Towards Memory-Efficient Neural Networks via Multi-Level in situ Generation

Jiaqi Gu, Hanqing Zhu, Chenghao Feng, Mingjie Liu, Zixuan Jiang, Ray T. Chen, David Z. Pan

ECE Department, University of Texas at Austin

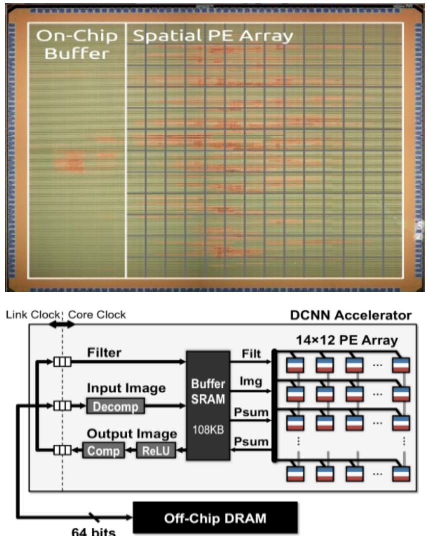
This work is supported in part by AFOSR MURI

jqgu@utexas.edu https://jeremiemelo.github.io

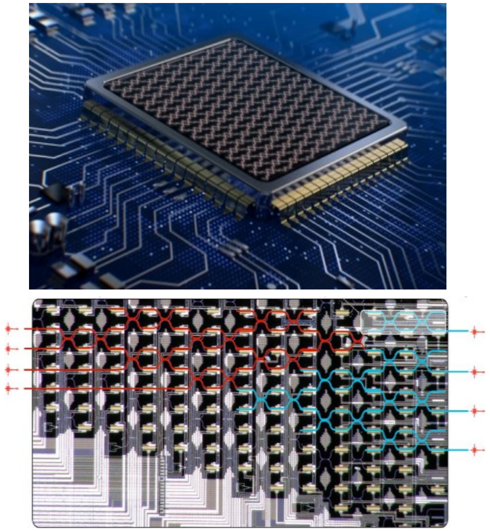
Efficient On-Device AI

- ML models/dataset keep increasing -> more computing capacity/efficiency
  - Low latency
  - Low power
  - High bandwidth

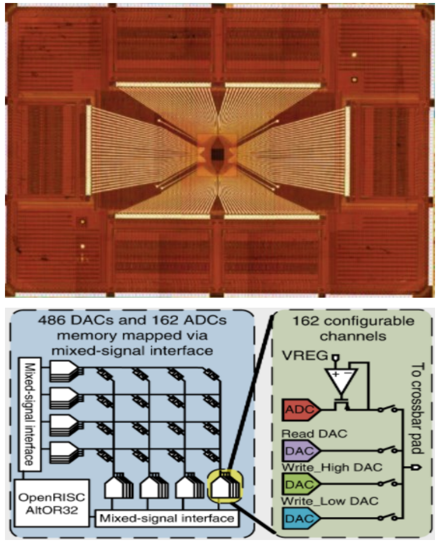
- Extensive efforts on efficient AI solutions



[Eyeriss, *ISSCC'2016*]
Electrical Digital



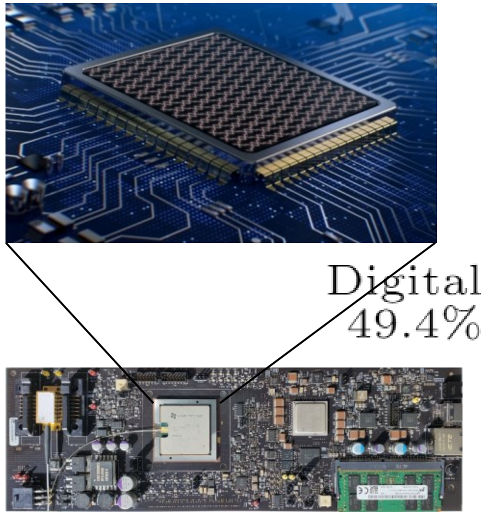
[Mars, *NaturePhotonics'2017*]
Photonics Analog

