SCIENCE CHINA
Information Sciences

• **Supplementary File** •

CMN: a co-designed neural architecture search for efficient computing-in-memory-based mixture-of-experts

Shihao HAN[1,†], Sishuo LIU[1,†], Shucheng DU[1,2,†], Mingzi LI[1,2,†], Zijian YE[1], Xiaoxin XU[3,4,5], Yi LI[1,2,3,4,*], Zhongrui WANG[1,2,6,*] & Dashan SHANG[3,4,5,*]

[1]*Department of Electrical and Electronic Engineering, the University of Hong Kong, Hong Kong, China;*
[2]*ACCESS - AI Chip Center for Emerging Smart Systems, InnoHK Centers, Hong Kong Science Park, Hong Kong, China;*
[3]*Key Lab of Fabrication Technologies for Integrated Circuits, Chinese Academy of Sciences, Beijing 100049, China;*
[4]*Laboratory of Microelectronics Devices and Integrated Technology, Institute of Microelectronics of the Chinese Academy of Sciences, Beijing 100029, China;*
[5]*University of Chinese Academy of Sciences, Beijing 100049, China;*
[6]*School of Microelectronics, Southern University of Science and Technology, Shenzhen 518055, China*

Appendix A Pseudocode of the search process

To clarify the search process, the algorithm pseudocode is presented in Figure A1. This pseudocode outlines the detailed steps of our nested optimization approach, which includes the model architecture search using an Evolutionary Algorithm (EA) and hardware optimization through Particle Swarm Optimization (PSO).

Algorithm 1 Hardware-aware NAS with nested optimization

Require: Fitness defined on latency, energy and performance according to application scenarios and associated constraints
Ensure: Optimized Mixture of Experts (MoE) model architecture and hardware design.

```
Initialize hardware scoring.
repeat[Outer Loop: Model Architecture Search by Evolutionary Algorithm (EA)]
    Step 1: Model Architecture Search
        Employ an EA to sample MoE architectures.
        Use a selection mechanism to identify architectures with optimal performance.
    for each selected architecture do
        repeat[Inner Loop: Hardware Search by Particle Swarm Optimization (PSO)]
            Step 2: Hardware Simulation and Scoring
                Extract details from the architecture.
                Simulate hardware performance for both RRAM and SRAM.
                Score the hardware designs based on performance metrics.
            Step 3: Hardware Optimization
                Apply PSO to search heuristically for the optimal hardware configuration.
        until convergence is achieved or the maximum iteration count is reached
            Record the temporary best hardware score and design.
    end for
    Feedback the temporary best score as the hardware loss for the Model Architecture Search.
    Apply crossover and mutation operations to generate new architectures.
until convergence is achieved or the maximum iteration count is reached
Output the best overall hardware loss and hardware design.
```

