AUTOHET: An Automated Heterogeneous ReRAM-Based Accelerator for DNN Inference

__Tong Wu__, Shuibing He, Jianxin Zhu, Weijian Chen, Siling Yang, Ping Chen, Yanlong Yin, Xuechen Zhang[#], Xian-He Sun[$], Gang Chen

浙江大学 Zhejiang University

Washington State University Vancouver

$ Illinois Tech

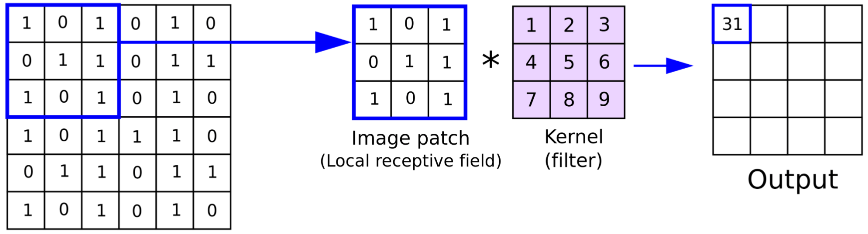
Background

Von Neumann Architecture

- Separate computing and storage units

Massive data movement

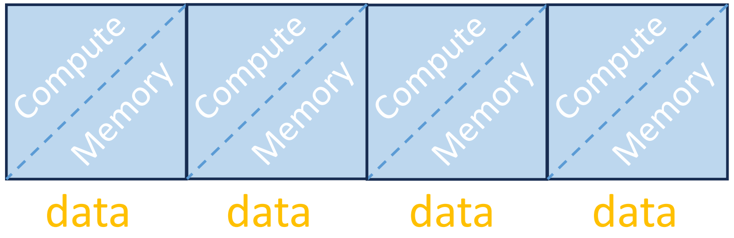
- DNN inference includes massive MVMs



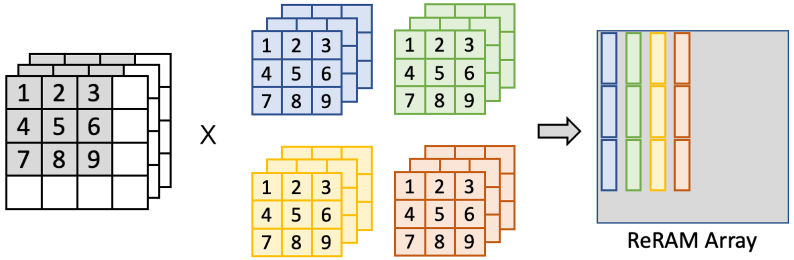
High latency and high energy

Processing in Memory (PIM)

- In-situ computing



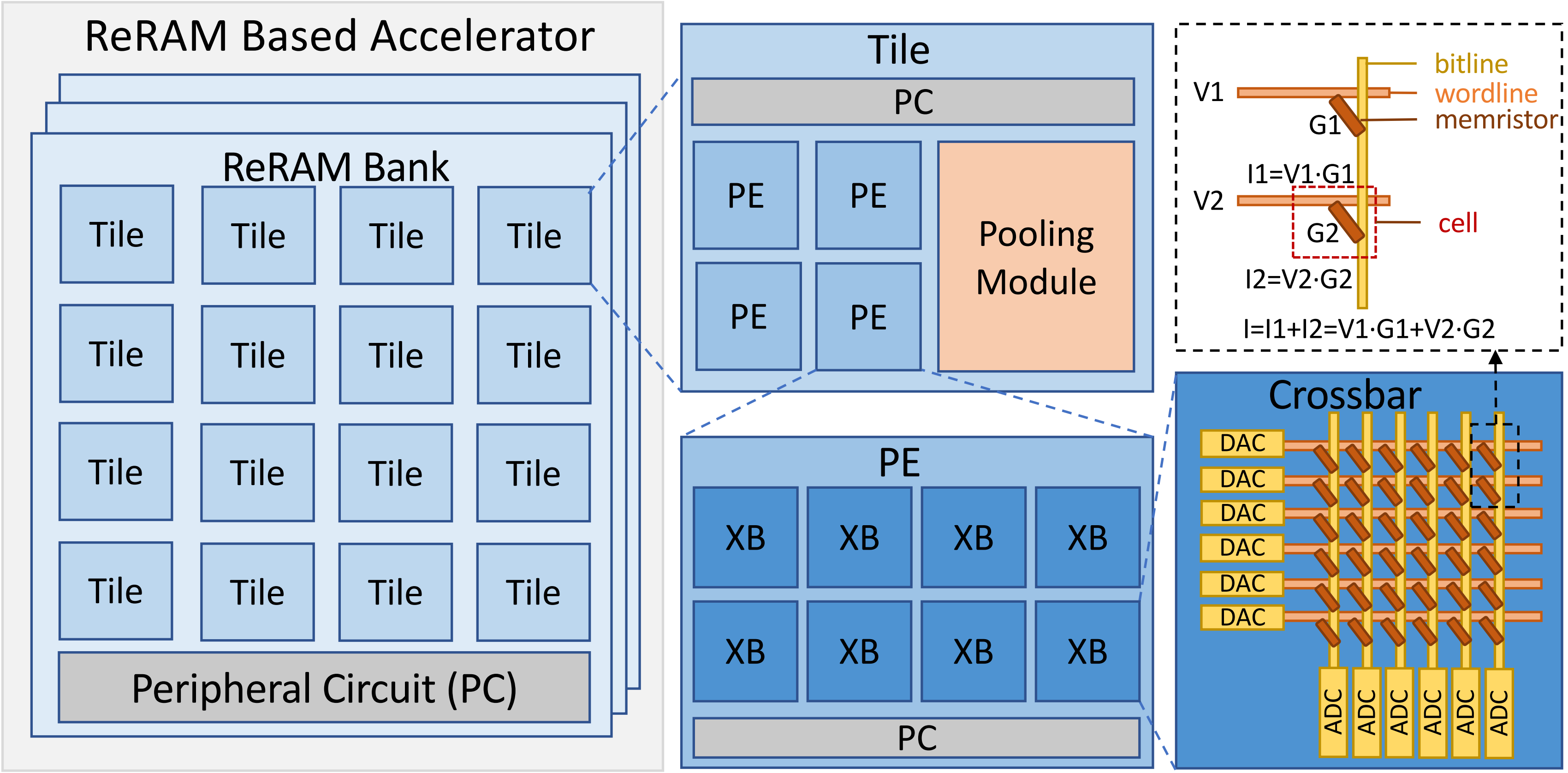
- High-parallel MVMs



Low energy and low latency

Background

- **Hierarchical topology**
- **In-situ computing**
- **Homogeneous crossbars**
- **Tile-based allocation**



