# LINEARIZING VISION-LANGUAGE VITS BEYOND THE TRAINING RESOLUTION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Linear recurrent visual encoders offer increasing computational advantages at high resolutions, yet representative linear CLIP models exhibit substantial performance degradation when the visual sequence length moves beyond their low training resolution. Strong pretrained vision-language ViTs retain high-quality representations beyond their training resolutions, whereas naively linearized recurrent encoders fail to inherit this capability. In this work, we introduce LADDER (**L**ayer **AD**aptive **D**istillation into **E**fficient Hybrid Encode**R**), a lightweight framework that converts pretrained vision-language ViTs into hybrid recurrent encoders while preserving their cross-resolution capabilities through architectural designs. LADDER combines attention and recurrence within blocks and across layers. For global feature aggregation, our proposed Vision-Delta pairs recurrent patch mixing with Attention-based CLS (AttnCLS), decoupling global readout from the block's recurrent propagation. For dense feature interaction, Resolution-Robust Layer-Adaptive Hybridization optimizes full-attention placement under a fixed budget via reinforcement learning guided by cross-resolution teacher patch-feature recovery. We primarily evaluate LADDER on TIPSv2, with additional validation on SigLIP2. With only 22 epochs of image-only training on ImageNet-1K, LADDER closely matches or surpasses its teachers across vision-language and linear probing evaluations, while matching the ViT teacher to within 1% point at Large scale. On high-resolution Cityscapes inputs, LADDER achieves $1.74\times$ the throughput of the FlashAttention teacher. Code and models will be released.

## 1 INTRODUCTION

Modern large-scale vision-language models are increasingly expected to operate across flexible input sizes and at high resolutions (Beyer et al., 2023; Tschannen et al., 2025). Vision Transformers (ViTs), driven by global self-attention, have emerged as the dominant visual backbone for these systems owing to their strong representation capacity and transferability (Dosovitskiy et al., 2021; Radford et al., 2021; Zhai et al., 2022; Yu et al., 2022). Yet, the quadratic $O(N^2)$ complexity of global self-attention imposes prohibitive computational overhead at high resolutions (Vaswani et al., 2017; Dao et al., 2022). By contrast, linear recurrent architectures scale linearly with sequence length (Katharopoulos et al., 2020; Gu & Dao, 2024; Dao & Gu, 2024; Peng et al., 2023; Yang et al., 2024b), making their computational advantage increasingly pronounced as resolution grows. This favorable scaling makes recurrent visual encoders particularly attractive for high-resolution vision-language models, where they can offer substantial efficiency gains.

However, favorable computational scaling alone does not ensure that vision-language representation quality is preserved at high resolutions. As shown in Fig. 1a, recent representative linear CLIP models like CLIMP, RWKV-CLIP (Shabtay et al., 2026; Gu et al., 2024) exhibit substantial degradation in zero-shot capacity when evaluated at practically real-world high-resolution scenarios. This exposes a fundamental paradox: *the high-resolution regime in which linear visual encoders yield their maximum computational benefits is precisely where pretrained linear models exhibit severe representational collapse.* By contrast, existing large-scale pretrained ViTs maintain both strong absolute performance and robust cross-resolution stability (Maninis et al., 2025; Cao et al., 2026). This contrast suggests a natural route. Rather than relearning such cross-resolution behavior through large-scale image-text pretraining, we can seek to inherit it from a strong pretrained ViT.



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.

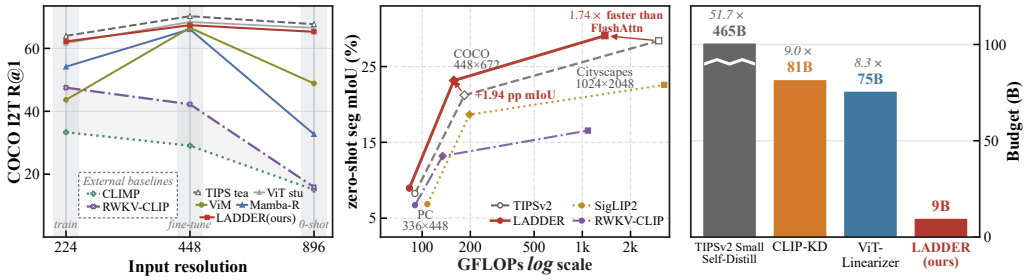



Figure 1: **Performance and efficiency comparison of LADDER. (a)** Zero-shot retrieval at resolutions of 224, 448, and 896. We convert TIPSv2-B into ViT and linear recurrent students using a naive protocol and into a hybrid linear encoder using LADDER, all under the same training budget. Linear CLIP baselines are evaluated using official checkpoints. **(b)** Zero-shot segmentation mIoU versus GFLOPs (log scale) across real-world datasets with different input resolutions at Base scale. **(c)** Training budgets measured in visual tokens of related methods.

Recent studies on Transformer linearization, cross-architecture distillation, and post-training hybridization suggest that strong pretrained models can be converted into more efficient architectures with limited additional training (Kasai et al., 2021; Zhang et al., 2024; Yang et al., 2024a; Wu et al., 2023; Zou et al., 2026). *However, whether straightforward linearization preserves the cross-resolution behavior of pretrained vision-language encoders remains unclear.* To investigate this, we conduct a controlled conversion study using TIPSv2 as the teacher (Fig. 1a). We linearize TIPSv2 into Vision Mamba (ViM) and Mamba-Reg (Zhu et al., 2024; Wang et al., 2025) using the same weight-transfer and distillation protocol, and include a ViT student trained with the same budget as an architectural control. The ViT student's attention QKV weights are excluded from transfer. Both recurrent students closely approach the teacher at the final fine-tuning resolution of 448, but lose performance at 224 and generalize poorly to the unseen resolution of 896. In contrast, the ViT student remains substantially closer to the teacher across all three resolutions. This architectural contrast under matched conversion conditions shows that naive linearization can largely recover performance at the final fine-tuning resolution, but falls substantially short of preserving the teacher's behavior across resolutions.

These findings motivate us to address cross-resolution generalization in VL-ViT linearization through hybrid attention architectures. In this work, we introduce LADDER (**L**ayer **AD**aptive **D**istillation into **E**fficient Hybrid Encode**R**), a lightweight framework that converts pretrained vision-language ViTs into hybrid recurrent encoders while preserving their cross-resolution behavior. Our design combines attention and recurrence within blocks and across layers. For global feature aggregation, we design Vision-Delta, a linear recurrent visual block that integrates **Attention-based CLS (AttnCLS)** with a Visual Delta Mixer. AttnCLS retains direct attention-based aggregation for CLS tokens alongside recurrent patch mixing, decoupling global readout from the block's recurrent scan. The Visual Delta Mixer adapts Gated DeltaNet (Yang et al., 2025) into a visual mixer tailored to pretrained ViT linearization and hybrid backbones.

To preserve dense representations across resolutions, we introduce **Resolution-Robust Layer-Adaptive Hybridization**. Full attention enables direct global interactions among spatial tokens. Previous hybrid visual backbones have shown that layer-wise attention hybridization can improve visual recognition performance (Hatamizadeh & Kautz, 2025; Zou et al., 2026; Li et al., 2026a), with its benefits depending on where full attention is retained. Under a fixed budget of full-attention blocks, we use reinforcement learning to jointly optimize a placement policy to seek optimal cross-resolution performance. Each candidate is evaluated as a complete discrete hybrid encoder with a specific attention placement and frozen weights, using a reward that measures how well the retained layers collectively recover teacher patch features across resolutions.

Using LADDER, we linearize the Base and Large ViT encoders of TIPSv2 into LADDER-B and LADDER-L, respectively. With only 22 epochs of image-only training on ImageNet-1K (Deng et al., 2009) (Fig. 1c), LADDER-L matches its teacher to within 1% point in image-text retrieval and maintains comparable linear-probing performance, while surpassing it on multiple zero-shot segmentation benchmarks. LADDER-B maintains near-teacher performance across resolutions and, at $896 \times 896$, outperforms CLIMP and RWKV-CLIP by 40.2–58.7 percentage points

in retrieval R@1, with gains of 13.7–19.1 points over naively linearized recurrent baselines. Additional experiments converting SigLIP2 into LADDER-Sig further demonstrate the framework's applicability across various teachers. On high-resolution Cityscapes inputs (Cordts et al., 2016), LADDER achieves up to $1.74\times$ the throughput of the FlashAttention teacher (Dao, 2023) while reducing FLOPs by more than half (Fig. 1b).

Our main contributions are as follows:
(1) We introduce LADDER, a lightweight framework for linearizing pretrained vision-language ViTs into hybrid recurrent encoders while preserving their cross-resolution capabilities.
(2) We design Vision-Delta, a lightweight linear visual block that combines AttnCLS for direct global aggregation with a Visual Delta Mixer for ViT linearization and hybrid backbones.
(3) We propose Resolution-Robust Layer-Adaptive Hybridization, which uses reinforcement learning to jointly select a fixed number of full-attention layers by evaluating complete hybrid encoders with cross-resolution teacher patch-feature feedback.
(4) Extensive experiments on TIPSv2 demonstrate that LADDER preserves vision-language and visual representation capabilities across resolutions and improves high-resolution efficiency with only 22 epochs of image-only training on ImageNet-1K. Additional experiments on SigLIP2 demonstrate its applicability beyond a single teacher family.

## 2 Related Work

### 2.1 ViT Linearization

Recently, linear recurrent visual architectures have demonstrated performance comparable to that of ViTs on standard vision tasks, while scaling linearly with sequence length. To avoid costly pretraining from scratch, pretrained ViT linearization offers a practical alternative. For instance, ViT-Linearizer (Wei & Chellappa, 2025) pairs activation matching with masked feature prediction, while ViT-AdaLA (Li et al., 2026b) employs progressive attention and feature alignment. Similarly, $T^5$ (Li et al., 2026c) exploits the architectural compatibility between softmax attention and test-time training for pretrained weight inheritance. While prior work primarily evaluates linearization on standard visual tasks, LADDER focuses on preserving vision-language alignment and zero-shot capabilities across input resolutions.

### 2.2 Transformer Hybrid Linearization and Layer Selection

Hybrid linearization combines full attention with efficient linear mixers, making attention placement a key design choice. Early works such as HALO and KL-LS use isolated layerwise scoring (Chen et al., 2026; Li et al., 2025), which fails to capture the joint effects of retaining multiple attention layers within a complete hybrid configuration. JetViT (Zou et al., 2026) evaluates complete configurations through downstream task guided beam search. However, its reliance on supernetwork training and repeated candidate evaluation increases the computational overhead of layer selection. More recently, FlashMorph (Lan et al., 2026) introduces *optimization-based* layer selection by jointly optimizing continuous coefficients that interpolate the outputs of frozen attention and linear branches. This captures inter-layer dependencies, but subsequent top-$K$ discretization introduces a structural discrepancy between the interpolated search model and the discrete deployment architecture. LADDER instead targets **cross-resolution robustness**, using teacher patch-feature recovery across resolutions to jointly learn shared layer and pairwise preferences through reinforcement learning. It directly evaluates complete discrete hybrid encoders, maintaining architectural consistency between search and deployment. Additional related work is discussed in the Appendix.

## 3 Method

In this section, we present LADDER, a framework for converting pretrained vision-language ViTs into hybrid recurrent visual encoders. LADDER addresses the loss of cross-resolution robustness in ViT linearization by hybridizing attention and recurrence within blocks and across layers. We first describe the overall linearization pipeline in Section 3.1, then detail its two main components.

### 3.1 LADDER: Layer Adaptive Distillation into Efficient Hybrid Encoder

Given a pretrained vision-language ViT, LADDER constructs a hybrid visual encoder by combining Vision-Delta blocks with selected full-attention blocks. We initialize the student with the teacher's

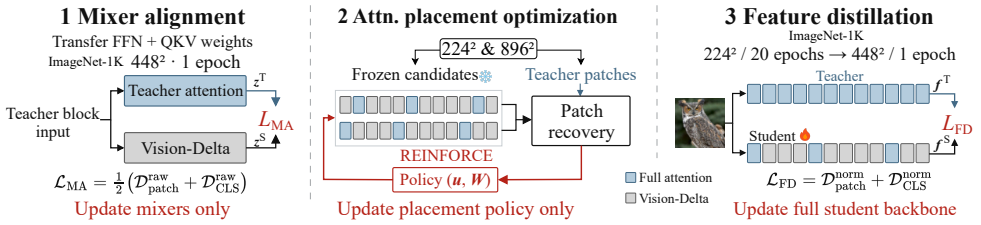



Figure 2: **LADDER linearization pipeline. (1)** Layerwise alignment of Vision-Delta mixers with teacher attention outputs. **(2)** Attention placement optimization using cross-resolution teacher patch-feature recovery with frozen encoder weights. **(3)** End-to-end distillation of final patch and CLS features with the selected attention placement fixed.

FFN and QKV weights and other compatible parameters. As illustrated in Fig. 2, LADDER comprises three stages: mixer alignment, attention placement optimization, and feature distillation. All stages use only images from the ImageNet-1K training set.