Figure A1 Pseudocode of the search process

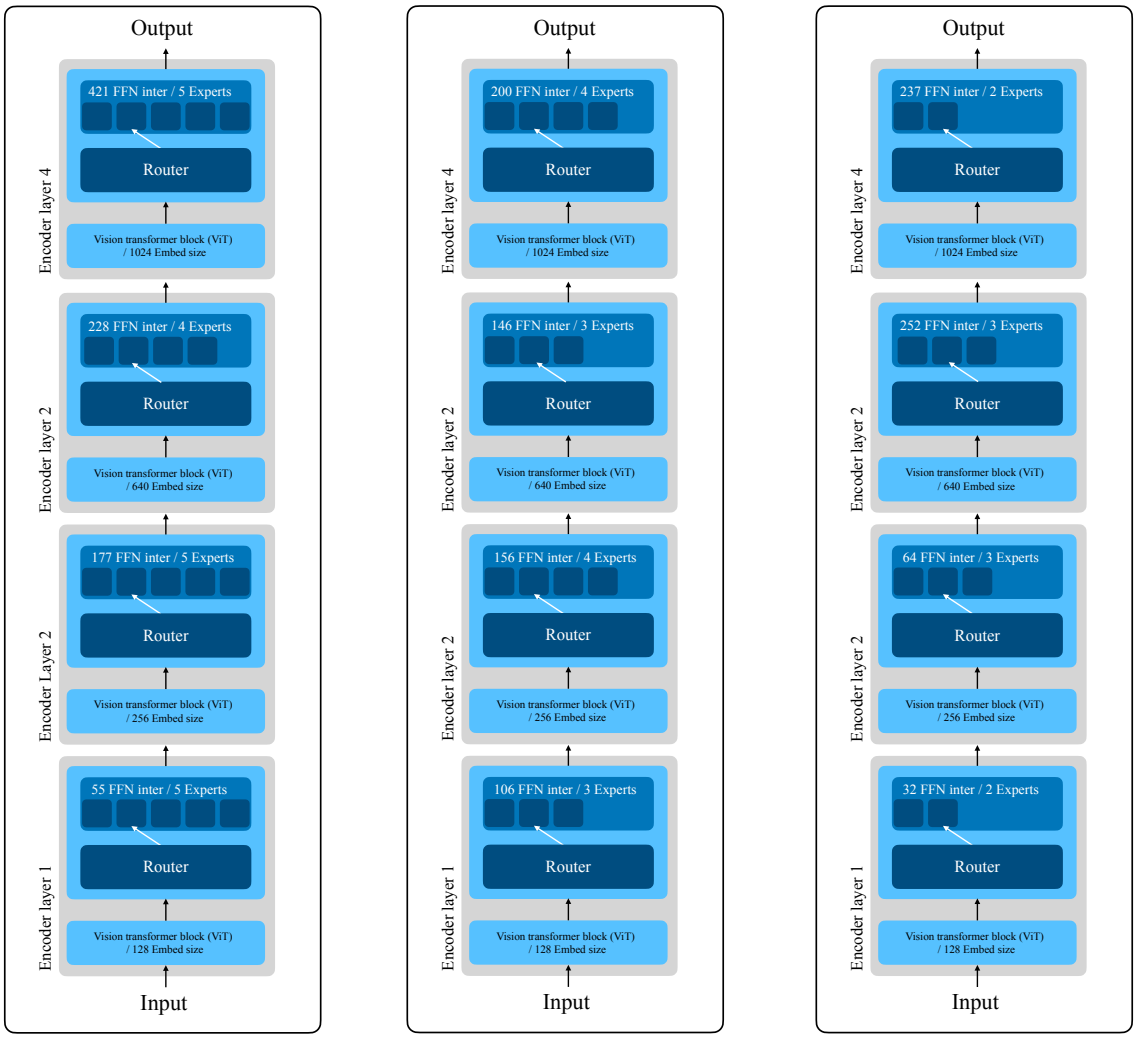
* Corresponding author (email: liyi126@hku.hk, wangzr@sustech.edu.cn, shangdashan@ime.ac.cn)
† Shihao Han, Sishuo Liu, Shucheng Du, and Mingzi Li have equal contributions to this work.

Appendix B Dynamic ViT-MoE searched results

We summarized the search result in the following Table B1. Figures B1 and B2 detail the searched results of the Dynamic ViT-MoE, showcasing the hierarchical organization, router and Vision Transformer (ViT) blocks, and Feed-Forward Network (FFN) configurations, thereby demonstrating the trend of the NAS to optimize for performance, latency, and energy consumption under different constraints.

