Efficiency Follows Global-Local Decoupling

Zhenyu Yang[1,3], Gensheng Pei[2*], Tao Chen[1,3], Yichao Zhou[1], Tianfei Zhou[4], Yazhou Yao[1,3*], Fumin Shen[5]
[1]Nanjing University of Science and Technology, [2]Department of Electrical and Computer Engineering, Sungkyunkwan University
[3]State Key Laboratory of Intelligent Manufacturing of Advanced Construction Machinery, [4]Beijing Institute of Technology
[5]University of Electronic Science and Technology of China, [*]Corresponding authors
https://github.com/NUST-Machine-Intelligence-Laboratory/ConvNeur

Abstract

*Modern vision models must capture **image-level context** without sacrificing **local detail** while remaining computationally affordable. We revisit this tradeoff and advance a simple principle: **decouple** the roles of global reasoning and local representation. To operationalize this principle, we introduce **ConvNeur**, a two-branch architecture in which a lightweight neural memory branch aggregates global context on a compact set of tokens, and a locality-preserving branch extracts fine structure. A learned gate lets global cues modulate local features without entangling their objectives. This separation yields subquadratic scaling with image size, retains inductive priors associated with local processing, and reduces overhead relative to fully global attention. On standard classification, detection, and segmentation benchmarks, ConvNeur matches or surpasses comparable alternatives at similar or lower compute and offers favorable accuracy versus latency trade-offs at similar budgets. These results support the view that efficiency follows global-local decoupling.*

1. Introduction

Recent vision models are increasingly asked to do two things at once: reason over the entire image and preserve fine spatial detail. High-level tasks such as recognition and detection benefit from image-level context, since objects are often defined by scene layout, co-occurrence, or long-range shape cues. Meanwhile, many decisions occur at the pixel or patch level, where edges, textures, and thin structures must remain intact. This creates a fundamental tension: expanding the receptive field to "*see the whole*" typically relies on heavier global interactions, whereas keeping features locally faithful favors lightweight, inductive operations that stay close to the image grid. As resolutions rise, heads become denser, and deployment budgets stay tight, the question is not only how to model global information, but how to do so without eroding locality or blowing up compute.

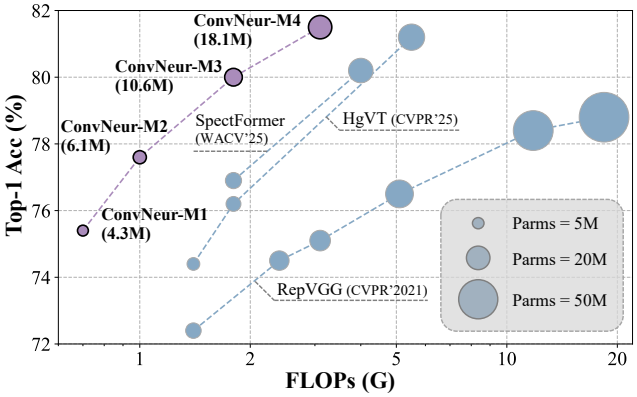

Figure 1. **Efficiency v.s. accuracy on the ImageNet-1K benchmark.** The proposed ConvNeur family establishes a new state-of-the-art frontier, delivering superior TOP-1 Acc while demanding significantly fewer FLOPs and parameters than existing methods.

To address this "*see the whole, keep the detail*" requirement, recent work has largely turned to Transformer-style global attention [19, 25, 53, 82]. Its cost, however, still grows roughly *quadratically* with resolution, which makes high-resolution or dense settings expensive. Windowed or sparse variants [37, 44] lower the cost but reintroduce locality and no longer view the full image. Convolutional models [3, 4, 18, 46, 52] remain efficient and preserve translation-friendly priors, yet their global context is limited or arrives late. Hybrid designs [13, 87] try to combine the two, but global and local computation are usually carried out in the same feature space and at similar resolution, so the two roles remain coupled and the model effectively pays for both.

We observe that the main source of cost inflation is not global modeling itself, but the fact that global reasoning and local representation are often entangled in a single pathway. Once a feature stream is required to see the whole image, preserve fine-grained structure, and stay within a tight FLOP budget, width, spatial resolution, and interaction range start to compete and none of them is optimal. This suggests a different design principle: learn global and local separately, then let the global path modulate the local

one instead of replacing it. A global path that only needs to guide can operate on compressed tokens and in chunked form, which keeps the computation subquadratic, while the local path can retain the inductive priors that make convolutions effective. In short, *efficiency follows global-local decoupling*, not the other way around.

We realize this principle in **ConvNeur**, a two-branch vision architecture. The first branch is a locality-preserving path that follows the convolutional paradigm and focuses on edges, textures, and small structures and does not attempt to solve global reasoning. In parallel, a global branch bottlenecks the features to a smaller channel dimension, chunks the features into a sequence, and feeds each chunk into a neural memory module that aggregates image-level context in a blockwise manner. The chunkwise readouts are reassembled to the spatial layout and lifted to the original channel size to produce a global context map. A learned gate uses this map to modulate the local features, so the global branch acts as guidance while the local branch keeps the image priors. This differs from typical attention or hybrid designs in three ways. First, the global branch is not a full-width attention layer but a neural memory operating on compressed and chunked tokens, which makes the cost grow *subquadratically* with image size. Second, fusion is done by gated modulation rather than naive concatenation or addition, so global signals do not wash out local evidence. Third, because the two roles are structurally decoupled, we can budget and compress the global branch independently of the local one, which is exactly what high-resolution vision needs. We evaluate ConvNeur on ImageNet-1K [15] classification, COCO 2017 [42] detection, and ADE20K [81] segmentation. Under comparable FLOPs and parameters, our models surpass recent efficient vision backbones, and the same module improves downstream tasks, supporting our claim that decoupling global and local leads to better accuracy-efficiency trade-offs. Our main contributions are summarized as follows:

- We identify global-local decoupling as a simple but effective route to efficiency in modern vision models.
- We introduce ConvNeur, a two-branch network that performs chunked neural memory aggregation on a compact global branch and uses learned gating to modulate a locality-preserving convolutional branch.
- On classification, detection, and segmentation tasks, the proposed ConvNeur achieves better accuracy-efficiency trade-offs than comparable convolutional and attention-based alternatives under similar budgets.

2. Related Works

Vision Transformers & Efficient Global Attention. Vision Transformers (ViT) [19] treated an image as a sequence of patches and applied fully global self-attention as the core vision operator, and DeiT [61] showed that with a stronger training recipe this paradigm is practical without massive external data. The limitation is structural, since all-to-all attention grows roughly quadratically with the number of tokens. Swin Transformer [44, 45] controlled this cost by restricting attention to local windows and shifting them across layers so that information can still flow globally. Follow-up windowed or hierarchical variants such as HEAL-Swin [7] and SHViT [80] further tailored this design for large fields of view or for better memory usage. Another line [41, 85] improves efficiency by reducing the active token set and learning sparse or adaptive attention patterns instead of enlarging windows. More recently, linearized and state-space vision backbones, including Vision Mamba [32, 38, 43, 86] and Vision RWKV [20, 77], replace quadratic attention with recurrent [54–56] or SSM-style updates [14, 26] to obtain long-range dependencies at near-linear cost. Across these directions, global interactions are made cheaper by being more local, more sparse, or more sequential, yet global and local information are still propagated in the same feature stream. Our approach is complementary: we place global reasoning on a compressed, chunked path and let it modulate a separate locality-preserving path, removing the coupling rather than only approximating attention.

