Efficient Edge Vision Transformer Accelerator with Decoupled Chunk Attention and Hybrid Computing-In-Memory

Yi Li[1,5,*], Zijian Ye[1,*], Xiangqu Fu[2,4,*], Songqi Wang[1,5], Shucheng Du[6], Ning Lin[1], Dashan Shang[2], Jinshan Yue[2,†], Zhongrui Wang[3,†], Xiaojuan Qi[1,†], Feng Zhang[2,†], Han Wang[1,5]

[1]Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong, China; [2]State Key Lab of Fabrication Technologies for Integrated Circuits and Key Laboratory of Microelectronic Devices and Integrated Technology, Institute of Microelectronics of the Chinese Academy of Sciences, Beijing, China; [3]School of Microelectronics, Southern University of Science and Technology, Shenzhen, China; [4]The School of Integrated Circuits, University of Chinese Academy of Science

[5]Center for Advanced Semiconductor and Integrated Circuits, The University of Hong Kong, Hong Kong, China

[6]ACCESS – AI Chip Center for Emerging Smart Systems, InnoHK Centers, Hong Kong Science Park, Hong Kong, China

[*]Equal contribution to this work.

[†]Corresponding author: yuejinshan@ime.ac.cn; wangzr@sustech.edu.cn; xjqi@eee.hku.hk; zhangfeng_ime@ime.ac.cn.

***Abstract*—Vision Transformers (ViTs) are new foundation models for vision applications. Edge-deploying ViTs to realize energy-saving, low-latency, and high-performance dense predictions have wide applications, such as autonomous driving and surveillance image analysis. However, the quadratic complexity of the self-attention mechanism renders ViTs slow and resource-intensive, particularly for pixel-level dense predictions that involve long contexts. Additionally, the pyramid-like architecture of modern ViT variants leads to an unbalanced workload, further reducing hardware utilization and decreasing the throughput of conventional edge devices. To this end, we propose an algorithm-hardware co-optimized edge ViT accelerator tailored for efficient dense predictions. At the algorithm level, we propose a decoupled chunk attention (DCA) mechanism implemented in a pipelined manner to reduce off-chip memory access, thereby enabling efficient dense predictions within limited on-chip memory. At the architecture level, we introduce a hybrid architecture that combines SRAM-based computing-in-memory (CIM) and nonvolatile RRAM storage to eliminate extensive off-chip memory access, with a fusion scheduling to balance workloads and minimize intermediate on-chip memory access. At the circuit level, a bit/element two-way-reconfigurable CIM macro is proposed to improve hardware utilization across pyramidal ViT blocks with varied matrix sizes. The experimental results on object detection, semantic segmentation, and depth estimation tasks demonstrate that our design can efficiently process patch lengths up to 16384 with a speedup of 18.5×-217.1×, a reduction in memory accesses of 1.7×-7.4×, and an improvement in energy efficiency of 1.8×, under less than 1% performance degradation.**

***Index Terms*—Vision Transformers, compute-in-memory, edge AI accelerator, dense prediction**

I. Introduction

Recently, Vision Transformers (ViTs) [1] have emerged as foundation models in computer vision (CV), outperforming convolutional neural networks (CNNs) in various downstream tasks. By partitioning input images into sequences of patches and leveraging self-attention mechanisms, ViTs excel at capturing global image features and have demonstrated superior performance in dense prediction tasks [2], such as object detection, semantic segmentation, and depth estimation. These tasks involve pixel-level category prediction rather than image-level classification, making them critical for applications that require precise visual interpretation. Moreover, dense prediction tasks have a wide range of potential applications, including autonomous driving, robotics, and surveillance.

Despite their success, deploying ViTs on edge devices presents significant challenges. Unlike cloud servers with vast computational resources, edge devices operate under constraints in power, memory, and storage, while still requiring high accuracy and energy efficiency. The quadratic computational complexity of the self-attention mechanism in ViTs exacerbates these issues, particularly for pixel-level dense predictions involving long sequence lengths and extensive computations.

Several studies have been proposed to mitigate the hardware costs of vanilla Transformers and make them more suitable for resource-constrained edge devices [3]. Software-wise, researchers focus on reducing the computational burden of self-attention in Transformer inference by exploiting the inherent sparsity [4]. Since each token is usually correlated with only a few other tokens, the attention matrix is often sparse, therefore, many computations can be skipped to minimize energy and latency overhead. Hardware-wise, one solution is to optimize data flow scheduling to enhance the efficiency of traditional computing platforms. For instance, the FLASHATTENTION [5] method utilizes tiling techniques to reduce data shuffling between the low-speed, high-bandwidth memory (HBM) of the graphics processing unit (GPU) and the high-speed on-chip static random-access memory (SRAM), resulting in lower system power consumption and reduced latency. Another promising solution is the exploitation of the emerging compute-in-memory (CIM) architecture. With the computational functionality built into the memory block itself, the CIM paradigm significantly reduces the energy and time costs associated with off-chip memory access during Transformer's attention calculations on von Neumann architecture-based edge devices [6].