Table B1 Summary of specific search results under different constraints

			Block1	Block2	Block3	Block4
Scenario1: Latency Constraint	RRAM	Number of Experts	2	5	3	2
		Hidden Size	97	99	258	671
	SRAM	Number of Experts	3	4	3	4
		Hidden Size	106	156	146	200
Scenario2: Energy Constraint	RRAM	Number of Experts	3	3	4	5
		Hidden Size	16	73	373	561
	SRAM	Number of Experts	2	3	3	2
		Hidden Size	32	64	252	237
Scenario3: Performance Constraint	RRAM	Number of Experts	5	5	4	5
		Hidden Size	55	177	228	421
	SRAM	Number of Experts	5	5	4	5
		Hidden Size	55	177	227	421



Dynamic ViT-MoE with performance-optimized for SRAM

Dynamic ViT-MoE with latency-optimized for SRAM

Dynamic ViT-MoE with energy-optimized for SRAM

Figure B1 Dynamic ViT-MoE searched results for SRAM

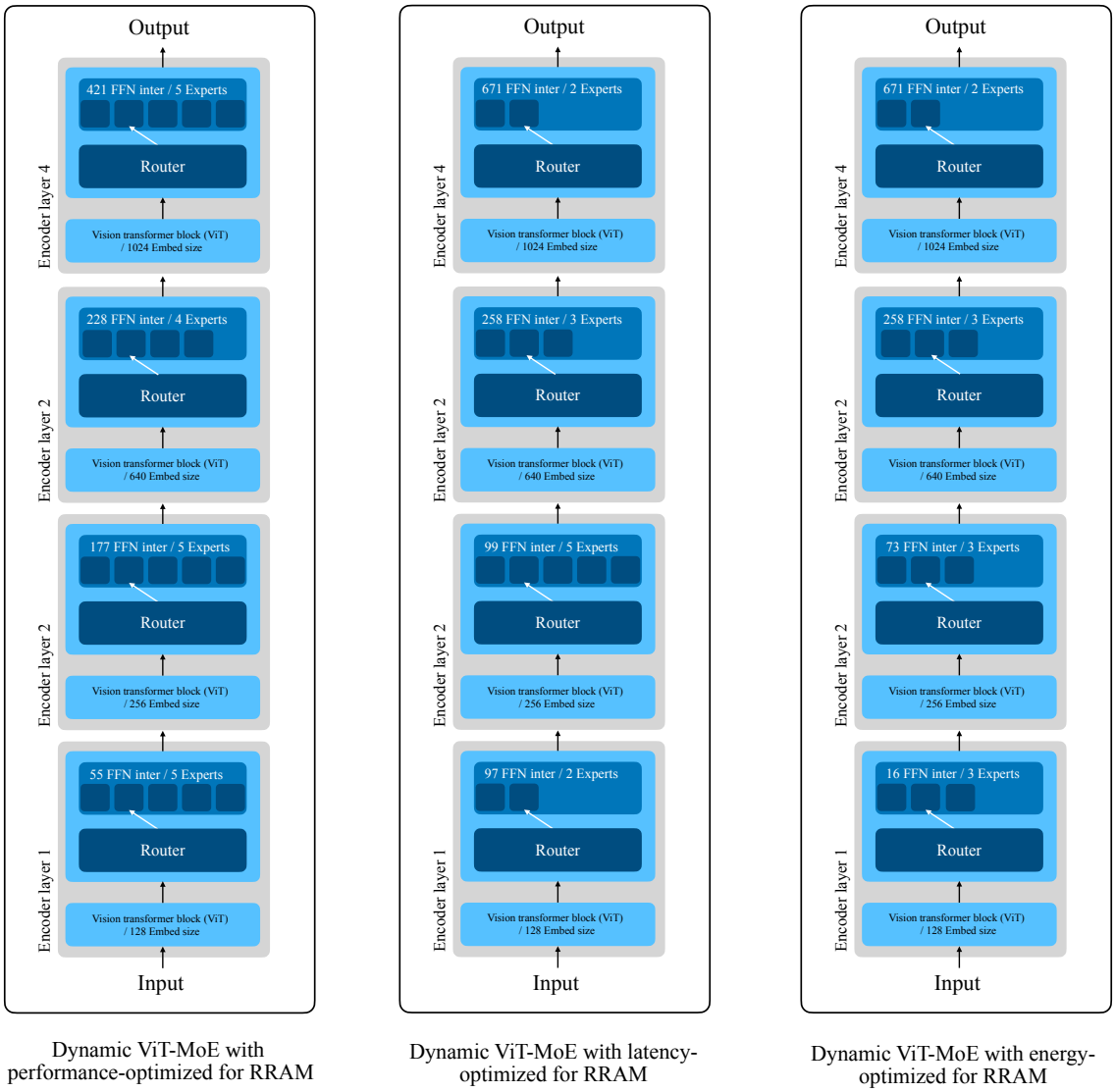


Figure B2 Dynamic ViT-MoE searched results for RRAM

Appendix C Hybrid system of SRAM and RRAM

To investigate the viability of an RRAM/SRAM hybrid system, we implemented a series of experiments utilizing a block-wise search policy. This methodology involved evaluating both RRAM and SRAM for each memory block to ascertain the optimal hardware