Optimizing Convolutional Weight Mapping for Energy-Efficient In-Memory CNN Inference

Johnny Rhe
Advisor: Dr. Jong Hwan Ko
IRIS Lab., Sungkyunkwan University, Suwon, Korea

Abstract

With their high energy efficiency, In-Memory Computing (IMC) architectures are increasingly being used for the convolutional neural networks (CNNs) inference. As most of the IMC-based computing energy is consumed by data conversions between analog and digital domains at every computing cycle, it is important to reduce the number of computing cycles to minimize the energy consumption. The objective of this research is to design mapping-aware optimization in IMC architectures for energy-efficient IMC-based CNN inference with algorithmic, network architectures and their training innovations.

Background

Convolutional Weight Mapping Methods

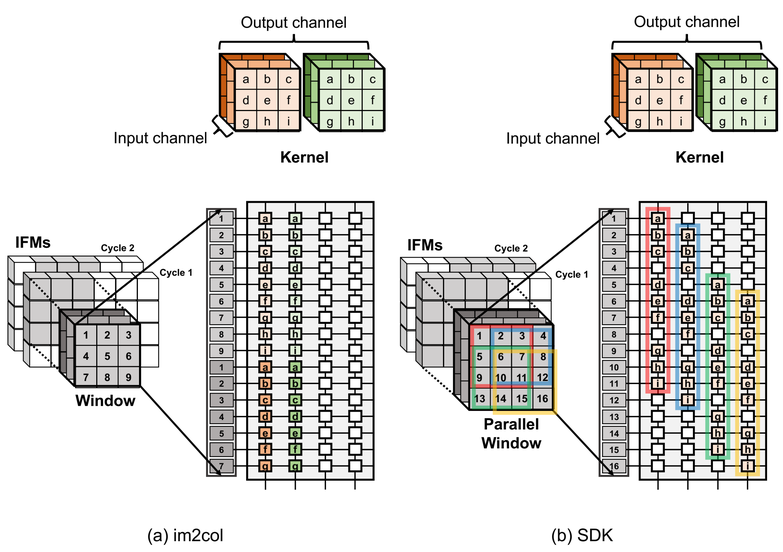

Figure 1. Existing weight mapping methods.

- **Image to Column (im2col)**
  - A widely used mapping method (Fig.1 (a)).
  - 3D-shaped kernel to an IMC array column.
    → The utilization depends on array and layer sizes.
- **Shift and Duplicate Kernel (SDK)**
  - Advanced mapping method (Fig.1 (b)).
  - Parallel window (PW) to reuse input data.
    → Multiple output data at a single cycle.

Motivation

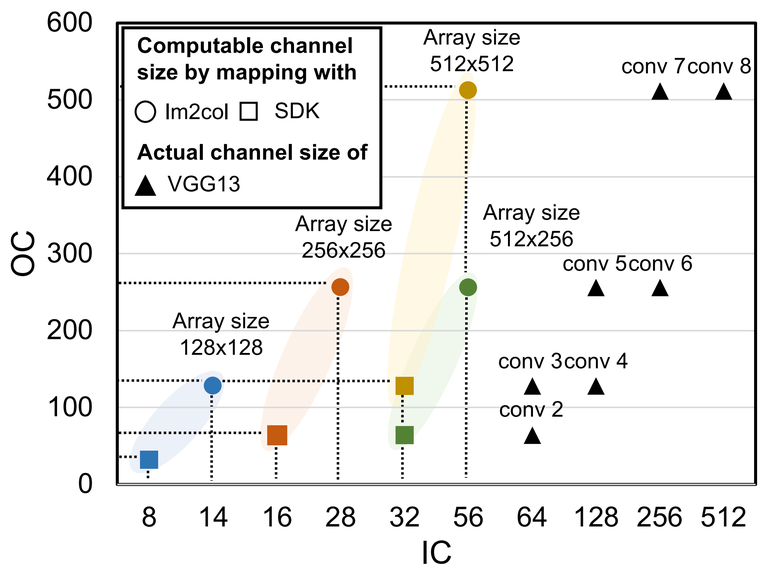

Figure 2. The figure indicates that the conventional mapping methods cannot map the entire channels of general layers into IMC arrays at one cycle.

1. Limited Weight Mapping Algorithms
- Non-optimal mapping method.
- Low utilization in the given IMC array.
  → Novel mapping method is needed.

2. How to Compress the Weight Matrix?
- Smaller IMC array sizes.
- Little pruning schemes for the mapping methods.
  → Novel weight compression schemes are needed.

Proposed Methods

1. Novel Mapping Method
- Various Window SDK (VW-SDK) [1](Fig. 3).
- Dividing the channels into several tiles (channel tiling)
  → Deploying duplicated kernels in limited array sizes.
- Various 2D shapes of the parallel window.
  → Extended search space of PW shapes.