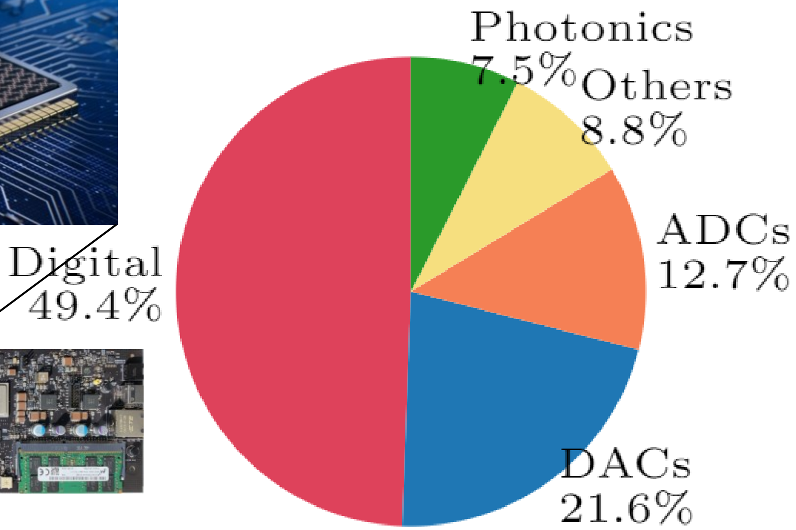


[ReRAM Xbar, *NatureElectronics'2019*]
Electrical Analog

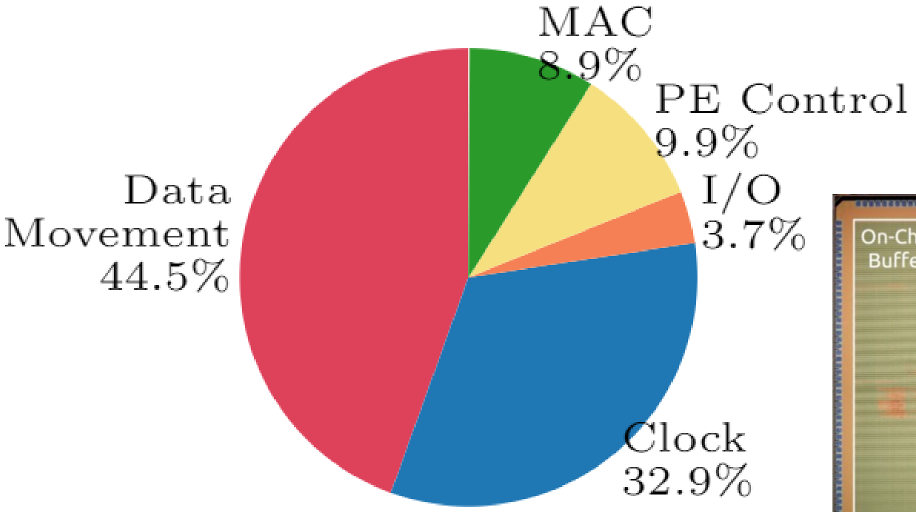
Challenges: Data Movement Bottleneck

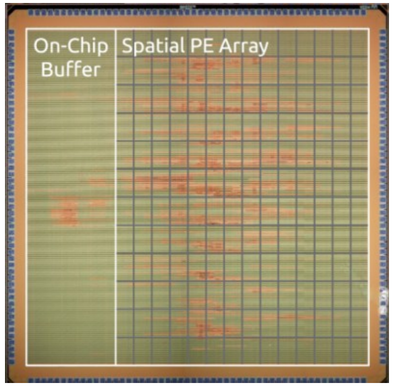
- Power consumption is mainly on memory storage/access
  - SRAM vs. Optical MVM: 50% vs. 8%
- Latency/performance is bottlenecked by data movement
  - SRAM: >10 ns and ~300 GB/s
  - Optical MVM: ~100 ps and ~3×10^6 GLOPS