configuration, considering system energy consumption, latency, and area efficiency. The experimental results, as illustrated in Figure C1, demonstrate the potential for synergistic integration of RRAM and SRAM technologies to achieve enhanced energy efficiency and latency under various constraints. This preliminary analysis elucidates the prospective advantages of judiciously combining SRAM and RRAM technologies in hybrid memory architectures. Such an approach facilitates tailored optimizations that leverage the inherent strengths of each memory type, thereby addressing diverse performance requirements in advanced computing systems.

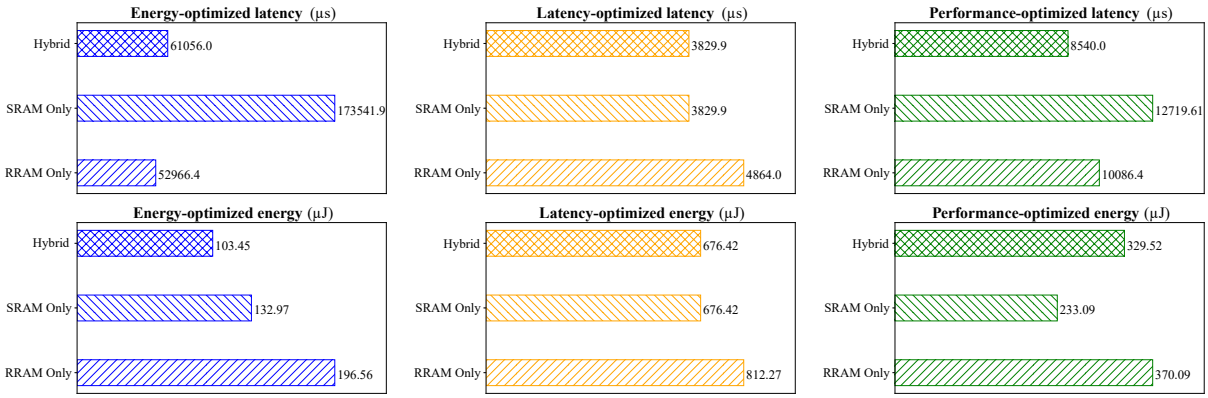


Figure C1 Dynamic ViT-MoE searched results for hybrid SRAM/RRAM system

Appendix D The analysis of discrepancy between the simulator and actual circuit

The proposed models for energy and latency in RRAM and SRAM align well with actual circuit behaviors [1, 2]. Although there is inevitable discrepancy in hardware performance estimation, the MoE-NAS remains robust and yields favorable outcomes within 10% discrepancy, ensuring its viability in practical applications (see Figure D1 and D2).