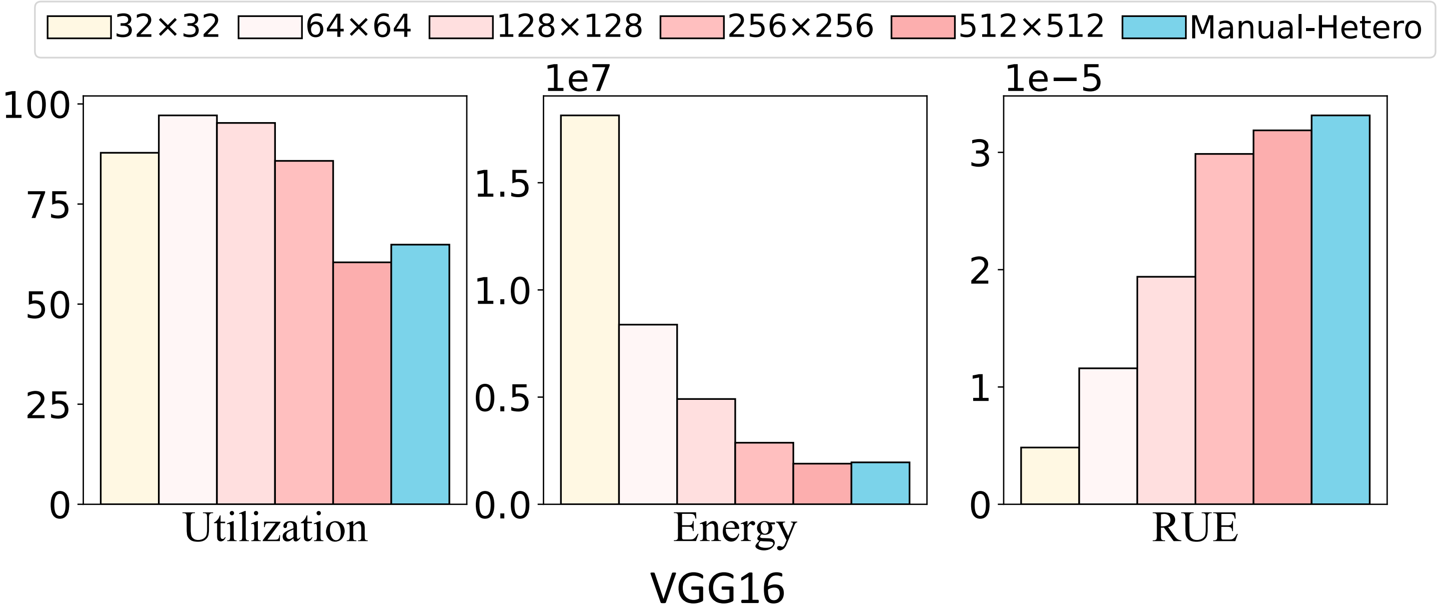
Crossbars and peripheral circuits(PCs) cooperate to perform DNN inference together.

[1] S. Mittal, "A Survey of ReRAM-based Architectures for Processing-in-memory and Neural Networks," Machine learning and Knowledge extraction, vol. 1, no. 1, pp. 75–114, 2019.
[2] Lixue Xia, Boxun Li, MNSIM: Simulation Platform for Memristor-based Neuromorphic Computing System, in IEEE TCAD, vol.37, No.5, 2018, pp.1009-1022.

Motivation

The homogeneous crossbar architecture causes sub-optimal resource utilization or energy efficiency.

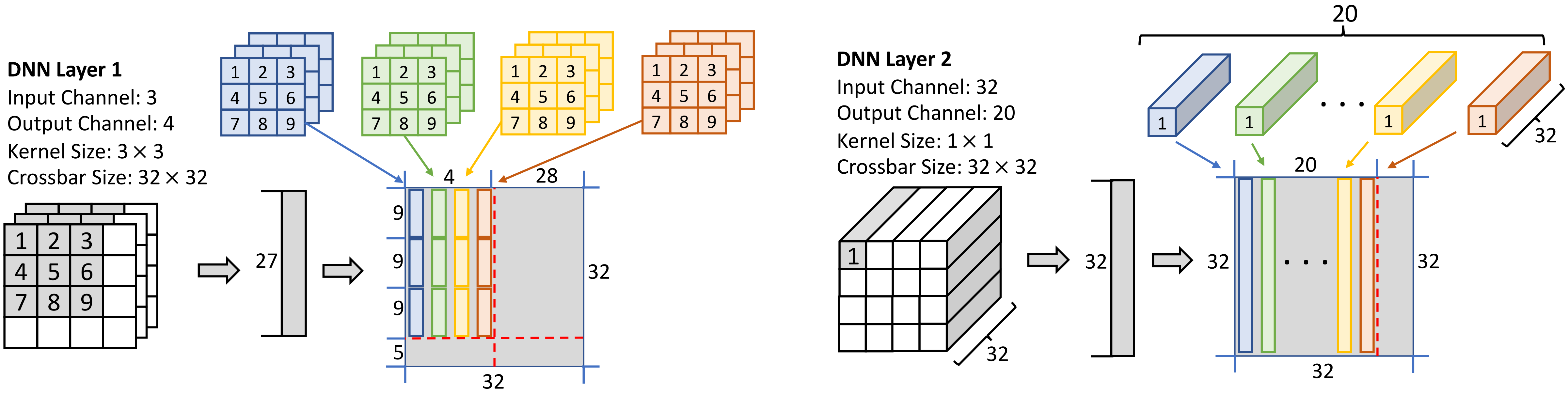
RUE=U/E



1. Homogeneous crossbars cannot cope with different DNN layers.
2. Square crossbars are not well-matched the most common kernels.