Nonetheless, deploying ViTs for dense predictions on edge devices remains a formidable challenge due to two key factors. First, pixel-level vision tasks often require significantly longer contexts compared to natural language processing (NLP) tasks, resulting in quadratic growth in memory and computational demands as the image resolution increases [7]. For example, when estimating depth

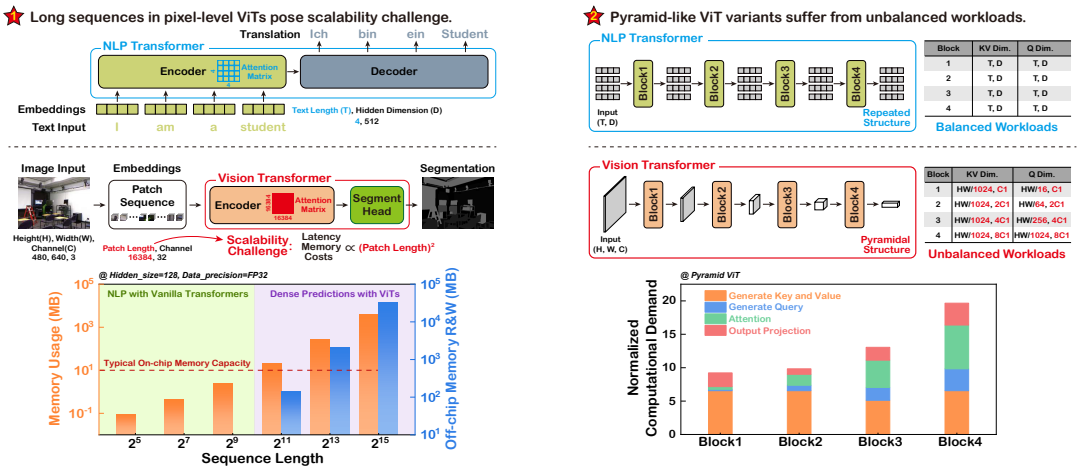

Fig. 1. Dense predictions with ViTs on edge devices face significant challenges, especially with scalability (left) and workload balance (right).

on the ADE20k dataset, the sequence length for the ViT encoder can reach up to 16,384 patches. Second, modern ViTs frequently employ pyramidal architectures to reduce computational costs while capturing multi-scale resolution features [8]. However, the varying feature sizes and matrix dimensions of pyramid ViT blocks not only challenge the optimal resource mapping due to the varying workload in each block but also incur hardware utilization concerns.

To address these challenges, we propose an algorithm-hardware co-optimized edge ViT accelerator tailored for efficient dense predictions. The contributions of this work can be summarized as follows:

- **Algorithmic Level:** We propose the decoupled chunk attention (DCA) mechanism, which rearranges patches to preserve spatial information and processes attention in pipelined chunks. This approach significantly reduces off-chip memory access, achieving 87.1% sparsity and 56.9% computational reduction during attention calculation, with less than 1% performance degradation compared to vanilla attention.
- **Architectural Level:** We introduce a hybrid architecture combining SRAM-based CIM and RRAM storage, where static weights are stored on-chip using RRAM to eliminate off-chip memory access, and a fusion scheduling strategy is proposed to balance workloads and minimize intermediate on-chip memory access. This reduces memory accesses by 1.7×-7.4×, and improves energy efficiency by 1.8× compared to using vanilla attention on SRAM-CIM with off-chip DRAM.
- **Circuit Level:** We develop a reconfigurable computing circuit that combines bit-level and element-level parallelism to enhance hardware utilization and throughput across pyramidal ViT blocks. This work achieves a total acceleration ratio of 18.5×-217.1×.

II. Preliminaries and Motivation

A. ViT

In 2020, Dosovitskiy et al. [1] introduced the ViT architecture, which allows for capturing global interactions between image patches without the constraints of local connectivity. Despite their good performance, vanilla ViTs use only single-scale low-resolution features and suffer from high computation costs on dense prediction tasks, such as semantic segmentation and depth estimation. To address these limitations, Wang et al. [8] leveraged the strengths of both CNNs and Transformers by substituting uniform encoder blocks with a progressively shrinking pyramid structure, thereby reducing the computational cost of large feature maps and enhancing the performance of many downstream CV tasks. Recent research on ViTs has increasingly focused on utilizing sparse techniques to further reduce the complexity of self-attention computations. Zheng et al. [9] collapsed the input or output dimensions based on the sparsity and skipped the zero-value attention map. Liu et al. [10] sparsely preserved the semantically relevant patches and condensed irrelevant ones into a representative patch.