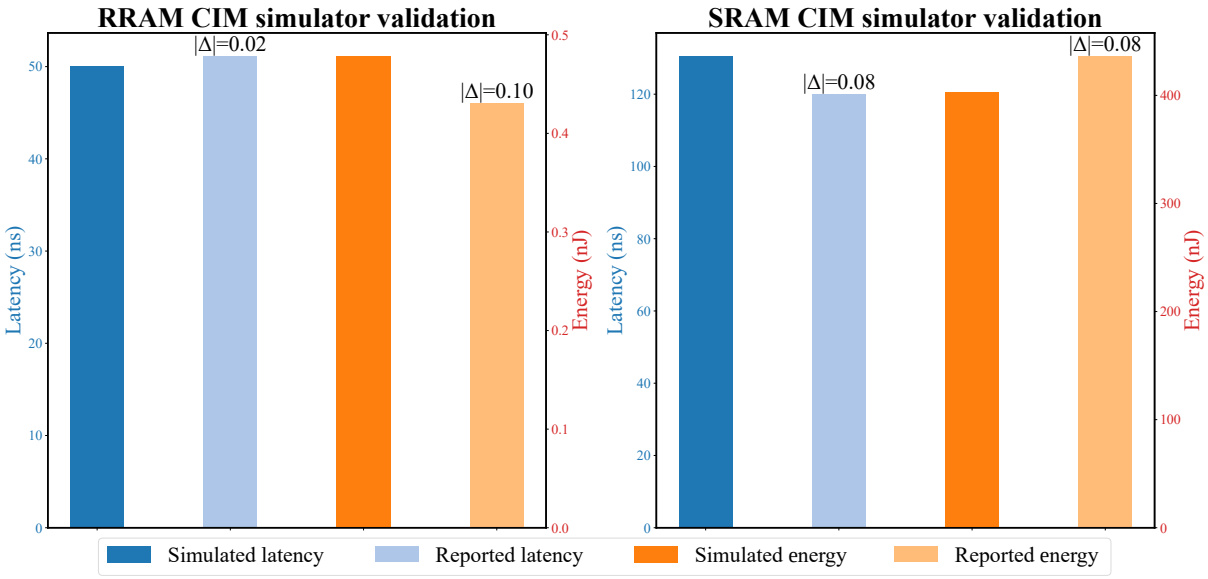


Figure D1 Comparison between reported CIM circuit and the simulator predictions

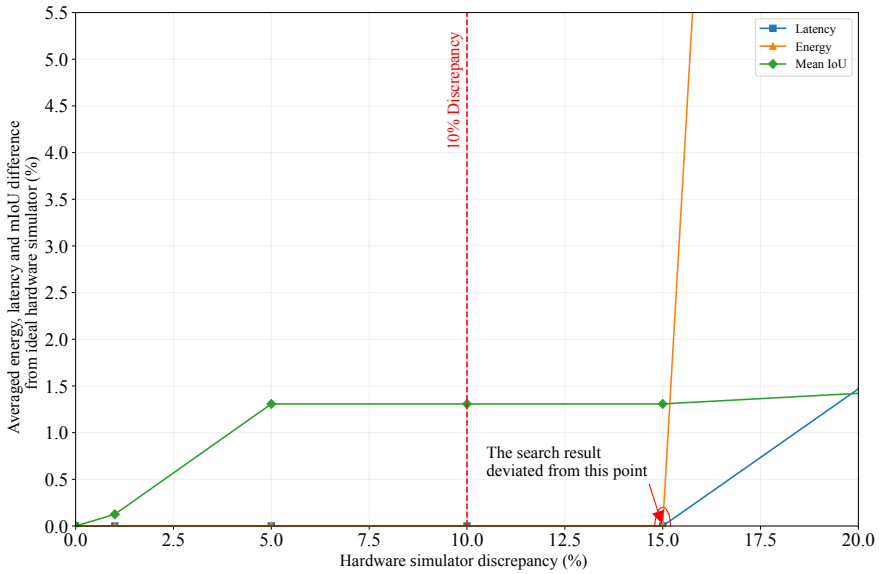


Figure D2 Percentage error in averaged searched latency, energy, and Mean IoU under different hardware simulator discrepancy

Appendix E The impact of RRAM read/write noise

It is noted that our hardware-aware NAS is robust to homogeneous variation, as shown in Figure E1. By integrating additional constraints for non-ideal factors, we can adapt our NAS algorithm to non-homogeneous variation.