Motivation

DNN layer mapping and crossbar allocation



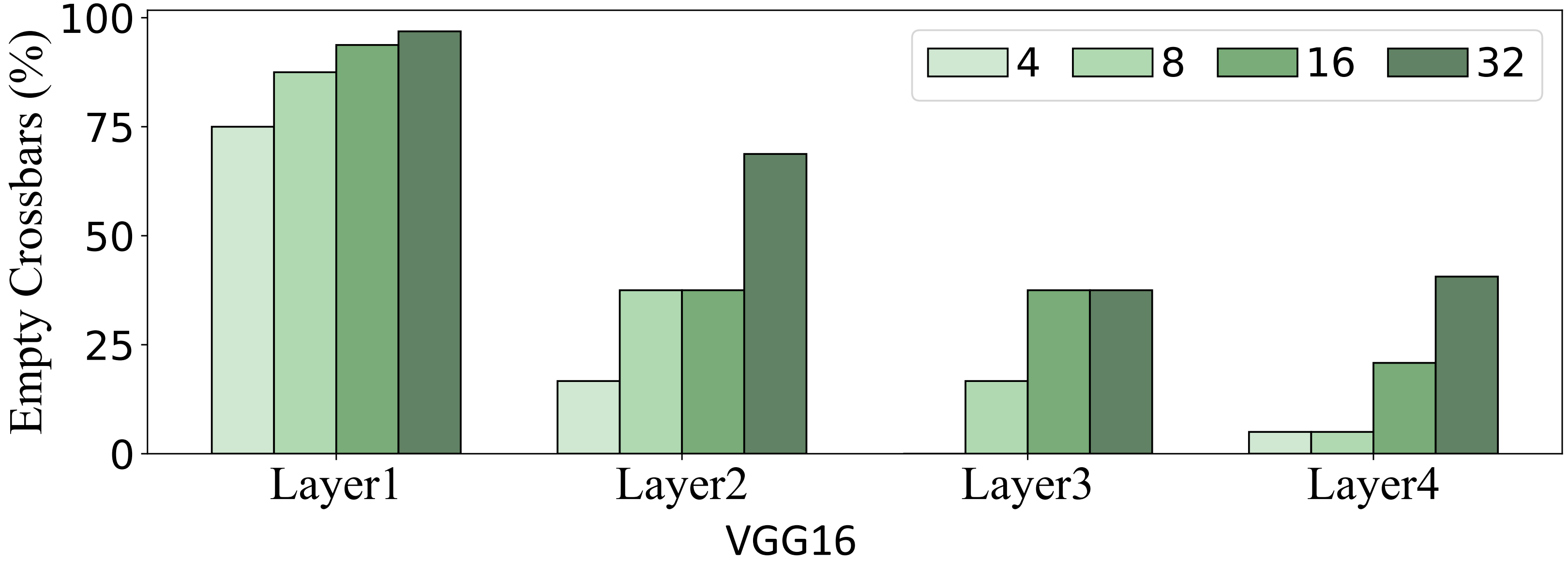
(a) Layer 1

(b) Layer 2

1. Homogeneous crossbars cannot cope with different DNN layers.
2. Square crossbars are not well-matched the most common kernels.

Motivation

☹ **The tile-based allocation scheme compromises resource utilization.**



3. Crossbar wastage in tiles reduces the utilization of crossbars, thereby decreasing the RUE value.

Goals and Challenges

- **Our goal:** Using heterogeneous crossbars for DNN inference to achieve higher RUE.

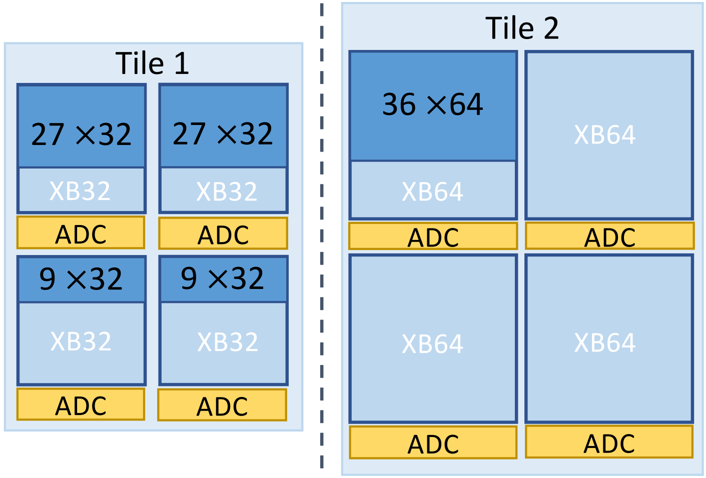
1. Homogeneous crossbars	1. Heterogeneous crossbars
2. Square crossbars	2. Square + Rectangular
3. Tile-based allocation	3. Tile-shared allocation

Goals and Challenges

- **Challenge1: The conflict between utilization and energy efficiency.**

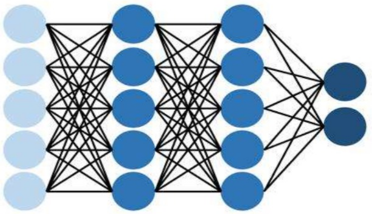
DNN Layer
Input Channel: 4
Output Channel: 64
Kernel Size: 3 × 3

