3PXNet: Pruned-Permuted-Packed XNOR Networks for Edge Machine Learning

WOJCIECH ROMASZKAN & TIANMU LI

NANOCAD LABORATORY

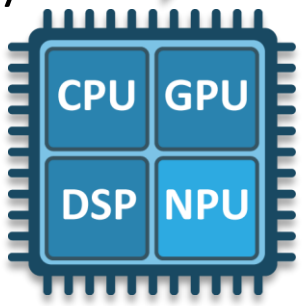
ELECTRICAL AND COMPUTER ENGINEERING DEPARTMENT

UNIVERSITY OF CALIFORNIA LOS ANGELES

Edge Machine Learning

Energy
Latency
Privacy

Edge Inference



High Model Complexity

10s-100s of MBs

1s-10s GOPS

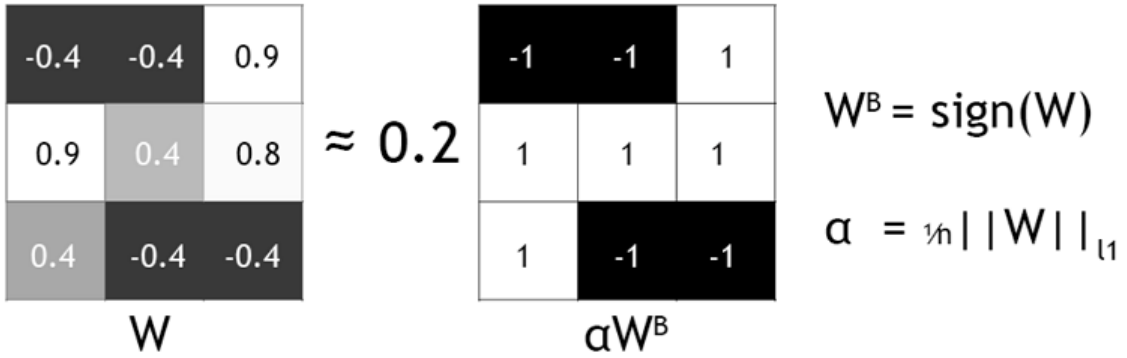
How do we bypass this?

Model Compression

Reduced Precision

Images: Freepik.com

Binarized Network & Pruning

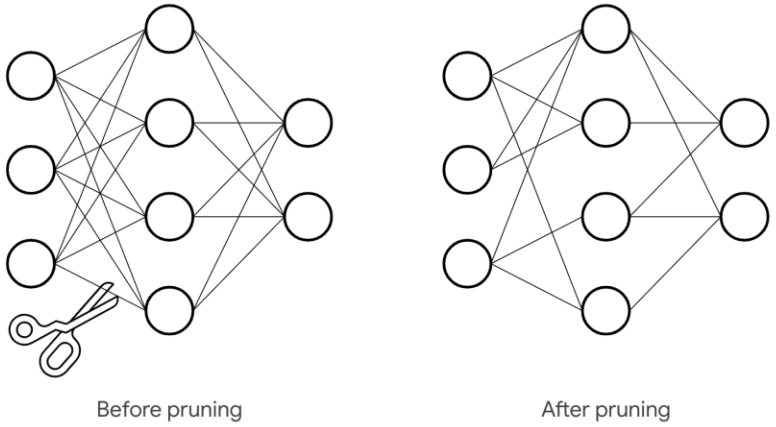


https://software.intel.com/en-us/articles/binary-neural-networks

Binarization

+

Pruning



https://medium.com/tensorflow/tensorflow-model-optimization-toolkit-pruning-api-42cac9157a6a

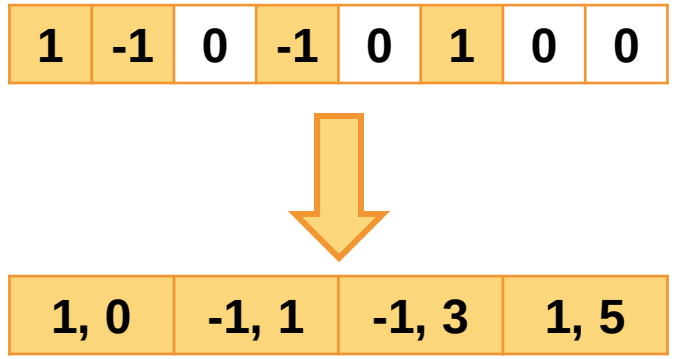
Challenges in Pruning XNOR Networks

- No zero – cannot pad missing values
- Cannot utilize SIMD XNOR multiplication – Naively-Pruned (NP)
- Need to align to word width of processor