Convolutional & Hybrid Vision Backbones. Early CNNs [34, 59, 60] struggle to capture image-level context due to their inherently local receptive fields. ConvNeXt [46, 70] addressed this with depthwise convolutions, layer-wise normalization, and later GRN and improved scaling. Large kernel models such as RepLKNet [18] further enlarged the receptive field to approximate global context, while LSNet [64] paired a large-context view with a detail-preserving path. In parallel, hybrid models like CoAtNet [13] and [87] used convolutions for local structure and attention or token mixing for global interactions. These approaches improve recognition, but global and local computation are still carried out in the same feature space, so the two remain coupled. Our method keeps the "conv for local, another operator for global" idea, but explicitly separates the two paths and fuses them only through a learned gate, which allows the global branch to be budgeted independently.

Memory-Augmented Models. Recent vision works [1, 76, 84] began to attach explicit memory to the network, especially for video and open-world settings. RMem [83] keeps a restricted memory bank and reads past object cues back into the current frame, while CUTIE [11] performs object-level memory reading to restore instance consistency over time. CMeRT [50] uses a context-enhanced encoder and a memory-refined decoder to exploit near-past features. Earlier memory-augmented models in vision, such as Non-local Networks [68], GCNet [6], and CBAM [69], also aggregate global signals first and then modulate local features. Video memory networks like STM [48] and AOT [75] keep a spatiotemporal memory that is read to guide the

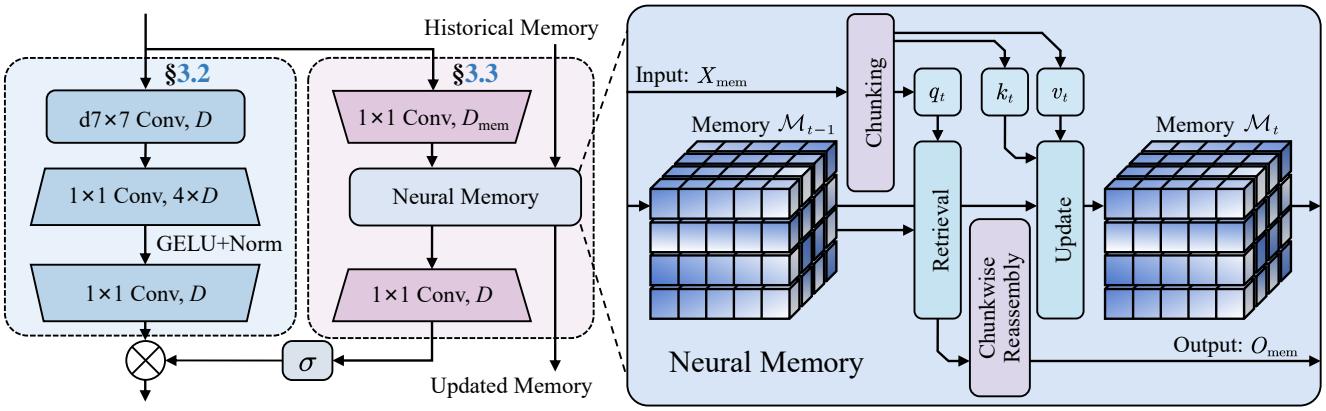

Figure 2. **The overall architecture of ConvNeur.** On the left, a locality-preserving convolutional branch extracts fine-grained features. In parallel, a compressed global branch performs chunked neural memory aggregation, and produces a gating map to modulate the local features. On the right, we show the neural-memory module: each chunk is linearly mapped to $\{q_t, k_t, v_t\}$. q_t reads from the previous memory state $\mathcal{M}_{t-1}$, while k_t and v_t compute a surprise loss to update the memory to $\mathcal{M}_t$.

current prediction. Titans [2] goes one step further by organizing memory into chunks, using surprise-driven updates, and treating memory as a learnable component rather than a fixed repository. Our work follows the same spirit of separating fast locality-preserving computation from a slower context-accumulating component, but we do this spatially in an explicit two-branch architecture.

3. Method

3.1. Overview Architecture of ConvNeur

Given an intermediate feature map $X \in \mathbb{R}^{C\times H\times W}$, our objective is to model image-level context and local structure separately and to let the former modulate the latter. The guiding principle is that efficiency follows global-local decoupling: global reasoning and locality-preserving representation should not compete for width, spatial resolution, or interaction range inside the same pathway.

As shown in Fig. 2, we design ConvNeur as a two-branch architecture. The locality-preserving convolutional branch follows the modern convolutional paradigm [46] and is responsible only for edges, textures, and other fine-scale patterns; it keeps the inductive priors of CNNs and does not attempt to solve global reasoning. In parallel, the compressed global-memory branch first projects X to a lower-dimensional memory space, converts the spatial map to a sequence, partitions it into fixed-size chunks, and for each chunk performs a *retrieval-update-reassemble* cycle that aggregates image-level context at subquadratic cost. This branch is inspired by recent neural memory [2] formulations, so we adopt the term *neural memory* and the surprise-driven update mechanism, but we place it inside a spatially decoupled vision framework so that global context is produced on a lightweight path and only used to gate the convolutional features rather than replacing them.

3.2. Locality-Preserving Convolutional Branch

The first branch keeps feature processing local. It applies a depthwise convolution with a relatively large kernel to gather nearby context, followed by lightweight channel mixing and normalization. This path is meant to capture edges, textures and small objects and to retain the convolutional inductive bias. We denote its output by F_{loc}. Because global reasoning is handled elsewhere, this branch does not expand its interaction range and therefore avoids the width-resolution-context entanglement discussed above.

3.3. Global Memory Branch

The global branch is built around an optimization-style view. The outer part of the network learns how global context should be queried and how memory updates should be parametrized, while an inner, chunk-wise procedure actually updates the memory state for the current image. This yields a structure that is close to bilevel optimization [22, 24, 57], where slow variables specify the update rule and fast variables adapt to the current example, and also to online or block-coordinate methods that process one block at a time to control complexity. Processing the image in blocks is exactly what keeps the cost below quadratic.

Bottlenecked tokenization. We first apply a pointwise projection to map the feature map X into a lower-dimensional memory space, obtaining $X_{\text{mem}} \in \mathbb{R}^{C_m\times H\times W}$, $C_m < C$. This gives the global branch a compressed representation on which reading and updating the memory is more efficient, while preserving the spatial layout for later reassembly.

Chunking. We flatten X_{mem} along the spatial dimensions to obtain $S \in \mathbb{R}^{(HW)\times C_m}$ and then partition this sequence into fixed-size chunks $\{S_t\}_{t=1}^{T}$, each of length L. Processing the image in chunks can be viewed as an online or block-coordinate pass over the spatial sequence. At each step the

Table 1. **ConvNeur variants used in our experiments.** Depths and dims are listed from high to low resolution. Memory dimension is matched to the stage width, and the chunk size is shared.

Model	Depths	Dims	Memory Dim C_m	Chunk Size L	FLOPs (G)
ConvNeur-M1	[2, 2, 6, 2]	[40, 80, 160, 320]	80	196	0.71
ConvNeur-M2	[2, 2, 6, 2]	[48, 96, 192, 384]	96	196	1.01
ConvNeur-M3	[2, 2, 6, 2]	[64, 128, 256, 512]	128	196	1.77
ConvNeur-M4	[2, 2, 8, 2]	[80, 160, 320, 640]	160	196	3.06

memory only receives a small subset of tokens, which prevents the global branch from operating on all tokens at once.