DNN Layer	XB32	XB64
Utilization	**9/16**	9/64
ADC	128	**64**



- **Challenge2: The search space for determining crossbar sizes can be vast.**

N DNN layers



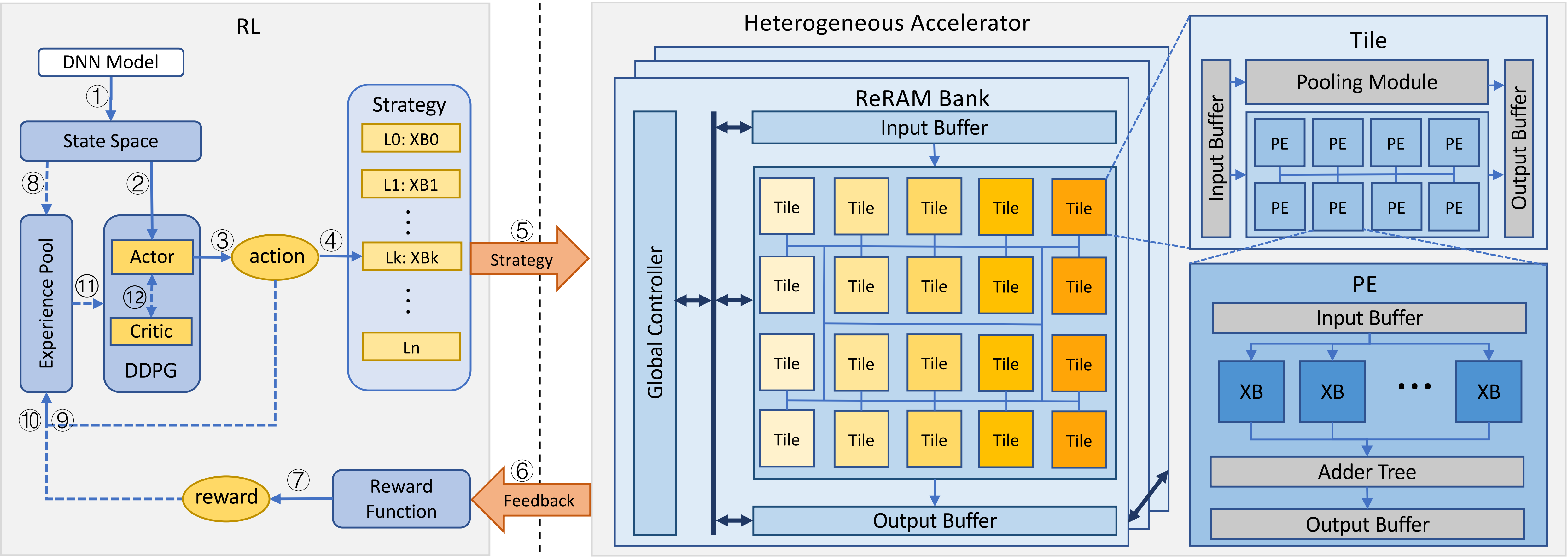
+

C crossbar candidates

Search space: C^N

Design: AUTOHET

AUTOHET: an automated heterogeneous ReRAM-based accelerator for DNN inference.