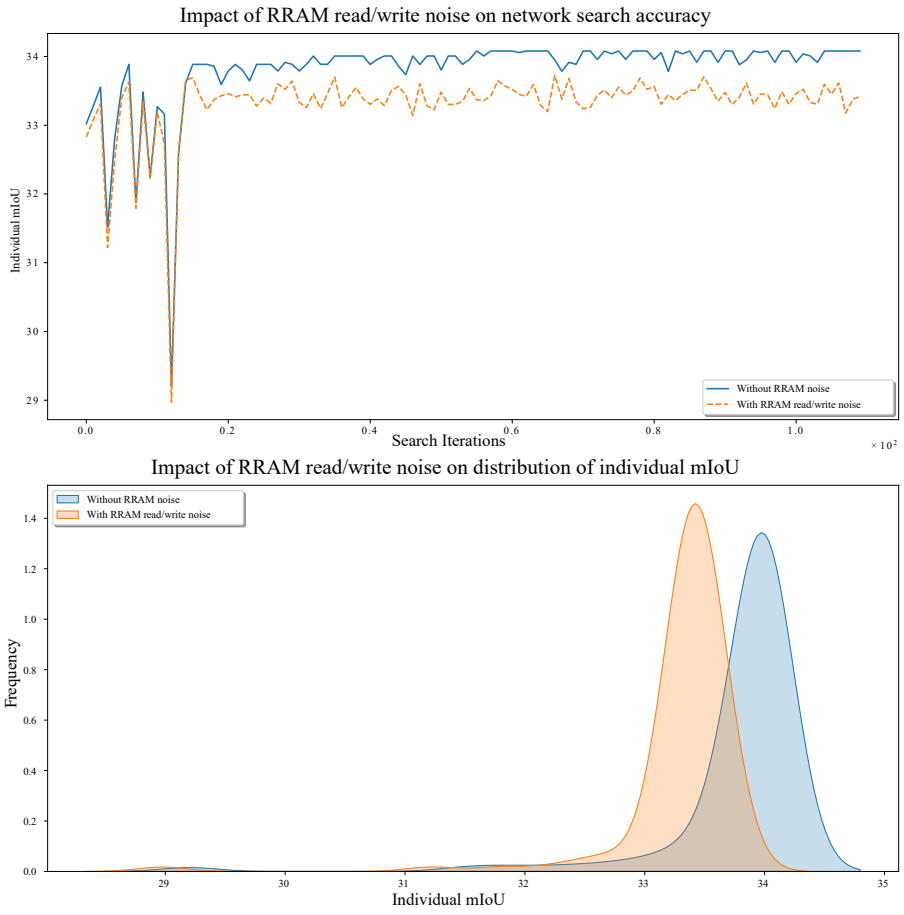


Figure E1 Impact of RRAM read/write noise on network search accuracy and distribution of individual search accuracy

Appendix F Capability of handling larger model deployments

Despite the 10 MB search limitation imposed on both SRAM and RRAM, our hardware simulator remains capable of handling larger model deployments. Also, the NAS shows, a nearly linear supernet training time increase and marginal search time increase relative to the LLM model size, as shown in the Figure F1 and F2.

