VISION TRANSFORMERS NEED CROSS AGGREGATION FOR PATCH SEMANTICS

Anonymous authors
Paper under double-blind review

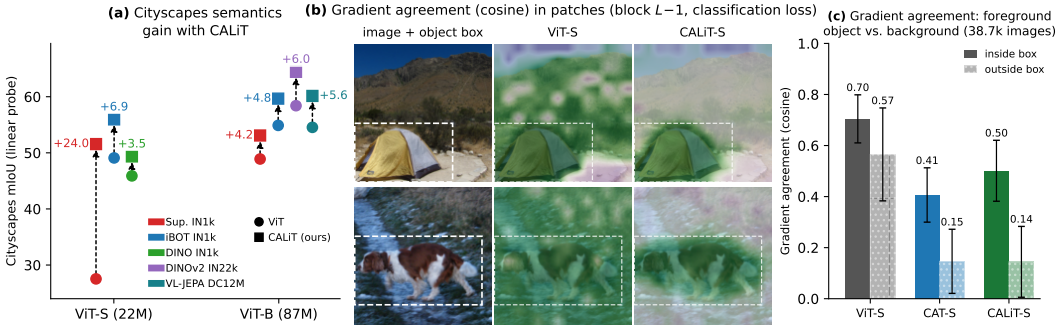

Figure 1: **CALiT improves patch gradients and thereby the patch tokens.** (a) Frozen-backbone linear probe in Cityscapes (mIoU) at S and B scale. (b) Cosine between mean gradient in the object bounding box and a given patch output at the block $L-1$ of L layer transformer, for supervised ViT-S and CALiT-S after 200 epochs (green: aligned, indigo: opposing). (c) Same gradient agreement of patches against object box, with mean cosine bar plot (and std) for inside and outside object bounding box in ViT-S, CAT-S, and full CALiT. Statistics was computed from *all* 38.7k ImageNet-val images. The plot reveals a low gradient contrast (as in (b)) in ViT-S training compared to CAT-S and CALiT-S.

ABSTRACT

A Vision Transformer (ViT) is tasked to learn both a useful global representation through the CLS token and distinct local semantics through patch tokens. Yet a loss on the CLS token does not always induce learning of *discriminative* patch tokens. We theoretically observe that a CLS-based loss induces low rank patch gradients and in practice, most patches receive similar gradients. Adding patch-based losses may not compensate for the strong unidirectional CLS induced gradients. Consequently, even under global-local objectives, patch updates can remain insufficiently differentiated and spatially correlated. We therefore introduce *cross-aggregation*, which redefines the CLS token by aggregating patch-level concept heads in the expanded FFN space of every block. This parameter-free operation adds only 0.6% FLOPs to ViT-S and more directly couples global semantics to patch tokens. Combined with local depth-wise convolutions, it further improves the patch semantics, in a variety of pre-training methods. Together, these components form the *Cross Aggregation and Local Mixing Transformer* (CALiT). CALiT consistently improves semantic probes under both global-only and global-local pre-training objectives and vision-language finetuning. In particular, CALiT improves thin class semantics mean IoU by double digits in many pre-training/linear probe pairs and provides moderate gains in all other local semantics.

1 INTRODUCTION

Vision Transformers (ViT) (Dosovitskiy et al., 2021; Touvron et al., 2021a; Wu et al., 2021; Dai et al., 2021; Tu et al., 2022) maintain two types of tokens: a single global CLS token with a learnable vector as input and N image patch tokens. A ViT uses the CLS token as a summary representation of the image and the patch tokens for the local image information. Both tokens are then processed

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.

symmetrically, in each of the two steps per block: self-attention and Feed Forward Network (FFN) with activations. Crucially, recent literature in ViT encoder pre-training (Oquab et al., 2023; Siméoni et al., 2025; Tschannen et al., 2025) put more emphasis on semantically meaningful patch tokens.