**Mixer alignment.** To adapt the recurrent mixers to pretrained representations, we first align each Vision-Delta mixer (Section 3.2) with the corresponding teacher attention branch at a fixed input resolution of $448 \times 448$. Each student mixer receives the corresponding teacher block input and matches the teacher's attention output before residual addition. The alignment objective is

$$\mathcal{L}_{\mathrm{MA}} = \frac{1}{2\sqrt{D}} \left( \mathbb{E}_{\mathrm{p}} \left[ \| \mathbf{z}^S_{\mathrm{p}} - \mathbf{z}^T_{\mathrm{p}} \|_2 \right] + \mathbb{E}_{\mathrm{cls}} \left[ \| \mathbf{z}^S_{\mathrm{cls}} - \mathbf{z}^T_{\mathrm{cls}} \|_2 \right] \right). \tag{1}$$

Here, $\mathbf{z}_{\mathrm{p}}$ and $\mathbf{z}_{\mathrm{cls}}$ denote patch and CLS token outputs, superscripts $S/T$ indicate the student/teacher, and $D$ is the feature dimension. The expectation averages over layers, images, and token positions within each group. Only the student mixers and their associated spatial parameters are updated. This provides an aligned Vision-Delta alternative at each layer for subsequent attention placement search.

**Attention placement optimization.** After alignment, we freeze the encoder weights and apply Resolution-Robust Layer-Adaptive Hybridization (Section 3.3) to identify attention placements that best preserve teacher patch representations across resolutions. We use reinforcement learning under a fixed full-attention budget and retain the placement with the highest learned policy score. The selected hybrid encoder inherits all weights of the retained full-attention blocks from the corresponding teacher blocks, while preserving the Vision-Delta parameters obtained through mixer alignment.

**Feature distillation.** We then train the selected hybrid encoder end-to-end through feature distillation. Teacher and student independently process the same image and output the full token sequence. We match their final patch and CLS features using

$$\mathcal{L}_{\mathrm{FD}} = \frac{1}{2} \left( \mathbb{E}_{\mathrm{p}} \left[ \| \hat{\mathbf{f}}^S_{\mathrm{p}} - \hat{\mathbf{f}}^T_{\mathrm{p}} \|_2^2 \right] + \mathbb{E}_{\mathrm{cls}} \left[ \| \hat{\mathbf{f}}^S_{\mathrm{cls}} - \hat{\mathbf{f}}^T_{\mathrm{cls}} \|_2^2 \right] \right). \tag{2}$$

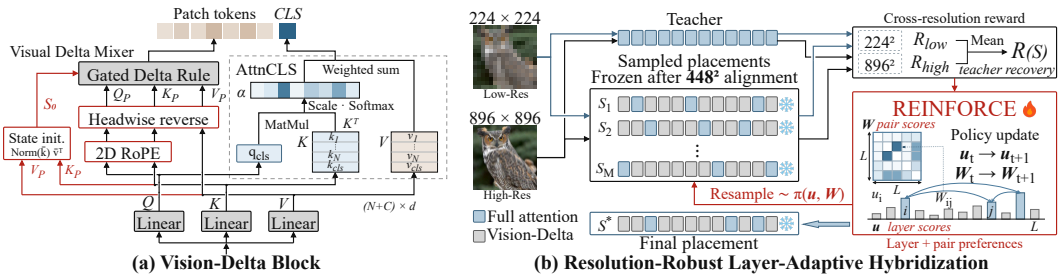
Here, $\hat{\mathbf{f}}$ denotes $\ell_2$-normalized final token features, and the expectation averages over images and token positions within each group. This objective transfers both dense patch representations and global vision-language features. The attention placement remains fixed while all trainable student parameters are updated. Training and implementation details are provided in the Appendix.

### 3.2 Vision-Delta

Vision-Delta is a lightweight recurrent visual block designed for pretrained ViT linearization. As illustrated in Fig. 3(a), it combines Attention-based CLS (AttnCLS) for global readout with a Visual Delta Mixer for patch processing.

#### 3.2.1 Attention-based CLS

In Transformer models, CLS tokens directly attend to the token sequence through an attention mechanism (Dosovitskiy et al., 2021; Touvron et al., 2021) to form global vision-language representations. A recurrent mixer, however, processes CLS tokens within the recurrent scan, where global readout relies on a fixed size state.




(a) Vision-Delta Block (b) Resolution-Robust Layer-Adaptive Hybridization

Figure 3: **Vision-Delta and Resolution-Robust Layer-Adaptive Hybridization. (a)** Vision-Delta combines recurrent patch mixing with direct CLS attention over shared pre-scan keys and values. **(b)** Each sampled attention placement is evaluated as a complete hybrid encoder at $224 \times 224$ and $896 \times 896$ with frozen weights and a fixed attention budget. The placement with the highest learned policy score is retained for subsequent distillation and deployment.

For an input-dependent linear recurrent update $\mathbf{S}_t = \mathbf{A}_t\mathbf{S}_{t-1} + \mathbf{B}_t$, a preceding write $\mathbf{B}_i$ contributes to the state at a CLS position $j$ through the transition product $\mathbf{\Phi}_{j,i} = \mathbf{A}_j \cdots \mathbf{A}_{i+1}$. Meanwhile, resolution changes can alter these propagation paths and reweight token contributions, making recurrent CLS readout susceptible to resolution shifts.

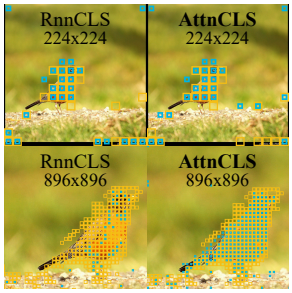



Figure 4: CLS attribution. Tea(yellow) & Stu(cyan).

To bridge this architectural gap, we introduce AttnCLS, which retains direct attention-based CLS aggregation alongside recurrent patch mixing. Using shared student projections, patch tokens enter the Visual Delta Mixer, while each CLS query attends to the pre-scan keys and values of all tokens, including CLS tokens. Specifically, $\mathbf{o}_{\text{cls}}^\top = \text{softmax}(\mathbf{q}_{\text{cls}}^\top \mathbf{K}^\top/\sqrt{d})\mathbf{V}$, where $d$ is the key dimension. The updated patch and CLS features are concatenated along the token dimension before shared gated normalization and output projection. This design decouples global readout from the current block's recurrent propagation, and the shared attention readout structure can further facilitate the transfer of the teacher's representations. With a fixed number of CLS tokens, this attention readout remains linear complexity. Figure 4 compares patch gradient attributions for CLS readout under a shared teacher feature target. Yellow and cyan boxes mark the highest attribution 10% of patches for the teacher and student. Both readouts show comparable attribution overlap with the teacher at $224\times224$, but only AttnCLS maintains substantial overlap at $896 \times 896$. Detailed derivations and visualization settings are provided in the Appendix.

### 3.2.2 Visual Delta Mixer

We adapt the gated delta rule (Yang et al., 2025) into a lightweight visual recurrent mixer for pretrained ViT linearization and efficient inference. For each head, the state update is $\mathbf{S}_t = \gamma_t(\mathbf{I} - \beta_t\mathbf{k}_t\mathbf{k}_t^\top)\mathbf{S}_{t-1} + \beta_t\mathbf{k}_t\mathbf{v}_t^\top$, with output $\mathbf{o}_t^\top = \mathbf{q}_t^\top\mathbf{S}_t$. Here, $\gamma_t$ and $\beta_t$ control state decay and writing, respectively, and $\mathbf{k}_t$ is the normalized scan key.

**Headwise reverse.** The causal scan in Gated DeltaNet provides only prefix context. Visual RNNs introduce bidirectional or multidirectional (Zhu et al., 2024; Liu et al., 2024) scanning to obtain global context, but repeating full-width scans increases recurrent computation. We therefore reverse the scan direction of half the $H$ heads while keeping the remaining heads in the forward direction. Queries, keys, values, and gates are reversed together, and outputs are restored to the original patch order. Each head scans once, providing bidirectional context within a single block with $HN$ recurrent updates in total. This supports layerwise hybridization without requiring alternating scan directions across layers. Since reversal operates on token order, it preserves the existing Q/K/V projection structure without introducing direction-specific parameters.

**State initialization.** The Visual Delta Mixer initializes each head with $\mathbf{S}_0 = \text{Normalize}(\bar{\mathbf{k}})\bar{\mathbf{v}}^\top$, where $\bar{\mathbf{k}}$ and $\bar{\mathbf{v}}$ are the mean patch keys and values after local preprocessing and projection. This image-conditioned initialization provides global context from the beginning of the scan.

We further apply 2D rotary embeddings (Su et al., 2024; Heo et al., 2024) to the patch queries and keys used in recurrent scanning. Together, these designs yield a lightweight visual recurrent

mixer with linear complexity in sequence length, tailored to pretrained ViT linearization and high-throughput inference.

### 3.3 Resolution-Robust Layer-Adaptive Hybridization

Attention layers allow each patch query to attend directly to the entire token sequence, enriching dense representations through global feature interaction. Prior studies have shown that attention placement affects hybrid model performance (Lan et al., 2026; Zou et al., 2026). We optimize attention placement through reinforcement learning, jointly learning individual layer preferences and pairwise retention preferences from cross-resolution teacher patch-feature recovery (Fig. 3b).

**Hybrid configurations.** For an encoder with $L$ layers, an attention placement is a $K$-element set $\mathcal{S} \subseteq \{1, \ldots, L\}$, whose elements are the indices of layers that retain full attention. Each layer executes full attention if its index belongs to $\mathcal{S}$, and Vision-Delta otherwise. All encoder weights remain frozen after mixer alignment at $448^2$ (Section 3.1). Evaluating each placement as a complete hybrid encoder captures the collective effects of retained attention layers and preserves architectural consistency between search and deployment.

**Policy parameterization.** To model interactions among multiple attention layers in a complete hybrid encoder, a direct approach is to learn an independent policy score for each of the $\binom{L}{K}$ possible placements, e.g., $\binom{24}{6} = 134,596$. Yet, the number of independent scores grows combinatorially, increasing the demands on optimization and exploration, and potentially leading to suboptimal placements (Section 4.3). Our method models each attention placement using shared scores for individual layers and pairs of layers. Specifically, $u_i$ is a learnable score for retaining layer $i$, while $W_{ij}$ is a learnable score for jointly retaining an unordered pair of layers $i$ and $j$ ($i < j$). These scores define the placement score and sampling rule:

$$F(\mathcal{S}) = \sum\nolimits_{i \in \mathcal{S}} u_i + \tfrac{1}{K-1} \sum\nolimits_{i<j,\, i,j \in \mathcal{S}} W_{ij}, \qquad \mathcal{S}_m \sim \pi(\mathbf{u}, \mathbf{W}), m = 1, \ldots, M. \tag{3}$$

The distribution $\pi(\mathbf{u}, \mathbf{W})$ is obtained by applying softmax over the scores of all valid placements. At each iteration, we sample $M$ complete placements independently with replacement and evaluate the corresponding frozen hybrid encoders. Their rewards update $\mathbf{u}$ and $\mathbf{W}$, after which we recompute the sampling probabilities and draw the next $M$ placements.

**Cross-resolution reward.** To select placements that preserve teacher representations under resolution shifts, we evaluate each frozen candidate at $224^2$ and $896^2$, both unseen during mixer alignment at $448^2$. All candidates use the same image batch at both resolutions. Let $s_r(\mathcal{S})$ denote the mean cosine similarity between final student and teacher patch features at corresponding spatial positions, averaged over images and patch positions at resolution $r^2$. The reward is $R(\mathcal{S}) = \frac{1}{2}\big(s_{224}(\mathcal{S})/C_{224} + s_{896}(\mathcal{S})/C_{896}\big)$, where $C_r > 0$ is a fixed, resolution-specific scale used to balance contributions across resolutions.

**Policy optimization.** We jointly optimize $\mathbf{u}$ and $\mathbf{W}$ to maximize the expected cross-resolution reward using REINFORCE (Williams, 1992) with a leave-one-out baseline (Ahmadian et al., 2024). For each of the $M > 1$ sampled placements, the baseline is the mean reward of the other $M - 1$ candidates evaluated on the same image batch. Entropy regularization encourages exploration, while a KL constraint limits changes between successive placement distributions. Only the $L + \binom{L}{2}$ parameters in $\mathbf{u}$ and $\mathbf{W}$ are updated, with no backpropagation through the frozen encoders. After optimization, we select the highest-scoring placement, $\mathcal{S}^\star = \arg\max_{|\mathcal{S}|=K} F(\mathcal{S})$. The policy is discarded, and this placement remains fixed during subsequent distillation and deployment. Calibration and optimization details are provided in the Appendix.