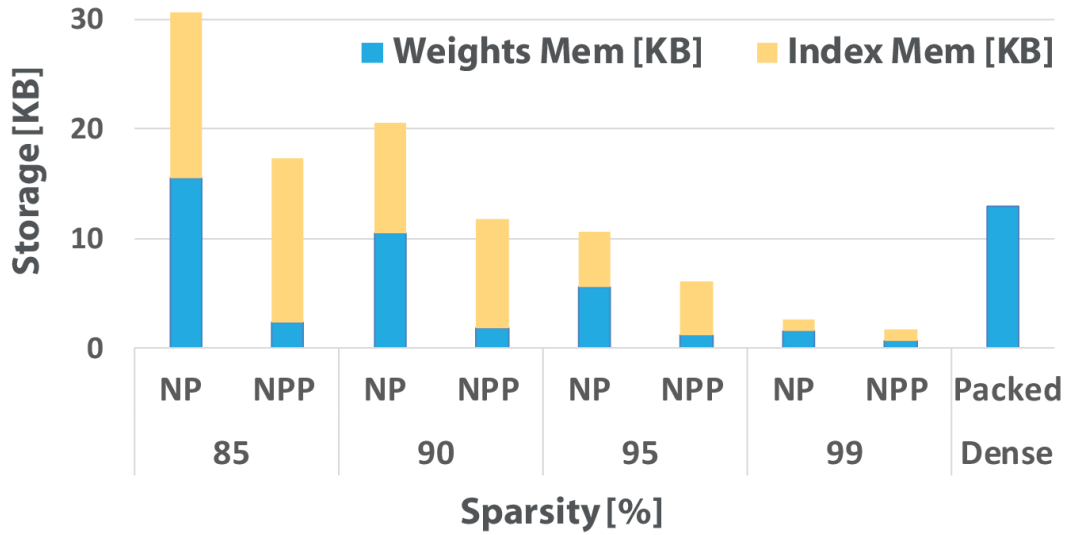
Naively-Pruned (NP)

Challenges in Pruning XNOR Networks

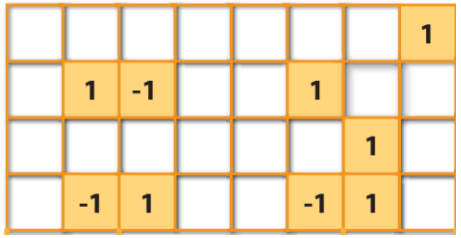
- Each non-zero weight requires an index
- 1-bit weight + 8(16)-bit index – high overhead



Naively-Pruned, Packed (NPP)



Pruning, Packing, and Permutation



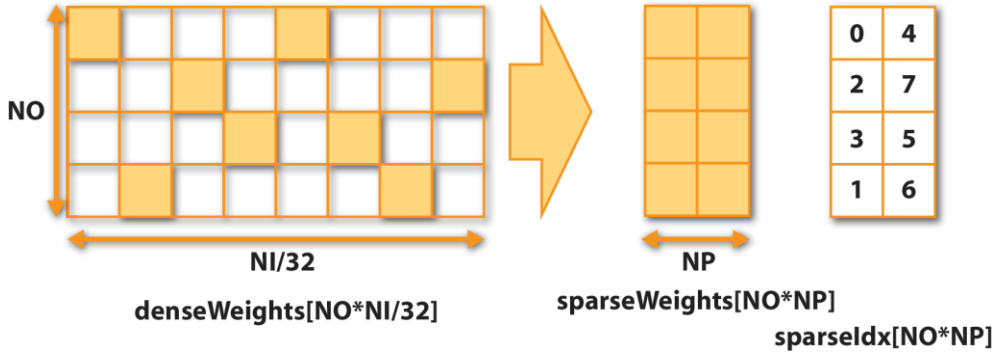
1) Prune: binarized network with zero weights