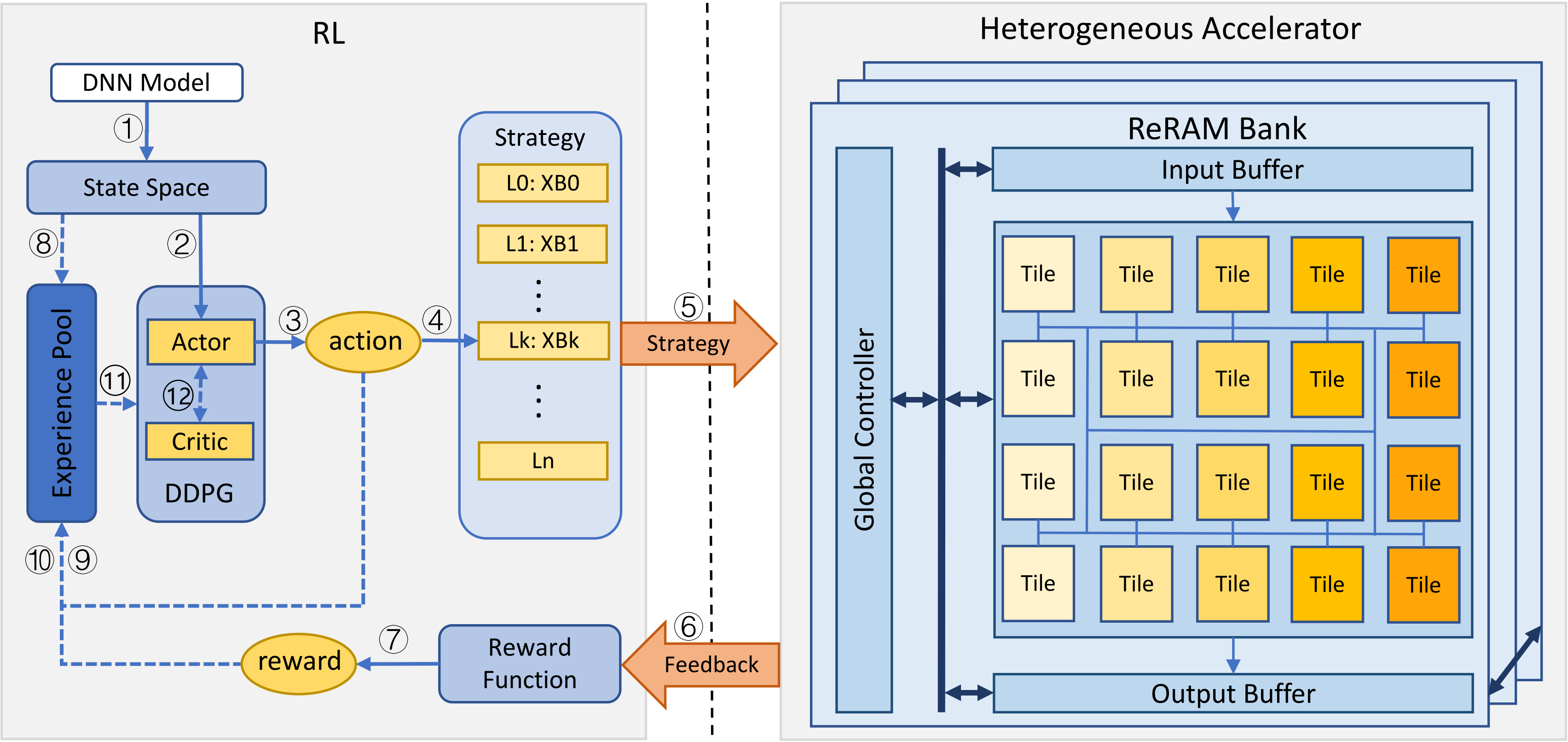


Design 1: RL-based Decision Scheme

Reinforcement learning (RL):

- **A well-established ML method for NAS**
- **Deep deterministic policy gradient (DDPG) algorithm**
- **Latency feedback**

Work flow:



[1] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. 2014. Deterministic Policy Gradient Algorithms. In Proceedings of the 31st International Conference on International Conference on Machine Learning (ICML). 387--395.

Design 1: RL-based Decision Scheme

- **Action space:**

a_k **-> Crossbar types**

- **Reward function:**

$$R = \frac{u}{e}$$

- **Experience pool:**

$$E_k = (S_k, S_{k+1}, a_k, R)$$

- **State space:**

$$S_k = (k, t, inc, outc, ks, s, w, ins, a_k, u_k)$$

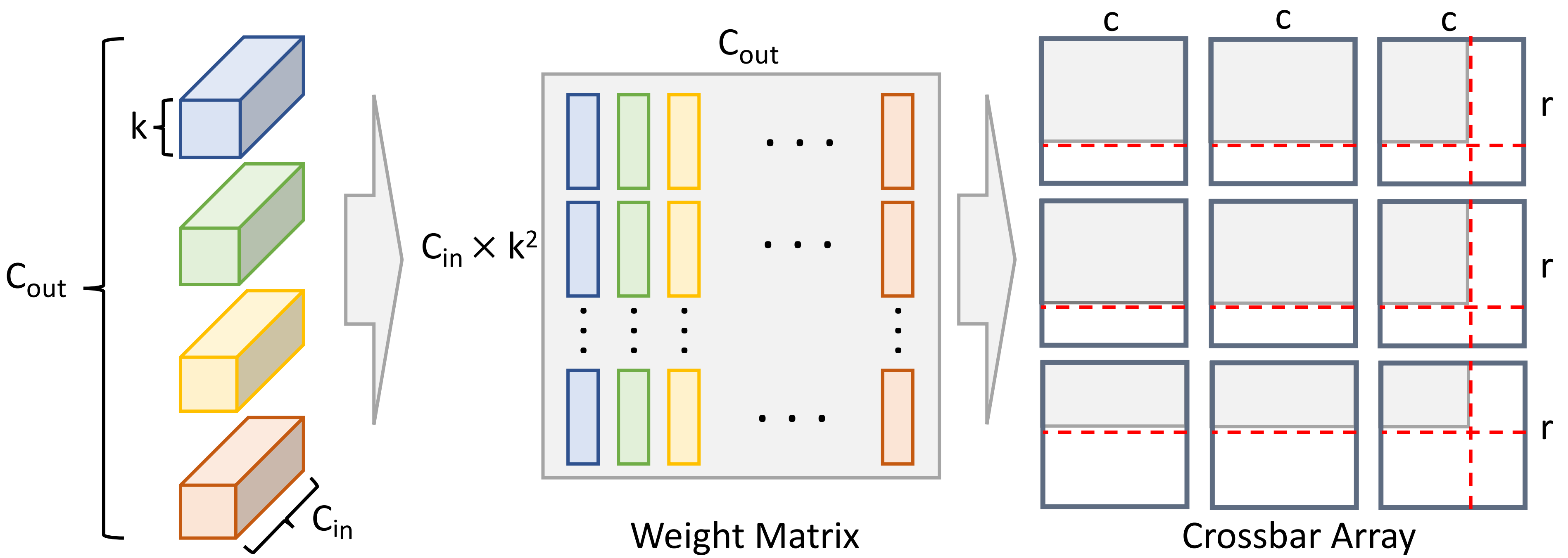
Table 1: Symbols used in the RL state space.

No.	Symbol	Meaning
1	k	layer index
2	t	layer type: CONV:1 ; FC: 0
3	inc	number of channels in the input feature map
4	$outc$	number of channels produced by the CONV
5	ks	number of elements of a convolution kernel
6	s	stride of the convolution
7	w	number of weights in layer k
8	ins	size of the input feature map
9	a_k	action of layer k
10	u_k	crossbar utilization of layer k

Design 2: Heterogeneous Crossbar Size Selection

- **Utilization equation:**

$$u = \frac{C_{\text{in}} \times k^2 \times C_{\text{out}}}{r \times \left\lceil C_{\text{in}} / \left\lfloor r/k^2 \right\rfloor \right\rceil \times c \times \left\lceil C_{\text{out}} / c \right\rceil}$$



Design 2: Heterogeneous Crossbar Size Selection

$$u = \frac{C_{\text{in}} \times k^2 \times C_{\text{out}}}{r \times \left\lceil C_{\text{in}} / \boxed{\left\lfloor r/k^2 \right\rfloor} \right\rceil \times c \times \boxed{\left\lceil C_{\text{out}} / c \right\rceil}}$$

r needs to be multiples of k^2

c needs to be divisible by C_{out}

Table 2: The structure of three popular DNN models.