## 4 Experiments

We primarily evaluate LADDER by linearizing the Base and Large visual encoders of TIPSv2 (Cao et al., 2026) into LADDER-B/L, and additionally linearize SigLIP2 (Tschannen et al., 2025) into LADDER-Sig to assess LADDER's applicability across different VL-ViT teachers. Unless otherwise stated, LADDER-B/L refer to the TIPSv2-derived students. Training uses only ImageNet-1K images, without text or class label supervision. For TIPSv2, we perform one epoch of mixer alignment at $448^2$, followed by attention placement optimization with frozen encoder weights and feature

Table 1: **Vision-language and visual representation evaluation at Large scale.** Zero-shot retrieval reports R@1, segmentation mIoU, and ImageNet classification top-1 accuracy. **Bold** and underlined values indicate the best and second-best results in each column, respectively.

| Model | I→T Ret. ↑ | | T→I Ret. ↑ | | Zero-shot seg. ↑ | | | Linear probing | | | ImageNet ↑ | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | COCO | Flickr | COCO | Flickr | ADE150 | PC59 | VOC21 | VOC↑ | ADE↑ | NYU↓ | KNN | Linear |
| CLIP-L/14 | 56.3† | 85.2† | 36.5† | 65.0† | 6.4 | 11.9 | 21.8 | 74.5 | 39.0 | 0.553 | 79.8 | 84.3 |
| OpenCLIP-L/14 | 63.3† | 89.0† | 45.7† | 73.4† | 8.3 | 16.4 | 22.4 | 73.9 | 39.8 | 0.550 | 81.8 | 84.7 |
| MobileCLIP2-L/14 | 68.9 | 92.2 | 51.6 | 77.8 | 4.1 | 7.9 | 11.3 | 75.1 | 41.2 | 0.518 | 83.3 | 85.7 |
| dino.txt-L/14 | 65.0 | 91.4 | 45.4 | 77.1 | 20.6 | 30.9 | 26.1 | 78.5 | 46.2 | 0.410 | **84.7** | 85.1 |
| SigLIP2-L/16(512) | 71.9 | **95.3** | 55.2 | **85.1** | 10.8 | 24.1 | 26.0 | 75.2 | 43.6 | 0.484 | 84.4 | **87.3** |
| **LADDER-Sig-L/16** | 71.1 | 93.9 | 55.7 | 83.1 | 13.8 | 25.3 | 24.9 | 76.4 | 43.2 | 0.493 | 84.4 | 87.1 |
| TIPSv2-L/14 | **72.9** | 94.3 | **57.5** | 82.5 | 24.5 | 37.1 | **44.4** | **82.7** | **50.7** | **0.345** | 83.5 | 85.7 |
| **LADDER-L/14** | 72.0 | 93.7 | 56.9 | 82.0 | **25.0** | **39.4** | 43.6 | 82.5 | 49.8 | 0.352 | 83.3 | 86.1 |

† Results from publicly report, remaining results are **obtained using our unified evaluation protocol**.

distillation for 20 epochs at $224^2$ and one epoch at $448^2$. We optimize the placement policy for only 256 updates. Full attention is retained in $1/4$ of the encoder layers. Evaluation protocols, model configurations, layer selection, and additional experiments are provided in the **Appendix**.

### 4.1 Vision-Language and Visual Representation Evaluation

Following TIPSv2's evaluation protocols, we assess vision-language and visual representation tasks with frozen visual encoders locally under the same protocols. Table 1 compares LADDER with its teachers and representative vision-language encoders of comparable size (Cherti et al., 2023; Gadre et al., 2023; Jose et al., 2025). With only 22 epochs of image-only conversion training, LADDER-L remains within 1% point of TIPSv2-L across all four retrieval metrics and closely matches its visual probing performance. It also achieves the highest zero-shot segmentation scores on ADE150 and PC59 among the compared models. These results show that lightweight conversion preserves strong vision-language alignment and transferable visual representations.

To assess generality beyond TIPSv2, we additionally convert SigLIP2-L/16 into LADDER-Sig-L/16. The converted model largely preserves its teacher's retrieval and visual representation performance, while improving zero-shot segmentation on ADE150 and PC59. These results demonstrate that LADDER can be applied to different pretrained vision-language encoder families.

### 4.2 Cross-Resolution Robustness

To assess cross-resolution robustness and the contributions of LADDER's architectural designs, we evaluate Base- and Large-scale encoders at resolutions of 224, 448, and 896 (Table 2). Vision Mamba and Mamba-Reg are converted using the same weight-transfer procedure (FFN & QKV), mixer alignment, and training budget as LADDER. Although both baselines approach the teacher at the final fine-tuning resolution of 448, their performance gaps widen substantially at 224 and 896, uncovering the limitations of naive linearization in preserving cross-resolution capabilities.

**Intra-block hybridization.** AttnCLS improves the cross-resolution stability of global representations. Vision-Delta (RnnCLS) exhibits a similar pattern of cross-resolution degradation to Vision Mamba. Replacing recurrent CLS readout with AttnCLS leaves Vision-Delta (AttnCLS)'s Flickr30K retrieval performance nearly unchanged at 448, yet substantially improves performance at both lower and higher resolutions (224 and 896), with gains of up to **18.3** percentage points in I2T retrieval. These results demonstrate that decoupling global feature aggregation from the linear recurrent scan improves the robustness of global representations to resolution shifts.

**Layer-wise hybridization.** Retaining full-attention blocks further improves dense representations. Uniform Hybrid replaces $K$ evenly spaced Vision-Delta blocks with full-attention blocks, improving dense representation over Vision-Delta (AttnCLS) across the three resolutions. To further evaluate **Resolution-Robust Layer-Adaptive Hybridization**, we compare LADDER with uniform hybrid baselines at both Base and Large scales. At Base scale, the benefits of adaptive attention placement are most evident in dense zero-shot segmentation, while retrieval gains remain modest.

Table 2: **Cross-resolution evaluation of Base- and Large-scale visual encoders.** We report zero-shot retrieval R@1 on Flickr30K, zero-shot segmentation mIoU on ADE150, and kNN classification at input resolutions of 224, 448, and 896. Retrieval and kNN uses TIPSv2 Web cls. * Converted by us using the same weight transfer and training procedure.

| Model | Flickr I→T ↑ | | | Flickr T→I ↑ | | | ADE150 0-shot ↑ | | | IN-1K kNN ↑ | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 224 | 448 | 896 | 224 | 448 | 896 | 224 | 448 | 896 | 224 | 448 | 896 |
| TIPSv2-B *(teacher)* | 86.9 | 91.9 | 90.0 | 72.9 | 79.6 | 78.3 | 15.0 | 17.4 | 17.6 | 78.9 | 80.7 | 79.0 |
| *external linear backbones under the same linearization protocol* | | | | | | | | | | | | |
| Vision Mamba-B* | 69.7 | 89.8 | 73.2 | 59.6 | 77.8 | 61.8 | 13.2 | 16.1 | 11.7 | 66.2 | 79.9 | 69.5 |
| Mamba-Reg-B* | 78.6 | 89.9 | 67.8 | 66.0 | 77.3 | 59.4 | 13.6 | 16.0 | 11.2 | 73.2 | **80.0** | 68.3 |
| *LADDER method ablations* | | | | | | | | | | | | |
| Vision-Delta (RnnCLS) | 76.0 | 88.8 | 60.8 | 66.2 | 77.5 | 58.1 | 13.3 | 16.5 | 15.0 | 74.5 | 79.7 | 64.8 |
| Vision-Delta (AttnCLS) | 82.1 | 88.7 | 79.1 | 69.4 | 77.3 | 69.7 | 13.5 | 16.4 | 14.5 | 75.9 | 79.8 | 72.2 |
| Uniform Hybrid-B | 84.5 | **90.2** | 85.1 | **70.9** | 77.8 | 75.1 | 16.2 | **20.0** | 17.3 | 76.0 | 79.7 | 75.8 |
| **LADDER-B** | **85.7** | 89.6 | **86.9** | 70.8 | **77.9** | **76.2** | **17.2** | 19.6 | **18.7** | **77.8** | 79.9 | **78.5** |
| TIPSv2-L *(teacher)* | 90.8 | 94.8 | 94.6 | 77.8 | 82.0 | 81.8 | 22.3 | 24.5 | 25.1 | 81.9 | 83.5 | 82.8 |
| Uniform Hybrid-L | 88.3 | 93.6 | 91.3 | 76.5 | **81.8** | 80.6 | **22.8** | 24.9 | 24.0 | 80.8 | **83.3** | 81.7 |
| **LADDER-L** | **90.3** | **93.7** | **93.9** | **77.3** | 81.4 | **81.9** | 22.7 | **25.0** | **25.5** | **81.5** | **83.3** | **82.6** |
| *externally pretrained linear CLIP models* | | | | | | | | | | | | |
| CLIMP (VMamba-B) | 59.8 | 55.4 | 28.2 | 46.6 | 46.4 | 32.7 | 9.7 | 12.3 | 12.0 | 74.4 | 75.0 | 59.8 |
| RWKV-CLIP-B | 71.0 | 67.8 | 33.8 | 58.9 | 57.5 | 36.0 | 12.5 | 13.0 | 9.4 | 58.8 | 62.3 | 33.9 |

These benefits become more pronounced in the *deeper Large-scale model*, where Resolution-Robust Layer-Adaptive Hybridization improves both high-resolution retrieval and segmentation. At 896, LADDER-L outperforms Uniform Hybrid-L by 2.6 and 1.3 percentage points in image-to-text and text-to-image retrieval, respectively, and by 1.5 mIoU on ADE150.

Externally pretrained CLIMP and RWKV-CLIP exhibit substantial high-resolution performance degradation, undermining their ability to provide effective visual representations precisely where linear architectures offer their greatest computational advantages. LADDER-B instead maintains near-teacher retrieval and kNN performance across resolutions while exceeding the teacher in zero-shot segmentation, demonstrating its effectiveness as a hybrid recurrent vision-language encoder for high-resolution inputs.

### 4.3 Hybridization Policy

We evaluate different RL policy parameterizations for attention placement on frozen Large-scale encoders after mixer alignment (Table 3). Following Section 3.3, we compare independent scores per complete placement ($a_{\mathcal{S}}$), shared layer scores ($\mathbf{u}$), and shared layer and pair scores ($\mathbf{u}, \mathbf{W}$), with uniformly spaced attention as a fixed reference. The shared score policies use 256 updates, whereas the per-placement policy remains unconverged after 4096 updates. This indicates that assigning an independent score to each placement causes the number of policy parameters to grow combinatorially, increasing the burden of exploration and optimization. With cross-resolution rewards, our policy improves patch cosine scores over uniform placement by 1.1 and 2.8 points at 224 and 896, respectively, with only a 0.1-point decrease at 448. Using 448 rewards instead yields 96.2 at 448, exceeding the uniform baseline of 95.8, demonstrating that the framework also supports single-resolution optimization. Compared with layer only policy, adding pairwise scores further improves upon layer-only parameterization under both reward settings, supporting explicit modeling of joint layer retention. Base-scale results and further analyses are provided in the Appendix.

Table 3: **Policy parameterization and reward resolution.** Teacher–student patch cosine similarity (%) for frozen Large-scale hybrid encoders. † more optimization steps.

| Policy | Reward res. | Patch cosine ↑ | | |
|---|---|---|---|---|
| | | 224 | 448 | 896 |
| Uniform | – | 93.2 | 95.8 | 86.7 |
| Per-placement ($a_{\mathcal{S}}$)† | 224, 896 | 92.1 | 95.2 | 85.7 |
| Layer only ($\mathbf{u}$) | 224, 896 | 93.8 | 95.7 | 89.3 |
| **Layer + pair** ($\mathbf{u}, \mathbf{W}$) | **224, 896** | **94.3** | 95.7 | **89.5** |
| Layer only ($\mathbf{u}$) | 448 | 93.2 | 96.1 | 86.9 |
| Layer + pair ($\mathbf{u}, \mathbf{W}$) | 448 | 93.5 | **96.2** | 88.0 |

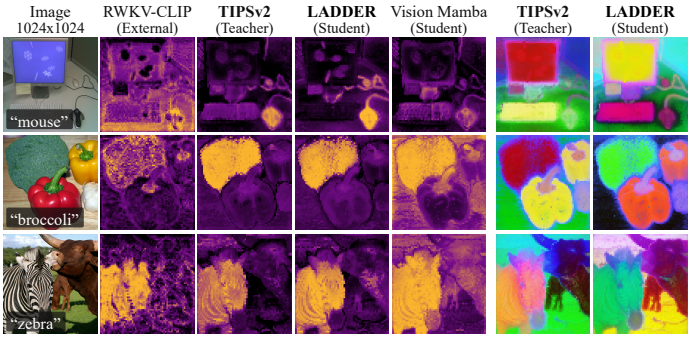



Figure 5: **High-resolution qualitative visualizations.** Patch-text activation map and PCA visualizations under 0-shot $1024^2$.

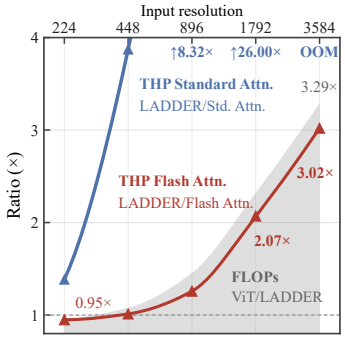



Figure 6: **Efficiency.** Ratio of Throughput&FLOPs(Base scale)

### 4.4 Further Analysis

**Qualitative visualizations.** Figure 5 presents patch-text activation maps (left) and PCA visualizations of patch features (right), with all images evaluated at $1024 \times 1024$. The activation maps reveal that LADDER retains the teacher's strong patch-text alignment, clearly localizing the small mouse and accurately capturing the spatial extent of the larger zebra. In contrast, the naively linearized Vision Mamba student struggles to precisely localize the mouse and produces spurious activations beyond target object regions.