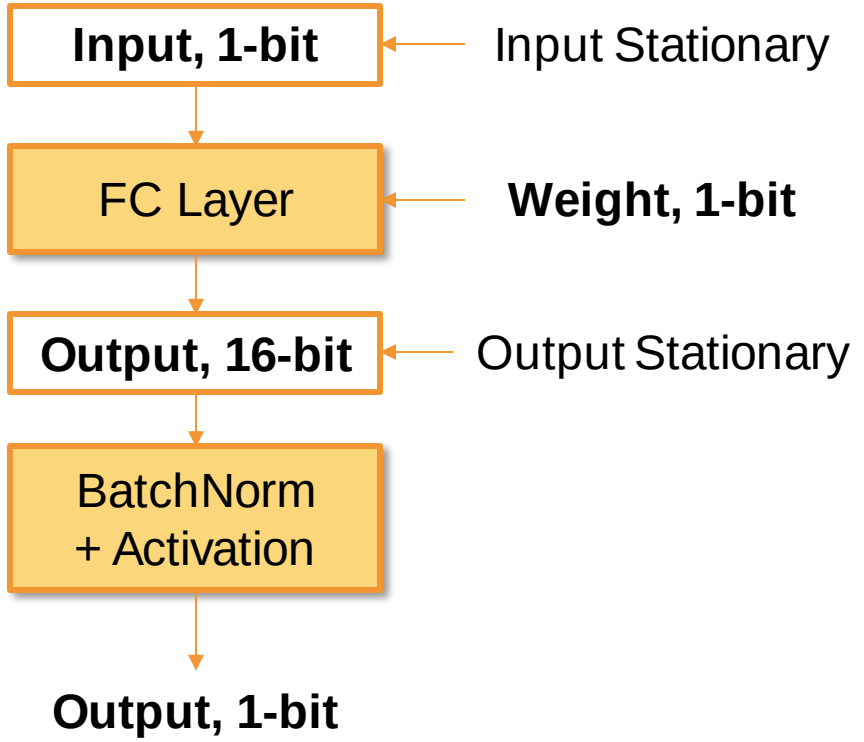
2) Pack: prune and keep weights in packs

3) Permute: permute columns to reduce pruning error

Implementation – Fully-Connected Layers

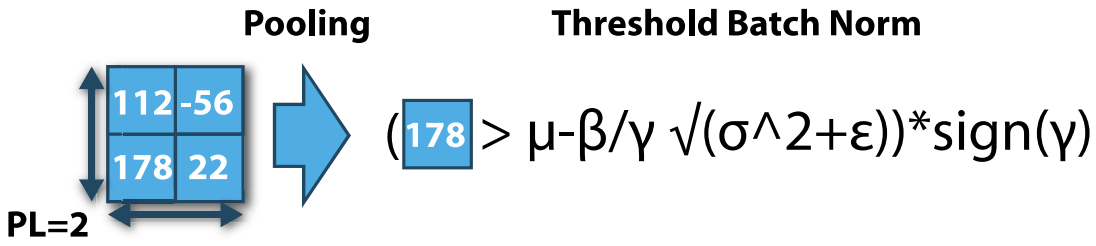
- Input stationary
  - Small enough to fit into cache
- Only store non-zero weights
- Outputs binarized on the fly