[Mars, *NaturePhotonics'2020*]
Photonics Analog

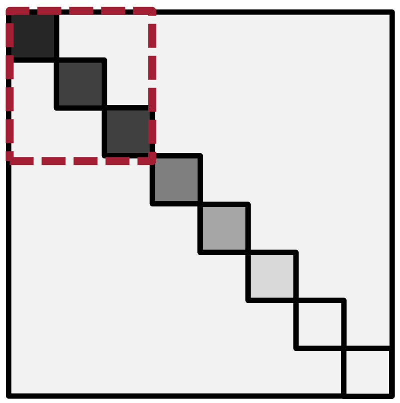


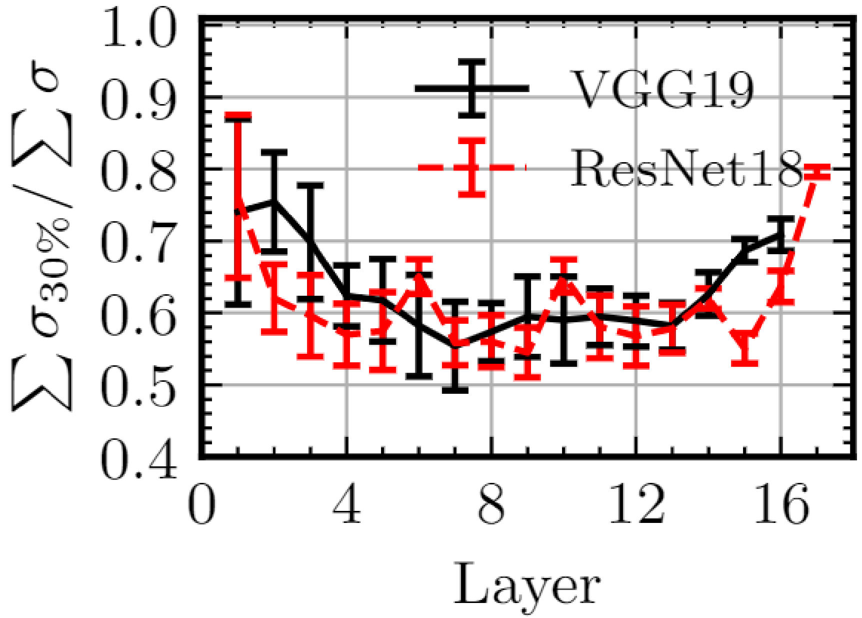


[Eyeriss, *ISSCC'2016*]
Electrical Digital

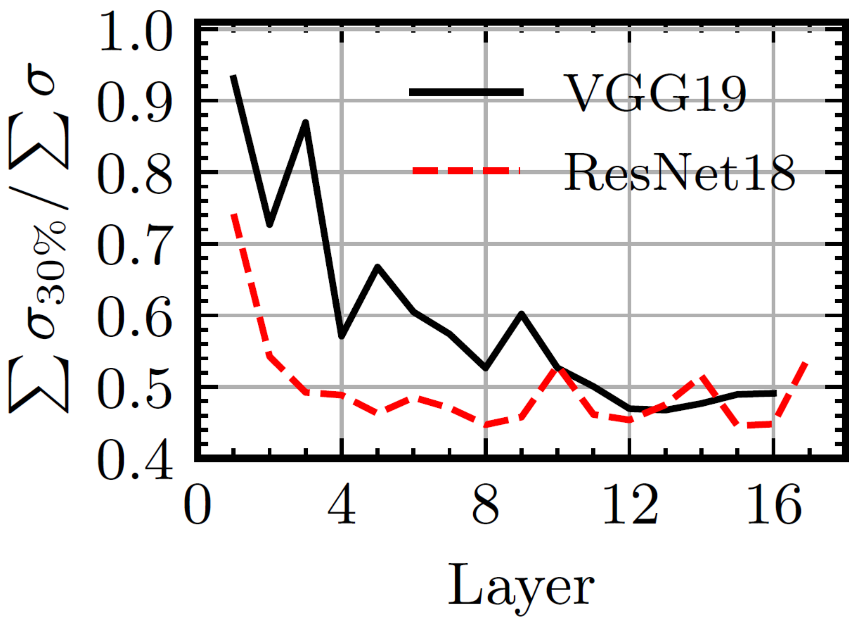
Opportunity: Multi-Level Redundancy in DNNs

- Observation of weight correlation via SVD
  - Intra-kernel correlation
  - Cross-kernel correlation
  - 50-90% total values on 30% top singular values