Network	Structure	
AlexNet	C3-64, C3-192, C3-384, 2C3-256, F4096, F4096, F10	62%
VGG16	2C3-64 , 2C3-128, 3C3-256, 6C3-512,F4096, F1000, F10	81%
ResNet152	C7-64, 3C1-64, 8C1-128, 40C1-256, 12C1-512, 37C1-1024, 4C1-2048, 3C3-64, 8C3-128, 36C3-256, 3C3-512, F1000	32%

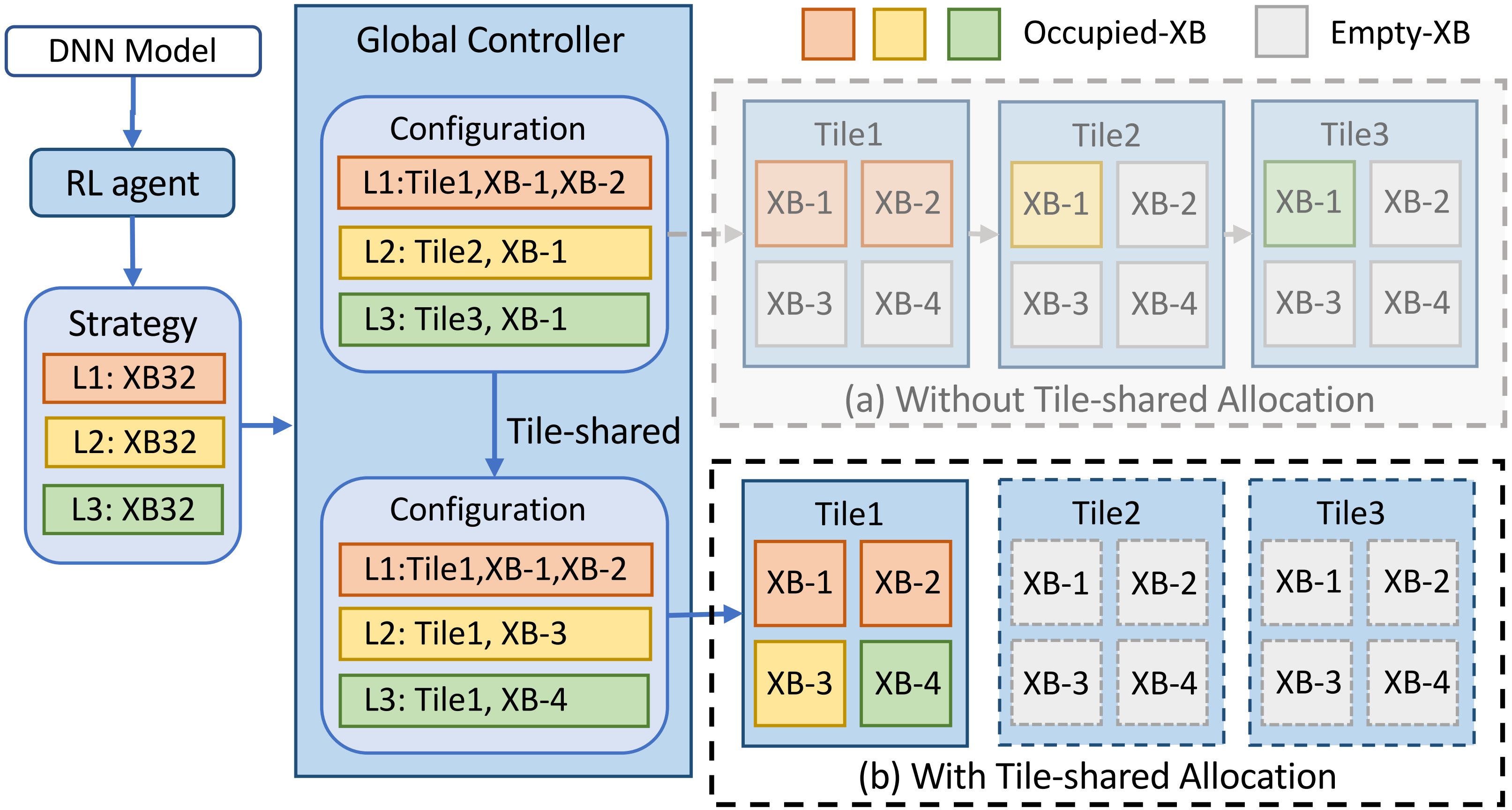
For CONV layers: k = 3, C_{out} = 2^n → **Rectangular crossbars**

For FC layers: k = 1 , C_{out} = 2^n → **Square crossbars**

Hybrid crossbars: 32 × 32, 36 × 32, 72 × 64, 288 × 256, 576 × 512

Design 3: Tile-shared Crossbar Allocation Scheme

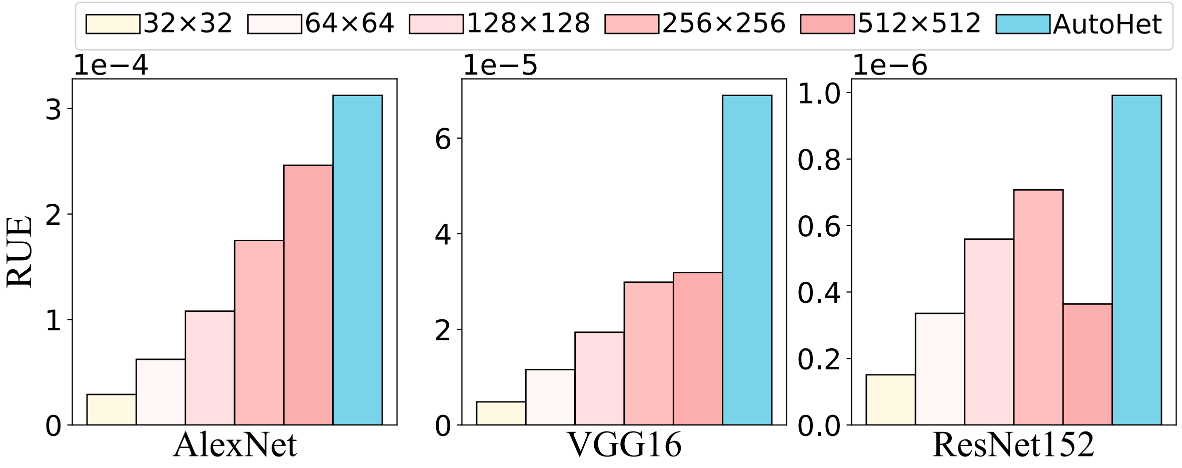
Key idea: allowing multiple DNN layers to be mapped onto the same tile, thereby reducing the number of empty crossbars by **tile sharing**.