Memory Retrieval & Update. For each chunk $S_t \in \mathbb{R}^{L\times C_m}$, we produce three sequences with linear projection:

$$q_t = W_q S_t, \quad k_t = W_k S_t, \quad v_t = W_v S_t, \tag{1}$$

where $\{W_q, W_k, W_v\}$ are learned together with the backbone and determine how the memory is queried and how it is updated. We first use q_t to read from the current memory state $\mathcal{M}_{t-1}$ and obtain a chunk-level global context $\hat{y}_t$. Given the current memory, this step selects the global information that is most useful for this part of the image:

$$\hat{y}_t = \mathcal{M}_{t-1}(q_t). \tag{2}$$

We then define for this chunk a reconstruction objective:

$$\mathcal{L}_t = \|\mathcal{M}_{t-1}(k_t) - v_t\|_2^2. \tag{3}$$

A set of learnable update generators converts this loss into an adaptive step size, a momentum coefficient, and a decay factor, and applies them to obtain the next memory state:

$$\mathcal{M}_t = \mathcal{U}(\mathcal{M}_{t-1}, \nabla_{\mathcal{M}}\mathcal{L}_t). \tag{4}$$

Chunks that are harder to reconstruct produce larger updates, which follows the surprise-driven idea in recent neural memory work [2]. Because we repeat this *Retrieval-Update* pattern for every chunk and for every head, different chunks can maintain partially independent fast weights and the memory behaves like a small ensemble.

Chunkwise reassembly. The retrieved contexts $\{\hat{y}_t\}_{t=1}^{T}$ are concatenated in the original token order to form a full sequence of length HW, which is then reshaped back to the spatial layout $O_{\text{mem}} \in \mathbb{R}^{C_m\times H\times W}$ to ensure that, although global modeling is carried out chunk by chunk, the final output is aligned with the original feature map X_{mem}.

Global map. After chunkwise reassembly, we restore the channel dimension with another pointwise convolution to obtain the global context map $G \in \mathbb{R}^{C\times H\times W}$.

The total cost of the global branch can be written as:

$$\mathcal{O}(CC_mHW) + \mathcal{O}(T \cdot \text{mem}(L, C_m)). \tag{5}$$

Since the memory dimension C_m and the chunk size L are small and fixed in our models, the overall cost grows roughly linearly with the number of spatial locations.

3.4. Gated Global-to-Local Fusion

We pass the global context map through a sigmoid function (σ) to obtain a spatial gate $A = \sigma(G)$, and use it to modulate the local features pointwise:

$$F_{\text{out}} = X + \texttt{DropPath}(A \otimes F_{\text{loc}}). \tag{6}$$

This can be viewed as amortized conditioning: the global branch first infers a context that is specific to the current image and then converts it into a mask that tells the local branch what to emphasize or suppress. The global signal therefore guides rather than overwrites the local representation, so convolutional priors are preserved. Compared with SE [36], CBAM [69] or non-local [68], which aggregate global information once and broadcast back, our gate is produced from an online-updated memory over spatial chunks, and thus reflects the current state of the global branch.

3.5. Model Variants and Scaling

To show that our design is not limited to a single width-depth configuration, we instantiate ConvNeur in several sizes. All variants share the same template: four stages with downsampling, a locality preserving branch in every block, and a global memory branch inserted at the start of each stage. As listed in Tab. 1, we scale the models by widening the stage channels and proportionally increasing the memory dimension so that the global path stays lightweight relative to the local path. We keep the chunk size fixed, which makes the memory cost predictable across variants.

4. Experiments

We evaluate the proposed ConvNeur on public benchmarks for image classification [15], object detection [42], and semantic segmentation [81] under standard protocols with matched compute and parameter budgets to rigorously and fairly assess effectiveness, scalability, and efficiency.

4.1. Image Classification

Settings. We train on ImageNet-1K with 224×224 random crops for 300 epochs. The optimizer is AdamW [47] with learning rate 4e-3 at a global batch size of 4096, epsilon 1e-8, weight decay 0.05 with cosine decay to a final value when specified, no gradient clipping, layer decay 1.0, and minimum learning rate 1e-6. The learning rate is warmed up for 20 epochs. When warmup steps are provided, the step schedule overrides the epoch count. Training uses 8 NVIDIA RTX 3090 GPUs. Data augmentation follows the widely used DeiT [61] style recipe: RandAugment, Mixup, CutMix, Random Erasing, and label smoothing.

Results. Tab. 2 shows the proposed ConvNeur advances the accuracy-efficiency frontier across budgets with only a few reference points needed. At the small end, M1 reaches

Table 2. **Comparison results of vision backbone performance on the ImageNet-1K [15] dataset pre-training.**

Method	Venue	Params (M) ↓	FLOPs (G) ↓	Top-1 Acc ↑
RepVGG-A0 [17]	CVPR 2021	8.3	1.4	72.4
DeiT-T [61]	ICML 2021	5.7	1.1	72.2
ConViT-Ti [21]	ICML 2021	5.7	1.0	73.1
CrossViT-Ti [8]	ICCV 2021	6.9	1.6	73.4
PVT-T [66]	ICCV 2021	13.2	1.9	75.1
GFNet-Ti [58]	NeurIPS 2021	7.0	1.3	74.6
PoolFormer-S7 [79]	CVPR 2022	8.6	1.1	73.0
ViG-Ti [30]	NeurIPS 2022	7.1	1.3	73.9
DependencyViT-Lite-T [16]	CVPR 2023	6.2	0.8	73.7
ViHGNN-Ti [31]	ICCV 2023	8.2	1.8	74.3
MobileAtt-DeiT-T [78]	ICML 2024	5.7	1.2	73.3
SLAB-DeiT-T [27]	ICML 2024	6.2	1.3	73.6
Agent-DeiT-T [29]	ECCV 2024	6.0	1.2	74.9
QuadMamba-Li [72]	NeurIPS 2024	5.4	0.8	74.2
KAT-T [74]	ICLR 2025	5.7	1.1	74.6
VRWKV-T [20]	ICLR 2025	6.2	1.2	75.1
HgVT-Mi [23]	CVPR 2025	5.8	1.4	74.4
ConvNeur-M1(ours)	-	**4.3**	**0.7**	**75.4**
RepVGG-A2 [17]	CVPR 2021	25.5	5.1	76.5
ConViT-Ti+ [21]	ICML 2021	10.0	2.0	76.7
CrossViT-9 [8]	ICCV 2021	8.8	2.0	77.1
PoolFormer-S12 [79]	CVPR 2022	12.0	1.8	77.2
EdgeViT-XS [49]	ECCV 2022	6.7	1.1	77.5
RIFormer-S12 [65]	CVPR 2023	12.0	1.8	76.9
MobileOne-S2 [62]	CVPR 2023	7.8	1.3	77.4
SLAB-PVT-T [27]	ICML 2024	13.4	1.9	76.0
Vim-Ti [86]	ICML 2024	7.0	1.5	76.1
LocalVim-T [38]	ECCV 2024	8.0	1.5	75.8
HgVT-Ti [23]	CVPR 2025	7.7	1.8	76.2
SpectFormer-T [51]	WACV 2025	9.0	1.8	76.9
ConvNeur-M2(ours)	-	**6.1**	**1.0**	**77.6**
RepVGG-B1g2 [17]	CVPR 2021	41.4	8.8	77.8
DeiT-S [61]	ICML 2021	22.1	4.6	79.9
PVT-S [66]	ICCV 2021	24.5	3.8	79.8
GFNet-XS [58]	NeurIPS 2021	16.0	2.9	78.6
FasterNet-T2 [9]	CVPR 2023	15.0	1.9	78.9
MobileOne-S4 [62]	CVPR 2023	14.8	3.0	79.4
Flatten-PVTv2-B1 [28]	CVPR 2023	19.2	2.2	79.5
GC ViT-XXT [33]	ICML 2023	12.0	2.1	79.9
Agent-PVT-T [29]	ECCV 2024	11.6	2.0	78.4
ConvNeur-M3(ours)	-	**10.6**	**1.8**	**80.0**
RepVGG-B3g4 [17]	CVPR 2021	75.6	16.1	80.2
CrossViT-S [8]	ICCV 2021	26.7	5.6	81.0
PVT-M [66]	ICCV 2021	44.2	6.7	81.2
GFNet-B [58]	NeurIPS 2021	43.0	7.9	80.7
PoolFormer-S24 [79]	CVPR 2022	21.0	3.4	80.3
ViG-S [30]	NeurIPS 2022	22.7	4.5	80.4
RIFormer-S24 [65]	CVPR 2023	21.0	3.4	80.3
Vim-S [86]	ICML 2024	26.0	5.3	80.5
LocalVim-S [38]	ECCV 2024	28.0	4.8	81.0
VRWKV-S [20]	ICLR 2025	23.8	4.6	80.1
KAT-S [74]	ICLR 2025	22.1	4.4	81.2
HgVT-S [23]	CVPR 2025	22.9	5.5	81.2
SpectFormer-XS [51]	WACV 2025	20.0	4.0	80.2
ConvNeur-M4(ours)	-	**18.1**	**3.1**	**81.5**

