RETHINKING CROSS-LAYER ATTENTION FOR CHANNEL AND SPATIAL RECALIBRATION IN CONVNETS

Anonymous authors
Paper under double-blind review

ABSTRACT

Cross-layer attention enables convolutional networks to access earlier representations, but retaining full feature histories becomes costly as depth increases. We propose Cross-Layer Attention Recalibration (CLAR), which performs historical selection over compact attention representations to construct channel and spatial gates. Within each residual stage, the current block attends to keys and values from preceding blocks and itself, and the aggregated values sequentially recalibrate its residual features while preserving the local shortcut. Historical source dropping regularizes this selection during training and always retains the current source. CIFAR-100 ablations support the complementary roles of channel and spatial recalibration and the combination of dynamic weighting with source dropping. On ImageNet-1K, CLAR achieves 78.71% Top-1 accuracy with ResNet-152 and outperforms the listed attention methods across all evaluated backbone depths. Experiments with Faster R-CNN and Mask R-CNN on COCO further demonstrate gains in overall detection and instance segmentation performance over MRLA-light. These results support explicit selection over historical attention representations as an effective mechanism for feature recalibration in deep convolutional networks.

1 INTRODUCTION

Residual networks pass information through local identity shortcuts, allowing deep convolutional models to refine a representation without relearning it at every block (He et al., 2016). Ever since ResNet was established as the design paradigm for convolutional neural networks, attention mechanisms have consistently served as a key principle for enhancing network performance. Squeeze-and-Excitation (SE) derives channel weights from a global summary, while the Convolutional Block Attention Module (CBAM) applies channel and spatial attentions in sequence (Hu et al., 2018; Woo et al., 2018). In these designs, attention weights are derived from the feature maps of the residual branch and subsequently applied to adjust the feature maps via operations such as element-wise multiplication.

Another line of network design involves enabling interaction with information from earlier layers. DenseNet concatenates the outputs of preceding blocks within a stage (Huang et al., 2017). Extending cross-layer access beyond convolutional feature concatenation, OmniNet lets Transformer tokens attend across both network width and depth, whereas Adaptive Cross-Layer Attention (ACLA) retrieves selected keys from earlier feature maps for image restoration (Tay et al., 2021; Wang et al., 2022). Such broad access comes with substantial computation and memory demands. Both OmniNet and ACLA therefore introduce restricted or adaptive access mechanisms. Their designs expose a central tension: earlier features may be useful, but retaining and comparing increasingly many full representations makes direct cross-layer interaction costly as depth grows.

Recent success with Attention Residuals has renewed interest in the operation performed across depth. In Transformers, it replaces fixed residual accumulation with input-dependent attention over preceding layer outputs (Kimi Team et al., 2026). We first investigated a direct convolutional adaptation that selected among full residual features. Under our controlled comparisons, this feature-level route incurs rapidly increasing computational costs as depth grows, without yielding consistent gains over the compared attention approaches (see Appendix A). This observation led us to ask a narrower

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

question: can cross-layer selection act on the *signals that generate attention gates*, rather than aggregate full feature tensors?

We answer this question with **Cross-Layer Attention Recalibration (CLAR)** (Figure 1). In each resolution stage, CLAR stores compact channel and spatial descriptors from preceding blocks. The current block computes separate channel and spatial queries, scores its own representation together with the stored sources, and aggregates values by a softmax over depth. The resulting channel gate first modulates the current residual branch; a spatial gate computed from the channel-gated feature follows. The local identity or projection shortcut is then added as usual. Historical sources can be stochastically omitted during training, while the current source is always available. CLAR therefore introduces cross-layer information into attention *recalibration* while preserving the form of the residual update.

Multi-head Recurrent Layer Attention (MRLA) and Dynamic Layer Attention (DLA) are the closest cross-layer attention approaches for CNNs (Fang et al., 2023; Wang et al., 2024). Both MRLA and DLA provide base and light variants. Their base variants compute attention over earlier feature representations, but the associated cost grows with network depth. In their ImageNet ResNet experiments, both papers report base variants only at depth 50. Our reproduction likewise encountered a memory failure when scaling MRLA-base to ResNet-101 under the same training configuration. The light variants address this cost through recurrently propagated summaries, thereby giving up explicit, layer-specific selection over earlier feature maps. CLAR instead performs historical selection on compact channel and spatial attention representations. This design supports training with backbones up to ResNet-152. In our reported ImageNet comparisons, CLAR achieves the highest Top-1 accuracy among the listed methods at all three evaluated depths; the COCO results also show gains in overall bounding-box and mask AP over MRLA-light (Section 4).