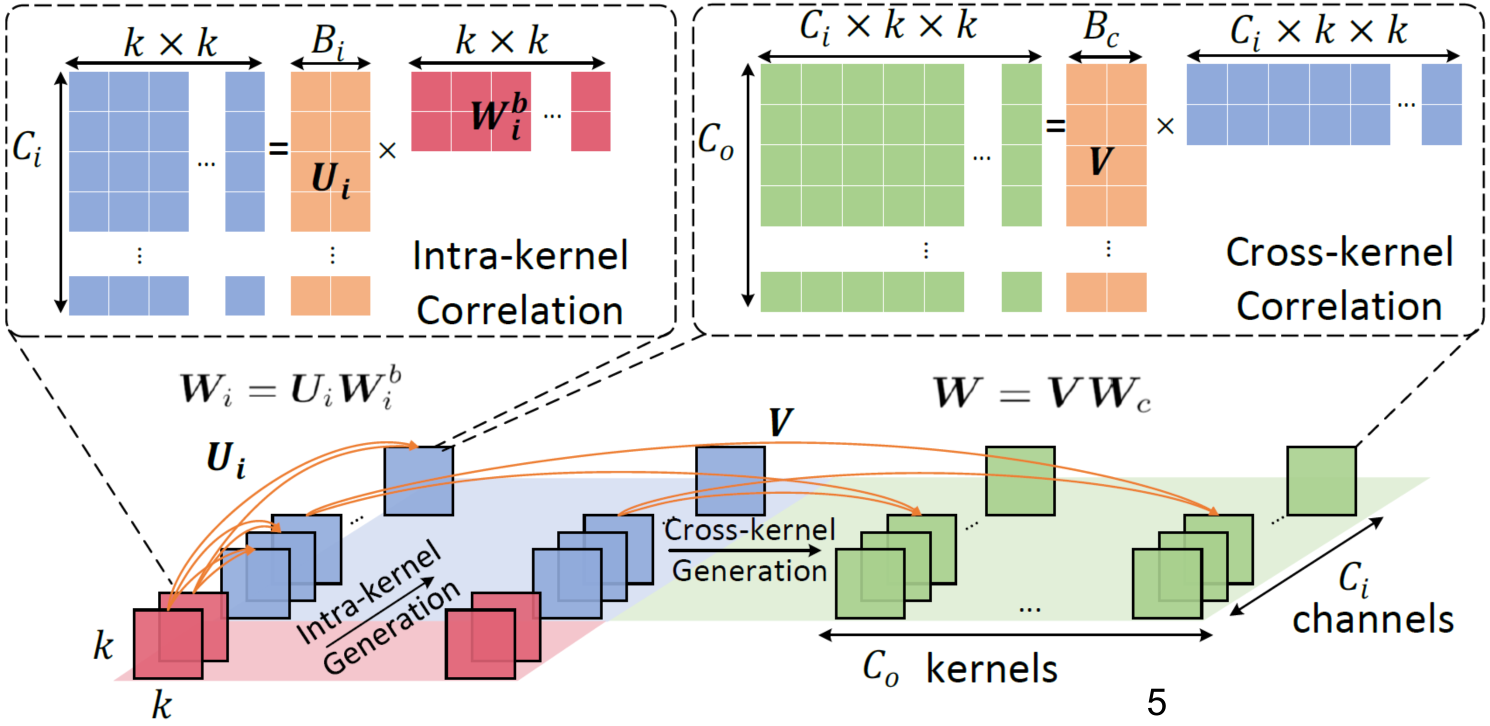
Intra-kernel correlation

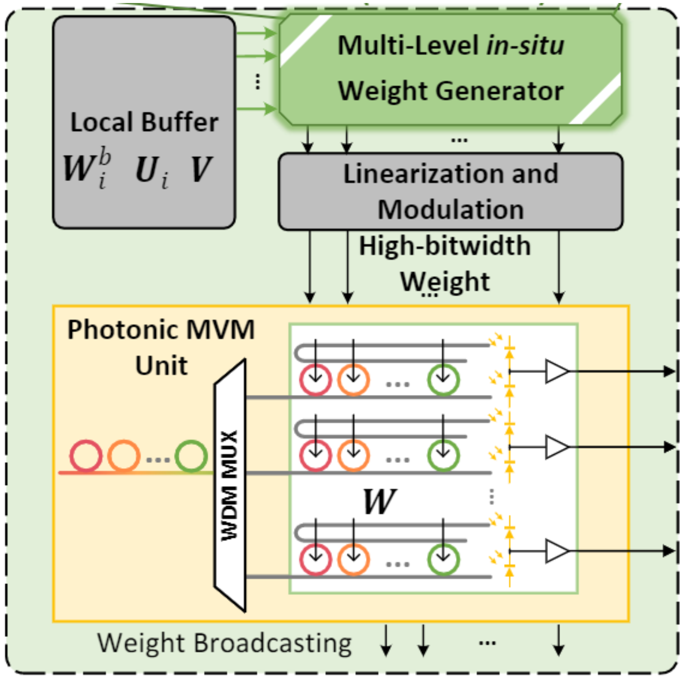


Cross-kernel correlation

Our Method: Multi-Level *in situ* Generation

- Trade expensive data movement for cheap computations
- A unified framework to generalize prior single-level low-rank NNs
- Kernel redundancy: Intra-/cross- kernel generation on-the-fly
- Bit-level redundancy: Precision-preserving mixed-precision basis





Proposed Memory-Efficient ONN Design

Explore Kernel Redundancy

- Intra-kernel generation (B_i)
  - Span all input channels from a small basis $\boldsymbol{W}^b$

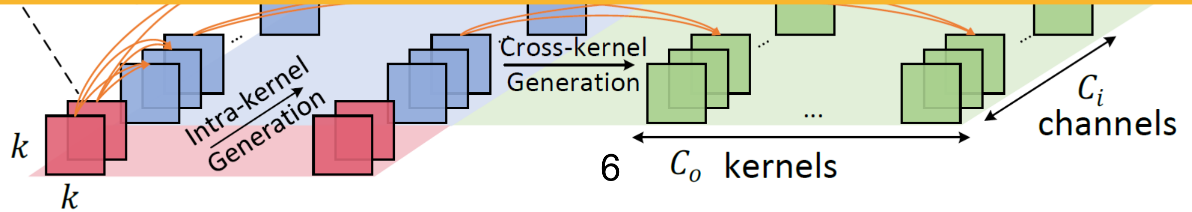
$$\boldsymbol{W}_i = \boldsymbol{U}_i \boldsymbol{W}_i^b, \quad \forall i \in [C_o]$$

- Cross-kernel generation (B_c)
  - Span all kernels from a kernel basis

$$\boldsymbol{W} = \boldsymbol{V} \boldsymbol{W}_c = \boldsymbol{V} \{\boldsymbol{U}_i \boldsymbol{W}_i^b\}_{i \in [B_c]}$$

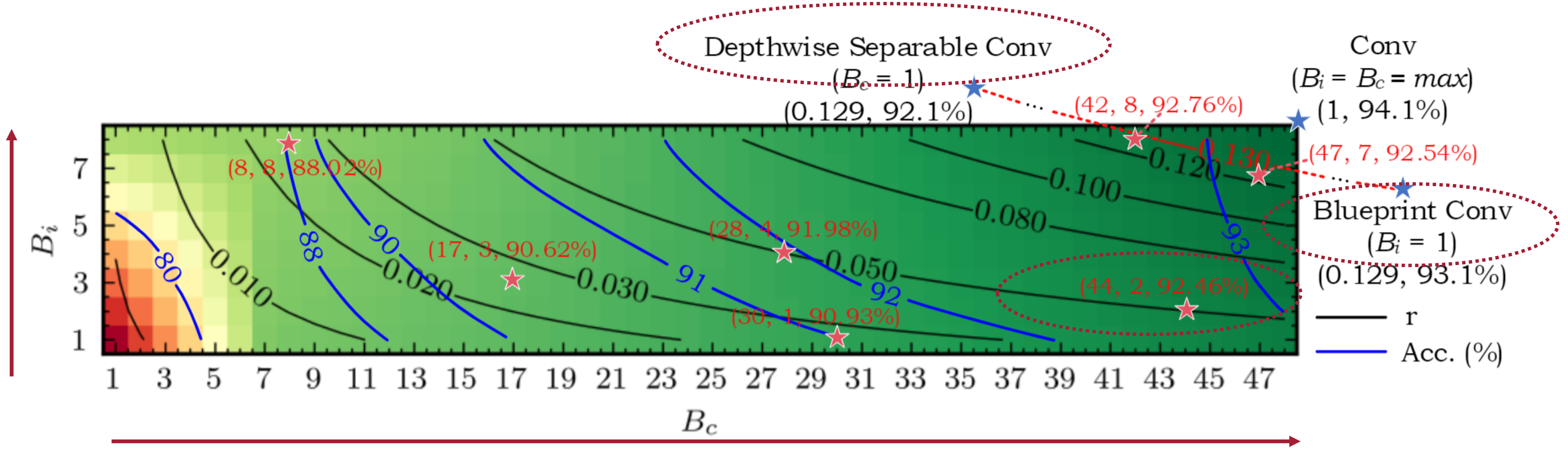
Params. reduction ratio

$$r = \frac{|\boldsymbol{V}| + \sum_{i \in [B_c]} (|\boldsymbol{U}_i| + |\boldsymbol{W}_i^b|)}{|\boldsymbol{W}|} = \frac{(C_o + B_i k^2 + C_i B_i) B_c}{C_o C_i k^2}$$



Performance/Efficiency Contour

- Explore the multi-level generation space
  - Generalize separable CONV [He+, *CVPR*'16] and Blueprint CONV [Haase+, *CVPR*'20]
  - **Small B_i + Medium B_c** → Good efficiency and performance trade-off