**Inference efficiency.** Figure 6 compares BF16 inference on a single NVIDIA A100, with batch size 32 for both LADDER and the FlashAttention baseline. LADDER is slightly slower than the FlashAttention baseline at 224, but its throughput advantage increases with input resolution, reaching $3.02\times$ at 3584. The corresponding FLOPs ratio reaches $3.29\times$. These results show that the computational savings translate into substantial inference gains for high-resolution inputs, where long visual sequences make recurrent computation particularly beneficial.

**Visual Delta Mixer ablations.** Table 4 evaluates the mixer designs after one epoch of alignment at 448. Teacher and student independently process each image, and cosine similarity is measured between their final patch features. Removing headwise reverse, image-conditioned state initialization, or 2D RoPE reduces feature similarity. Compared with bidirectional scanning, the complete VDM achieves a similar cosine similarity of 0.958 versus 0.961, while increasing throughput from 392 to 489 images/s. This $24.7\%$ throughput improvement supports headwise reverse as an efficient design for visual recurrent conversion.

Table 4: **Visual Delta Mixer ablations.** Vision-Delta-Base variants undergo 1 epoch of mixer alignment at $448 \times 448$. Patch cosine similarity is measured between independently computed teacher and student final patch features.

| Headwise reverse | State init. | 2D RoPE | Bi-dir. | THP ↑ (img/s) | Patch cos. ↑ |
|---|---|---|---|---|---|
| ✓ | ✓ | ✓ | ✗ | 489 | 0.958 |
| ✗ | ✓ | ✓ | ✗ | 507 | 0.918 |
| ✓ | ✗ | ✓ | ✗ | 496 | 0.931 |
| ✓ | ✓ | ✗ | ✗ | 510 | 0.927 |
| ✗ | ✓ | ✓ | ✓ | 392 | 0.961 |

The first row is the full VDM. The last row replaces headwise reverse with bidirectional scanning.

## 5 Conclusion

We present LADDER, a lightweight framework for linearizing pretrained vision-language ViTs into hybrid recurrent encoders while preserving their cross-resolution capabilities. Our work reveals a paradox in VL-ViT linearization: *naively linearizing VL-ViT can loss teacher's cross-resolution image-text representation.* We address this limitation through architectural hybrid attention design. Vision-Delta pairs AttnCLS, which bridges the teacher–student gap in global readout, with a Visual Delta Mixer tailored to efficient ViT linearization. Resolution-Robust Layer-Adaptive Hybridization uses reinforcement learning to select attention placements, directly evaluating complete hybrid placements to maintain architectural consistency between selection and deployment. With short image-only training, LADDER closely matches various teachers across vision-language and visual representation evaluations while improving high-resolution inference efficiency.

### AI USE STATEMENT

In this work, AI tools were used in an auxiliary capacity for language polishing, literature search, LaTeX formatting, and assistance with experimental code implementation and debugging. We expressly declare that no AI tools were used to formulate research hypotheses, design the theoretical framework, implement the core methodological algorithms, interpret experimental conclusions, or generate any synthetic datasets. All core contributions are entirely the product of human effort, and the authors take full responsibility for the accuracy, originality, and integrity of the final manuscript.

## ETHICS STATEMENT

This work focuses on improving the efficiency of pretrained vision-language encoders using publicly available models and established benchmarks. We encourage responsible use of the proposed method and resulting models.

### REPRODUCIBILITY STATEMENT

We provide model configurations and attention placements in Appendix A, policy optimization details in Appendix D, and training settings in Appendix E. Evaluation protocols and efficiency measurement settings are described in Appendix E.5. All conversion experiments use publicly available teacher models and ImageNet-1K images. Code and trained models will be released.

## REFERENCES

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting REINFORCE-style optimization for learning from human feedback in LLMs. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12248–12267, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.662. URL https://aclanthology.org/2024.acl-long.662/.

Lucas Beyer, Pavel Izmailov, Alexander Kolesnikov, Mathilde Caron, Simon Kornblith, Xiaohua Zhai, Matthias Minderer, Michael Tschannen, Ibrahim Alabdulmohsin, and Filip Pavetic. Flexivit: One model for all patch sizes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14496–14506, June 2023.

Bingyi Cao, Koert Chen, Kevis-Kokitsi Maninis, Kaifeng Chen, Arjun Karpur, Ye Xia, Sahil Dua, Tanmaya Dabral, Guangxing Han, Bohyung Han, Joshua Ainslie, Alex Bewley, Mithun Jacob, René Wagner, Washington Ramos, Krzysztof Choromanski, Mojtaba Seyedhosseini, Howard Zhou, and Andre Araujo. Tipsv2: Advancing vision-language pretraining with enhanced patch-text alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29325–29335, June 2026.

Yingfa Chen, Zhen Leng Thai, Zihan Zhou, Zhu Zhang, Xingyu Shen, Shuo Wang, Chaojun Xiao, Xu Han, and Zhiyuan Liu. Hybrid linear attention done right: Efficient distillation and effective architectures for extremely long contexts, 2026. URL https://arxiv.org/abs/2601.22156.

Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2829, June 2023.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning, 2023. URL https://arxiv.org/abs/2307.08691.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10041–10071. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/dao24a.html.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness, 2022. URL https://arxiv.org/abs/2205.14135.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. IEEE, 2009.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021. URL https://arxiv.org/abs/2010.11929.

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 Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 83166–83182, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/ce65173b994cf7c925c71b482ee14a8d-Paper-Conference.pdf.

Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, Eyal Orgad, Rahim Entezari, Giannis Daras, Sarah Pratt, Vivek Ramanujan, Yonatan Bitton, Kalyani Marathe, Stephen Mussmann, Richard Vencu, Mehdi Cherti, Ranjay Krishna, Pang Wei Koh, Olga Saukh, Alexander J Ratner, Shuran Song, Hannaneh Hajishirzi, Ali Farhadi, Romain Beaumont, Sewoong Oh, Alex Dimakis, Jenia Jitsev, Yair Carmon, Vaishaal Shankar, and Ludwig Schmidt. Datacomp: In search of the next generation of multimodal datasets. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 27092–27112. Curran Associates, Inc., 2023. doi: 10.52202/075280-1179. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/56332d41d55ad7ad8024aac625881be7-Paper-Datasets_and_Benchmarks.pdf.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces, 2024. URL https://arxiv.org/abs/2312.00752.

Tiancheng Gu, Kaicheng Yang, Xiang An, Ziyong Feng, Dongnan Liu, Weidong Cai, and Jiankang Deng. RWKV-CLIP: A robust vision-language representation learner. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 4799–4812, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.276. URL https://aclanthology.org/2024.emnlp-main.276/.

Yuxian Gu, Qinghao Hu, Haocheng Xi, Junyu Chen, Shang Yang, Song Han, and Han Cai. Jet-nemotron: Efficient language model with post neural architecture search. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 47191–47218. Curran Associates, Inc., 2025. doi: 10.52202/085713-1572. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/4364fbe773a77fb373df5c7f1d4cbb53-Paper-Conference.pdf.

Ali Hatamizadeh and Jan Kautz. Mambavision: A hybrid mamba-transformer vision backbone. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25261–25270, June 2025.

Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. Rotary position embedding for vision transformer. In *European Conference on Computer Vision (ECCV)*, 2024.

Weiquan Huang, Yifei Shen, and Yifan Yang. Clip-mamba: Clip pretrained mamba models with ood and hessian evaluation, 2024. URL https://arxiv.org/abs/2404.19394.

Cijo Jose, Théo Moutakanni, Dahyun Kang, Federico Baldassarre, Timothée Darcet, Hu Xu, Daniel Li, Marc Szafraniec, Michaël Ramamonjisoa, Maxime Oquab, Oriane Siméoni, Huy V. Vo, Patrick Labatut, and Piotr Bojanowski. Dinov2 meets text: A unified framework for image- and pixel-level vision-language alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24905–24916, June 2025.

Jungo Kasai, Hao Peng, Yizhe Zhang, Dani Yogatama, Gabriel Ilharco, Nikolaos Pappas, Yi Mao, Weizhu Chen, and Noah A. Smith. Finetuning pretrained transformers into RNNs. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 10630–10643, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.830. URL https://aclanthology.org/2021.emnlp-main.830/.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5156–5165. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/katharopoulos20a.html.

Disen Lan, Jianbin Zheng, Yuxi Ren, Xin Xia, Xuanda Wang, Xuefeng Xiao, Xipeng Qiu, and Yu Cheng. Morphing into hybrid attention models. *arXiv preprint arXiv:2606.30562*, 2026.

Ruibang Li, Guan Luo, Yiwei Zhang, Jin Gao, Bing Li, and Weiming Hu. Sola-vision: Fine-grained layer-wise linear softmax hybrid attention. *arXiv preprint arXiv:2601.11164*, 2026a.

Yanhong Li, Songlin Yang, Shawn Tan, Mayank Mishra, Rameswar Panda, Jiawei Zhou, and Yoon Kim. Distilling to hybrid attention models via kl-guided layer selection, 2025. URL https://arxiv.org/abs/2512.20569.

Yifan Li, Seunghyun Yoon, Viet Dac Lai, Franck Dernoncourt, Jason Kuen, Yu Kong, and Trung Bui. Vit-adala: Adapting vision transformers with linear attention, 2026b. URL https://arxiv.org/abs/2603.16063.

Yining Li, Dongchen Han, Zeyu Liu, Hanyi Wang, Yulin Wang, and Gao Huang. Linearizing vision transformer with test-time training, 2026c. URL https://arxiv.org/abs/2605.02772.

Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. Vmamba: Visual state space model. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 103031–103063. Curran Associates, Inc., 2024. doi: 10.52202/079017-3273. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/baa2da9ae4bfed26520bb61d259a3653-Paper-Conference.pdf.

Kevis-Kokitsi Maninis, Kaifeng Chen, Soham Ghosh, Arjun Karpur, Koert Chen, Ye Xia, Bingyi Cao, Daniel Salz, Guangxing Han, Jan Dlabal, Dan Gnanapragasam, Mojtaba Seyedhosseini, Howard Zhou, and André Araujo. TIPS: Text-Image Pretraining with Spatial Awareness. In *ICLR*, 2025.

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 Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 14048–14077, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.936. URL https://aclanthology.org/2023.findings-emnlp.936/.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Nimrod Shabtay, Itamar Zimerman, Eli Schwartz, and Raja Giryes. Climp: Contrastive language-image mamba pretraining. *arXiv preprint arXiv:2601.06891*, 2026.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, February 2024. ISSN 0925-2312. doi: 10.1016/j.neucom.2023.127063. URL http://dx.doi.org/10.1016/j.neucom.2023.127063.

Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 32–42, October 2021.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Feng Wang, Jiahao Wang, Sucheng Ren, Guoyizhe Wei, Jieru Mei, Wei Shao, Yuyin Zhou, Alan Yuille, and Cihang Xie. Mamba-reg: Vision mamba also needs registers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14944–14953, June 2025.

Guoyizhe Wei and Rama Chellappa. Vit-linearizer: Distilling quadratic knowledge into linear-time vision models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20737–20747, October 2025.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3):229–256, May 1992. ISSN 1573-0565. doi: 10.1007/BF00992696. URL https://doi.org/10.1007/BF00992696.

Kan Wu, Houwen Peng, Zhenghong Zhou, Bin Xiao, Mengchen Liu, Lu Yuan, Hong Xuan, Michael Valenzuela, Xi (Stephen) Chen, Xinggang Wang, Hongyang Chao, and Han Hu. Tinyclip: Clip distillation via affinity mimicking and weight inheritance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21970–21980, October 2023.

Chuanguang Yang, Zhulin An, Libo Huang, Junyu Bi, Xinqiang Yu, Han Yang, Boyu Diao, and Yongjun Xu. Clip-kd: An empirical study of clip model distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15952–15962, June 2024a.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 56501–56523. PMLR, 21–27 Jul 2024b. URL https://proceedings.mlr.press/v235/yang24ab.html.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving mamba2 with delta rule. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=r8H7xhYPwz.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models, 2022. URL https://arxiv.org/abs/2205.01917.

Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18123–18133, June 2022.

Michael Zhang, Kush Bhatia, Hermann Kumbong, and Christopher Re. The hedgehog & the porcupine: Expressive linear attentions with softmax mimicry. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 53551–53580, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/ebba182cb97864368fdb6ae00773a5e4-Paper-Conference.pdf.

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 Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 62429–62442. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/zhu24f.html.

Dongyun Zou, Zhuoyang Zhang, Junyu Chen, Wenkun He, Qinhe Peng, Hanrong Ye, Yao Lu, Hongxu Yin, Yu Wang, Song Han, and Han Cai. Jetvit: Efficient high-resolution vision transformer with post-training attention search, 2026. URL https://arxiv.org/abs/2605.26636.

# A Model Configurations

Table A1 summarizes the main LADDER models. LADDER-B and LADDER-L use the official TIPSv2-B/14 and TIPSv2-L/14 teachers, while LADDER-Sig-L uses SigLIP2-L/16 at $512^2$. Attention placements remain fixed during feature distillation and inference.