Our contributions are threefold:

1. We formulate cross-layer interaction as channel and spatial attention recalibration, retaining explicit selection over earlier sources through compact stage-local representations while preserving the local residual shortcut.
2. We introduce historical source dropping to regularize depth-wise selection. Controlled CIFAR-100 ablations show that its combination with dynamic weighting improves mean Top-1 accuracy over dynamic weighting without dropping and over the static learned and uniform controls.
3. We evaluate CLAR across ResNet-50, ResNet-101, and ResNet-152 on CIFAR-100 and ImageNet-1K, and assess transfer with Faster R-CNN and Mask R-CNN on COCO. The reported results demonstrate gains over the strongest listed ImageNet comparisons and improvements in overall detection and segmentation AP over MRLA-light.

2 Related Work

2.1 Within-Layer Attention in Convolutional Networks

Feature attention can alter the importance of channels, locations, or parallel operations within a convolutional block. SE predicts channel weights from a globally pooled descriptor (Hu et al., 2018), and ECA models local cross-channel dependencies without the dimensionality reduction used by SE (Wang et al., 2020). Selective Kernel Networks use input-dependent weights to combine convolutional branches with different receptive fields (Li et al., 2019). Non-Local Neural Networks compute interactions among spatial or spatiotemporal positions within a feature representation (Wang et al., 2018). These mechanisms place attention on channels, positions, or branches of the representation available at a given point in the network.

CBAM combines two complementary gates in a fixed order: channel attention based on global average and max pooling, followed by spatial attention based on channelwise average and max maps (Woo et al., 2018). This decomposition makes channel and spatial modulation explicit while keeping the gate computation local to the block. It also supplies a useful reference for distinguishing a *gate value* from the feature tensor to which that gate is applied.

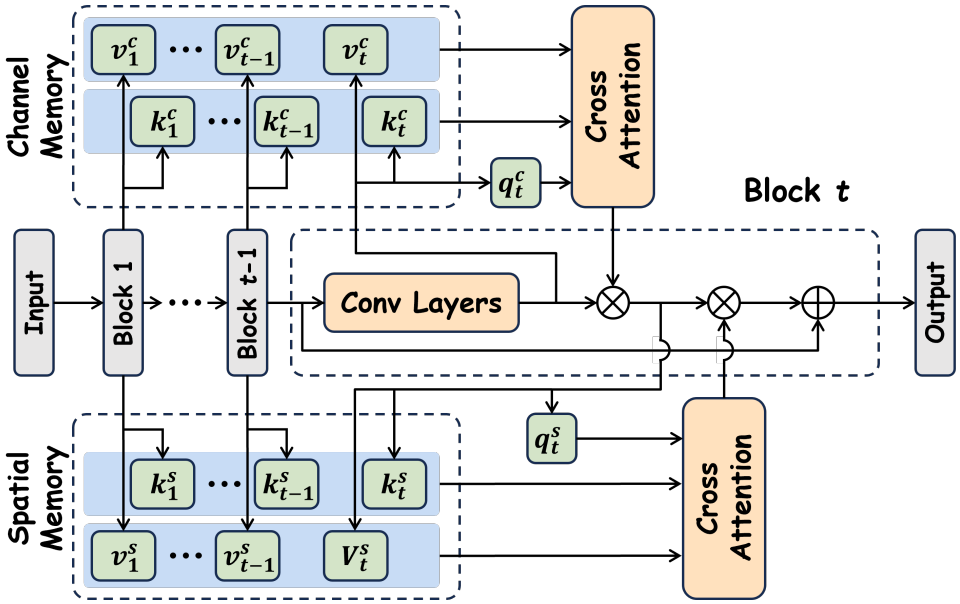


Figure 1: CLAR overview: stage-local channel and spatial memories recalibrate the current residual feature before shortcut addition.

2.2 Cross-Layer Interaction

Convolutional networks. DenseNet exposes earlier features through concatenation, while Deep Layer Aggregation merges representations through iterative and hierarchical structures (Huang et al., 2017; Yu et al., 2018). DIANet shares a recurrent attention unit within a stage so that a hidden state carries information from preceding blocks into later feature recalibration (Huang et al., 2020). ACLA uses query locations to select keys from preceding layers and adaptively limits the number of keys used for image restoration (Wang et al., 2022). MRLA formalizes attention over layer outputs and offers both an explicit base form and a recurrent light approximation (Fang et al., 2023). DLA further extracts context along the forward depth path and refreshes earlier features before applying its base or light layer-attention form (Wang et al., 2024). These methods differ in whether history is retained as separate feature maps, compressed into a recurrent state, or modified before retrieval. Here "DLA" denotes *Dynamic Layer Attention* (2024); *Deep Layer Aggregation* (2018) is cited by its full name.