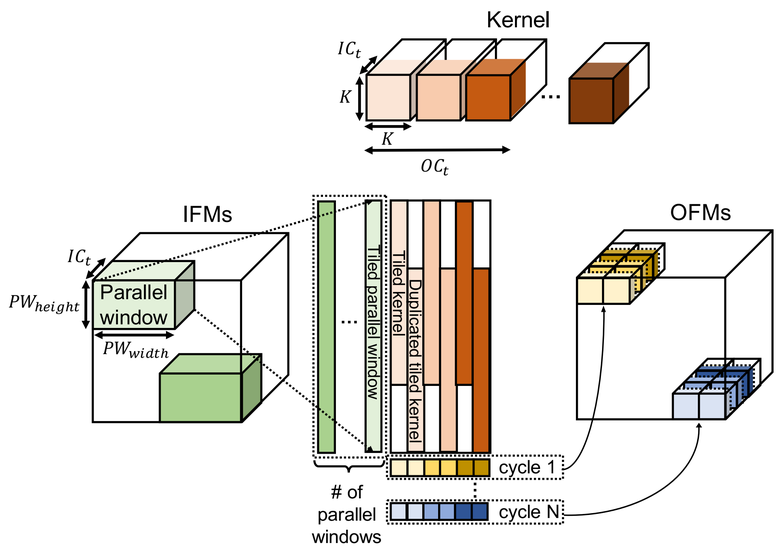

Figure 3. Mapping concept of VW-SDK.

2. Weight Compression w/ Pruining
- Variable Windows and Channels SDK [2].
- VW-SDK + channel pruning.
- Removes some channels that lead to low utilization.
  → Further optimize the VW-SDK mapping method.
- Pruning-Aided Row-Skipping (PAIRS) [3](Fig. 4).
- Pattern-based pruning for IMC architectures.
- SDK mapping-aware pruning pattern design.
  → Maximum weight matrix compression rate.

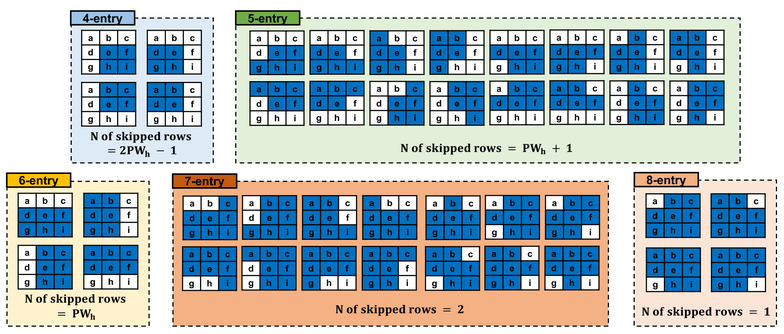

Figure 4. The proposed patterns for the SDK mapping.

3. Weight Compression w/o Pruning
- Kernel Shape Kerntrol (KERNTROL) [4](Fig. 5).
- Omit the weight element (weight omission).
  → Prevents pruning the important weight element.
- Every column corresponds to its own unique kernel.
- Novel training technique.
  → Prevents pruning the important weight element.

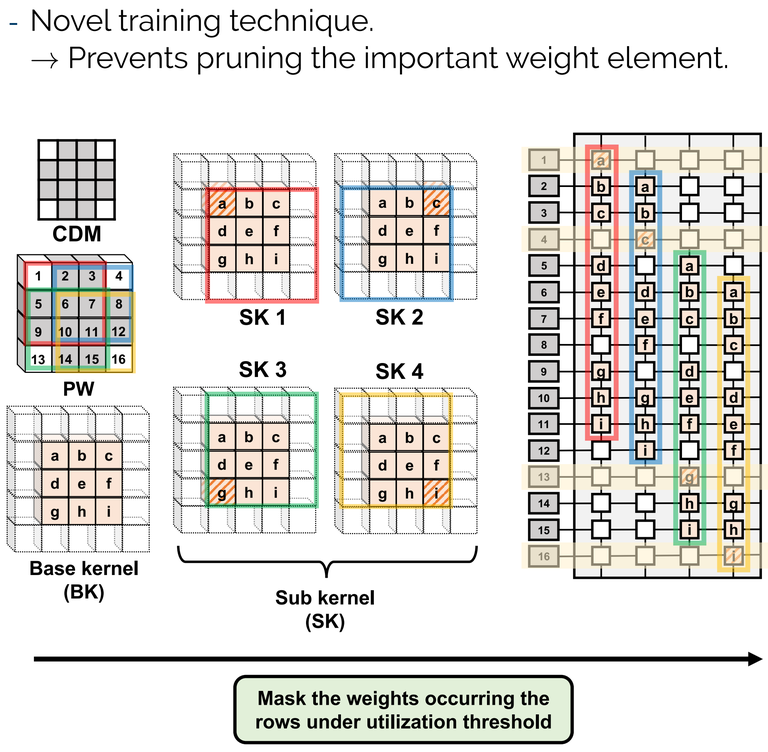

Figure 5. Concept of the proposed KERNTROL.