CLS-based losses provide the primary supervision in image classification (Deng et al., 2009; Ridnik et al., 2021) pre-training (Dosovitskiy et al., 2021; Touvron et al., 2021a; 2022; Hassani et al., 2021), language-aligned encoder pretraining (Zhai et al., 2023; Tschannen et al., 2025; Kuhn et al., 2026) as well as in some self-supervised learning (Caron et al., 2021; Kuhn et al., 2026; Maes et al., 2026). Regardless, except for rare cases (Assran et al., 2023; Liu et al., 2025), most vision encoders (Oquab et al., 2023; Zhou et al., 2022; Siméoni et al., 2025) use either CLS or CLS and patch-based losses for pre-training. A recent work (Marouani et al., 2026) argues for loosening the CLS-patch parameter coupling, in order to improve the patch tokens in either loss type. We instead consider the gradient perspective and attempt to create a better patch-discriminative signal from the CLS-loss regardless of whether any patch-based losses are present.

Our work starts from a simple observation. Differentiating the CLS-loss with respect to the patch inputs in the last layer reveals that the gradients are rank-limited to 2 times the number of heads for all patches. This limited corrective information is then propagated back through the whole model. Moreover, in practice, when the same gradient is measured in trained models, we find it to be largely unidirectional, analogous to the forward component (Nait Saada et al., 2025), as shown in Fig. 1(b). The impact is measurable and expected: supervised or CLS-loss approaches perform poorly on patch tokens and model pre-training often require lengthy schedules.

We therefore, propose to add an additional information flow from patches to the CLS in the post-activation expanded Feed Forward Network (FFN) space. Specifically, we divide the post-activation hidden features of each token in the Feed Forward Network (FFN) into slots and aggregate all patch heads in a slot into a CLS head. Intuitively, this parameter-free *cross-aggregation* (CAg) layer adds an inductive bias in the high dimensional space on the CLS token. In the Natural Language Processing (NLP) literature, the post-activation expanded space has been described as having a privileged basis (Dai et al., 2022; Suau et al., 2020; 2021; Gurnee et al., 2023; Elhage et al., 2022), in which individual neurons are more likely to align with semantics. A recent finding in vision and language models uses exactly these post-activation neurons to investigate mono and poly-semanticity (Dravid et al., 2026). This further justifies CAg as a semantically meaningful information aggregation.

This single change has three important consequences. First, the *forward pass* now provides CLS with a richer summary of the patch concepts, supplementing the self-attention's single bidirectional pass. Second, the *backward pass* now lets a CLS-only loss flow into patches through the concept aggregation directly impacting the patch tokens. Finally, the CAg directly resolves the smoothing tendency of local convolutions in the FFN (Li et al., 2021), effectively allowing further patch discrimination. We therefore directly test this as our final architecture CALiT: Cross Aggregation and Local Mixing Transformer. CALiT improves semantic segmentation probe performance consistently in supervised learning as well as self-supervised learning, with double digit gains on the segmentation of thin classes. We summarize our main contributions below:

- We frame the *weak image patch embeddings* of ViTs as an architectural issue and explain them as a consequence of low-rank unidirectional gradients coming from CLS-based losses.
- We propose **CALiT**(*Cross Aggregation and Local Mixing Transformer*), a lightweight architectural modification to plain ViTs (+1.4% FLOPs and +0.8% parameters for ViT-S). Our main contribution *Cross Aggregation* (CAg) is a parameter-free, multi-query aggregation in the FFN concept space. CAg further enables effective *Local Mixing* (LMix) (Li et al., 2021) to capture local patch relations, further improving the patch semantics.
- We empirically validate the shortcomings of ViT on patch embeddings in both training schemes of CLS and CLS-patch losses, with ablations and analysis. We provide evidence of better patch semantics from CALiT in a number of experimental setups.