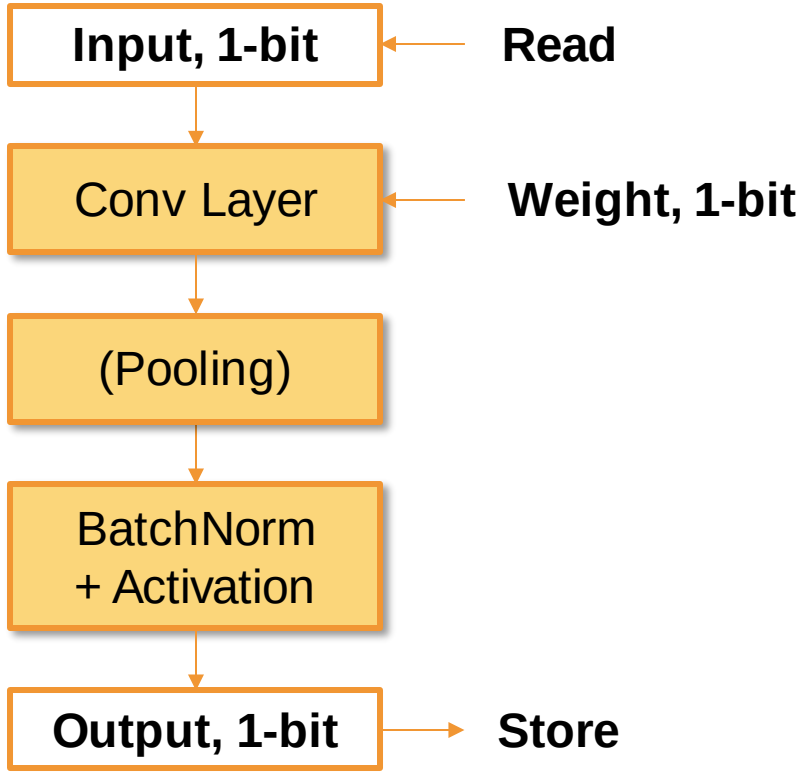




Implementation – Convolutional Layers

- Direct PressedConv approach
- Channel-last layout
- Weight stationary
- Fused pooling + batch normalization

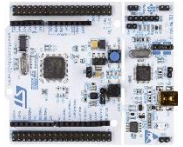




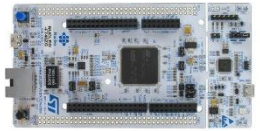
Evaluation

- Platforms

Nucleo F031K6: Cortex M0, 4KB SRAM, 32KB Flash, 48MHz



Nucleo F103RB: Cortex M3, 20KB SRAM, 128KB Flash, 72MHz



Nucleo F746ZG: Cortex M7, 320KB SRAM, 1MB Flash, 216 MHz

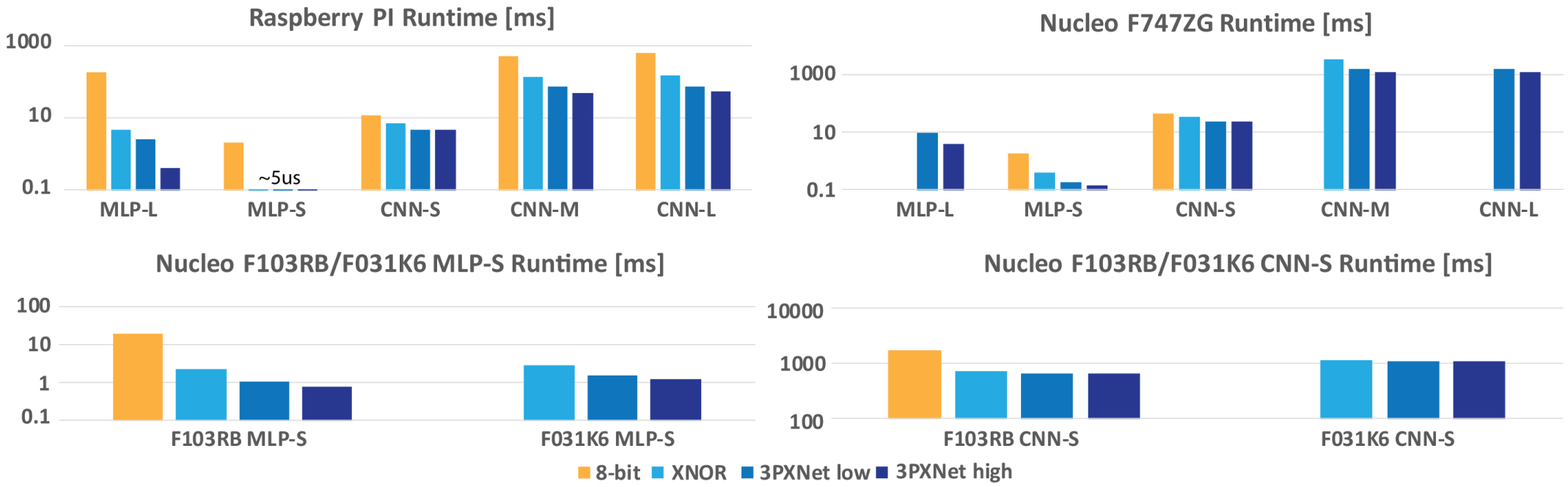
Raspberry PI B+ - Cortex A53, 2MB L2, 1GB DRAM, 1.2GHz