B. CIM

Data movement consumes the majority of time and energy in the inference process, therefore reducing data movement is crucial for efficiently deploying Transformers on edge devices [6]. The CIM paradigm, which integrates computation within the memory itself, offers an energy-efficient and high-speed computing architecture well-suited for Transformer processing. Recent CIM-based Transformer accelerators predominantly employ either analog RRAM-CIM or digital SRAM-CIM architectures [11], [12]. Analog RRAM-CIM provides high-density, nonvolatile storage and facilitates matrix multiplications via inherent physical properties. However, RRAM's intrinsic stochasticity and expensive programming overhead may render it unsuitable for dynamically generated, high-precision attention computations. Conversely, digital SRAM-CIM architecture incurs minimal memory write overhead and supports lossless computations, ensuring efficient and high-accuracy Transformer processing. Nevertheless, the volatile nature of SRAM necessitates loading weights from off-chip memory, resulting in a slow wakeup-to-response time for edge devices. Moreover, SRAM cells

require more area compared to RRAM cells, limiting their memory capacity.

C. Motivation

Although sparse attention and the utilization of CIM hardware architectures have reduced data movement and decreased the time-energy overheads associated with self-attention. However, the computation of self-attention still relies on the entire sequence, failing to decouple the attention size from the sequence length, and continues to face scalability issues in pixel-level dense predictions with longer contexts. As illustrated in the left part of Fig. 1, when the sequence length exceeds 1024, the memory usage of the attention block surpasses the typical on-chip storage capacity of CIM systems, resulting in frequent reads and writes to off-chip memory. Moreover, with each quadrupling of the sequence length, the frequency of accesses to the off-chip storage increases by an order of magnitude. Additionally, the pyramid-like architecture of modern ViT variants leads to an unbalanced workload. As shown in the right part of Fig. 1, the computational demands of each block vary due to differences in block size and spatial reduction mechanisms. This variability renders the fixed data flow and scheduling in the NLP transformer inefficient, leading to suboptimal hardware utilization. To the best of our knowledge, no existing work has simultaneously addressed these issues. Therefore, a novel solution is required to deploy ViTs for dense predictions on edge devices efficiently.

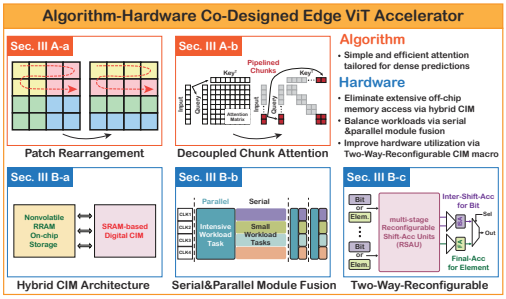

Fig. 2. Overview of the co-designed edge ViT accelerator.

III. Algorithm-Hardware Co-Designed Edge ViT Accelerator

Fig. 2 depicts the overview of the proposed algorithm-hardware co-designed edge ViT accelerator. Algorithmically, we propose a simple and efficient DCA mechanism that computes attention scores using fixed-size chunks in a pipelined manner, thereby minimizing the frequent access of query (Q), key (K), and value (V) elements. Hardware-wise, we introduce a hybrid architecture that combines SRAM-based CIM with nonvolatile on-chip RRAM storage to eliminate extensive off-chip memory access, with a fusion scheduling to minimize intermediate on-chip memory access for the varied workload ratios in each pyramid block. Further, we design reconfigurable CIM computing circuits that combine bit-level and element-level parallelism to improve resource utilization.

A. DCA Mechanism

a) Patch Rearrangement: In contrast to the long-range dependencies between tokens typically observed in NLP tasks, CV tasks generally exhibit stronger correlations between adjacent patches [13]. However, the sequential flattening adopted from NLP Transformers overlooks the two-dimensional spatial characteristics of images, causing adjacent patches to be scattered across widely spaced locations in the attention matrix (Fig. 3, upper part). Although the attention mechanism remains effective in capturing these long-range dependencies, the increased intervals necessitate the storage of longer patch sequences during attention computation, leading to costly off-chip memory accesses. To preserve the spatial information of adjacent patches, we divide the patch embeddings $\in \mathbb{R}^{H\times W}$ into windows $\in \mathbb{R}^{\frac{H}{4}\times\frac{W}{4}}$ and rearrange the patches akin to sequential flattening. This rearrangement effectively reduces the attention distance between adjacent patches, resulting in the clustering of high attention scores and forming a chunk-like attention pattern (Fig. 3, lower part). Therefore, we can still obtain sufficient information from the image even if we focus solely on the attention values within these chunks and ignore interactions with distant patches.