75.4 with 0.7 GFLOPs and 4.3M parameters, outperforming PVT-T [66] at 75.1 with 1.9 GFLOPs while using less than half the compute. In the mid range, M3 attains 80.0 with 1.8 GFLOPs and 10.6M parameters, matching or exceeding DeiT-S [61] at 79.9 with 4.6 GFLOPs and PVT-S at 79.8 with 3.8 GFLOPs at a much lower cost. At the high end, M4 delivers 81.5 with 3.1 GFLOPs and 18.1M parameters, surpassing CrossViT-S [8] at 81.0 with 5.6 GFLOPs and PVT-M at 81.2 with 6.7 GFLOPs. Across all scales, ConvNeur also outperforms pure convolutional baselines such as RepVGG [17], from A0 at 72.4 with 1.4 GFLOPs to B3g4 at 80.2 with 16.1 GFLOPs, highlighting the importance of injecting explicit global context rather than deepening or widening a single local stream. These gains follow the design. The locality-preserving branch maintains translation-friendly priors and fine detail. The global branch aggregates scene-level cues in a compact memory space on chunked tokens, so the cost of global reasoning remains modest. A learned gate then modulates the local stream rather than overwriting it, avoiding feature drift while still exploiting global evidence. Because the memory path is bounded by its bottleneck width and chunk length, its cost grows slowly as models scale, which explains why ConvNeur stays ahead of pooling-based token mixers, frequency-domain globalizers, and hierarchical Transformers that operate global interactions at full width.

Table 3. **Comparison results of object detection and instance segmentation performance on the COCO 2017 [42] validation set using Cascade Mask R-CNN [5] under the 1x schedule.** All methods share the same detector, training recipe, and data pipeline.

Task	Method	AP↑	AP_{50}↑	AP_{75}↑
Object Detection	ResNet50 [34]	38.0	58.6	41.4
	PVT-T [66]	36.7	59.2	39.3
	PVTv2-B0 [67]	38.2	60.5	40.7
	BiViT [35]	40.8	59.2	44.1
	ConvNeur-M2 (ours)	**41.2**	**60.9**	**44.5**
	ResNet101 [34]	40.4	61.1	44.2
	ResNeXt101-32x4d [73]	41.9	62.5	45.9
	PVT-S [66]	40.4	62.9	43.8
	PVTv2-B1 [67]	41.8	64.3	45.9
	ConvNeur-M3 (ours)	**42.4**	**64.8**	**46.1**
Instance Segmentation	ResNet50 [34]	34.4	55.1	36.7
	PVT-T [66]	35.1	56.7	37.3
	PVTv2-B0 [67]	36.2	**57.8**	**38.6**
	BiViT [35]	35.7	56.5	38.2
	ConvNeur-M2 (ours)	**36.4**	57.7	**38.6**
	ResNet101 [34]	36.4	57.7	38.8
	ResNeXt101-32x4d [73]	37.5	59.4	40.2
	PVT-S [66]	37.8	60.1	40.3
	PVTv2-B1 [67]	38.8	**61.2**	41.6
	ConvNeur-M3 (ours)	**39.0**	61.0	**41.8**

4.2. Object Detection

Settings. We train on COCO 2017 [42] and report results on the validation set. We adopt Cascade Mask R-CNN [5] in MMDetection [10] as the detector and train with the 1x schedule. Images are resized with the shorter side at 800 and the longer side capped at 1333. Training uses 4 NVIDIA RTX 3090 GPUs with 2 images per GPU, giving a total batch size of 8. The 1x schedule runs for 12 epochs with a 500 iteration linear warm-up, followed by

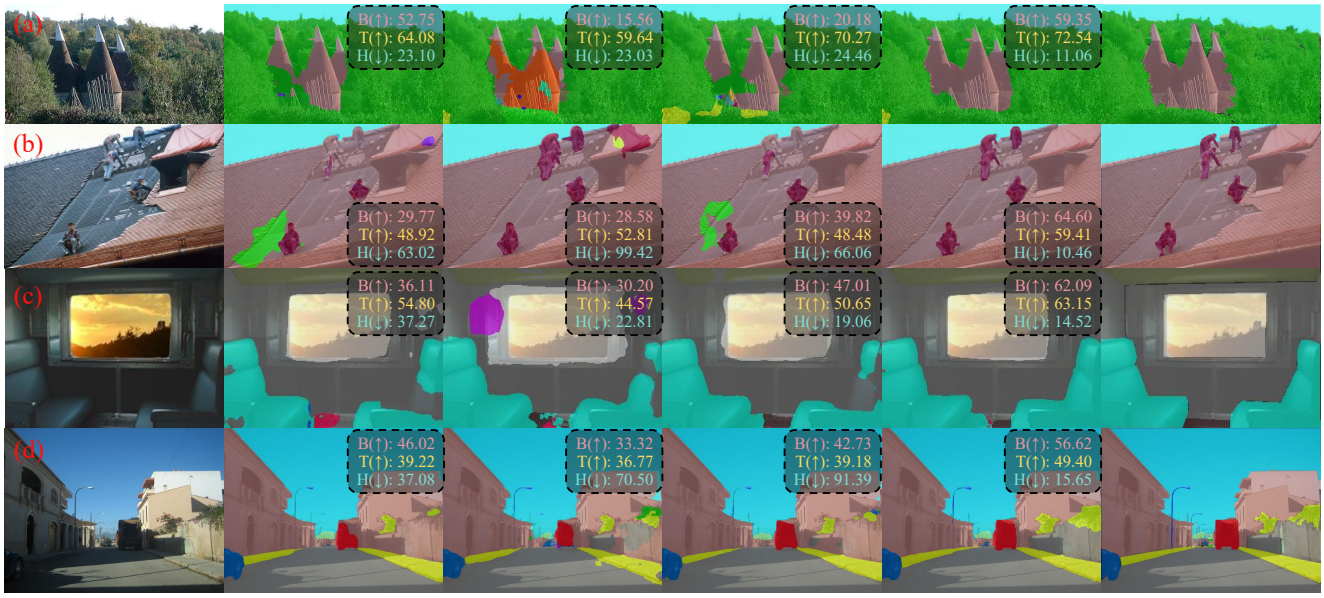

Figure 3. **Qualitative results on the ADE20K [81] dataset.** All examples are from the validation set. In the figure, B represents Boundary F1 score, T represents Trimap-based mIoU, and H represents Hausdorff distance.