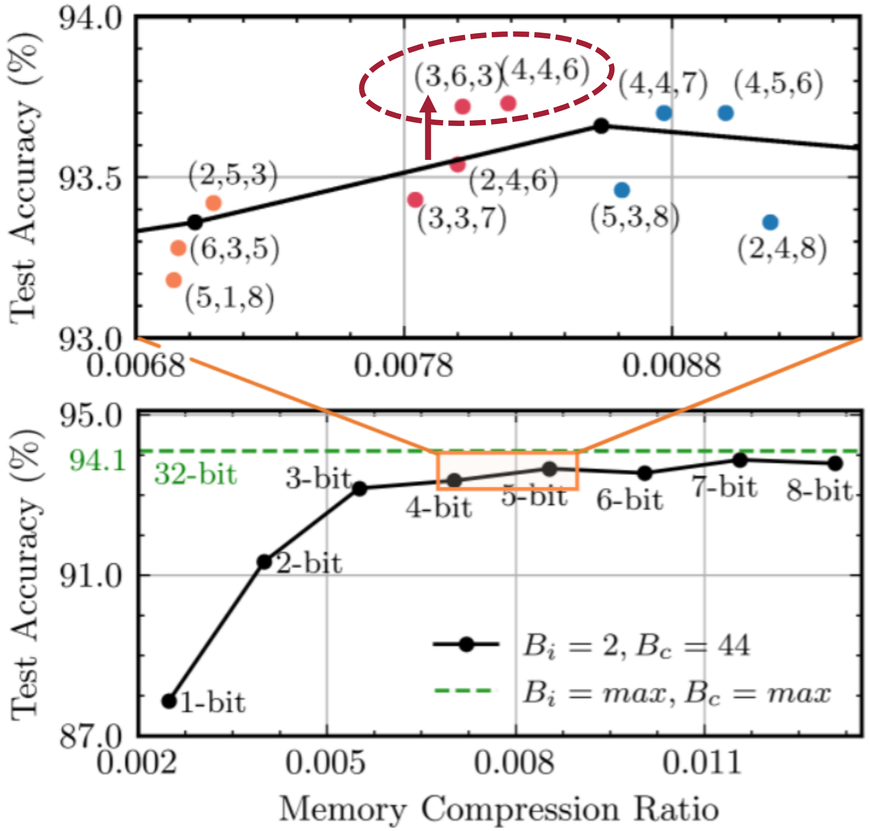
Explore Bit-Level Redundancy

- Augmented mixed-precision kernel generation
- Assign different bits to basis and coefficients

$$(q_b, q_u, q_v) \rightarrow (\boldsymbol{W}_b, \boldsymbol{U}, \boldsymbol{V})$$

- Precision-preserving using analog generators

$$\sup(q_c) = (q_b + q_u + \log_2 B_i)$$
$$\sup(q) = (q_v + \sup(q_c) + \log_2 B_o)$$



Memory compression ratio

$$r_m = \frac{B_c B_i k^2 q_b + B_c C_i B_i q_u + C_o B_c q_v}{C_o C_i k^2 q_w}$$

Effective Training Flow for in-situ Generation

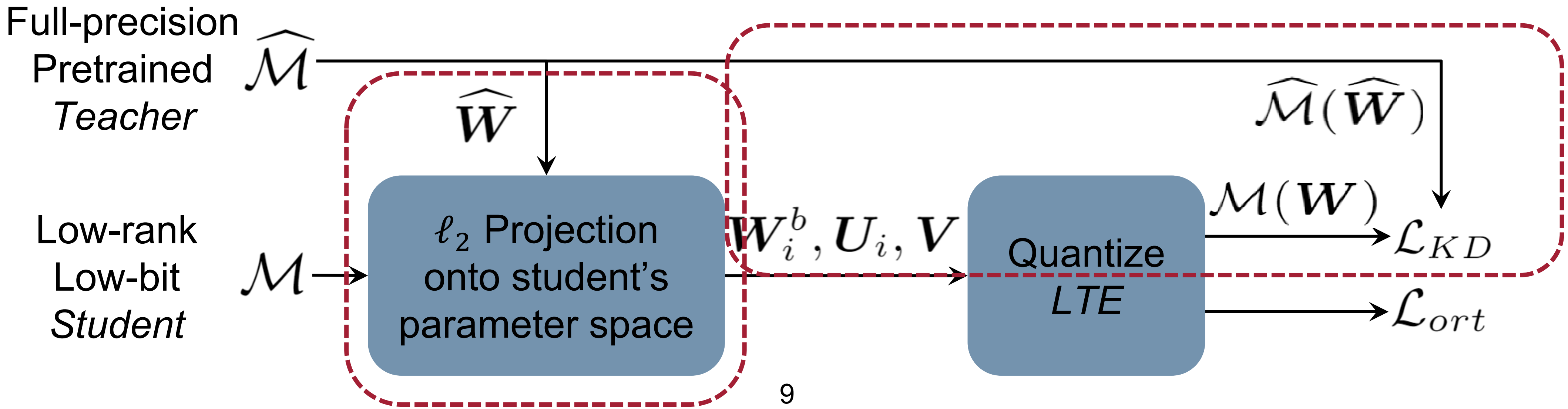
- Project teacher to student parameter space

$$\boldsymbol{W}_i^b, \boldsymbol{U}_i, \boldsymbol{V} \leftarrow \operatorname{argmin} \ \|\widehat{\boldsymbol{W}} - \boldsymbol{V}\{\boldsymbol{U}_i \boldsymbol{W}_i^b\}_{i \in [B_c]}\|_2^2$$