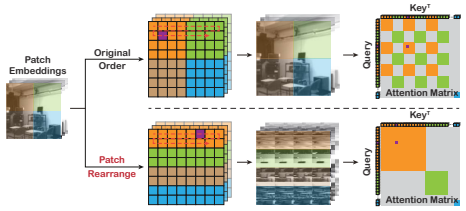

Fig. 3. Patch rearrangement diagram.

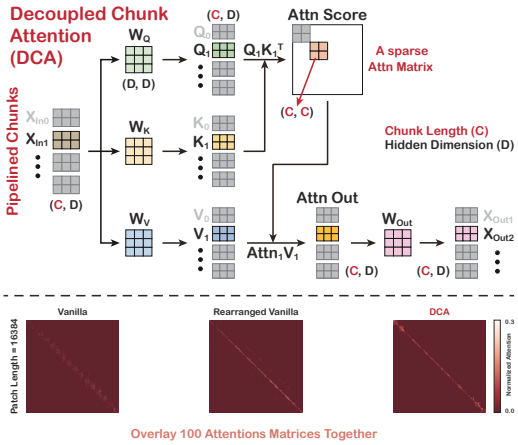

Fig. 4. Flowchart of the DCA. The bottom section displays the attention matrices of three different methods: vanilla attention (left), rearranged vanilla attention (middle), and the proposed decoupled chunk attention (right).

b) Hardware-friendly Chunk Attention: Inspired by the FLASHATTENTION, we propose a DCA algorithm tailored for dense predictions at the edge (Fig. 4, upper part), where the Q, K, and V matrices are partitioned into small chunks based on the on-chip memory capacity (e.g., a chunk length of 1024 corresponds to an on-chip

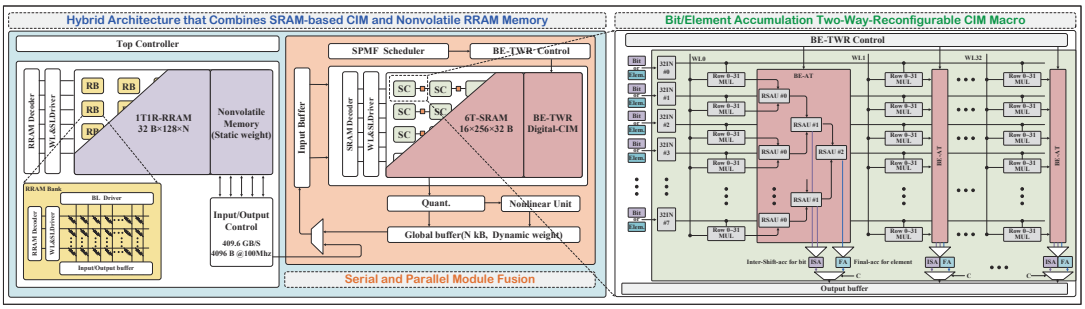

Fig. 5. Architecture overview of the proposed hybrid CIM ViT accelerator.

memory capacity of 128 KB), and the attention output is computed based on these chunks. Thanks to the pre-rearranged patches that concentrate important and relevant information in diagonal chunks, we avoid traversing the entire K and V matrices during the attention computation, in contrast to the FLASHATTENTION. Simultaneously, the pipelined architecture of the attention calculation minimizes the need for expensive off-chip memory accesses to store the complete input and intermediate attention results, thereby significantly reducing both the time and energy overhead associated with the attention computation.

To demonstrate the efficacy of our method, we visualize the overlay attention matrices of 100 randomly selected images, comparing the vanilla attention pattern with our DCA approach (Fig. 4, lower part). In the vanilla attention, bright spots indicating high correlation are scattered throughout the attention map. After patch rearrangement, most of these spots are concentrated on the diagonal strip, exhibiting an attentional pattern similar to our DCA method.