multi-step decay at epochs 8 and 11 with gamma 0.1. The optimizer is SGD with a learning rate of 0.02, momentum 0.9, and weight decay 1e-4. All other components follow MMDetection defaults, and both training and inference are single-scale without test-time augmentation. For the global branch, the higher input resolution in COCO 2017 motivates a larger chunk size of 4096 so that each memory update aggregates more tokens, which stabilizes the surprise-driven updates. We also reduce the inner-loop step size from 1 to 1e-3 to discourage overly frequent memory writes under sparse box and mask supervision.

Results. As shown in Tab. 3, ConvNeur improves both boxes and masks while keeping the detector unchanged. With the small backbone, ConvNeur-M2 reaches 41.2 AP and 44.5 AP_{75} for detection, surpassing ResNet-50 [34] at 38.0 and PVT-T [66] at 36.7. The gain comes from scene-level cues retrieved by the global memory that help disambiguate scale and context, while the gate preserves edges and small parts so localization tightens rather than blurs. Upder the same setting, M2 attains 36.4 AP mask, stronger than ResNet-50 at 34.4 and PVT-T at 35.1, indicating that modulated global guidance sharpens boundaries in crowded regions. Scaling the backbone reinforces the trend. ConvNeur-M3 delivers 42.4 AP, exceeding ResNeXt-101-32x4d [73] at 41.9 and PVTv2-B1 [67] at 41.8. The improvement is consistent with the per-stage placement of memory, which aligns global guidance with the FPN pyramid and provides cross-scale hints without inflating head compute. For masks, M3 reaches 39.0 AP mask, higher than PVTv2-B1 at 38.8, suggesting that chunked memory helps separate overlapping instances and maintain fine contours. Across both scales, ConvNeur shows higher AP_{75} at comparable AP_{50}, which points to better localization quality rather than only higher recall, matching the design goal of decoupled global-local modeling with learned gating.

4.3. Semantic Segmentation

Settings. We train on ADE20K [81] with Semantic FPN [40] and UperNet [71] in MMSegmentation [12]. Images are resized so that the long side is 2048 and the short side is 512. During training we apply a random crop of 512^2. Experiments run on 4 NVIDIA RTX 3090 GPUs with 4 images per GPU for a total batch size of 16. Optimization uses AdamW [47] with base learning rate 6e-5, betas 0.9 and 0.999, and weight decay 0.01. We also apply gradient clipping with max norm 1.0, and use layer-wise learning rate multipliers of 10 for the neck and decode head. The schedule warms up linearly for 1500 iterations, then follows a polynomial decay until 80k iterations.

Quantitative Results. Tab. 4 reports ADE20K under identical heads and pipelines. With Semantic FPN, ConvNeur-M2 attains 39.17 mIoU at 106 GFLOPs, outperforming ResNet-50 [34] at 36.59 with 183 GFLOPs and PVT-T [66] at 36.57 with 158 GFLOPs, while ConvNeur-M3 pushes to 41.42 at 123 GFLOPs. The gains reflect scene cues retrieved by the global memory that guide the local stream toward complete regions and clean boundaries, which a purely local backbone with S-FPN tends to miss. With UperNet, ConvNeur-M4 reaches 45.72 mIoU and 57.61 mAcc at 908 GFLOPs, exceeding Swin-T [44] at 44.51 with

Table 4. **Comparison results of semantic segmentation performance on the ADE20K [81] validation set.** The FLOPs are computed with an input image at the resolution of 512×2048. In the table, S-FPN is short for the Semantic FPN [40] model.

Head	Method	FLOPs(G) ↓	mIoU ↑	mAcc ↑
S-FPN [40]	ResNet50 [34]	183	36.59	46.85
	PVT-T [66]	158	36.57	46.72
	FLatten-PVT-T [28]	169	37.21	48.95
	ConvNeur-M2(ours)	**106**	39.17	49.85
	ConvNeur-M3(ours)	123	**41.42**	**52.35**
UperNet [71]	ResNet50 [34]	953	42.10	52.91
	DeiT-S [61]	1217	43.92	54.33
	Swin-T [44]	945	44.51	55.61
	FLatten-Swin-T [28]	946	44.82	57.01
	ConvNeur-M4(ours)	**908**	**45.72**	**57.61**

945 GFLOPs and Flatten-Swin-T [28] at 44.82 with 946 GFLOPs, and also surpassing DeiT-S [61] at 43.92 with 1217 GFLOPs. Improvements are strongest at tighter budgets, consistent with keeping global reasoning in a compact memory space. The concurrent rise in mAcc indicates fewer class confusions in cluttered scenes, supporting the view that decoupling global reasoning from local representation benefits dense prediction without inflating head compute.

Qualitative Results. Fig. 3 compares ConvNeur with recent backbones on four challenging scenes selected by low mIoU of prior methods, including (a) occlusion where hillside trees partially hide houses, (b) background clutter where colors closely match the target, (c) high illumination contrast area, and (d) cast shadows that erase part of the object semantics. ConvNeur recovers more complete masks and cleaner edges across all panels, proving that the locality-preserving path keeps fine structures intact, and the global memory produces a context map that suppresses distractors and fills in missing parts by borrowing evidence from distant, unoccluded regions. This global-to-local guidance is especially visible on thin facades behind foliage, furniture against similarly colored walls, reflective surfaces, and shadowed object halves. We also annotate three boundary metrics next to each prediction, including Boundary F1 score, Trimap-based mIoU, and Hausdorff distance. ConvNeur improves all three, suggesting that global guidance together with a decoupled global and local design promotes crisp, well-aligned boundaries. These examples show that separating global reasoning from local representation improves instance completeness and boundary fidelity in complex scenes.

4.4. Ablation Study

We ablate on ConvNeur-M3 using ImageNet-1K with the same schedule as in Sec. 4.1. We analyze the contribution of each core component and rigorously evaluate the model's computational efficiency, with results summarized in Tab. 5.

Global Memory (*cf.* ❶ ❷). We compare three settings: a local-only branch (❶), adding neural memory once at the beginning of each stage, and adding neural memory at every block (❷). Relative to the local-only baseline, the per-stage memory lifts Top-1 from 78.2 to 80.0 while adding only 0.4 GFLOPs. This confirms the value of decoupling: the local path preserves inductive priors, and the memory supplies image-level cues that a single stream cannot capture under the similar budget. Activating memory at every block reaches 80.4 but requires 2.3 GFLOPs, which is a small gain for a clear increase in cost. The modest improvement aligns with the high redundancy across adjacent blocks within a stage, where global context changes slowly and the gate acts as a slowly varying conditioner. In contrast, stage boundaries introduce larger shifts due to downsampling and width changes, so a single update per stage provides most of the benefit. We therefore adopt the per-stage design by default.