2 Related Work

CLS variations in ViTs. The CLS token in a classical ViT (Touvron et al., 2021a; 2022; Dosovitskiy et al., 2021) is a single learned token concatenated to the patches; the same Q/K/V/FFN parameters

process both. The design was inspired from BERT (Devlin et al., 2019) which introduced the CLS token on the original transformer architecture (Vaswani et al., 2017), in a language decoding task. Since then, the most common implementation of ViT encoders (Dosovitskiy et al., 2021; Touvron et al., 2021a; Zhou et al., 2022; Oquab et al., 2023; Siméoni et al., 2025; Zhai et al., 2023; Tschannen et al., 2025) use the CLS token symmetric to the patch tokens. Several works have proposed alternative to such an approach. CaiT (Touvron et al., 2021b) freezes patch tokens in the last two blocks and runs class-attention layers where CLS is the query and CLS and patches the keys/values. Jumbo-CLS (Fuller et al., 2025) widens the CLS token, splits it into multiple sub-tokens while (Darcet et al., 2024) simply add extra register tokens unattached to any losses. CrossViT (Chen et al., 2021) uses CLS-as-query cross-attention to exchange information across two parallel patch streams at different scales (the cross-attention block omits the FFN). Most relevant to our work, recently (Marouani et al., 2026) proposed separate Q/K/V projections and layer-norm parameters for the CLS token in order to improve patch tokens.

CLS via global pooling. ViT (Dosovitskiy et al., 2021) explored an alternative construction of the CLS token at the output head via global average pooling (GAP) of patches and showed that it can match the performance of the standard CLS token (Dosovitskiy et al., 2021; Touvron et al., 2022) if provided with a separate learning rate. Swin Transformer (Liu et al., 2021) drops CLS entirely and uses patch GAP for the classifier head. MAE's linear-probing protocol (He et al., 2022) likewise uses GAP rather than the CLS token. These works perform GAP only at the output *head*: there is no CLS token in each block when GAP is present. Other forms of information aggregation in transformers are present in Perceiver IO (Jaegle et al., 2022), the Set Transformer's PMA (Lee et al., 2019), TokenLearner (Ryoo et al., 2021), and Slot Attention (Locatello et al., 2020). To our knowledge no published ViT applies a fully-parameter-free aggregation of patch features in order to obtain the CLS token in every block.

FFN analysis and interpretations. Most ViT modifications (Li et al., 2021; Wang et al., 2021; 2022; Liu et al., 2024) treat the FFN as a black-box component. However, several works in the NLP literature have analyzed the FFN internals, providing useful interpretations. Closer to our own interpretation, (Dai et al., 2022; Suau et al., 2020; 2021; Elhage et al., 2022) define the activation space in the FFN as a source of privileged basis or concept-specific neurons. In the same structure, (Gurnee et al., 2023) and more relevantly in vision models (Dravid et al., 2026) analyze the monosemantic vs. polysemantic nature of neurons. In other words, the space is likely to provide semantically meaningful neurons (Gurnee et al., 2023; Park et al., 2025). Our work uses these interpretations of the FFN internal space, thus justifying the weighted retrieval of patch heads as the CLS feature.

FFN-internal attentions. A variety of FFN modifications have been proposed, more so for large language models, in order to improve transformer efficiency or expressive power. Gated Attention Unit (GAU) (Hua et al., 2022) fuses attention with the FFN expansion of an LLM. (Wu et al., 2024; Zhang et al., 2025) use multi-head FFNs for LLMs. FlashMHF (Zhang et al., 2025) additionally uses an I/O-aware fused kernel for memory-efficient training. With similar motivation as GAU (Hua et al., 2022), MetaMixer (Yun et al., 2024) abstracts attention and the FFN into a unified Q-K-V "MetaMixer" framework and replaces self-attention with FFN-style large-kernel depthwise convolutions. UMoE (Yang et al., 2025) reformulates multi-head attention into a token-mixing and token-wise FFN form. Several approaches replace self-attention (Liu et al., 2022; Hou et al., 2024; Scribano et al., 2026) entirely with operations in the FFN. Our FFN change is instead a localized modification in standard ViT. Unlike (Hua et al., 2022; Yun et al., 2024; Yang et al., 2025), we do not instantiate Q, K, V projections in the FFN attention.