B. RRAM-storage SRAM-CIM Hybrid Hardware Architecture

a) Overall Hardware Architecture: As shown in Fig. 5, the proposed hybrid architecture consists of a top controller, 4 MByte 1T1R RRAM banks for static weights storage, 16 SRAM CIM (SC) macros supporting bit/element two-way reconfiguration, a non-linear unit, a small global buffer for dynamic weights, and a configurable serial and parallel module fusion (SPMF) scheduler. The high-density RRAM bank stores the static weights of each ViT block, and the reconfigurable SRAM-CIM macro efficiently performs matrix-matrix multiplications in DCA. The hybrid CIM architecture eliminates off-chip access through two means: first, the high-density RRAM banks allow for full on-chip storage of the lightweight Segformer-B0 model (3.4 MB) [7]; second, the proposed SPMF scheduling method eliminates off-chip access for dynamic weights and intermediate results.

The SPMF scheduler addresses the unbalanced workloads of the pyramidal ViTs depicted in Fig. 1. By decoupling the unbalanced workload in each ViT block using a combination of serial computation and parallel module fusion, the SPMF scheduler reduces intermediate memory access, ensuring efficient processing across the Segformer model. Additionally, we propose a bit/element two-way-reconfigurable CIM (BE-TWR-CIM) adder tree in the CIM macros to efficiently process matrix multiplications of different sizes (32, 64, 256, 1024, etc.). The BE-TWR controller can configure the adder tree into a high-input-parallelism single-bit accumulation mode for large-size matrix multiplication or a low-input-parallelism multi-bit accumulation mode for small-size matrix multiplication, thereby maximizing computation resource utilization.

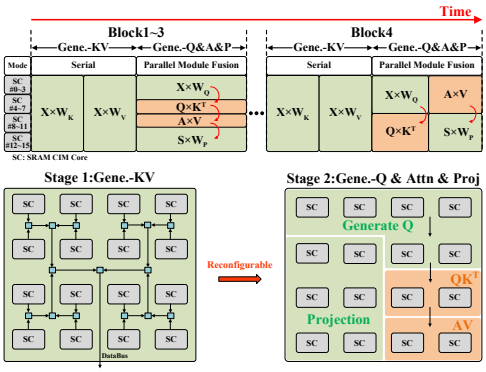

Fig. 6. Serial and parallel module fusion.

b) Serial and Parallel Module Fusion Scheduling: The fundamental principle of the SPMF scheduling method is to optimize computation resource utilization while minimizing memory access and storage requirements. As depicted in Fig. 4, the parallel execution of several operators (e.g., $Q \cdot K^T$ and $A \cdot V$) can eliminate intermediate memory access and storage space through direct data forwarding. However, the available computation resources may not suffice to support the parallel execution of all pipelineable operators. Consequently, the SPMF scheduling method is employed to determine the optimal parallel and serial workflows within each block, as illustrated in Fig. 6. In Blocks 1 to 3, the matrix dimensions of K^T and V are relatively small, allowing W_Q and W_p to be divided into smaller blocks without data dependency. This enables the full pipelining of the four operators ($X \cdot W_Q$, $Q \cdot K^T$, $A \cdot V$, and $S \cdot W_p$), thereby conserving memory access and storage space for the intermediate matrices Q, A, and S. In Block 4, the increased matrix dimensions of K^T and V preclude their mapping onto the limited CIM computation resources in a fully pipelined manner without

data dependency. Consequently, the pipeline is segmented into two sequential stages. The first stage involves the generation of large K and V matrices that cannot be efficiently integrated into a pipelined structure with adjacent operators, thereby executing serially. A reconfiguration schematic of the 16 SC macros between serial generation mode and parallel attention and projection mode is also provided in Fig. 6.

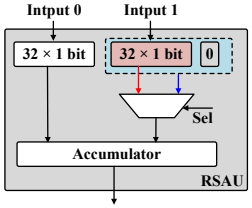

Fig. 7. Bit/element two-way-reconfigurable CIM adder tree.

c) Bit/element Two-Way-Reconfigurable CIM Macro: Each of the 16 SRAM-CIM macros comprises 256×256 SRAM cells and their associated peripherals. Each column of 256 SRAM cells is connected to a bit/element two-way reconfigurable adder tree (BE-AT), which is constructed from multi-stage reconfigurable shift accumulation units (RSAUs), as depicted in Fig. 7. In high-input-parallelism single-bit (HP-SB) accumulation mode, configured by the BE-TWR controller for large-scale matrix multiplication (e.g., 256×256×32, 1 bit), the RSAU utilizes a 2-1 multiplexer to load Input1 (32×1 bit) into the accumulator directly. By leveraging multi-stage direct accumulation, the RSAU efficiently performs single-bit accumulation for large-size matrix multiplications. In low-input-parallelism multi-bit (LP-MB) accumulation mode, also configured by the BE-TWR controller for smaller-scale matrix multiplications (e.g., 32×32×32, 4 bit), the RSAU employs the 2-1 multiplexer to load the shifted Input1 into the Accumulator. Through multi-stage shift accumulation, the BE-AT efficiently performs multi-bit accumulation for smaller-scale matrix multiplications.