Table 5. **Ablation studies on the ImageNet-1K [15] dataset.** Rows highlighted in purple indicate the default configuration. The highest value is highlighted in **bold**.

Experiment	Varient #	Method	FLOPs (G) ↓	Top-1 Acc ↑
Component				
Global Memory	❶	Local-only	**1.4**	78.2
	-	**Per-stage (ours)**	1.8	80.0
	❷	Per-layer	2.3	**80.4**
Global Branch Type	❸	CBAM [69]	**1.5**	78.4
	❹	Non-local [68]	3.0	79.1
	❺	Self Attention [63]	1.8	79.8
	❻	IR-RWKV [39]	2.0	77.6
	❼	Titans [2]	3.1	79.6
	-	**Ours**	1.8	**80.0**
Fusion Mechanism	❽	Addition	**1.8**	79.6
	❾	Concatenate	1.9	79.5
	-	**Gating (ours)**	**1.8**	**80.0**
Hyperparameter				
Mem Dim C_m	①	64	**1.5**	78.6
	-	**128 (ours)**	1.8	80.0
	②	256	2.7	**80.3**

Global Branch Type (*cf.* ❸ ❹ ❺ ❻ ❼). We next ask what form the global branch should take. For a fair comparison, all variants use the same bottleneck scaffold and the same global dimension at the same insertion points. CBAM [69] is light at 1.5 GFLOPs but reaches only 78.4, a gain of 0.2 over the local-only baseline, which indicates that reweighting alone does not provide persistent global context. Non-local [68] reaches 79.1 with 3.0 GFLOPs. Its cost stems from building dense affinities across all spatial positions and applying the response matrix, which introduces quadratic compute and heavy memory traffic. Self attention [63] attains 79.8 at 1.8 GFLOPs, close to our budget, yet its activation memory and attention maps grow quadratically with tokens, so the footprint rises quickly at higher resolutions. The linear RWKV variant [39] yields 77.6 at 2.0 GFLOPs. This likely reflects two factors: recurrence imposes a scan order over the spatial sequence and breaks isotropy in 2D, and a single low-dimensional recurrent state

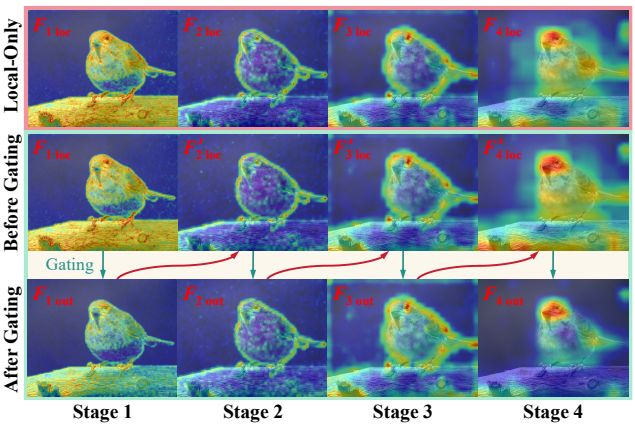

Figure 4. **Stage-wise visualization of global to local modulation.** Columns correspond to the four stages from shallow to deep. Row 1 shows local-only features F_{loc} produced by the locality-preserving branch without the global path. Row 2 shows the local features in ConvNeur before gating. Row 3 shows the gated features F_{out}. Green arrows mark the gating step.

is a narrow conduit that tends to forget long-range cues across distant chunks. Our neural memory instead keeps multiple fast weights and retrieves chunk-level associations without a fixed ordering, which aligns better with the chunked setting. A Titans [2] style memory reaches 79.6 at 3.1 GFLOPs. Our design removes segmented attention and retains neural memory as the sole global mechanism, reducing FLOPs by more than 40% while slightly improving accuracy to 80.0. Overall, the proposed neural memory offers the most favorable balance under matched scaffolding.

Fusion Mechanism (*cf.* ❽ ❾). With the global branch fixed to neural memory, we compare three fusion schemes under matched settings. Simple addition reaches 79.6 at 1.8 GFLOPs. Concatenation reaches 79.5 at 1.9 GFLOPs and introduces 0.7M extra parameters due to the required pointwise reduction. Learned gating attains 80.0 at 1.8 GFLOPs. The small but consistent gain of gating likely comes from its multiplicative calibration, which preserves the residual local path and selectively amplifies or suppresses features per location and per channel. Additive fusion injects global activations directly into the residual stream and can blur fine structures by shifting feature statistics, especially when the global response is strong. Concatenation pays extra compute to learn a similar rescaling after the merge, yet the bottleneck leaves limited headroom for improvement. We therefore adopt gating as the default fusion.

Memory Dimension C_m (*cf.* ① ②). The global branch runs in a bottlenecked space, so C_m sets its compute share. Raising C_m from 64 to 128 increases FLOPs from 1.5 to 1.8 and lifts Top-1 from 78.6 to 80.0, which is a strong return for a small cost. Pushing to 256 raises FLOPs to 2.7 but reaches only 80.3, showing diminishing gains. Moreover, relative to variant ❷, which deepens the global path by applying memory at every block, simply widening to 256 yields less improvement at higher cost. We set $C_m = 128$ as the default since it offers the best balance of accuracy and compute.

4.5. Mechanism Visualization

We further probe how the global branch guides the local stream by visualizing three feature views per stage, as shown in Fig. 4, including local-only activations, pre-gate activations in ConvNeur, and the gated result. The influence map is the absolute change between pre-gate and post-gate features, summed over channels and upsampled to the image. The pattern is consistent across stages. The global gate works like a weighted selector that picks the truly informative parts of the local features from a scene-level viewpoint. Early layers show background suppression and cleaner edges. Thin structures and small parts gain contrast while textured clutter fades. Deeper layers become object-centric and highlight semantically relevant regions such as distinctive parts of the target (see $F'_{4\,\text{loc}} \rightarrow F_{4\,\text{out}}$ in Fig. 4). The gate amplifies evidence that supports the current hypothesis and attenuates distractors with similar color or texture. These observations support the design principle. Global reasoning and local representation are learned on separate paths, and a learned gate converts global cues into pixelwise modulation. This preserves local priors while sharpening boundaries and improving region completeness.

5. Conclusion

In this paper, we revisited the long-standing tension between *seeing the whole image* and *keeping local detail*, and argued for a simple design rule: learn global reasoning and local representation separately, then let global cues modulate the local stream. Building on this principle, we introduced **ConvNeur**, a two-branch architecture with a locality-preserving convolutional path and a compact global path that performs chunked neural memory retrieval and surprise-driven update, followed by learned gating. This decoupling keeps inductive priors intact, makes global aggregation lightweight, and turns global information into targeted guidance rather than a wholesale rewrite. Experiments on classification, detection, and segmentation show consistent accuracy-efficiency gains under comparable budgets, and ablations confirm that each piece matters: per-stage memory beats local-only baselines, neural memory is a stronger globalizer than common alternatives at matched cost, and gated modulation outperforms naive fusion. We see ConvNeur as a minimal template for efficient global-local modeling. Future work includes multi-scale and dynamic chunking, extending the memory to video and streaming settings, and refining the theory of the inner fast-weight updates. We hope this perspective encourages architectures that separate roles first and let efficiency follow.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62506169, 62472222), Natural Science Foundation of Jiangsu Province (No. BK20240080), National Defense Science and Technology Industry Bureau Technology Infrastructure Project (JSZL2024606C001).