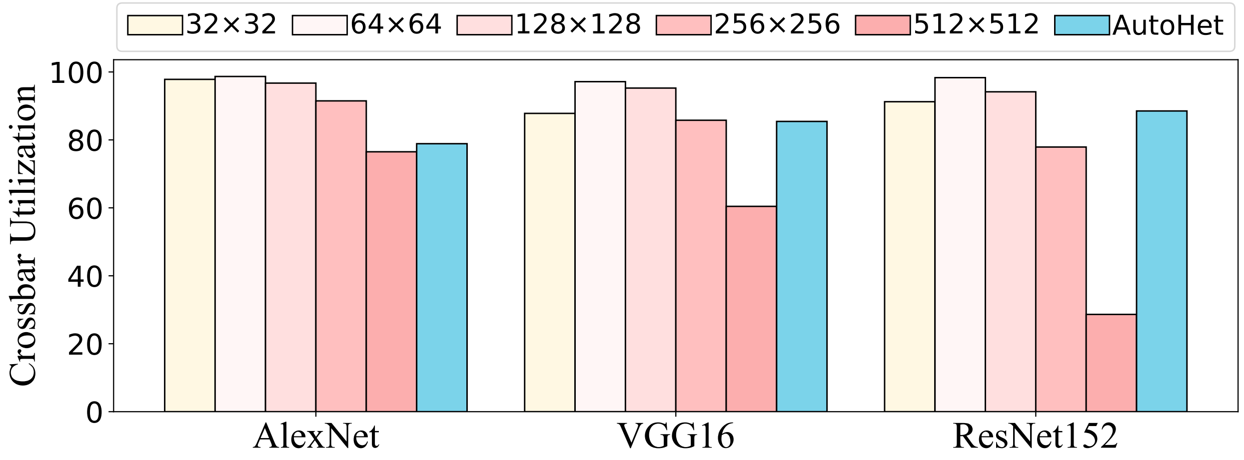
Experiments

Experimental Setup		
Experimental Platform	Software	Ubuntu+anaconda_envs(Pytorch)
	Hardware	MNSIM_Python(simulator)
Models and Datasets	DNN Model	AlexNet, VGG16, ResNet152
	Dataset	MNIST, CIFAR10, ImageNet
Baselines	Homogeneous	32*32, 64*64, 128*128, 256*256, 512*512
	AUTOHET	32*32, 36*32, 72*64, 288*256, 576*512