IV. Evaluation

A. Experiment Setup

a) Networks and Dataset: To demonstrate the efficacy and generality of our design, we evaluate the performance of DCA using SegFormer-B0 as the backbone across three representative dense prediction tasks: object detection on the COCO dataset [14], semantic segmentation on the ADE20k dataset [15], and depth estimation on the NYUDV2 dataset [16]. SegFormer is a state-of-the-art pyramidal ViT framework that uses a series of mixed Transformer encoders to extract hierarchical feature representations across multiple scales. During the experiments, input images are resized to a resolution of 512×512 and embedded into 128×128 patches, corresponding to a sequence length of 16384. The hierarchical feature maps for the four encoder blocks in SegFormer-B0 are (128, 128, 32), (64, 64, 64), (32, 32, 160), and (16, 16, 256). Subsequently, a lightweight full multilayer perceptron decoder aggregates the information from different layers and generates pixel-wise predictions. The simple yet powerful SegFormer backbone is ideally suited for dense predictions on resource-constrained edge devices.

b) Hardware Baselines: To better illustrate the enhancements introduced by each proposed technique, we utilize three hardware baselines for comparative analysis. Baseline0 maintains the same SRAM CIM computational performance (excluding the BE-TWR adder tree) and uses off-chip DRAM for intermediate data storage while executing the original Segformer model. Baseline1 incorporates the DCA algorithm optimization based on Baseline0. OP0 refers to deploying DCA on the proposed RRAM-storage SRAM-CIM hybrid architecture, whereas OP1 further integrates the SPMF scheduling and bit/element two-way-reconfigurable accumulation technique. Additionally, a GPU baseline is evaluated with the RTX A30.

c) Hardware Components and Power/Area Breakdown: We implement the proposed hybrid CIM ViT accelerator using Verilog and synthesize the design using Synopsys Design Compiler under a 28 nm technology at 250 MHz. Table I shows the hardware components and power/area breakdown. It contains 16 SRAM CIM macros, each consisting of 256×256 SRAM cells and 256 BE-AT adder trees. The BE-AT unit only increases power consumption by 2.3% compared to a normal adder tree. The global SRAM buffer and fixed RRAM banks are set to 640 KB and 4 MB, respectively. The total area is 10.54 mm^2 with 834 mW peak power.

TABLE I
Hardware Components and Area/Power Breakdown

Module	Description	Area (mm^2)	Power (mW)
SRAM-CIM: cells	16×256×256	4.30	141
SRAM-CIM: BE-AT	16 × 256	1.98	414
Top control	-	0.32	3.64
Nonlinear unit	-	0.50	24.0
Global buffer	640 KB	1.28	10.1
RRAM	4 MB	2.16	241
Total	-	10.54	834

B. Experiment Results

a) DCA algorithm: Table II presents the performance results of our DCA design in comparison with the baseline SegFormer-B0 model across three different tasks. Our DCA approach achieves an 87.1% sparsity in the attention matrix and reduces attention FLOPs by 56.9% compared to the baseline model. Despite the simplicity of the DCA mechanism, it proves to be highly effective, achieving a mIoU of 37.1% on the ADE20K, a RMSE of 0.62 on the NYUDV2, and a box mAP of 36.6% and a mask mAP of 34.4% on the COCO, with an average performance degradation of only 0.7% compared to the conventional attention method. The visualization result of three tasks in Fig. 8 further demonstrates the high performance. For object recognition tasks, the DCA method shows results close to the baseline, albeit with slightly lower confidence scores for certain objects. For semantic segmentation tasks, the DCA method yields similar results consistent with the baseline, particularly for objects occupying large areas, with only a minor reduction in accuracy at the boundaries

between different classes. In the depth estimation task, the depth prediction results are comparable to the baseline prediction, with a small decrease in accuracy in delineating object boundaries.

b) Evaluation of Acceleration Ratio: Fig. 9(a) presents the acceleration ratio compared to the GPU and hardware baselines. Among the four ViT blocks in Segformer, barely using CIM shows 8.5×-16.1× acceleration. Utilizing the DCA Segformer algorithm optimization, the acceleration reaches 10.9×-20.9×, which varies depending on the algorithm-level computation compression on each ViT block. Note that the hardware resource cannot be fully utilized due to the unbalanced workload ratios among operators in each block and also the varying matrix dimensions. Utilizing the SPMF scheduling and BE-TWR CIM adder tree, the total acceleration ratio finally reaches 18.5×-217.1×.