Transformers and related residual streams. OmniNet enlarges the attention receptive field to tokens from multiple layers, using efficient attention and partitioning to manage the resulting cost (Tay et al., 2021). RealFormer propagates raw attention *scores* from one Transformer layer to the next before softmax (He et al., 2021). Hyper-Connections creates multiple residual states and learns how to mix them across depth and width (Zhu et al., 2025). Attention Residuals instead applies attention to the outputs accumulated in a Transformer residual stream and provides a blockwise variant that limits the amount of retained history (Kimi Team et al., 2026). Cross-layer interaction in these models consequently acts on different objects: token representations, attention scores, parallel residual states, or preceding layer outputs.

3 Cross-Layer Attention Recalibration (CLAR)

Figure 1 summarizes CLAR within a residual stage. At block t, separate channel and spatial queries select among attention keys and values from blocks $1, \ldots, t$, and the retrieved values sequentially recalibrate the current residual feature before shortcut addition. Sections 3.2 and 3.3 specify the two branches, while Sections 3.4 and 3.5 describe source dropping and stage-local memory.

3.1 Preliminary: Revisiting Channel and Spatial Attention

For a residual network, let $F_t \in \mathbb{R}^{B\times C\times H\times W}$ be the output of the convolutional branch of its t-th residual block, before addition to its shortcut branch I_t. CBAM sequentially computes a channel attention map G_t^c and a spatial attention map G_t^s (Woo et al., 2018). The procedure of attention application can be described as:

$$F_t' = F_t \odot G_t^c, \qquad F_t'' = F_t' \odot G_t^s, \qquad X_t = \mathrm{ReLU}(F_t'' + I_t),$$

where $\odot$ denotes element-wise multiplication and X_t is the output of the residual block. During multiplication, G_t^c and G_t^s are broadcast to match the shape of F_t. The channel attention map is derived from global average and max pooling across the spatial dimensions of the feature map, followed by a shared two-layer bottleneck MLP ϕ_t and sigmoid σ:

$$a_t = \mathrm{AvgPool}(F_t), \qquad m_t = \mathrm{MaxPool}(F_t), \qquad G_t^c = \sigma\big(\phi_t(a_t) + \phi_t(m_t)\big),$$

The spatial branch concatenates the mean and maximum of F_t' across the channel dimension and applies a 7×7 convolution ψ_t:

$$D_t = \big[\mathrm{mean}_C(F_t'), \max_C(F_t')\big], \qquad G_t^s = \sigma\big(\psi_t(D_t)\big).$$

The channel gate has shape $B \times C \times 1 \times 1$; the spatial gate has shape $B \times 1 \times H \times W$. CLAR retains this channel-then-spatial order and the local residual shortcut, while allowing the gate-generating values to come from earlier layers.

3.2 Channel Attention Recalibration across Depth

The channel attention map in CLAR is derived by recalibrating the descriptors of the current layer and historical layers through the cross-attention mechanism. Index blocks within a stage by $t \in \{1, \ldots, L\}$. Each block j stores a channel key and value derived from its own residual branch feature F_j:

$$k_j^c = \rho_j^k(a_j), \qquad v_j^c = \phi_j(a_j).$$

The target block computes $q_t^c = \rho_t^q(m_t)$. Query and key RMS normalizers have separate learnable channel scales: $\rho(z;\gamma) = \gamma \odot z/\sqrt{\mathrm{mean}(z^2) + \epsilon}$, with $\epsilon = 10^{-6}$. Their rescaling operation follows Zhang & Sennrich (2019). The current block contributes its own key and value alongside those retained from earlier blocks. For $t > 1$, the source score, depth weight, and resulting channel gate are

$$e_{t,j}^c = \frac{\langle q_t^c, k_j^c\rangle}{\tau\sqrt{C}}, \quad \alpha_{t,j}^c = \frac{\exp(e_{t,j}^c)}{\sum_{i=1}^t \exp(e_{t,i}^c)}, \quad G_t^c = \sigma\left(\sum_{j=1}^t \alpha_{t,j}^c v_j^c\right).$$

The positive temperature τ is shared with the spatial branch and set to 1 in the implementation. Scores are normalized over layer depth for each image. One scalar $\alpha_{t,j}^c$ weights all channels of a source value, after which the aggregated channel vector is broadcast over spatial positions.