- Quantization-aware knowledge distillation to guide optimization

$$\min \ \mathcal{L}_{KD} = \beta T^2 \mathcal{D}_{KL}(q_T, p_T) + (1-\beta) H(q, p_{T=1})$$

$$\text{s.t. } \ p_T = \frac{\exp(\frac{\mathcal{M}(\boldsymbol{W})}{T})}{\sum \exp(\frac{\mathcal{M}(\boldsymbol{W})}{T})}, q_T = \frac{\exp(\frac{\widehat{\mathcal{M}}(\widehat{\boldsymbol{W}})}{T})}{\sum \exp(\frac{\widehat{\mathcal{M}}(\widehat{\boldsymbol{W}})}{T})},$$

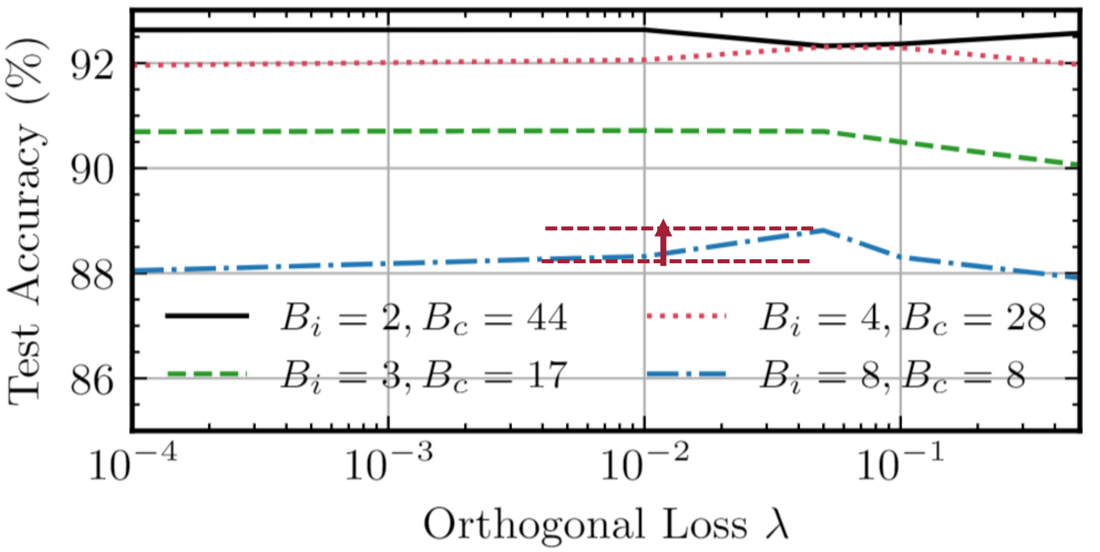


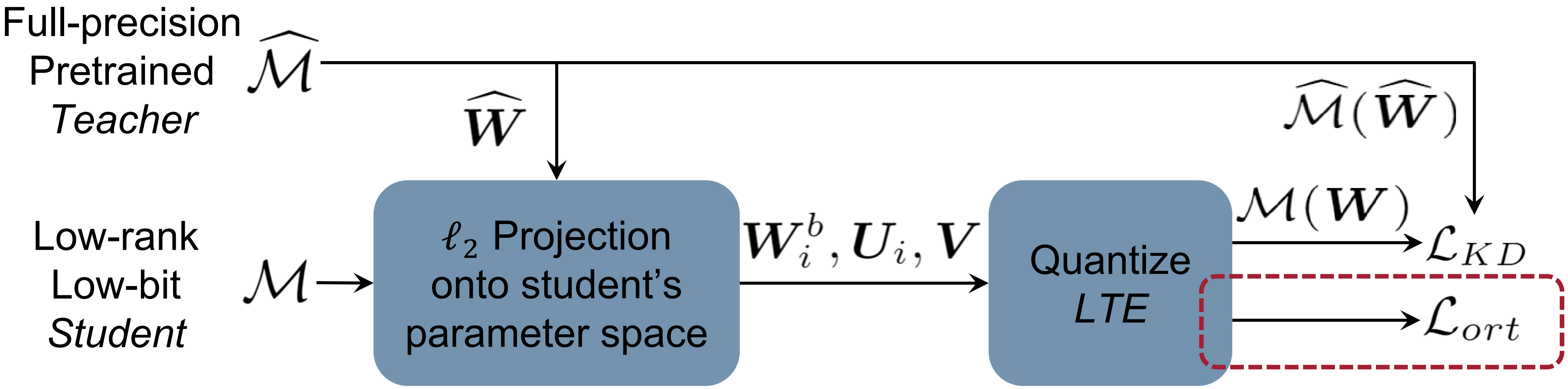
Effective Training Flow for in-situ Generation

- Encourage the rank of spanned kernel
- Multi-level orthonormality regularization

$$\sum_{i=1}^{B_c}\left(\|\boldsymbol{W}_i^b(\boldsymbol{W}_i^b)^T-\boldsymbol{I}\|_2^2+\|\tilde{\boldsymbol{U}}_i^T\tilde{\boldsymbol{U}}-\boldsymbol{I}\|_2^2\right)+\|\tilde{\boldsymbol{V}}^T\tilde{\boldsymbol{V}}-\boldsymbol{I}\|_2^2,$$