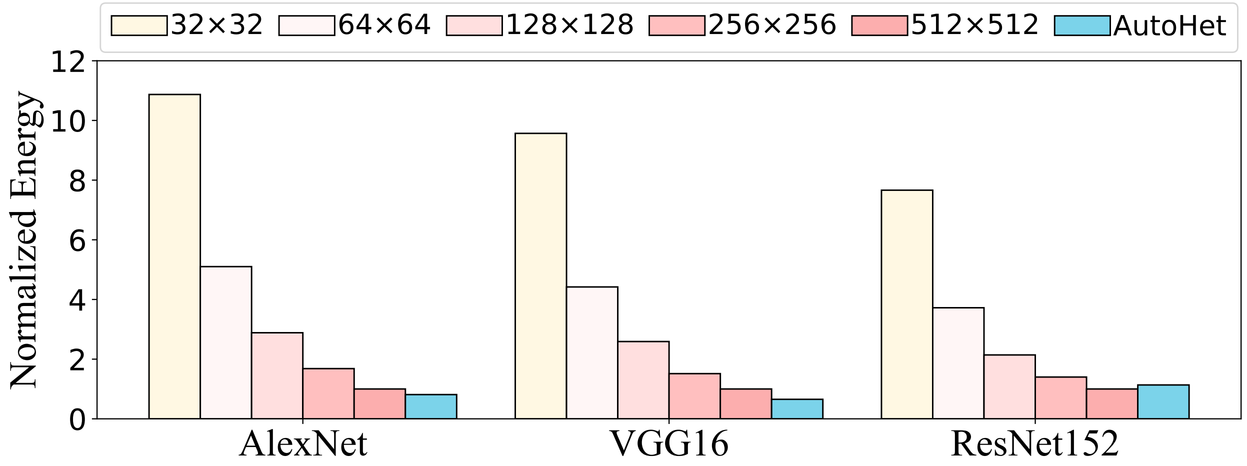
Experiments: Overall Performance



(a) RUE



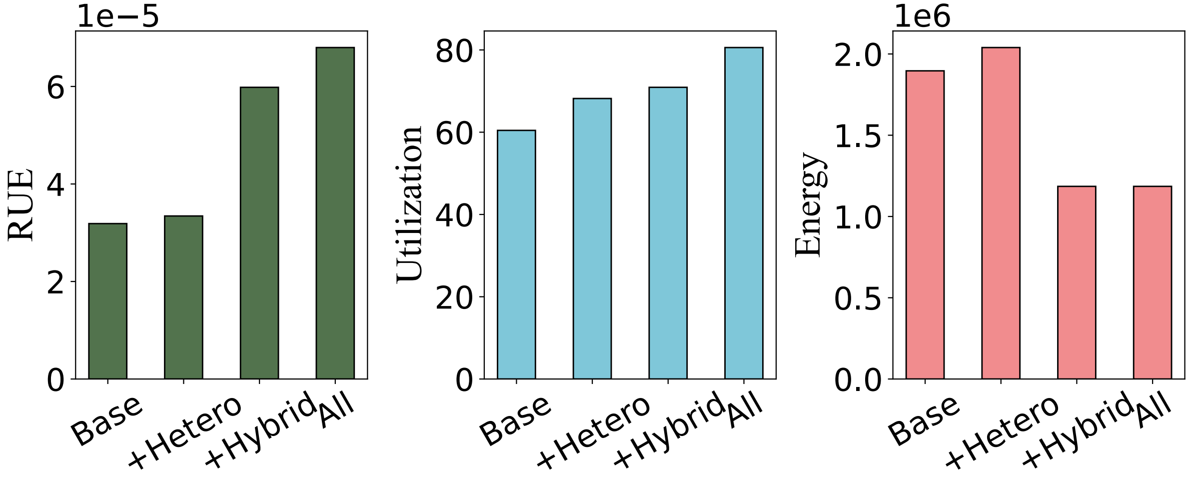
(b) Crossbar Utilization



(c) Energy Consumption

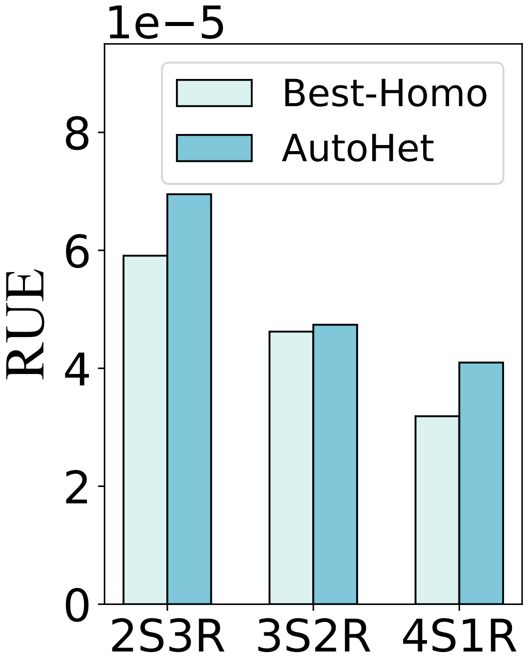
Experiments: Impact of Individual Techniques

Base: the best homogeneous SXB
+Hetero: RL+SXBs
+Hybrid: RL+SXBs+RXBs
All: RL+SXBs+RXBs+tile_shared

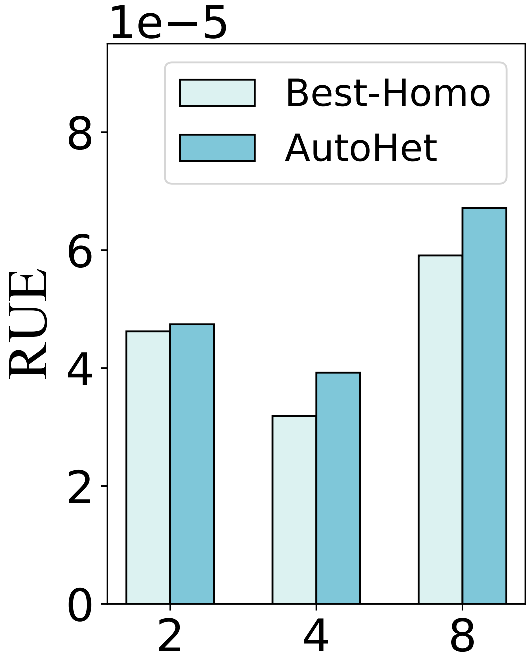


VGG16

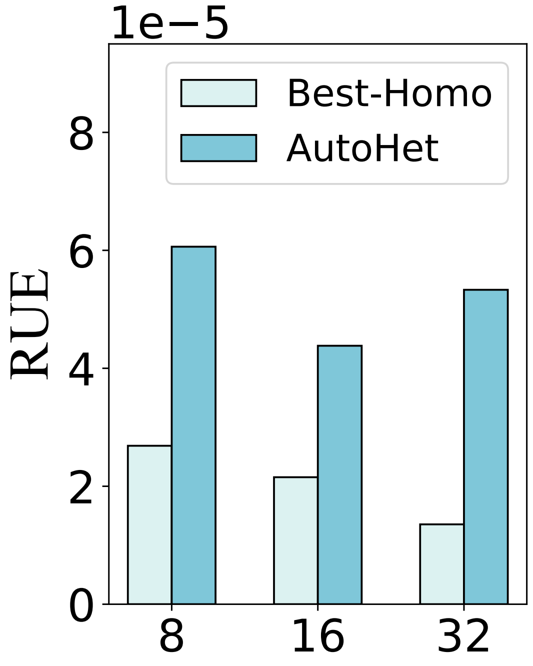
Experiments: Sensitivity Analysis



Various ratios of SXBs and RXBs



Various numbers of crossbar candidates



Various numbers of Pes in each tile

Conclusion

$\Rightarrow$ We propose **AUTOHET,** an automated heterogeneous ReRAM-based accelerator for DNN inference.

$\Rightarrow$ We introduce **hybrid crossbar shapes** (i.e., square and rectangle crossbars) to enhance the matching between the weight matrices and crossbars.

$\Rightarrow$ We propose the **tile-shared allocation scheme** to improve crossbar utilization further.

$\Rightarrow$ Experimental results demonstrate that AUTOHET effectively improves the crossbar utilization by up to **3.1×** while reducing the energy consumption by up to **94.6%**.

AutoHet: An Automated Heterogeneous ReRAM-Based Accelerator for DNN Inference

Thanks for your attention!

Contact me at: wu.tong@zju.edu.cn

浙江大学
Zhejiang University

Washington State University
Vancouver

Illinois Tech