3.3 Spatial Attention Recalibration across Depth

Similar to CBAM, the spatial branch receives $F_t' = G_t^c \odot F_t$ and forms $D_t = [\mathrm{mean}_C(F_t'), \max_C(F_t')]$. At every block, three separate 7×7 convolutions generate query, key, and value maps:

$$Q_t = W_t^q * D_t, \qquad K_t = W_t^k * D_t, \qquad V_t = W_t^v * D_t,$$

where $*$ denotes convolution and each map has one channel and spatial size $H \times W$. To enable cross-attention computation, a flattening operator vec acts on Q_t and K_t only. Let $q_t^s = \rho(\mathrm{vec}(Q_t))$ and $k_j^s = \rho(\mathrm{vec}(K_j))$. Here the spatial normalizer ρ has no learnable scale and uses $\epsilon = 10^{-6}$. For $t > 1$, the spatial routing weights and gate are

$$e_{t,j}^s = \frac{\langle q_t^s, k_j^s \rangle}{\tau\sqrt{HW}}, \quad \alpha_{t,j}^s = \frac{\exp(e_{t,j}^s)}{\sum_{i=1}^t \exp(e_{t,i}^s)}, \quad G_t^s = \sigma\left(\sum_{j=1}^t \alpha_{t,j}^s V_j\right).$$

Each $\alpha_{t,j}^s$ is a scalar per image and source, whereas V_j retains the original $H \times W$ resolution. The gate therefore varies spatially even though depth weights are shared across positions.

3.4 Historical Source Dropping

We introduce historical source dropping to regularize attention recalibration across depth. During training, each earlier source is independently retained with probability $1-p$, while the current source remains available. For branch $b \in \{c, s\}$ at target block $t > 1$, we sample

$$m_{t,j}^b \sim \mathrm{Bernoulli}(1-p), \qquad j < t,$$

and define the active source set as

$$\mathcal{S}_t^b = \{j < t : m_{t,j}^b = 1\} \cup \{t\}.$$

The depth weights defined in Sections 3.2 and 3.3 are normalized over this active set:

$$\hat{\alpha}_{t,j}^c = \frac{\exp(e_{t,j}^c)}{\sum_{i \in \mathcal{S}_t^c} \exp(e_{t,i}^c)}, \qquad \hat{\alpha}_{t,j}^s = \frac{\exp(e_{t,j}^s)}{\sum_{i \in \mathcal{S}_t^s} \exp(e_{t,i}^s)}.$$

Consequently, the retained sources form a convex combination without additional rescaling. Each mask is shared across images in the local minibatch, whereas the channel and spatial branches sample their masks independently.

Dropping affects source access at the current block: the complete memory is preserved, so an omitted source remains available to subsequent blocks. This mechanism is intended to discourage reliance on a fixed subset of historical sources while preserving a current-block contribution. At inference, dropping is disabled and all preceding sources participate in recalibration.

3.5 Implementation Details

Stage-local memory. CLAR maintains separate channel and spatial memories within each residual stage. Before processing block t, these memories contain

$$\mathcal{M}_t^c = \{(k_j^c, v_j^c)\}_{j=1}^{t-1}, \qquad \mathcal{M}_t^s = \{(k_j^s, V_j)\}_{j=1}^{t-1}.$$

The current block generates its own keys and values, includes them in the candidate sources for recalibration, and appends them to the corresponding memories for subsequent blocks. Each stage begins with empty memories, and no entries are carried across stage boundaries or forward passes. Restricting access to a stage ensures compatible channel dimensions and spatial resolutions without additional alignment operations.

The retained entries comprise channel vectors, spatial key descriptors, and single-channel spatial value maps. For a stage with L blocks, the explicit memory contains $O(L(C + HW))$ elements per image. Thus, historical routing uses compact attention representations without requiring full historical feature tensors of size $C \times H \times W$.

Initialization at stage boundaries. As the first block has no preceding sources, it applies the local channel and spatial attention following the typical CBAM design. The block also generates and

Table 1: Ablation on Channel and spatial recalibration.

Configuration	ResNet-50				ResNet-101			
	Params (M)	FLOPs (G)	Top-1 (%)	Top-5 (%)	Params (M)	FLOPs (G)	Top-1 (%)	Top-5 (%)
ResNet	23.70	2.60	78.83 ± 0.26	94.22 ± 0.09	42.69	5.02	78.55 ± 0.34	93.95 ± 0.16
+ Channel only	26.25	2.60	79.18 ± 0.09	94.67 ± 0.12	47.51	5.03	79.37 ± 0.31	94.54 ± 0.11
+ Spatial only	23.71	2.60	78.11 ± 0.28	94.06 ± 0.25	42.71	5.02	78.48 ± 0.36	94.31 ± 0.28
+ CLAR	26.26	2.61	79.25 ± 0.13	94.46 ± 0.15	47.52	5.04	79.59 ± 0.27	94.62 ± 0.13

stores its channel and spatial keys and values for later retrieval. To align channel count and spatial resolution at a stage boundary, I_1 uses a 1×1 convolution followed by batch normalization.