Table A1: **Model configurations.** Parameter counts cover the visual encoder, including its global readout, and exclude the text encoder and selection policy. Attention placements use one-based layer indices.

| | LADDER-B | LADDER-L | LADDER-Sig-L |
|---|---|---|---|
| Teacher | TIPSv2-B/14 | TIPSv2-L/14 | SigLIP2-L/16 |
| Patch size | 14 | 14 | 16 |
| Encoder layers | 12 | 24 | 24 |
| Feature dimension | 768 | 1024 | 1024 |
| Attention heads $\times$ dimension | $12 \times 64$ | $16 \times 64$ | $16 \times 64$ |
| Delta heads $\times$ dimension | $12 \times 64$ | $16 \times 64$ | $16 \times 64$ |
| FFN hidden dimension | 3072 | 4096 | 4096 |
| FFN activation | GELU | GELU | GELU-tanh |
| Attention / Vision-Delta blocks | 3 / 9 | 6 / 18 | 6 / 18 |
| Visual parameters (M) | 92.07 | 324.37 | 337.04 |
| Global readout | Web / Syn CLS | Web / Syn CLS | Native MAP head |
| Attention placement | $\{4, 5, 12\}$ | $\{6, 7, 8, 11, 12, 24\}$ | $\{2, 7, 8, 10, 12, 20\}$ |

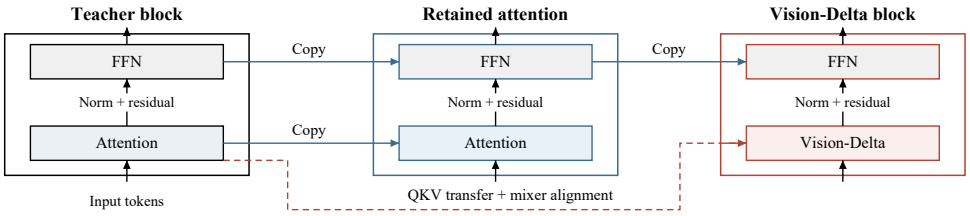



Figure A1: **Weight inheritance in LADDER.** Retained attention blocks copy teacher weights. Vision-Delta blocks inherit compatible parameters and align their mixers to the teacher.

**Teacher interfaces.** TIPSv2 students inherit the Web and Syn CLS tokens. Within each Vision-Delta block, patches enter the Visual Delta Mixer, while CLS tokens use AttnCLS over the shared keys and values before recurrent scanning. LADDER-Sig-L also includes auxiliary tokens updated by AttnCLS that act as register tokens within the backbone, while retaining SigLIP2's native multi-head attention pooling (MAP) head for global readout.

**Weight inheritance.** The student inherits compatible patch embeddings, positional embeddings, normalization layers, FFNs, and QKV projections (Fig. A1). Vision-Delta preserves the teacher's head count and dimensions, allowing direct QKV transfer. Retained attention blocks inherit the corresponding teacher blocks; other layers use the aligned Vision-Delta mixers and inherited FFNs. TIPSv2 students output the final token sequence without an additional projection. Positional embeddings are interpolated when the patch grid changes.

**Resolution settings.** TIPSv2 students use $448^2$ for mixer alignment, $224^2/896^2$ for placement search, and $224^2$ followed by $448^2$ for feature distillation. LADDER-Sig-L uses $256^2/512^2/1024^2$ in place of $224^2/448^2/896^2$. Both settings yield 256, 1,024, and 4,096 patch tokens, respectively, with a fourfold increase between successive resolutions. These settings provide representative low, intermediate, and high resolutions for studying cross-resolution behavior. The framework does not require these particular input sizes. For TIPSv2 conversion, encoder weights are trained at $224^2$ and $448^2$, while $896^2$ is used only to evaluate frozen candidates during attention placement search. Thus, $896^2$ is unseen during encoder weight training but participates in architecture selection. Whole-image zero-shot segmentation on Cityscapes at $1024 \times 2048$ further demonstrates that the selected placement remains effective beyond the resolutions used for both weight training and placement search (Fig. 1b).

# B Extended Related Work

## B.1 Linear Visual Encoders and Linear CLIP Models

Linear recurrent and state space architectures reduce the cost of visual token mixing through compact recurrent states. Vision Mamba uses bidirectional state space modeling, VMamba introduces two-dimensional selective scanning, and Vision-RWKV combines recurrent aggregation with spatial token shifting (Zhu et al., 2024; Liu et al., 2024; Duan et al., 2025). These models provide efficient alternatives to full-attention ViTs for visual recognition. CLIP-Mamba and RWKV-CLIP extend recurrent visual encoders to contrastive vision-language pretraining (Huang et al., 2024; Gu et al., 2024). RWKV-CLIP also enriches textual supervision using web descriptions, generated captions, and image tags. CLIMP adopts Mamba in both image and text towers, reporting efficiency and retrieval gains over its Transformer baselines at high resolutions (Shabtay et al., 2026). However, representative recurrent CLIP models still exhibit substantial zero-shot degradation in this regime (Table 2). LADDER instead converts a strong pretrained vision-language ViT through lightweight image-only training, improving high-resolution efficiency while preserving cross-resolution capabilities.

## B.2 Pretrained ViT Linearization and Distillation

Converting strong pretrained ViTs offers an alternative to training new efficient encoders from scratch. ViT-Linearizer combines intermediate activation matching with masked feature prediction (Wei & Chellappa, 2025). Its activation matching explicitly constructs pairwise token similarity matrices, introducing quadratic supervision cost that grows with input resolution. Its standard distillation schedule also uses 300 ImageNet-1K epochs, comparable in epoch count to common pretraining schedules. ViT-AdaLA progressively aligns attention outputs and final features before supervised downstream adaptation (Li et al., 2026b). $T^5$ exploits the structural compatibility between softmax attention and test-time training to inherit pretrained weights (Li et al., 2026c). CLIP-KD and TinyCLIP further demonstrate the effectiveness of feature matching, affinity transfer, and weight inheritance for vision-language distillation (Yang et al., 2024a; Wu et al., 2023).

Existing ViT linearization studies primarily evaluate visual recognition and downstream adaptation. Preserving image-text alignment after architectural conversion, particularly across input resolutions, remains insufficiently studied. LADDER examines this question through both global image-text alignment and dense patch-text representations. Our controlled conversions distinguish recovery at the final training resolution from preservation of the teacher's cross-resolution behavior.

## B.3 Hybrid Conversion and Attention Placement

Hybrid conversion retains a small number of full-attention layers while replacing the remaining mixers with efficient alternatives. HALO and KL-LS estimate layer importance through individual replacements or restorations, using task performance or teacher–student divergence (Chen et al., 2026; Li et al., 2025). Such isolated scores do not directly capture the complementarity and redundancy of multiple retained attention layers. PostNAS and JetViT train supernetworks and use beam search to evaluate complete hybrid configurations (Gu et al., 2025; Zou et al., 2026). This accounts for the collective behavior of each candidate, but requires supernetwork training and repeated candidate evaluation. JetViT guides search directly with downstream metrics, such as segmentation mIoU and depth estimation performance. LADDER instead selects attention placements through teacher patch-feature recovery across resolutions, without task-specific heads or labels.

FlashMorph jointly optimizes continuous coefficients that interpolate the outputs of frozen attention and linear branches, then retains the layers with the largest coefficients under a fixed attention budget (Lan et al., 2026). Joint optimization captures dependencies among layers, but discretization creates a structural discrepancy between the interpolated search model and the deployed encoder. LADDER uses reinforcement learning to jointly optimize shared scores for individual layers and pairs of layers. It samples and evaluates complete discrete hybrid encoders, with one block executed at each layer, maintaining architectural consistency between search and deployment.

Table A2 summarizes the selection criteria and mechanisms of these methods.

Table A2: **Comparison of hybrid attention placement methods.** For non-isolated selection, ✓ denotes joint optimization of placement variables and ✗ individual layer scoring. △ denotes stage-wise search over complete candidates; intermediate pruning may exclude combinations that become favorable only after further layer changes. Optim.-based refers to direct gradient optimization of placement variables. Discrete evaluation uses one mixer per layer during selection. – denotes no search.

| Method | Selection criterion | Selection mechanism | Non-isolated selection | Optim.-based | Discrete evaluation |
|---|---|---|---|---|---|
| Uniform | – | Fixed pattern | – | – | – |
| HALO | Recall / reasoning | Layer replacement | ✗ | ✗ | ✓ |
| KL-LS | Teacher–student KL | Layer restoration | ✗ | ✗ | ✓ |
| PostNAS | Downstream task metrics | Beam search | △ | ✗ | ✓ |
| JetViT | Downstream task metrics | Beam search | △ | ✗ | ✓ |
| FlashMorph | Retrieval feature alignment | Continuous gates | ✓ | ✓ | ✗ |
| **LADDER** | Cross-resolution patch recovery | RL (layer + pair) | ✓ | ✓ | ✓ |

### B.4 Global Readout and Visual Hybridization

Attention-based global aggregation has established precedents. CaiT separates patch self-attention from subsequent class-attention layers, where CLS queries aggregate token information while patch features remain unchanged within the readout stage (Touvron et al., 2021). Our AttnCLS addresses a different computational arrangement: each Vision-Delta block performs recurrent patch mixing alongside direct CLS attention to shared, pre-scan keys and values. Its contribution lies in this integration for recurrent conversion and its cross-resolution behavior, rather than a new attention aggregation formula. Across layers, MambaVision and SoLA-Vision demonstrate the value of combining efficient mixers with attention for visual recognition (Hatamizadeh & Kautz, 2025; Li et al., 2026a). SoLA-Vision specifically investigates the placement of sparse global attention within a linear backbone. LADDER builds on the motivation for visual hybridization while addressing post-training conversion of vision-language encoders and learning attention placements for cross-resolution representation preservation.

## C Attention-based CLS

### C.1 Recurrent Propagation and Global Readout

We analyze the recurrent CLS alternative used in Table 2. Its two CLS tokens follow the patch sequence and participate in the recurrence. For reverse heads, patches are traversed in reverse order while CLS tokens remain at the tail. Thus, both scan directions aggregate patch information before the global readout.

For the input-dependent update $\mathbf{S}_t = \mathbf{A}_t\mathbf{S}_{t-1} + \mathbf{B}_t$, define $\mathbf{\Phi}_{j,i} = \mathbf{A}_j \cdots \mathbf{A}_{i+1}$ and $\mathbf{\Phi}_{j,j} = \mathbf{I}$. Expanding the recurrence gives

$$\mathbf{S}_j = \mathbf{\Phi}_{j,0}\mathbf{S}_0 + \sum_{i=1}^{j} \mathbf{\Phi}_{j,i}\mathbf{B}_i, \qquad \mathbf{o}_j^\top = \mathbf{q}_j^\top \mathbf{\Phi}_{j,0}\mathbf{S}_0 + \sum_{i=1}^{j} \mathbf{q}_j^\top \mathbf{\Phi}_{j,i}\mathbf{B}_i. \tag{C.1}$$

Each write reaches the readout through a product of subsequent transitions. For the gated delta update, $\mathbf{A}_t = \gamma_t(\mathbf{I} - \beta_t\mathbf{k}_t\mathbf{k}_t^\top)$ and $\mathbf{B}_t = \beta_t\mathbf{k}_t\mathbf{v}_t^\top$. Changing the patch grid changes the sequence length, visual features, and transition products. These changes can reweight patch contributions to the global readout, making recurrent CLS aggregation sensitive to resolution shifts.