References

[1] Ben Agro, Sergio Casas, Patrick Wang, Thomas Gilles, and Raquel Urtasun. Mad: Memory-augmented detection of 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1449–1460, 2025. 2

[2] Ali Behrouz, Peilin Zhong, and Vahab Mirrokni. Titans: Learning to memorize at test time. *arXiv preprint arXiv:2501.00663*, 2024. 3, 4, 7, 8

[3] Xinhao Cai, Qiuxia Lai, Yuwei Wang, Wenguan Wang, Zeren Sun, and Yazhou Yao. Poly kernel inception network for remote sensing detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27706–27716, 2024. 1

[4] Xinhao Cai, Liulei Li, Gensheng Pei, Tao Chen, Jinshan Pan, Yazhou Yao, and Wenguan Wang. Beyond frequency: Scoring-driven debiasing for object detection via blueprint-prompted image synthesis. In *International Conference on Learning Representations*, 2026. 1

[5] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: High quality object detection and instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(5):1483–1498, 2019. 5

[6] Yue Cao, Jiarui Xu, Stephen Lin, Fangyun Wei, and Han Hu. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 2

[7] Oscar Carlsson, Jan E Gerken, Hampus Linander, Heiner Spieß, Fredrik Ohlsson, Christoffer Petersson, and Daniel Persson. Heal-swin: A vision transformer on the sphere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6067–6077, 2024. 2

[8] Chun-Fu Richard Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 357–366, 2021. 5

[9] Jierun Chen, Shiu-hong Kao, Hao He, Weipeng Zhuo, Song Wen, Chul-Ho Lee, and S-H Gary Chan. Run, don't walk: chasing higher flops for faster neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12021–12031, 2023. 5

[10] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 5

[11] Ho Kei Cheng, Seoung Wug Oh, Brian Price, Joon-Young Lee, and Alexander Schwing. Putting the object back into video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3151–3161, 2024. 2

[12] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 6

[13] Zihang Dai, Hanxiao Liu, Quoc V Le, and Mingxing Tan. Coatnet: Marrying convolution and attention for all data sizes. *Advances in Neural Information Processing Systems*, 34:3965–3977, 2021. 1, 2

[14] Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. In *International Conference on Machine Learning*. PMLR, 2024. 2

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE, 2009. 2, 4, 5, 7

[16] Mingyu Ding, Yikang Shen, Lijie Fan, Zhenfang Chen, Zitian Chen, Ping Luo, Joshua B Tenenbaum, and Chuang Gan. Visual dependency transformers: Dependency tree emerges from reversed attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14528–14539, 2023. 5

[17] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13733–13742, 2021. 5

[18] Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11963–11975, 2022. 1, 2

[19] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 1, 2

[20] Yuchen Duan, Weiyun Wang, Zhe Chen, Xizhou Zhu, Lewei Lu, Tong Lu, Yu Qiao, Hongsheng Li, Jifeng Dai, and Wenhai Wang. Vision-rwkv: Efficient and scalable visual perception with rwkv-like architectures. In *International Conference on Learning Representations*, 2025. 2, 5

[21] Stéphane d'Ascoli, Hugo Touvron, Matthew L Leavitt, Ari S Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. In *International Conference on Machine Learning*, pages 2286–2296. PMLR, 2021. 5

[22] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR, 2017. 3

[23] Joshua Fixelle. Hypergraph vision transformers: Images are more than nodes, more than edges. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9751–9761, 2025. 5

[24] Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *International Conference on Machine Learning*, pages 1568–1577. PMLR, 2018. 3

[25] Anthony Fuller, Daniel Kyrollos, Yousef Yassin, and James Green. Lookhere: Vision transformers with directed attention generalize and extrapolate. *Advances in Neural Information Processing Systems*, 37:19683–19739, 2024. 1

[26] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*, 2024. 2

[27] Jialong Guo, Xinghao Chen, Yehui Tang, and Yunhe Wang. Slab: Efficient transformers with simplified linear attention and progressive re-parameterized batch normalization. In *International Conference on Machine Learning*, 2024. 5

[28] Dongchen Han, Xuran Pan, Yizeng Han, Shiji Song, and Gao Huang. Flatten transformer: Vision transformer using focused linear attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5961–5971, 2023. 5, 7

[29] Dongchen Han, Tianzhu Ye, Yizeng Han, Zhuofan Xia, Siyuan Pan, Pengfei Wan, Shiji Song, and Gao Huang. Agent attention: On the integration of softmax and linear attention. In *European Conference on Computer Vision*, pages 124–140. Springer, 2024. 5

[30] Kai Han, Yunhe Wang, Jianyuan Guo, Yehui Tang, and Enhua Wu. Vision gnn: An image is worth graph of nodes. *Advances in Neural Information Processing Systems*, 35:8291–8303, 2022. 5

[31] Yan Han, Peihao Wang, Souvik Kundu, Ying Ding, and Zhangyang Wang. Vision hgnn: An image is more than a graph of nodes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19878–19888, 2023. 5

[32] Ali Hatamizadeh and Jan Kautz. Mambavision: A hybrid mamba-transformer vision backbone. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 25261–25270, 2025. 2

[33] Ali Hatamizadeh, Hongxu Yin, Greg Heinrich, Jan Kautz, and Pavlo Molchanov. Global context vision transformers. In *International Conference on Machine Learning*, pages 12633–12646. PMLR, 2023. 5

[34] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 2, 5, 6, 7

[35] Yefei He, Zhenyu Lou, Luoming Zhang, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. Bivit: Extremely compressed binary vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5651–5663, 2023. 5

[36] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7132–7141, 2018. 4

[37] Youbing Hu, Yun Cheng, Anqi Lu, Zhiqiang Cao, Dawei Wei, Jie Liu, and Zhijun Li. Lf-vit: Reducing spatial redundancy in vision transformer for efficient image recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2274–2284, 2024. 1

[38] Tao Huang, Xiaohuan Pei, Shan You, Fei Wang, Chen Qian, and Chang Xu. Localmamba: Visual state space model with windowed selective scan. In *European Conference on Computer Vision*, pages 12–22. Springer, 2024. 2, 5

[39] Juntao Jiang, Jiangning Zhang, Weixuan Liu, Muxuan Gao, Xiaobin Hu, Zhucun Xue, Yong Liu, and Shuicheng Yan. Rwkv-unet: Improving unet with long-range cooperation for effective medical image segmentation. *arXiv preprint arXiv:2501.08458*, 2025. 7

[40] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6399–6408, 2019. 6, 7

[41] Lujun Li, Zimian Wei, Peijie Dong, Wenhan Luo, Wei Xue, Qifeng Liu, and Yike Guo. Attnzero: Efficient attention discovery for vision transformers. In *European Conference on Computer Vision*, pages 20–37. Springer, 2024. 2

[42] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer, 2014. 2, 4, 5

[43] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. Vmamba: Visual state space model. *Advances in Neural Information Processing Systems*, 37:103031–103063, 2024. 2

[44] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 1, 2, 6, 7

[45] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12009–12019, 2022. 2

[46] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022. 1, 2, 3

[47] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 4, 6

[48] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9226–9235, 2019. 2

[49] Junting Pan, Adrian Bulat, Fuwen Tan, Xiatian Zhu, Lukasz Dudziak, Hongsheng Li, Georgios Tzimiropoulos, and Brais