4 Experiments

We evaluate CLAR on the standard benchmarks: CIFAR-100 and ImageNet-1K for image classification; MS COCO for object detection and instance segmentation. To thoroughly investigate the best design of our final module, we first perform ablation experiments including channel and spatial recalibration, alternative ways of weighting historical sources, and historical source dropping. Classification experiments cover ResNet-50, ResNet-101, and ResNet-152 and compare CLAR with SE, CBAM, MRLA-L, and DLA-L. The transfer evaluation uses Faster R-CNN and Mask R-CNN with ResNet-50 and ResNet-101 backbones.

4.1 Ablation Studies on CIFAR-100

Experimental setting. Limited by computational resources, we perform all ablation studies on CIFAR-100. CIFAR-100 contains 50,000 training images and 10,000 test images of resolution 32×32, spanning 100 classes (Krizhevsky, 2009). We evaluate attention variants on ResNet-50 and ResNet-101 using five random seeds. These variants are trained for 200 epochs with SGD, momentum 0.9, Nesterov acceleration, weight decay 5×10^{-4}, batch size 256, and label smoothing 0.1. The learning rate increases from 0.01 to 0.2 over five warmup epochs and is reduced by a factor of ten at epochs 100 and 150. We report test Top-1 and Top-5 accuracy as mean $\pm$ sample standard deviation across seeds. Classification FLOPs are profiled at inference with one multiplication and one addition counted as two operations; the counter includes convolutions, linear operations, and matrix products, but excludes pooling, normalization, activations, elementwise operations, and softmax.

Channel and spatial recalibration. We compare the original ResNet with channel-only, spatial-only, and combined channel–spatial recalibration. The three attention variants use the same dropping probability, $p = 0.4$. Table 1 reports their accuracy and model costs at both depths.

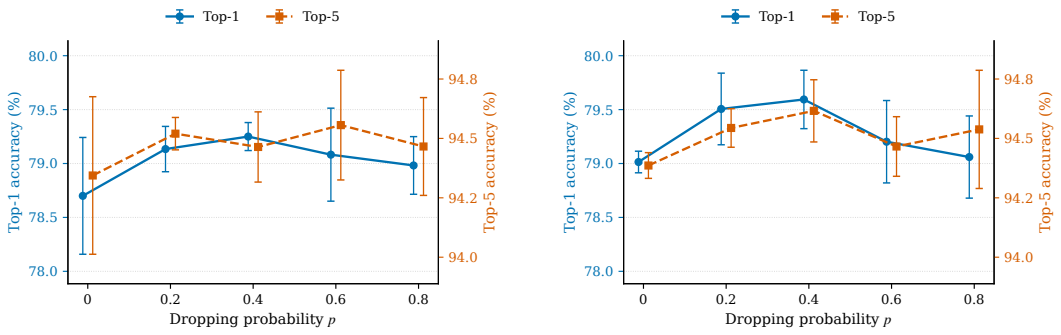
The combined design achieves Top-1 accuracies of 79.25% and 79.59% on ResNet-50 and ResNet-101, exceeding the channel-only variants by 0.07 and 0.22 percentage points and the spatial-only variants by 1.14 and 1.11 percentage points, respectively. The effects differ across metrics: channel-only recalibration achieves the highest Top-5 accuracy on ResNet-50, whereas the combined design achieves the highest Top-5 accuracy on ResNet-101. These results describe the contribution of adding each branch to the complete configuration; comparisons with local attention in Section 4.2 provide additional context for the role of historical interaction.

Baseline provenance. The ResNet reference entries use the available runs with batch size 128 and initial learning rate 0.1, unlike the attention variants above. Their statistics aggregate 20 seeds for ResNet-50 and six seeds for ResNet-101. Accordingly, differences from this reference are descriptive comparisons rather than controlled estimates of the attention effect under an identical training recipe.

Historical source weighting. We replace input-dependent query–key scores with two alternatives. Static learned weighting assigns trainable, input-independent logits to the sources at each target block and normalizes them with softmax. Uniform weighting assigns equal weight to every available source. Both alternatives retain the channel and spatial value representations and aggregate them before sigmoid. Table 2 compares these alternatives with dynamic weighting without source

Table 2: Ablation on historical source weighting.

Source weighting	ResNet-50				ResNet-101			
	Params (M)	FLOPs (G)	Top-1 (%)	Top-5 (%)	Params (M)	FLOPs (G)	Top-1 (%)	Top-5 (%)
Static learned	26.26	2.61	78.95 ± 0.32	94.35 ± 0.26	47.52	5.04	79.39 ± 0.47	94.53 ± 0.17
Uniform	26.26	2.61	79.12 ± 0.35	94.53 ± 0.07	47.52	5.04	79.27 ± 0.42	94.51 ± 0.11
Dynamic	26.26	2.61	79.25 ± 0.13	94.46 ± 0.15	47.52	5.04	79.59 ± 0.27	94.62 ± 0.13



(a) ResNet-50 (b) ResNet-101

Figure 2: Historical source dropping on CIFAR-100: ResNet-50 (left) and ResNet-101 (right). Error bars show sample standard deviations over five seeds; Top-1 and Top-5 use separate vertical scales.

dropping and also includes the regularized CLAR configuration used in Table 1. Static and uniform weighting access all available sources during training, so the dynamic $p = 0$ row provides the direct comparison without source dropping.