TABLE II
PERFORMANCE COMPARISON BETWEEN VANILLA ATTENTION AND DCA ON DENSE PREDICTION TASKS

SegFormer B0 (3.4 M)	**Attn Sparsity** (% ↑[a])	**Attn FLOPs** (M ↓)	**COCO** (mAP ↑)	**ADE20k** (mIoU ↑)	**NYUDV2** (RMSE ↓)	**Average Loss** (% ↓)
Vanilla Attention	/	1.938	36.9 (Box) 34.7 (Mask)	37.4	0.618	/
DCA	87.1	0.836	36.6 (Box) 34.4 (Mask)	37.1	0.62	0.7

[a]Arrow direction indicates a better result.

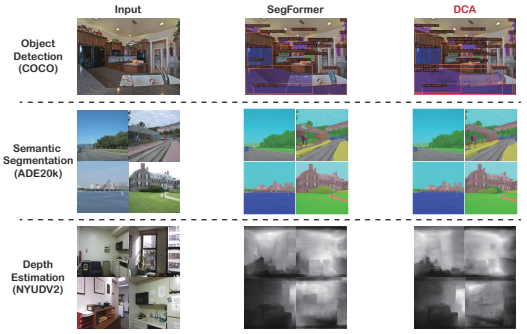

Fig. 8. Dense prediction results.

c) Evaluation of Memory Access: Fig. 9(b) shows that the proposed optimizations, OP1 and OP2, significantly reduce both on-chip and off-chip memory access compared to baseline0 and baseline1. For on-chip memory access, this work integrates SPMF scheduling and the BE-TWR accumulation techniques to reduce memory access by 1.7×-7.4× across four blocks compared to Baseline0. For off-chip memory access, this work utilizes the DCA mechanism and the heterogeneous RRAM-storage SRAM-CIM architecture to reduce memory access across four blocks to zero compared to Baseline0.

d) Evaluation of Performance and Energy Efficiency: Table III presents the comparison with the state-of-the-art ViT accelerators. The proposed hybrid CIM ViT accelerator achieves 1.6× and 1.8× energy efficiency improvement compared to the recent TranCIM [17] and P^{3}ViT [18], respectively. Utilizing the high-density 4 MB RRAM for fixed weight and 640 KB SRAM for intermediate data, this work avoids off-chip access compared to previous CIM chips. Furthermore, thanks to the DCA mechanism,

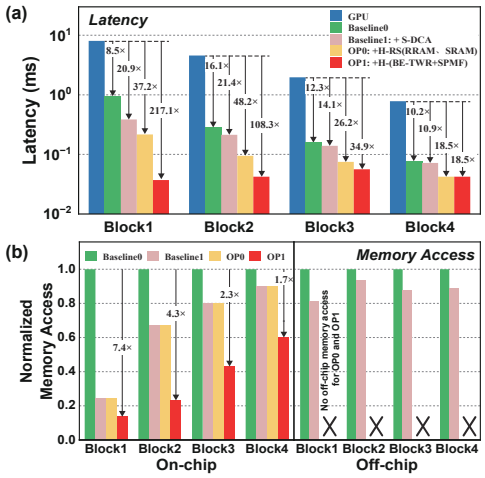

Fig. 9. Evaluation of (a) acceleration ratio and (b) memory access.

this work enables a pixel-level dense prediction algorithm with a maximum sequence length of 16384 within limited hardware resources.

TABLE III
COMPARISON TO THE STATE-OF-THE-ART VIT ACCELERATORS

	P^{3}ViT [18]	**TranCIM** [17]	**This Work**
MAC Implementation	Digital CIM	Digital CIM	Digital CIM
Technology (nm)	28 nm	28 nm	28 nm
On-chip memory	100 KB	192 KB	**4736 KB**
Need off-chip access	Yes	Yes	**No**
Frequency (MHz)	50-200 MHz	80-240 MHz	250 MHz
Die Area (mm^2)	2.0	6.83	9.46
Precision	INT4/8/16	INT8/16	INT8
Performance (TOPS)	1.6(INT8)	1.48(INT8)	6.55(INT8)
Energy Efficiency (TOPS/W)	23.2(INT8)	20.5(INT8)	**36.9(INT8)**
Prediction Level	Image-Level	Token-Level	**Pixel-Level**
Max Sequence Length	197	4096	**16384**

V. CONCLUSION

In this study, we implement an efficient yet powerful DCA on a hybrid CIM architecture, incorporating fusion scheduling and reconfigurable computation circuits to reduce both on- and off-chip memory accesses. Tailored for dense predictions, our proposed design supports a patch length of up to 16384, with a speedup of 18.5×-217.1×, a reduction in memory accesses of 1.7×-7.4×, and an improvement in energy efficiency of 1.8× under less than 1% performance loss across three representative pixel-level tasks. This ViT accelerator offers a promising solution for various dense prediction applications at the edge.