Experimental Results

- VW-SDK achieves an array utilization up to **73.8%**.

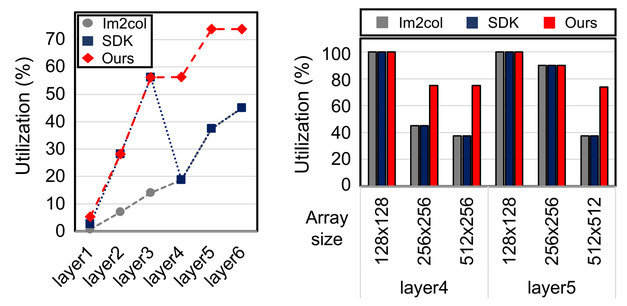

Figure 6. Comparison of the utilization rate in VGG13.

- Compared to other patterns, PAIRS achieves
- up to **4**× higher compression rate.

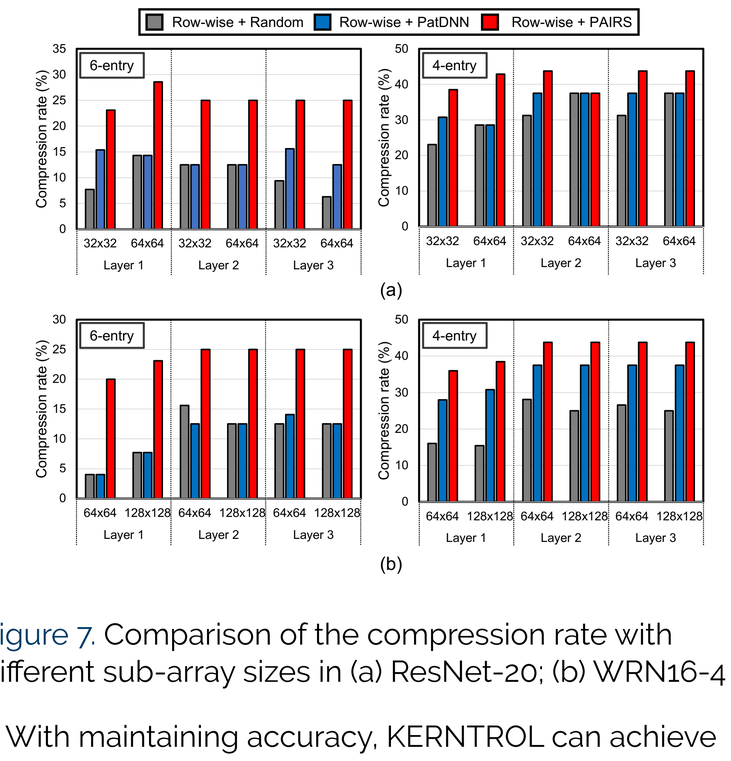

Figure 7. Comparison of the compression rate with different sub-array sizes in (a) ResNet-20; (b) WRN16-4

- With maintaining accuracy, KERNTROL can achieve
- up to **36.4%** improvement in the compression rate.
- up to **38.6%** improvement in the array utilization.

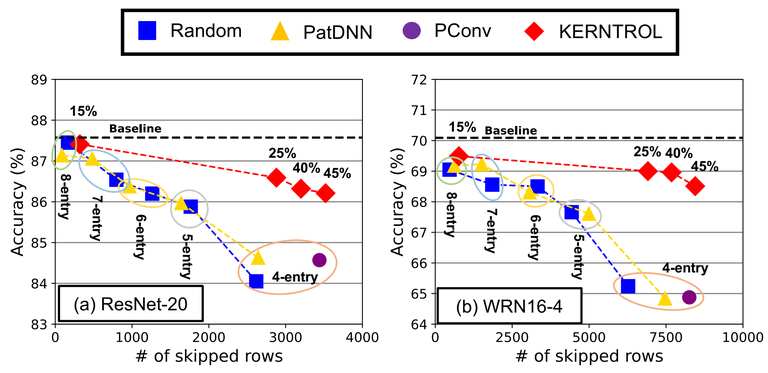

Figure 8. Performance of the proposed KERNTROL.

Conclusion

This research focuses on how to facilitate convolution operation in the IMC array for energy-efficient IMC-based CNN inference. To achieve this, various techniques are proposed to find the optimal mapping method and compress the weight matrix considering the mapping method. The proposed techniques show better hardware performance and inference accuracy compared to previous works.

References

[1] Rhe et al., VW-SDK, DATE 2021.
[2] Rhe et al., VWC-SDK, JETCAS 2022.
[3] Rhe et al., PAIRS, ISLPED 2023.
[4] Rhe et al., KERNTROL, ICCAD 2023.

Author's Information