Without source dropping, static learned and uniform weighting both exceed dynamic weighting in mean Top-1 accuracy at the two depths. Dynamic weighting with $p = 0.4$ reaches the highest mean Top-1 accuracy among these configurations, although uniform weighting on ResNet-50 has slightly higher Top-5 accuracy. Thus, the observed advantage concerns the combination of dynamic selection and source dropping; this comparison does not establish that input-dependent weighting alone is superior to the alternatives.

Historical source dropping. We vary p over $\{0, 0.2, 0.4, 0.6, 0.8\}$ while retaining both recalibration branches. Figure 2 presents separate panels for ResNet-50 and ResNet-101, each with Top-1 accuracy on the left axis and Top-5 accuracy on the right axis. Among the tested values, $p = 0.4$ gives the highest mean Top-1 accuracy at both depths, improving over $p = 0$ by 0.55 and 0.58 percentage points, respectively. The highest mean Top-5 accuracy occurs at $p = 0.6$ for ResNet-50 and $p = 0.4$ for ResNet-101. Top-1 accuracy decreases again at larger dropping probabilities, indicating that the benefit is not monotonic in p.

4.2 Image Classification

Datasets and comparisons. We evaluate ResNet backbones at depths 50, 101, and 152 on CIFAR-100 and ImageNet-1K. ImageNet-1K contains approximately 1.28 million training images and 50,000 validation images across 1,000 classes (Deng et al., 2009; Russakovsky et al., 2015). The comparisons include the original ResNet, the local attention methods SE and CBAM, and the cross-layer methods MRLA-L and DLA-L (Hu et al., 2018; Woo et al., 2018; Fang et al., 2023; Wang et al., 2024). Table 3 places the two datasets side by side and reports parameters, FLOPs, Top-1 accuracy, and Top-5 accuracy. Classification costs are profiled with 32×32 and 224×224 inputs for CIFAR-100 and ImageNet-1K, respectively. Accuracy entries are means over repeated runs or observed accuracies for single runs. CLAR uses $p = 0.4$ for CIFAR-100 ResNet-50/101 and $p = 0.2$ for CIFAR-100 ResNet-152 and ImageNet.

Table 3: Image classification on CIFAR-100 and ImageNet-1K. Accuracy is reported in percent.

Model	CIFAR-100				ImageNet-1K			
	Params (M)	FLOPs (G)	Top-1 (%)	Top-5 (%)	Params (M)	FLOPs (G)	Top-1 (%)	Top-5 (%)
ResNet-50	23.70	2.60	78.83	94.22	25.55	8.18	75.31	92.41
+ SE	26.22	2.60	78.93	94.32	28.07	8.18	76.03	92.97
+ CBAM	26.22	2.61	79.04	94.39	28.07	8.19	76.18	93.02
+ MRLA-L	23.89	2.63	76.75	93.29	25.74	8.28	77.10	93.63
+ DLA-L	26.73	2.70	77.39	93.31	28.58	8.42	77.25	93.32
+ CLAR	26.26	2.61	79.25	94.46	28.11	8.20	77.44	93.66
ResNet-101	42.69	5.02	78.55	93.95	44.55	15.60	76.02	92.82
+ SE	47.44	5.03	79.39	94.37	49.29	15.61	77.32	93.56
+ CBAM	47.44	5.04	79.19	94.58	49.30	15.62	77.55	93.78
+ MRLA-L	43.09	5.07	76.45	93.69	44.94	15.76	77.82	93.89
+ DLA-L	45.93	5.20	77.44	93.86	47.78	16.00	77.97	93.92
+ CLAR	47.52	5.04	79.59	94.62	49.37	15.63	78.26	94.02
ResNet-152	58.34	7.45	78.91	94.12	60.19	23.03	76.93	93.54
+ SE	64.92	7.46	79.27	94.63	66.77	23.04	78.12	93.66
+ CBAM	64.92	7.47	79.12	94.53	66.78	23.06	78.15	93.75
+ MRLA-L	58.92	7.52	77.00	93.56	60.77	23.27	78.45	94.18
+ DLA-L	61.76	7.71	77.88	93.75	63.61	23.61	78.52	94.20
+ CLAR	65.03	7.47	79.79	94.86	66.89	23.06	78.71	94.26