ACKNOWLEDGMENT

This work was supported in part by National Key Research and Development Program of China under Grant 2023YFB4402400; National Natural Science Foundation of China under Grants 92464201, U2341218, 92464203, 62374181, and 62204256; Shenzhen Science and Technology Innovation Commission (Grant No.SGDX20220530111405040); Beijing Natural Science Foundation (Grant No.Z210006); Hong Kong Research Grant Council (Grant Nos.27209621, 17205922, 17212923); RGC YCRG Grant no.C7003-24Y and RGC GRF Grant no.17205922; Joint Laboratory of Microelectronics (JLFS/E-601/24); This research was partially supported by ACCESS – AI Chip Center for Emerging Smart Systems, sponsored by InnoHK funding, Hong Kong SAR.

References

[1] A. Dosovitskiy, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[2] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *ICCV*, 2021, pp. 12 179–12 188.

[3] Y. Chen, L. Zhang, J. Shang, Z. Zhang, T. Liu, S. Wang, and Y. Sun, "DHA: learning decoupled-head attention from transformer checkpoints via adaptive heads fusion," in *NIPS*, 2024.

[4] W. Fedus, J. Dean, and B. Zoph, "A review of sparse expert models in deep learning," *arXiv preprint arXiv:2209.01667*, 2022.

[5] T. Dao, D. Fu, S. Ermon, A. Rudra, and C. Ré, "Flashattention: Fast and memory-efficient exact attention with io-awareness," *Adv. Neural Inform. Process. Syst.*, vol. 35, pp. 16 344–16 359, 2022.

[6] M. Zhou, W. Xu, J. Kang, and T. Rosing, "Transpim: A memory-based acceleration via software-hardware co-design for transformer," in *HPCA*, 2022, pp. 1071–1085.

[7] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and efficient design for semantic segmentation with transformers," *Adv. Neural Inform. Process. Syst.*, vol. 34, pp. 12 077–12 090, 2021.

[8] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *ICCV*, 2021, pp. 568–578.

[9] Q. Zheng, S. Li, Y. Wang, Z. Li, Y. Chen, and H. H. Li, "Accelerating sparse attention with a reconfigurable non-volatile processing-in-memory architecture," in *DAC*, 2023, pp. 1–6.

[10] T. Liu, X. Liu, S. Huang, L. Shi, Z. Xu, Y. Xin, Q. Yin, and X. Liu, "Sparse-tuning: Adapting vision transformers with efficient fine-tuning and inference," *arXiv preprint arXiv:2405.14700*, 2024.

[11] X. Yang, B. Yan, H. Li, and Y. Chen, "Retransformer: Reram-based processing-in-memory architecture for transformer acceleration," in *ICCAD*, 2020, pp. 1–9.

[12] F. Tu, Z. Wu, Y. Wang, W. Wu, L. Liu, Y. Hu, S. Wei, and S. Yin, "Multcim: Digital computing-in-memory-based multimodal transformer accelerator with attention-token-bit hybrid sparsity," *IEEE J. Solid-State Circuits*, 2023.

[13] A. Khan, Z. Rauf, A. Sohail, A. Rehman, H. Asif, A. Asif, and U. Farooq, "A survey of the vision transformers and its cnn-transformer based variants. arxiv 2023," *arXiv preprint arXiv:2305.09880*, 2023.

[14] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, pp. 740–755.

[15] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ade20k dataset," in *CVPR*, 2017, pp. 633–641.

[16] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *ECCV*, 2012, pp. 746–760.

[17] F. Tu, Z. Wu, Y. Wang, L. Liang, L. Liu, Y. Ding, L. Liu, S. Wei, Y. Xie, and S. Yin, "Trancim: Full-digital bitline-transpose cim-based sparse transformer accelerator with pipeline/parallel reconfigurable modes," *IEEE J. Solid-State Circuits*, vol. 58, no. 6, pp. 1798–1809, 2022.

[18] X. Fu, Q. Ren, H. Wu, F. Xiang, Q. Luo, J. Yue, Y. Chen, and F. Zhang, "P^{3}vit: A cim-based high-utilization architecture with dynamic pruning and two-way ping-pong macro for vision transformer," *IEEE Trans. Circuits Syst. I Regul. Pap*, 2023.