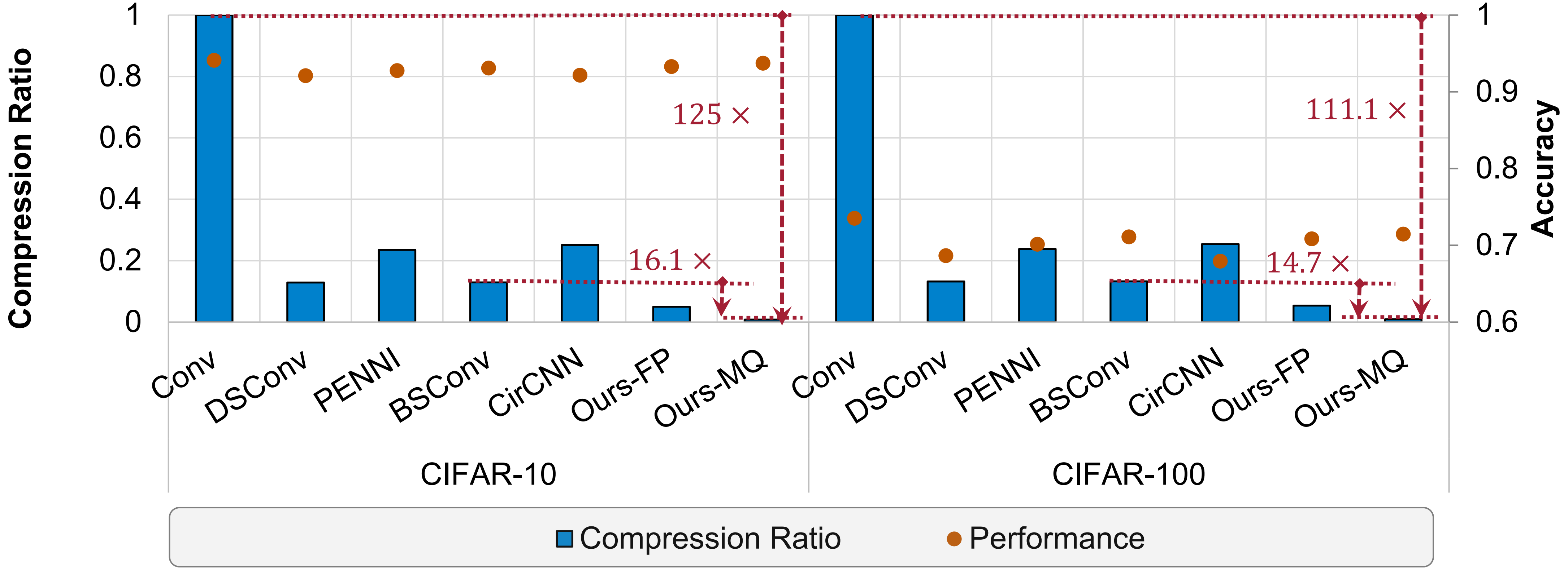
$$\tilde{\boldsymbol{U}}_i=\left(\frac{u_0}{\|u_0\|_2^2}\cdots\frac{u_0}{\|u_{B_i-1}\|_2^2}\right),\tilde{\boldsymbol{V}}=\left(\frac{v_0}{\|v_0\|_2^2}\cdots\frac{v_0}{\|v_{B_c-1}\|_2^2}\right)$$





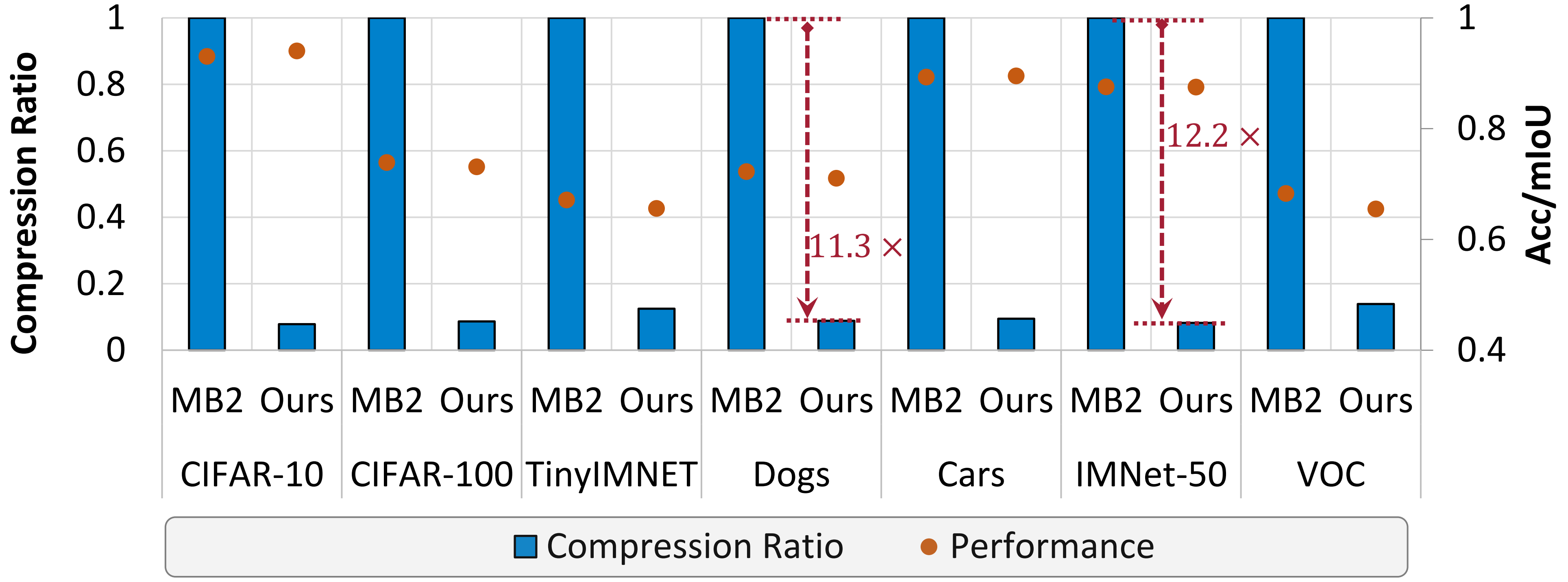
Experimental Results

- ResNet-18 + CIFAR10/100
  - **>100×** memory cost reduction on baseline architecture
  - **~15×** more efficient and **0.5%** higher accuracy than best baselines