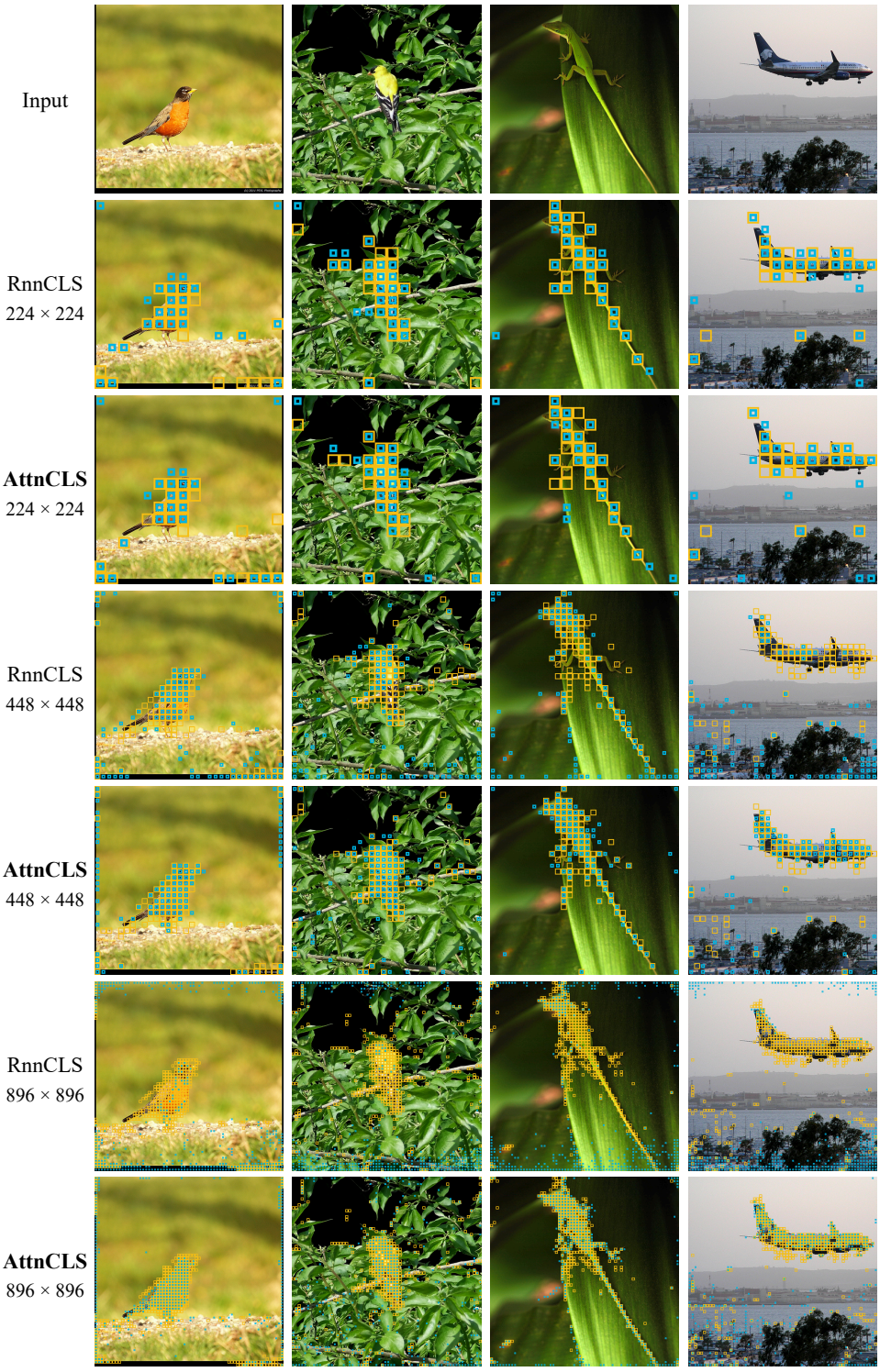



Figure A2: **CLS attribution across resolutions.** RnnCLS and AttnCLS students aligned at $448^2$ are evaluated at $224^2$, $448^2$, and $896^2$. Yellow and cyan boxes mark the top 10% of teacher and student patch attributions under the same teacher target. AttnCLS maintains closer agreement with teacher regions as resolution changes.

### C.2 Direct Attention Alongside Recurrent Patch Mixing

AttnCLS separates the two token roles within each Vision-Delta block. For $N$ patch tokens and $C$ CLS tokens, let $\mathbf{Q}_{\text{cls}} \in \mathbb{R}^{C\times d}$ contain the CLS queries, and let $\mathbf{K} \in \mathbb{R}^{(N+C)\times d}$ and $\mathbf{V} \in \mathbb{R}^{(N+C)\times d_v}$ be the shared projections before recurrent scanning, where $d$ and $d_v$ denote the key and value dimensions. For each head, the global output is

$$\mathbf{O}_{\text{cls}} = \text{softmax}\left(\frac{\mathbf{Q}_{\text{cls}}\mathbf{K}^\top}{\sqrt{d}}\right)\mathbf{V}. \tag{C.2}$$

CLS queries read all input keys and values, including those of the CLS tokens themselves, while patch tokens enter the Visual Delta Mixer. The outputs are concatenated before shared gated normalization and output projection. AttnCLS reads the projected features before recurrent scanning, decoupling global readout from propagation through the current block's recurrent state. Earlier blocks can still influence these input features.

### C.3 Gradient Attribution Visualization

Figure A2 compares RnnCLS and AttnCLS students aligned at $448^2$, with evaluation at $224^2$, $448^2$, and $896^2$. At $224^2$ and $448^2$, both readouts highlight many regions emphasized by TIPSv2-B/14. At $896^2$, RnnCLS attribution shifts toward image boundaries and background regions, while AttnCLS maintains closer agreement with the teacher across the displayed objects. These observations support the role of direct CLS attention in preserving global aggregation across resolutions, complementing Table 2.

**Visualization protocol.** Both students use the Base configuration. We analyze the final Web CLS after LayerNorm and compute attribution at the input patch tokens of the last block. Each image is center-cropped to a square and bicubically resized to the three resolutions, using identical inputs for teacher and students. For each image and resolution, we fix the teacher direction $\mathbf{t} = \text{sg}(\mathbf{c}_T/\|\mathbf{c}_T\|_2)$ and use the target $s_m = \mathbf{c}_m^\top \mathbf{t}$, where $\mathbf{c}_m$ is the final Web CLS of model $m$ and sg stops gradients. The CLS features receive no additional L2 normalization. Given the last block's input patch features $\mathbf{X}_m$, patch $j$ receives the attribution

$$I_{m,j} = \sum\nolimits_{d=1}^{D} \left| X_{m,j,d} \frac{\partial s_m}{\partial X_{m,j,d}} \right|. \tag{C.3}$$

Here, $D$ is the feature dimension; absolute values are taken before summing over channels. All models remain frozen, and gradients pass through the last block and final normalization. Yellow and cyan boxes mark the top 10% of teacher and student patch attributions, respectively. Both students share the same teacher mask, recomputed at each resolution. These maps show gradient attribution to global readout rather than attention weights.

## D Resolution-Robust Layer-Adaptive Hybridization

We implement the policy in Section 3.3 by computing $F(\mathcal{S})$ for all valid placements and applying softmax to their scores. There are 220 placements for $(L, K) = (12, 3)$ and 134,596 for $(24, 6)$, but only sampled placements require visual forward passes. The shared layer and pair scores reduce the number of policy parameters to 78 and 300, respectively. All scores are initialized to zero, giving a uniform initial distribution. We learn one $W_{ij}$ for each unordered layer pair and display $\mathbf{W}$ as a symmetric matrix with a zero diagonal. The factor $1/(K-1)$ in Eq. 3 accounts for each retained layer participating in $K-1$ pairs.

### D.1 Reward Calibration and Policy Optimization

At each search resolution, teacher and candidate encoders process the same image batch using identical center-square crops without random flipping. Following Section 3.3, $s_r(\mathcal{S})$ denotes their mean patch cosine similarity at resolution $r^2$. We calibrate each normalization scale $C_r$ as the mean loss $1 - s_r$ of the all-Vision-Delta encoder on 1,024 randomly sampled ImageNet-1K training images. These scales remain fixed during search (Table A3), balancing the different representation errors at the two resolutions.

In the implementation, we subtract the similarity of the all-Vision-Delta reference on the same image batch, denoted by $s_r^{\Delta}$. For TIPSv2, the reward is

$$\widetilde{R}(\mathcal{S}) = \frac{1}{2}\left[\frac{s_{224}(\mathcal{S}) - s_{224}^{\Delta}}{C_{224}} + \frac{s_{896}(\mathcal{S}) - s_{896}^{\Delta}}{C_{896}}\right]. \tag{D.1}$$

This measures recovery of teacher patch representations relative to the recurrent reference. The subtracted reference is shared by every placement evaluated on the batch and cancels after subtracting the leave-one-out baseline. Thus, Eq. D.1 produces the same policy update as the cosine reward in the main text. SigLIP2 uses resolutions 256 and 1024 in place of 224 and 896. Search images are randomly drawn from the ImageNet-1K training set; the mixer alignment resolution is excluded from the reward.

Table A3: **Fixed reward normalization scales.** Each value is the mean all-Vision-Delta patch loss $1 - s_r$ over 1,024 calibration images.

| Model | Search resolutions | Lower-resolution $C_r$ | Higher-resolution $C_r$ |
|---|---|---|---|
| LADDER-B | 224, 896 | 0.39450091 | 0.48146208 |
| LADDER-L | 224, 896 | 0.12021451 | 0.22581605 |

**Policy optimization and final selection.** At each update, we sample $M = 8$ complete placements independently with replacement and evaluate them on the same image batch. Let $R_m = \widetilde{R}(\mathcal{S}_m)$ be the reward of the $m$th sample. Its baseline, $b_m = \frac{1}{M-1}\sum_{n\neq m} R_n$, is the mean reward of the other samples. REINFORCE (Williams, 1992; Ahmadian et al., 2024) updates $\mathbf{u}$ and $\mathbf{W}$ by minimizing

$$\mathcal{L}_{\text{policy}} = -\frac{1}{M}\sum_{m=1}^{M}(R_m - b_m)\log \pi(\mathcal{S}_m; \mathbf{u}, \mathbf{W}) - \lambda H(\pi), \tag{D.2}$$

where rewards and baselines are held constant when computing gradients. The reward difference encourages higher probability for placements that outperform the other samples, while the entropy $H(\pi)$ over all valid placements encourages exploration. We set $\lambda = 0.002$. Duplicate samples share a visual forward pass but retain their multiplicity in the loss. No gradients pass through the frozen encoders.

We use AdamW with zero weight decay for 256 updates; model-specific settings are listed in Table A4. Each update is constrained by $D_{\text{KL}}(\pi_{\text{old}} \| \pi_{\text{new}}) \leq 0.001$. If this bound is exceeded, we restore the policy and optimizer state and halve the proposed learning rate, allowing up to six retries. An update that still violates the bound is skipped, and the next iteration starts with the configured learning rate. At the end of search, we select $\mathcal{S}^{\star} = \arg\max_{|\mathcal{S}|=K} F(\mathcal{S})$. The policy is discarded, and this attention placement remains fixed during feature distillation and deployment.

Table A4: **Attention placement optimization.** All models use eight samples per update, 256 updates, entropy coefficient 0.002, and KL bound 0.001. Batch denotes the number of images per update.

| Model | $(L, K)$ | Policy parameters | Batch | Learning rate |
|---|---|---|---|---|
| LADDER-B | $(12, 3)$ | 78 | 8 | 0.02 |
| LADDER-L | $(24, 6)$ | 300 | 32 | 0.02 |

### D.2 Policy Evolution and Placement Stability

Figures A3 and A4 show the evolution of $\mathbf{u}$ and $\mathbf{W}$ during Base and Large conversion. Each snapshot pairs a bar chart of individual layer scores with a heatmap of scores for retaining two layers together. Blue bars mark layers in the complete placement with the highest $F(\mathcal{S})$ at that update. The same axis limits and color scale are used throughout each figure, so changes can be compared directly across updates.

For Base, the highlighted placement is $\{4, 5, 12\}$ from the first nonzero snapshot at update 32 onward. Its layer scores become increasingly prominent as optimization proceeds. The pair scores also become more selective: by update 128, the three strongest pairs are $(4, 5)$, $(4, 12)$, and $(5, 12)$, all within the selected group. Thus, the policy develops consistent preferences for these layers both individually and together.

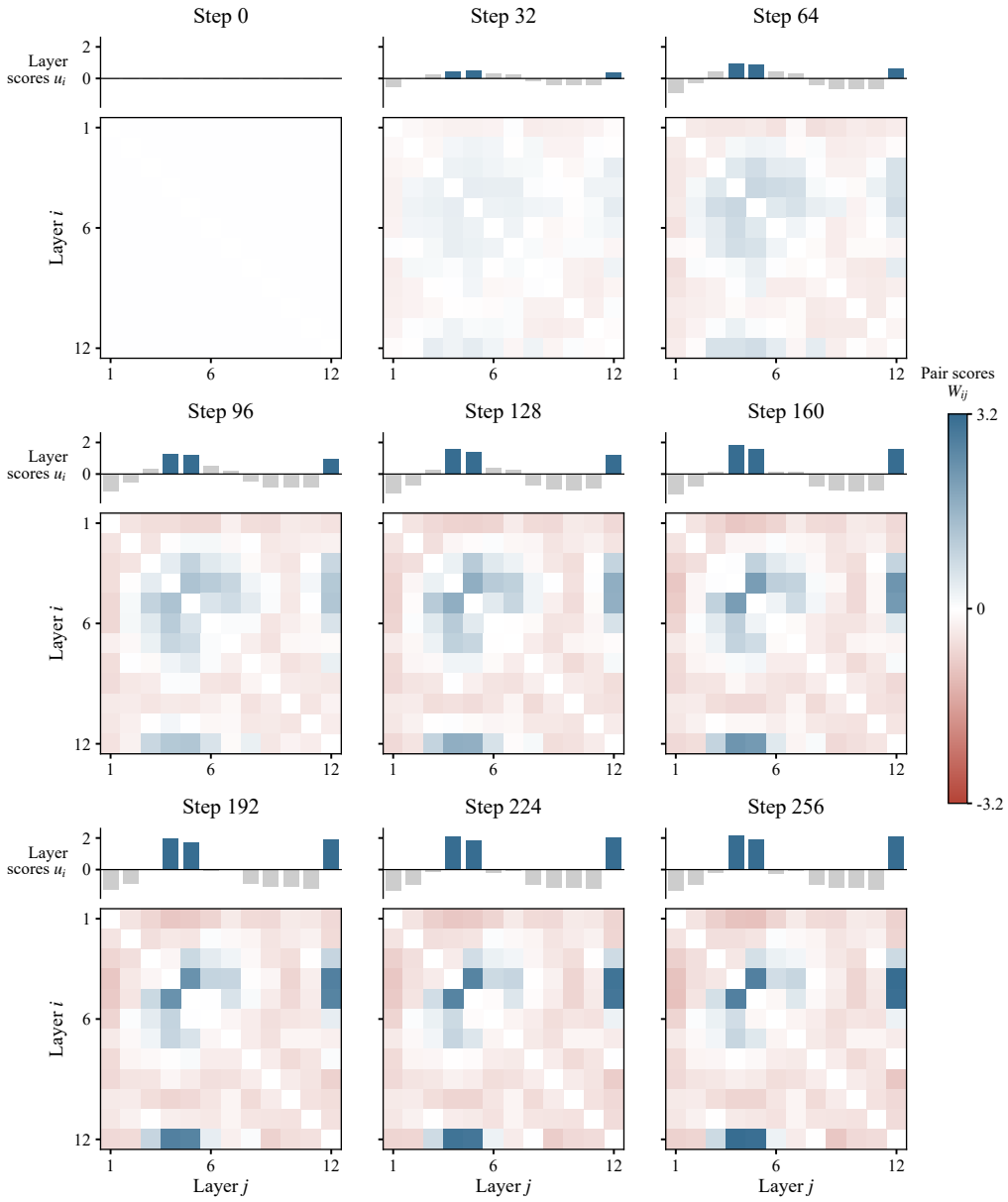



Figure A3: **Evolution of layer and pair scores for Base.** Bars show $\mathbf{u}$, with blue indicating the current highest-scoring complete placement. Heatmaps show $\mathbf{W}$ in the same layer order. Snapshots span updates 0–256 with fixed axis limits and color scale. Layer indices start at 1.