Results on CIFAR-100. CLAR achieves Top-1 accuracies of 79.25%, 79.59%, and 79.79% at depths 50, 101, and 152. At depth 50, CLAR matches the listed ResNet Top-1 accuracy and trails SE by 0.08 percentage points, while improving over CBAM by 0.21 points. At depths 101 and 152, CLAR has the highest Top-1 and Top-5 accuracies among the listed methods; its Top-1 gains over the strongest alternatives are 0.20 and 0.52 points, respectively. Relative to CBAM, CLAR improves Top-1 accuracy by 0.40 and 0.67 points at these depths, supporting the use of historical attention representations in deeper networks. The depth-152 CLAR result currently represents one run. The baseline and CLAR CIFAR runs use different batch sizes and initial learning rates, so the main-table differences describe the reported configurations rather than isolating the effect of historical interaction under a common recipe.

Results on ImageNet-1K. CLAR achieves Top-1 accuracies of 77.44%, 78.26%, and 78.71% for ResNet-50, ResNet-101, and ResNet-152, respectively, and has the highest Top-1 and Top-5 accuracies among the listed methods at each depth. Its Top-1 improvements over the strongest alternative, DLA-L, are 0.19, 0.29, and 0.19 percentage points. Relative to CBAM, the gains are 1.26, 0.71, and 0.56 points. The added model cost over CBAM is small: approximately 35K, 75K, and 110K parameters and less than 0.01G additional FLOPs at the three depths, using the unrounded profiling values. CLAR uses more parameters than MRLA-L but fewer profiled FLOPs at each depth. These comparisons characterize model size and counted inference arithmetic, rather than measured training memory or runtime. The available ImageNet runs differ in training duration, batch size, and warmup; repeated comparisons with matched training settings would further assess the size and consistency of the observed gains.

4.3 Object Detection and Instance Segmentation

We assess transfer to dense prediction using Faster R-CNN for object detection and Mask R-CNN for object detection and instance segmentation on COCO (Lin et al., 2014; Ren et al., 2015; He et al., 2017). Tables 4 and 5 compare ResNet-50 and ResNet-101 backbones with SE, MRLA-L, and CLAR. ResNet, SE, and MRLA-L accuracy entries are published reference results, while CLAR accuracy entries are from our updated transfer results. Bounding-box and mask performance are denoted by AP^{bb} and AP^{m}, respectively.

Table 4: Object detection results of different methods on COCO val2017 using Faster R-CNN as a framework.

Model	Params (M)	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{bb}_{S}	AP^{bb}_{M}	AP^{bb}_{L}
ResNet-50	41.53	36.4	58.2	39.2	21.8	40.0	46.2
+ SE	44.02	37.7	60.1	40.9	22.9	41.9	48.2
+ MRLA-L	41.70	40.4	61.5	44.0	24.2	44.1	52.7
+ CLAR	44.05	40.6	61.7	44.1	24.4	44.2	52.9
ResNet-101	60.52	38.7	60.6	41.9	22.7	43.2	50.4
+ SE	65.24	39.6	62.0	43.1	23.7	44.0	51.4
+ MRLA-L	60.90	42.0	63.1	45.7	25.0	45.8	55.4
+ CLAR	65.31	42.2	63.3	45.8	25.3	46.0	55.4

Table 5: Object detection and instance segmentation results of different methods on COCO val2017 using Mask R-CNN as a framework.

Model	Params (M)	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{m}	AP^{m}_{50}	AP^{m}_{75}
ResNet-50	44.18	37.2	58.9	40.3	34.1	55.5	36.2
+ SE	46.67	38.7	60.9	42.1	35.4	57.4	37.8
+ MRLA-L	44.34	41.2	62.3	45.1	37.1	59.1	39.6
+ CLAR	46.70	41.4	62.4	45.0	37.2	59.3	39.7
ResNet-101	63.17	39.4	60.9	43.3	35.9	57.7	38.4
+ SE	67.89	40.7	62.5	44.3	36.8	59.3	39.2
+ MRLA-L	63.54	42.8	63.6	46.5	38.4	60.6	41.0
+ CLAR	67.96	42.9	63.6	46.7	38.6	60.1	41.2

Faster R-CNN. Table 4 reports bounding-box AP, AP at IoU thresholds 0.50 and 0.75, and AP for small, medium, and large objects. This framework does not predict instance masks, so all six accuracy columns concern bounding boxes.

Mask R-CNN. Table 5 reports both bounding-box and instance-mask AP, following the layout of the MRLA comparison. The reference values are taken from the source papers, rather than presented as our reproductions.