Martinez. Edgevits: Competing light-weight cnns on mobile devices with vision transformers. In *European Conference on Computer Vision*, pages 294–311. Springer, 2022. 5

[50] Zhanzhong Pang, Fadime Sener, and Angela Yao. Context-enhanced memory-refined transformer for online action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8700–8710, 2025. 2

[51] Badri N Patro, Vinay P Namboodiri, and Vijay S Agneeswaran. Spectformer: Frequency and attention is what you need in a vision transformer. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 9543–9554. IEEE, 2025. 5

[52] Gensheng Pei, Tao Chen, Xiruo Jiang, Huafeng Liu, Zeren Sun, and Yazhou Yao. Videomac: Video masked autoencoders meet convnets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22733–22743, 2024. 1

[53] Gensheng Pei, Tao Chen, Yujia Wang, Xinhao Cai, Xiangbo Shu, Tianfei Zhou, and Yazhou Yao. Seeing what matters: Empowering clip with patch generation-to-selection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24862–24872, 2025. 1

[54] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Leon Derczynski, Xingjian Du, Matteo Grella, Kranthi Gv, Xuzheng He, Haowen Hou, Przemyslaw Kazienko, Jan Kocon, Jiaming Kong, Bartłomiej Koptyra, Hayden Lau, Jiaju Lin, Krishna Sri Ipsit Mantri, Ferdinand Mom, Atsushi Saito, Guangyu Song, Xiangru Tang, Johan Wind, Stanisław Woźniak, Zhenyuan Zhang, Qinghua Zhou, Jian Zhu, and Rui-Jie Zhu. Rwkv: Reinventing rnns for the transformer era. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14048–14077, 2023. 2

[55] Bo Peng, Daniel Goldstein, Quentin Gregory Anthony, Alon Albalak, Eric Alcaide, Stella Biderman, Eugene Cheah, Teddy Ferdinan, Kranthi Kiran GV, Haowen Hou, Satyapriya Krishna, Ronald McClelland Jr., Niklas Muennighoff, Fares Obeid, Atsushi Saito, Guangyu Song, Haoqin Tu, Ruichong Zhang, Bingchen Zhao, Qihang Zhao, Jian Zhu, and Rui-Jie Zhu. Eagle and finch: Rwkv with matrix-valued states and dynamic recurrence. In *First Conference on Language Modeling*, 2024.

[56] Bo Peng, Ruichong Zhang, Daniel Goldstein, Eric Alcaide, Xingjian Du, Haowen Hou, Jiaju Lin, Jiaxing Liu, Janna Lu, William Merrill, et al. Rwkv-7 "goose" with expressive dynamic state evolution. *arXiv preprint arXiv:2503.14456*, 2025. 2

[57] Ieva Petrulionytė, Julien Mairal, and Michael Arbel. Functional bilevel optimization for machine learning. *Advances in Neural Information Processing Systems*, 37:14016–14065, 2024. 3

[58] Yongming Rao, Wenliang Zhao, Zheng Zhu, Jiwen Lu, and Jie Zhou. Global filter networks for image classification. *Advances in Neural Information Processing Systems*, 34:980–993, 2021. 5

[59] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[60] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015. 2

[61] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021. 2, 4, 5, 7

[62] Pavan Kumar Anasosalu Vasu, James Gabriel, Jeff Zhu, Oncel Tuzel, and Anurag Ranjan. Mobileone: An improved one millisecond mobile backbone. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7907–7917, 2023. 5

[63] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017. 7

[64] Ao Wang, Hui Chen, Zijia Lin, Jungong Han, and Guiguang Ding. Lsnet: See large, focus small. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9718–9729, 2025. 2

[65] Jiahao Wang, Songyang Zhang, Yong Liu, Taiqiang Wu, Yujiu Yang, Xihui Liu, Kai Chen, Ping Luo, and Dahua Lin. Riformer: Keep your vision backbone effective but removing token mixer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14443–14452, 2023. 5

[66] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 568–578, 2021. 5, 6, 7

[67] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3):415–424, 2022. 5, 6

[68] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7794–7803, 2018. 2, 4, 7

[69] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *European Conference on Computer Vision*, pages 3–19, 2018. 2, 4, 7

[70] Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16133–16142, 2023. 2

[71] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *European Conference on Computer Vision*, pages 418–434, 2018. 6, 7

[72] Fei Xie, Weijia Zhang, Zhongdao Wang, and Chao Ma. Quadmamba: Learning quadtree-based selective scan for visual state space model. *Advances in Neural Information Processing Systems*, 37:117682–117707, 2024. 5

[73] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1492–1500, 2017. 5, 6

[74] Xingyi Yang and Xinchao Wang. Kolmogorov-arnold transformer. In *International Conference on Learning Representations*, 2025. 5

[75] Zongxin Yang, Yunchao Wei, and Yi Yang. Associating objects with transformers for video object segmentation. *Advances in Neural Information Processing Systems*, 34:2491–2502, 2021. 2

[76] Zhenyu Yang, Gensheng Pei, Yazhou Yao, Tianfei Zhou, Lizhong Ding, and Fumin Shen. ChangeTitans: Toward remote sensing change detection with neural memory. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–14, 2025. 2

[77] Zhenyu Yang, Gensheng Pei, Tao Chen, Xia Yuan, Haofeng Zhang, Xiangbo Shu, and Yazhou Yao. Beyond quadratic: Linear-time change detection with rwkv. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11811–11819, 2026. 2

[78] Zhiyu Yao, Jian Wang, Haixu Wu, Jingdong Wang, and Mingsheng Long. Mobile attention: mobile-friendly linear-attention for vision transformers. In *International Conference on Machine Learning*, 2024. 5

[79] Weihao Yu, Mi Luo, Pan Zhou, Chenyang Si, Yichen Zhou, Xinchao Wang, Jiashi Feng, and Shuicheng Yan. Metaformer is actually what you need for vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10819–10829, 2022. 5

[80] Seokju Yun and Youngmin Ro. Shvit: Single-head vision transformer with memory efficient macro design. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5756–5767, 2024. 2

[81] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 633–641, 2017. 2, 4, 6, 7

[82] Bo Zhou, Liulei Li, Yujia Wang, Huafeng Liu, Yazhou Yao, and Wenguan Wang. Unialign: Scaling multimodal alignment within one unified model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 29644–29655, 2025. 1

[83] Junbao Zhou, Ziqi Pang, and Yu-Xiong Wang. Rmem: Restricted memory banks improve video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18602–18611, 2024. 2

[84] Kaichen Zhou, Lanqing Hong, Xinhai Chang, Yingji Zhong, Enze Xie, Hao Dong, Zhihao Li, Yongxin Yang, Zhenguo Li, and Wei Zhang. Splatmesh: Interactive 3d segmentation and editing using mesh-based gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 305–316, 2025. 2

[85] Shihao Zhou, Duosheng Chen, Jinshan Pan, Jinglei Shi, and Jufeng Yang. Adapt or perish: Adaptive sparse transformer with attentive feature refinement for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2952–2963, 2024. 2

[86] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. In *International Conference on Machine Learning*, pages 62429–62442, 2024. 2, 5

[87] Lei Zhu, Xinjiang Wang, Wayne Zhang, and Rynson Lau. Revisiting the integration of convolution and attention for vision backbone. *Advances in Neural Information Processing Systems*, 37:42941–42964, 2024. 1, 2