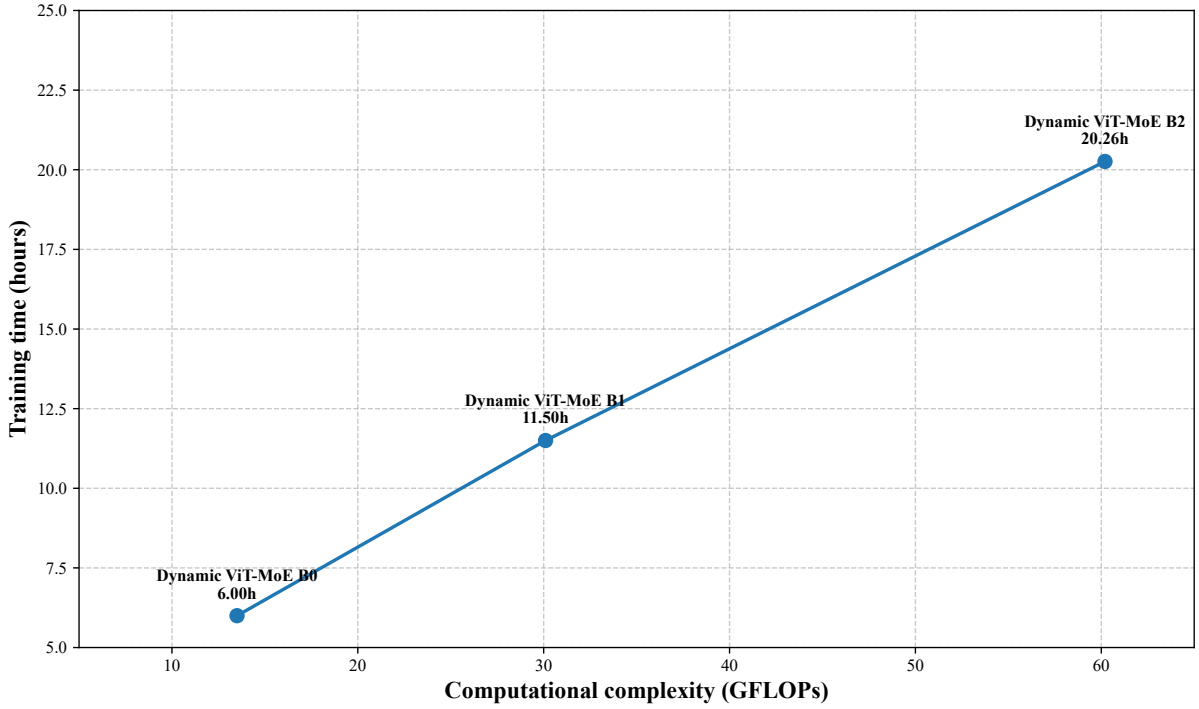


Figure F1 Supernet training time of the dynamic ViT-MoE B0 to B2

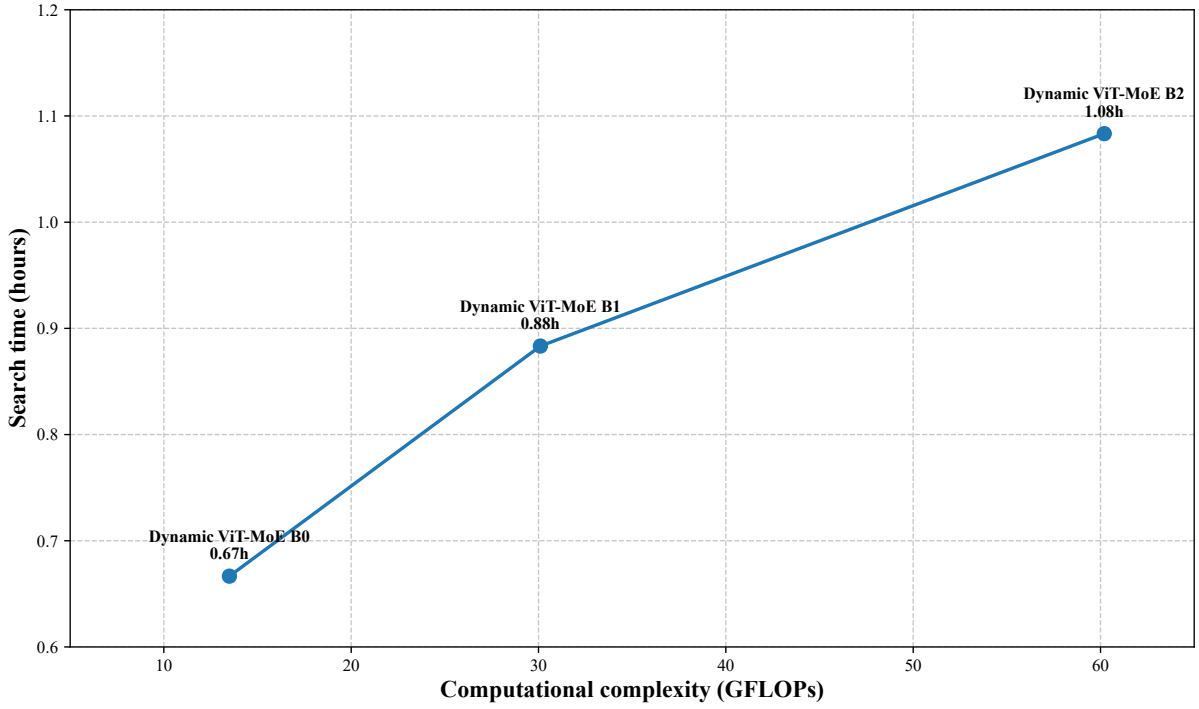


Figure F2 Evolutionary searching time of the dynamic ViT-MoE B0 to B2

Appendix G Comparison with post-training quantization and knowledge distillation

Compared to simple post-training quantization and knowledge refinement, our hardware-NAS exhibits comparable performance on similarly sized models while accounting for hardware constraints, as shown in Table G1 and G2

Table G1 Comparison of post training quantization to a sparse ViT-MoE model with the proposed NAS (*The best performance across all scenarios)

Approach	Accuracy	Hardware	Performance		
			mIoU (%)	Energy (μJ)	Latency (μs)
NAS	INT8	SRAM	35.21*	92.88*	3810.00*
		RRAM		132.97*	4870.00*
PTQ	INT8	SRAM	34.25	480.57	18188.7
		RRAM		389.16	15872
	INT4	SRAM	12.81	240.36	9167.1
		RRAM		200.42	7936
	INT2	SRAM	8.32	120.22	4620.22
		RRAM		103.22	3968