Large exhibits a longer adjustment process. At update 32, the selected group is concentrated in the middle layers, $\{6, 7, 8, 9, 10, 12\}$. By update 96, the complete placement includes the final layer, and it eventually settles on $\{6, 7, 8, 11, 12, 24\}$. The final pair scores are particularly strong for combinations such as $(8, 24)$, $(12, 24)$, and $(8, 12)$, indicating learned preferences for retaining these layers together. These entries describe the policy's preferences rather than direct attention connections between layers.

The intermediate snapshots also show how pair scores affect selection. At Large update 96, choosing the six largest $u_i$ values would give $\{6, 7, 8, 9, 11, 12\}$, whereas maximizing the complete score selects $\{6, 7, 8, 10, 12, 24\}$. The pair contribution therefore changes which layers are retained. Although the two selections agree at the final update, the layer scores at that point have been learned jointly with $\mathbf{W}$, rather than through a separate layer-only policy.

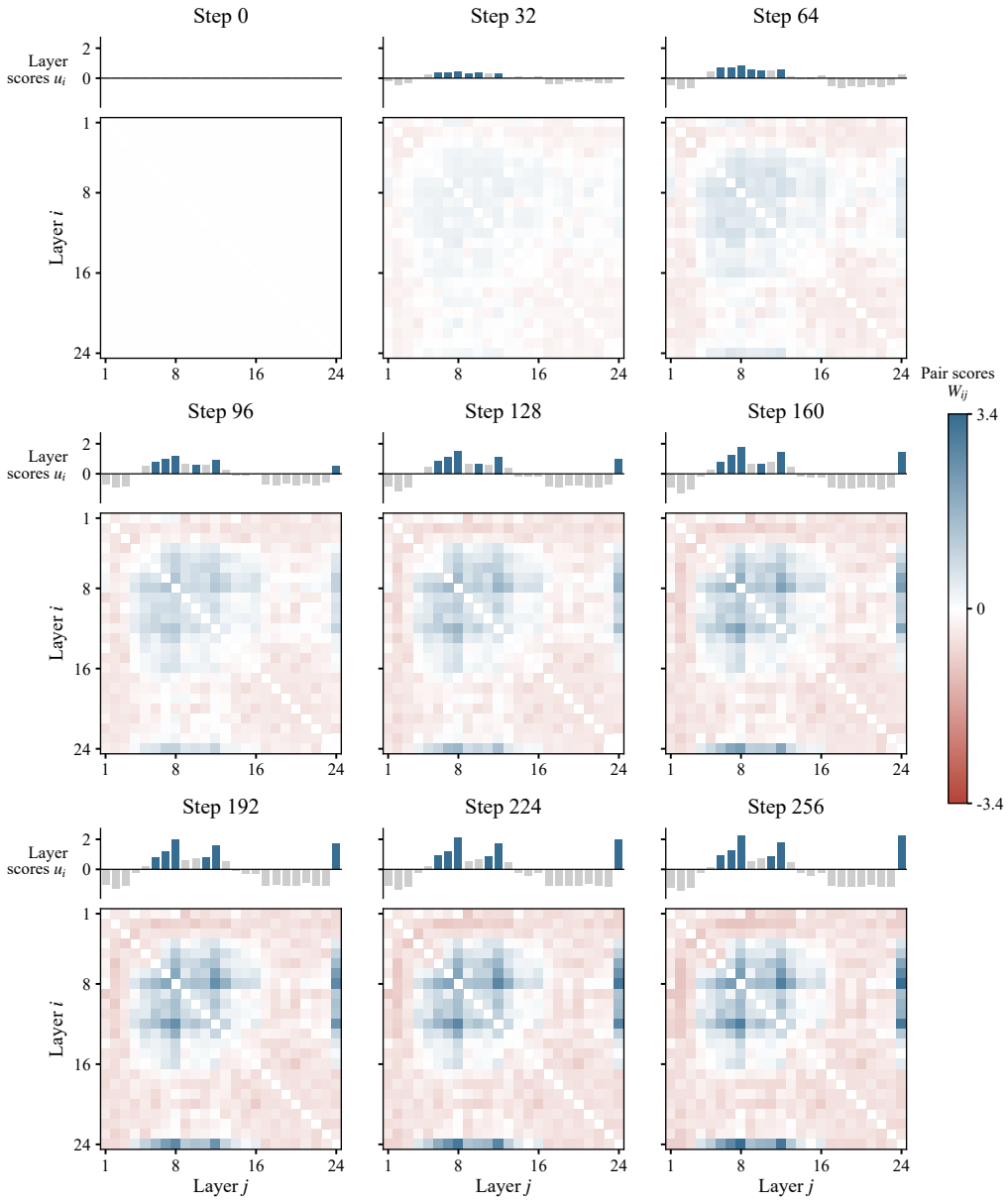



Figure A4: **Evolution of layer and pair scores for Large.** Bars show $\mathbf{u}$, with blue indicating the current highest-scoring complete placement. Heatmaps show $\mathbf{W}$ in the same layer order. Snapshots span updates 0–256 with fixed axis limits and color scale. Layer indices start at 1.

Figure A5 follows the resulting attention placement and its sampling probability. The Base placement remains unchanged from update 11 onward; Large stabilizes at update 187. Once the final placement becomes the highest-scoring candidate, the two probability curves coincide, and subsequent updates continue to adjust its sampling probability. Thus, the selected placements stabilize within the 256-update search, even though the full probability distribution continues to evolve. These figures illustrate one search trajectory for each scale; the final placements are reproduced across five random seeds.

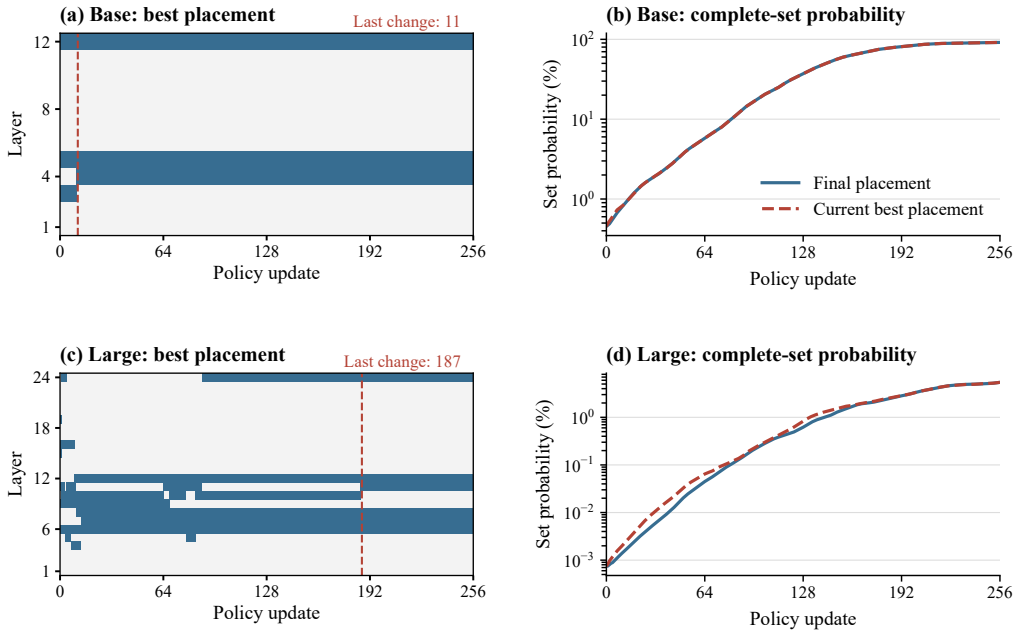



Figure A5: **Attention placement during optimization.** Left: layers in the highest-scoring complete placement at each update. Right: probability of the final placement and the highest probability among current placements. Dashed vertical lines mark the last placement change. Layer indices start at 1.

### D.3 Comparison of Policy Parameterizations

Table A5 summarizes the selected placements for Base and Large under different policy parameterizations. The shared-score policies yield consistent placements across five random seeds. Uniform placement is a fixed reference without learned parameters. Layer only uses shared individual scores, while Layer + pair additionally uses shared pairwise terms. Both shared-score policies use 256 updates. The independent per-placement policy uses 256 updates for Base and an extended 4,096-update search for Large, but remains unconverged within the evaluated search budget.

Table A5: **Selected attention placements across model scales.** The shared-score policies yield consistent placements across five random seeds. The independent per-placement policy remains unconverged within the evaluated search budget; we report the highest-scoring placement at the end of search. The per-placement policy uses an extended search for Large.

| Scale | Policy | Parameters | Updates | Selected placement |
|---|---|---|---|---|
| Base | Uniform | – | – | $\{4, 8, 12\}$ |
| | Per-placement | 220 | 256 | $\{3, 4, 6\}$ |
| | Layer only | 12 | 256 | $\{4, 5, 12\}$ |
| | **Layer + pair** | 78 | 256 | $\{4, 5, 12\}$ |
| Large | Uniform | – | – | $\{4, 8, 12, 16, 20, 24\}$ |
| | Per-placement | 134,596 | 4,096 | $\{1, 2, 4, 8, 12, 14\}$ |
| | Layer only | 24 | 256 | $\{7, 8, 9, 10, 12, 24\}$ |
| | **Layer + pair** | 300 | 256 | $\{6, 7, 8, 11, 12, 24\}$ |

For Base, Layer only and Layer + pair select the same placement and obtain an evaluation reward of 0.393224 on 1,024 images, compared with 0.350710 for Uniform and 0.340431 for Per-placement. In the recorded trajectories, Layer + pair reaches its final placement at update 11, whereas Layer only reaches it at update 140. For Large, the two shared policies select different placements. As reported in Table 3, adding pair scores improves patch cosine similarity from 93.8 to 94.3 at 224 and from 89.3 to 89.5 at 896, while maintaining 95.7 at 448. These results show faster placement stabilization for Base and improved representation recovery for Large.

**Independent per-placement scores.** An independent score for every placement can represent arbitrary preferences among complete layer combinations, but it does not share layer or pair information across configurations. For Large, this requires 134,596 parameters, compared with 300 for Layer + pair. Figure A6 shows the probability mass assigned to the top 1% of candidates under each policy. After 256 updates, Layer + pair concentrates 93.13% of its probability on these candidates, whereas Per-placement assigns only 1.18%. Even after 4,096 updates, the latter reaches only 2.24%, remaining close to its diffuse initial distribution.

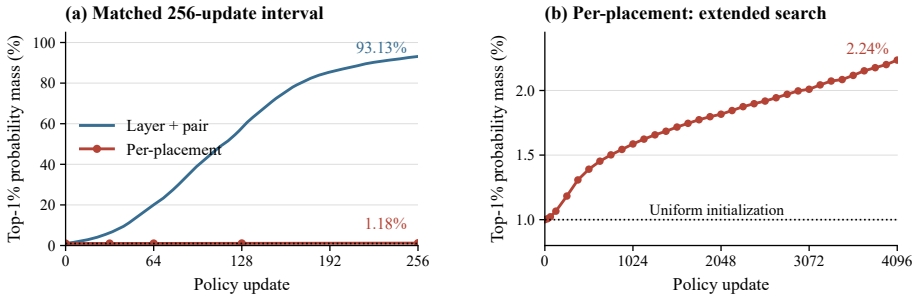



Figure A6: **Concentration of placement probabilities.** The plotted mass sums the probabilities of the 1,346 most probable candidates at each update, approximately the top 1% of the Large candidate space. Left: the first 256 updates. Right: the independent per-placement policy extended to 4,096 updates.

The placement trajectory further illustrates this slow convergence (Fig. A7). Per-placement changes its highest-scoring combination 86 times, with the last recorded change at update 4,015. In contrast, Layer + pair stabilizes within 256 updates. Together with the lower patch similarities in Table 3, the diffuse probabilities and late placement changes show that independent scores do not yield effective search in this setting, despite the longer optimization. Sharing layer and pair preferences allows observations from sampled encoders to inform other placements containing the same layers and pairs.