Transfer results. With Faster R-CNN, CLAR reaches 40.6 and 42.2 bounding-box AP for ResNet-50 and ResNet-101, improving over the corresponding ResNet references by 4.2 and 3.5 AP and over MRLA-L by 0.2 AP at both depths. CLAR also improves AP for small and medium objects over MRLA-L; large-object AP improves by 0.2 at depth 50 and is unchanged at depth 101. With Mask R-CNN, CLAR achieves 41.4/37.2 bounding-box/mask AP for ResNet-50 and 42.9/38.6 for ResNet-101. Relative to MRLA-L, the gains are 0.2/0.1 AP and 0.1/0.2 AP, respectively. The gains concern overall AP rather than every component metric: ResNet-50 bounding-box AP at IoU 0.75 is 0.1 lower, and ResNet-101 mask AP at IoU 0.50 is 0.5 lower than MRLA-L. Together, the results show that CLAR transfers to both detection and instance segmentation, with modest improvements in the overall metrics over the strongest listed reference.

5 Conclusion

We presented CLAR, a mechanism that recalibrates channel and spatial attention through explicit selection across residual blocks. Compact attention memories make earlier sources individually accessible within each stage, and historical source dropping regularizes their use while preserving the current source and local residual shortcut. CIFAR-100 ablations support the combination of dynamic weighting and source dropping; ImageNet-1K comparisons show the highest reported Top-1 and Top-5 accuracies among the listed methods at all three backbone depths. COCO evaluations further show improvements in overall bounding-box and mask AP over MRLA-light with Faster R-CNN and Mask R-CNN. These findings support cross-layer attention representations as a useful source for feature recalibration.

AI USE STATEMENT

In this work, we used generative AI tools to polish the manuscript's language, assist with code completion, and support literature searches. All AI-generated content was manually reviewed and verified by the authors. We checked that language edits preserved the intended technical meaning, reviewed and tested AI-assisted code for correctness, and verified references and their relevance against the original sources. We take full responsibility for the final content of this work, including all text, claims, code, and other artifacts produced with the aid of generative AI.

REFERENCES

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 Conference on Computer Vision and Pattern Recognition*, 2009.

Yanwen Fang, Yuxi Cai, Jintai Chen, Jingyu Zhao, Guangjian Tian, and Guodong Li. Cross-layer retrospective retrieving via layer attention. In *International Conference on Learning Representations*, 2023.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

Ruining He, Anirudh Ravula, Bhargav Kanagal, and Joshua Ainslie. Realformer: Transformer likes residual attention. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP*, 2021.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, 2018.

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2261–2269, 2017. doi: 10.1109/CVPR.2017.243.

Zhongzhan Huang, Senwei Liang, Mingfu Liang, and Haizhao Yang. Dianet: Dense-and-implicit attention network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

Kimi Team, Guangyu Chen, Yu Zhang, Jianlin Su, Weixin Xu, Siyuan Pan, Yaoyu Wang, Yucheng Wang, Guanduo Chen, Bohong Yin, Yutian Chen, Junjie Yan, Ming Wei, Y. Zhang, Fanqing Meng, Chao Hong, Xiaotong Xie, Shaowei Liu, Enzhe Lu, Yunpeng Tai, Yanru Chen, Xin Men, Haiqing Guo, Y. Charles, Haoyu Lu, Lin Sui, Jinguo Zhu, Zaida Zhou, Weiran He, Weixiao Huang, Xinran Xu, Yuzhi Wang, Guokun Lai, Yulun Du, Yuxin Wu, Zhilin Yang, and Xinyu Zhou. Attention residuals, 2026.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang. Selective kernel networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 510–519, 2019.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context. arXiv:1405.0312, 2014.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, 2015.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 2015.

Yi Tay, Mostafa Dehghani, Vamsi Aribandi, Jai Gupta, Philip Pham, Zhen Qin, Dara Bahri, Da-Cheng Juan, and Donald Metzler. Omninet: Omnidirectional representations from transformers. In *Proceedings of the International Conference on Machine Learning*, 2021.

Kaishen Wang, Xun Xia, Jian Liu, Zhang Yi, and Tao He. Strengthening layer interaction via dynamic layer attention. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 5073–5081, 2024. doi: 10.24963/ijcai.2024/561.

Qilong Wang, Banggu Wu, Pengfei Zhu, Peihua Li, Wangmeng Zuo, and Qinghua Hu. Eca-net: Efficient channel attention for deep convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7794–7803, 2018.

Yancheng Wang, Ning Xu, and Yingzhen Yang. Adaptive cross-layer attention for image restoration. arXiv:2203.03619, 2022.

Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. CBAM: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision*, pp. 3–19, 2018.

Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2403–2412, 2018.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. In *Advances in Neural Information Processing Systems*, pp. 12360–12371, 2019.

Defa Zhu, Hongzhi Huang, Zihao Huang, Yutao Zeng, Yunyao Mao, Banggu Wu, Qiyang Min, and Xun Zhou. Hyper-connections. In *International Conference on Learning Representations*, 2025.