Table G2 Comparison of improvement from NAS, Knowledge Distillation, and NAS + Knowledge Distillation to sparse MoE Model (*performance-optimized)

Approach	Hardware	Performance		
		mIoU (%)	Energy (μJ)	Latency (μs)
-	SRAM	34.25	587.05	16253.5
	RRAM		480.01	17981.7
NAS*	SRAM	34.12	233.09	12719. 6
	RRAM		244.34	10086.4
Knowledge Distillation	SRAM	29.15	293.53	8126.76
	RRAM		240.30	8990.85
NAS+Knowledge Distillation	SRAM	33.03	185.97	10139.5
	RRAM		216.75	8935.15

References

1 Wenbin Zhang, Peng Yao, Bin Gao, Qi Liu, Dong Wu, Qingtian Zhang, Yuankun Li, Qi Qin, Jiaming Li, Zhenhua Zhu, et al. Edge learning using a fully integrated neuro-inspired memristor chip. *Science*, 381(6663):1205–1211, 2023.

2 Hyunjoon Kim, Qian Chen, Taegeun Yoo, Tony Tae-Hyoung Kim, and Bongjin Kim. A 1-16b precision reconfigurable digital in-memory computing macro featuring column-mac architecture and bit-serial computation. In *ESSCIRC 2019 - IEEE 45th European Solid State Circuits Conference (ESSCIRC)*, pages 345–348, 2019.