Experimental Results

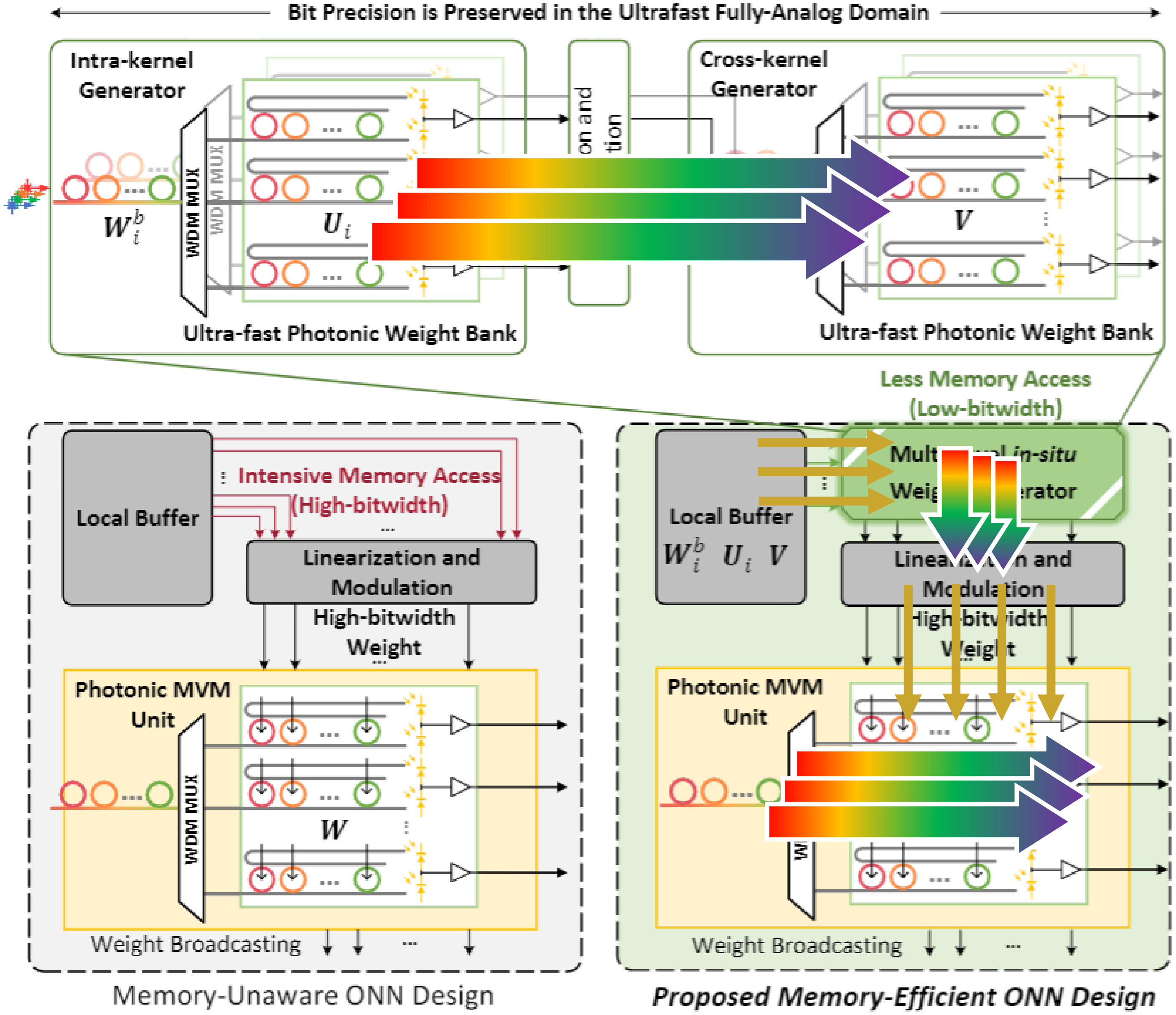
- MobileNetV2 + Various vision benchmarks
  - **>10×** memory cost reduction on *compact* architecture
  - Marginal performance drop on classification and object detection



Photonics NN Simulation

- Photonic MVM cores
- Ultra-fast optical weight generator

- ResNet-18/ImageNet
- Latency (**27.2%** ↓)
  - 56.46 ms → **41.11** ms
- Energy (**85.7%** ↓)
  - 25.77 mJ → **3.69** mJ
- Energy-delay product (**9.6**× ↓)



Conclusion and Future Work

- A unified multi-level in-situ generation framework for memory-efficient NNs
- **10~100× compression**: channel-level + kernel-level + bit-level
- **<1% performance drop**: projection + distillation + ortho regularization
- **~10× energy-delay reduction**: ultra-fast generator on emerging accelerators

- Future work
  - Automatic search of per-layer cardinality and mixed precision settings
  - Sparsity exploration on the in-situ generation

Thank you

jqgu@utexas.edu https://jeremiemelo.github.io