Source: stm.com, raspberrypi.org

- Models/Datasets:
  - MNIST:
    - Large MLP: 784-4k-4k-4k-10
    - Small MLP: 784-128-10
    - Small CNN: 32CONV5-MP2-32CONV5-MP2-10
  - CIFAR-10:
    - Medium (Large) CNN: 128CONV3x2-MP2-256CONV3x2-MP2-512CONV3x2-MP2-(1024x2)-10

Result - Runtime

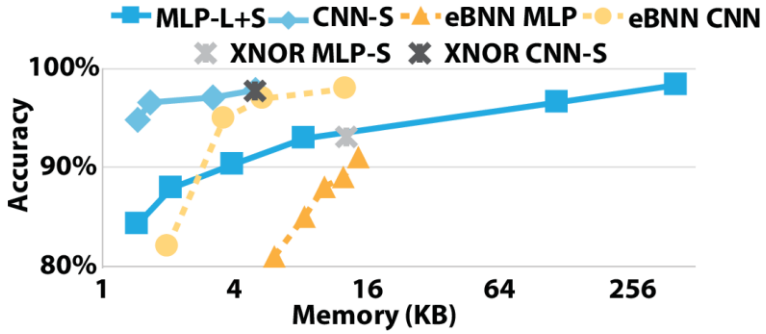
- Up to 3X(25X) faster compared to dense binarized (8-bit) model
- Enables model execution
- Speedup limited by:
  - Input binarization & batch normalization
  - First layer in CNNs



Result - Accuracy

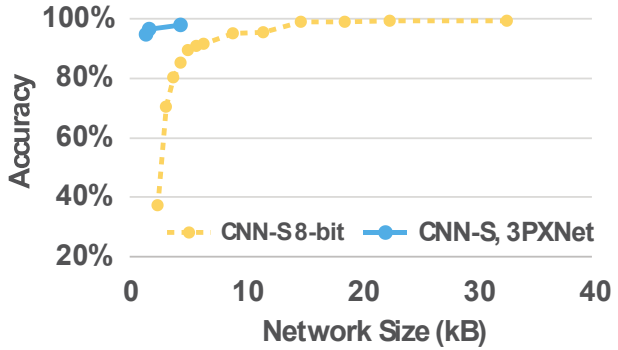
vs. eBNN

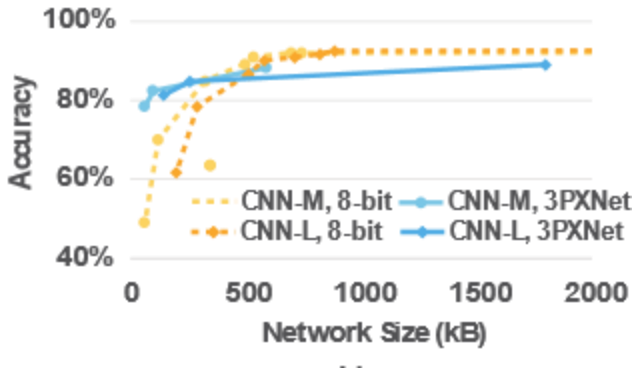
- Better than using a smaller network



vs. 8-bit weight pruning

- Smaller initial size
- Smaller indexing overhead





Summary

- Packing-aware pruning of XNOR networks
- 22X-307X size reduction vs. 8-bit
- $first$ software implementation of sparse binarized networks
- Real-time inference on IoT platforms