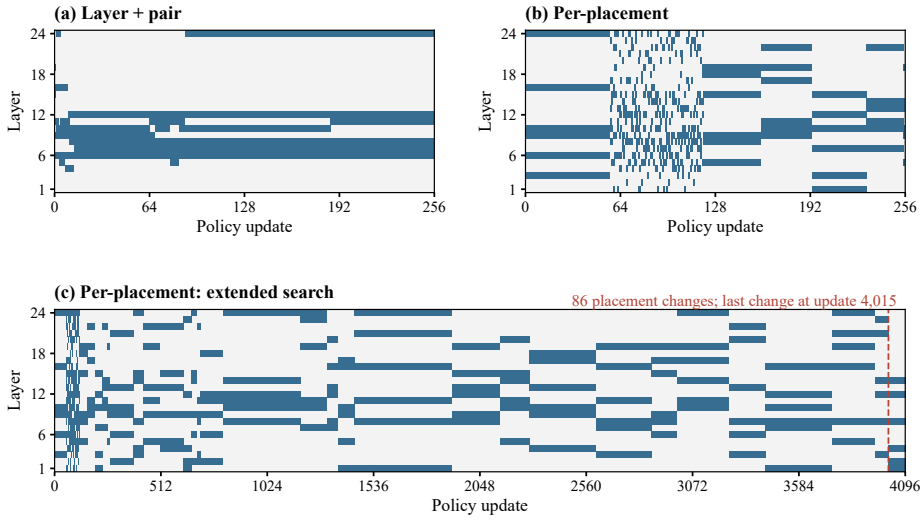



Figure A7: **Placement stability during Large search.** Blue cells indicate layers in the highest-scoring complete placement. Layer + pair stabilizes within 256 updates, while Per-placement continues to change late in the 4,096-update search. The last recorded change occurs at update 4,015.

# E Training and Evaluation Details

## E.1 LADDER Linearization Schedule

All conversion stages use ImageNet-1K images without text or class labels, and teacher parameters remain frozen. Table A6 lists the settings for mixer alignment, feature distillation, and final fine-tuning. Weight training uses eight NVIDIA A100 GPUs with BF16 mixed precision and AdamW with warmup followed by cosine learning rate decay. Optimizers and schedules are reset at each stage. Attention placement is optimized after mixer alignment using the frozen encoders, as described in Appendix D. The selected student combines aligned Vision-Delta blocks with the corresponding teacher attention blocks before feature distillation.

Table A6: **Image-only TIPSv2 conversion settings.** SigLIP2 uses token-equivalent $256^2/512^2$ inputs. Micro-batches are per GPU; the effective batch includes accumulation over eight GPUs.

| Setting | Mixer alignment | Feature distillation | Final fine-tuning |
|---|---|---|---|
| Input resolution | $448^2$ | $224^2$ | $448^2$ |
| Epochs | 1 | 20 | 1 |
| Effective batch | 1024 | 1024 | 256 |
| Base micro-batch $\times$ accum. | $32 \times 4$ | $128 \times 1$ | $32 \times 1$ |
| Large micro-batch $\times$ accum. | $16 \times 8$ | $64 \times 2$ | $16 \times 2$ |
| Peak learning rate | $10^{-3}$ | $2 \times 10^{-4}$ | $5 \times 10^{-5}$ |
| Minimum learning rate | $10^{-5}$ | $10^{-5}$ | $5 \times 10^{-6}$ |
| Warmup | 10 steps | 1 epoch | 0.1 epoch |
| Initial warmup learning rate | $10^{-5}$ | $10^{-5}$ | $5 \times 10^{-6}$ |
| AdamW $(\beta_1, \beta_2)$ | $(0.9, 0.95)$ | $(0.9, 0.999)$ | $(0.9, 0.999)$ |
| Weight decay | 0 | 0.05 | 0.05 |
| Crop scale | $[0.08, 1]$ | $[0.08, 1]$ | $[0.8, 1]$ |

All weight training stages use random resized crops, horizontal flipping with probability 0.5, and drop path 0.1. SigLIP2 follows the Large training configuration, changing only the input resolutions to the token-equivalent settings in Appendix A.

## E.2 Controlled Conversion Baselines

**Vision Mamba.** Our converted Vision Mamba uses patch size 14, 12 blocks, width 768, and FFN width 3072. Its BiMamba-v2 mixer has state dimension 16, convolution width 4, and expansion factor 2. Both recurrent baselines inherit the teacher's QKV weights and other compatible parameters using the same transfer procedure as LADDER. Web and Syn CLS tokens are placed in the middle of the patch sequence and participate in scanning. The original Vision Mamba uses 24 blocks without separate FFNs (Zhu et al., 2024); our implementation includes an FFN in each of 12 blocks to match the teacher structure. The asterisk in Table 2 identifies this controlled conversion baseline.

**Mamba-Reg.** The Mamba-Reg baseline also uses 12 blocks, width 768, and patch size 14. Following its register design (Wang et al., 2025), it introduces 12 pairs of Web and Syn registers and forms the global representation through a neck, ordered register concatenation, and a projector. Compatible teacher modules are transferred. Both recurrent baselines follow the same mixer alignment and feature distillation budget as LADDER.

**ViT architectural control.** The ViT control in Fig. 1a uses 12 attention blocks, width 768, 12 heads, patch size 14, and FFN width 3072. It inherits compatible teacher parameters except for attention QKV weights, which start from their native initialization and undergo mixer alignment. Excluding QKV transfer ensures that the student is not simply a copy of the teacher before conversion training.

## E.3 Architectural Ablations

Vision-Delta (RnnCLS) and Vision-Delta (AttnCLS) contain no full-attention blocks and differ in their CLS readout (Appendix C). Uniform Hybrid retains the same number of attention blocks as

LADDER, evenly placed at $\{4, 8, 12\}$ for Base and $\{4, 8, 12, 16, 20, 24\}$ for Large. Layer indices start at 1.

The Visual Delta Mixer ablations in Table 4 use one epoch of mixer alignment at $448^2$. Teacher and student then process images independently, and cosine similarity is measured between their final patch features. The bidirectional alternative follows Vision Mamba (Zhu et al., 2024) with gated delta updates: all heads scan in both directions, reverse outputs are restored to patch order, and the two outputs are combined before the shared projection. It replaces headwise reverse while retaining the other listed designs.

### E.4 Training Budget

We count student input patch tokens during mixer alignment and feature distillation. Each TIPSv2 image contributes 256 patches at $224^2$ and 1,024 at $448^2$. With $n = 1{,}281{,}167$ ImageNet-1K training images, the budget is $n(20 \times 256 + 2 \times 1024) \approx 9.1834 \times 10^9$ patch tokens, rounded to 9B in Fig. 1c. SigLIP2 has the same token count under its equivalent resolution schedule. The count excludes CLS tokens and frozen search passes and is not multiplied by encoder depth or teacher forward passes.

TIPSv2 (Cao et al., 2026) obtains its smaller ViT-B, ViT-L, and SO models by distilling a pretrained ViT-g teacher. Assuming that family distillation follows the reported training schedule, we estimate its budget from 90k steps at batch size 8192 and 9k steps at batch size 4096. Including one global crop and six local crops per image, these stages contribute $256+6\times49$ and $1024+6\times100$ patches per image, respectively. The resulting budget is $90{,}000 \times 8192 \times 550 + 9{,}000 \times 4096 \times 1624 \approx 465.37$B student patch tokens, rounded to 465B in Fig. 1c. This estimate excludes teacher pretraining and teacher forward passes.

CLIP-KD (Yang et al., 2024a) trains a Base/16 student at $224^2$ for 32 epochs on CC3M and CC12M; its implementation reports 12.86M downloaded image-text pairs, giving $12.86 \times 10^6 \times 32 \times 196 \approx 80.66$B tokens. ViT-Linearizer (Wei & Chellappa, 2025) distills a Base/16 student for 300 epochs on ImageNet-1K at $224^2$, giving $n \times 300 \times 196 \approx 75.33$B. Table A7 summarizes these conversion budgets, excluding teacher pretraining and subsequent supervised fine-tuning. We omit JetViT (Zou et al., 2026) from the budget comparison because its paper does not provide sufficient information to determine the total training budget.

Table A7: **Visual-token accounting for Fig. 1c.** Budgets count student input patches over conversion training.

| Method | Training images | Epochs × patches/image | Tokens (B) |
|---|---|---|---|
| CLIP-KD | CC3M + CC12M (12.86M) | $32 \times 196$ | 80.66 |
| ViT-Linearizer | ImageNet-1K | $300 \times 196$ | 75.33 |
| **LADDER** | ImageNet-1K | $20 \times 256 + 2 \times 1024$ | 9.18 |

### E.5 Evaluation Protocols

We follow the official TIPSv2 evaluation protocols (Cao et al., 2026) with frozen visual encoders. For vision-language evaluation, the converted LADDER visual encoder directly reuses its teacher's text encoder without additional text training. Linear probing trains only the task head and does not update the visual backbone. Public CLIP and OpenCLIP retrieval results are marked in Table 1; the remaining entries follow the local evaluation protocol.

**Vision-language evaluation.** For retrieval in Table 1, TIPSv2 models and their students use Syn CLS, while SigLIP2 models use native MAP global features and lowercased text inputs. Zero-shot segmentation follows TIPSv2 on ADE150, PC59, and VOC21. Both main tables use whole-image segmentation inference, preserving the long visual sequences of high-resolution inputs.

**Visual representation evaluation.** ImageNet linear classification and dense probing on VOC, ADE, and NYUv2 follow the official TIPSv2 feature extraction, probe training, and evaluation settings. ImageNet linear classification uses Web CLS for TIPSv2 students and the native global readout for SigLIP2 students. For cross-resolution kNN, we use a fixed ImageNet-1K training bank extracted at $448^2$ and vary only the validation query resolution. SigLIP2 and LADDER-Sig-L use

a $512^2$ bank matching their main-table inputs. Features are L2-normalized, with $k = 20$ neighbors and temperature 0.07 for weighted voting.

**Inference Efficiency** Visual encoder inference is measured on a single NVIDIA A100-SXM4-40GB GPU in BF16. LADDER and the FlashAttention teacher use the same batch size of 32 at every input resolution. For the standard-attention teacher, the batch size is reduced as needed at higher resolutions to fit GPU memory. FlashAttention uses CUDA kernels, while recurrent mixing uses Triton kernels. The deployed hybrid encoder uses a fixed attention placement without executing the selection policy. Figure 6 reports throughput and FLOPs ratios across resolutions, while Fig. 1b reports accuracy and computation at the dataset-specific input sizes. The maximum speedup at $3584^2$ and the Cityscapes result correspond to different input settings.

# F Further Discussions

## F.1 Standalone Vision-Delta Backbone

We additionally evaluate Vision-Delta-B as a standalone visual backbone through 300 epochs of supervised ImageNet-1K training. As summarized in Table A8, the model uses the same depth, feature dimensions, head configuration, and FFNs as LADDER-B, with all 12 layers implemented as Vision-Delta blocks. This standalone model uses patch size 16 at $224^2$ and retains no full-attention blocks. It therefore tests the recognition capability of the recurrent architecture beyond pretrained ViT conversion.

Table A8: **Standalone Vision-Delta-B configuration.** All encoder layers use Vision-Delta blocks.

| Setting | Vision-Delta-B |
|---|---|
| Input resolution | $224^2$ |
| Patch size | 16 |
| Vision-Delta / full-attention blocks | 12 / 0 |
| Feature dimension | 768 |
| Delta heads $\times$ dimension | $12 \times 64$ |
| FFN hidden dimension | 3072 |
| FFN activation | GELU |
| Parameters (M) | 93 |
| GFLOPs | 17.6 |
| Supervised training epochs | 300 |

We measure inference throughput on a single NVIDIA RTX 4090 following the protocol in Table 1 of ViT-Linearizer (Wei & Chellappa, 2025). The throughput and accuracy of DeiT-Base and the supervised Adventurer-Base baseline are taken from that table.

As shown in Table A9, Vision-Delta-Base achieves 83.3% top-1 accuracy, exceeding DeiT-Base and Adventurer-Base by 1.5 and 0.7 percentage points, respectively. It processes 1,046 images/s, achieving $1.42\times$ the throughput of Adventurer-Base. These results support Vision-Delta as an efficient standalone visual backbone, complementing its role in pretrained ViT linearization.

Table A9: **Supervised ImageNet-1K results at $224^2$.** Throughput is measured on a single RTX 4090. Baseline throughput and accuracy are taken from Table 1 of ViT-Linearizer (Wei & Chellappa, 2025).

| Model | Mixer | Params. (M) | THP ↑ (img/s) | GFLOPs | Top-1 (%) |
|---|---|---|---|---|---|
| DeiT-Base | Attention | 86 | 613 | 17.5 | 81.8 |
| Adventurer-Base | Recurrent | 99 | 736 | – | 82.6 |
| **Vision-Delta-Base** | Recurrent | 93 | **1046** | 17.6 | **83.3** |

## F.2 Data Usage and Future Directions

Using only ImageNet-1K images controls data variation in our architectural study. LADDER's retrieval performance still falls short of the teacher on benchmarks such as COCO, particularly at Base scale. Future work should extend image-only training to more diverse sources, including images from CC3M and CC12M, which we expect to improve absolute performance.