximIntegratedAI
- Open-source training repo: https://github.com/MaximIntegratedAI/ai8x-training/
- Open-source synthesis repo: https://github.com/MaximIntegratedAI/ai8x-synthesis
- Data-folding paper: L3U-net: Low-Latency Lightweight U-net Based Image Segmentation Model for Parallel CNN Processors https://arxiv.org/pdf/2203.16528.pdf
- B2RGBNet paper: Learning Deep Convolutional Networks for Demosaicing https://arxiv.org/pdf/1802.03769.pdf