Local depth-wise convolutions in Transformers. Spatial token-mixing is a widely explored idea in ViTs (Yu et al., 2023; Yun et al., 2024; Li et al., 2021; Wu et al., 2021). LocalViT (Li et al., 2021) introduces a depthwise convolution (DWConv) between the FFN's $\mathbf{W}_1$ and $\mathbf{W}_2$ with same operation replacing self-attention in (Scribano et al., 2026). CMT (Guo et al., 2022) uses an Inverted-Residual FFN with a similar depth-wise convolutions but in a hierarchical, CLS-free architecture. MOAT (Yang et al., 2023) adds the DWConv block before self-attention which also performs downsampling. In these works, the DWConv weights are learned without Dirac or identity initialization (Zagoruyko & Komodakis, 2018; Zhang et al., 2019). More importantly, as our experiments show, DWConv behaves very differently for low and high resolution images in the absence of a pooling based CLS.

3 VIT AND ITS GRADIENTS

We briefly describe the ViT encoder architecture and its patch gradients for a global CLS-based loss. A ViT has two important learning sub-blocks: the self-attention (SA) mechanism with its linear projections and the feed-forward network (FFN) with non-linearity. For the standard blocks considered here, the SA sub-block learns relationships between the tokens using one-third of the parameters, whereas the FFN learns an overall non-linear function independently for each token using two-thirds of the architecture's parameters. Each block of SA + FFN, processes a token sequence $\mathbf{x} \in \mathbb{R}^{(1+N)\times D}$ (with CLS first):

$$\mathbf{y} = \mathbf{x} + \mathrm{SA}(\mathbf{x}) \tag{1}$$

$$\mathbf{z} = \mathbf{y} + \mathrm{FFN}(\mathbf{y}), \qquad \mathrm{FFN}(\mathbf{y}) = \mathbf{W}_2\, \sigma(\mathbf{W}_1 \mathbf{y}), \tag{2}$$

where $\mathbf{W}_1 \in \mathbb{R}^{4D\times D}$ and $\mathbf{W}_2 \in \mathbb{R}^{D\times 4D}$ are the linear functions of FFN and $\sigma = \mathrm{GELU}$ (Hendrycks & Gimpel, 2016) is the activation.

3.1 GLOBAL LOSS WITH CLS AND ITS GRADIENTS

Consider a training objective $\mathcal{L} = \ell(\mathbf{c}_L)$ defined on the final CLS token $\mathbf{c}_L$. Such CLS-based supervision is used as the sole global objective in image classification Touvron et al. (2021a; 2022), DINO Caron et al. (2021), vision-language pre-training LeVLJEPA Kuhn et al. (2026); Zhai et al. (2023), and some self-supervised pre-training Balestriero & LeCun (2025); Maes et al. (2026). Even when additional patch-level objectives are present, the gradient contribution from the CLS loss follows a constrained structure, which we characterize below. We differentiate the loss w.r.t. the patches entering the final block to define the gradients on patches. The same gradients are also used to obtain the plots in Fig. 1 (b), (c).

Observation. *For a CLS-only loss, each patch entering the final block can influence the CLS output of the final SA layer through only two paths per attention head: the Value (V), which determines what information it contributes, and the Key (K), which determines how strongly CLS attends to it. Consequently, within each head, the gradients of all patches lie in the span of at most two shared directions. Across n_h heads, this totals to a gradient subspace of dimension at most $2n_h$.*

We prove this result in App. A.2 step-by-step and confirm it empirically: for ViT-S with $n_h = 6$, the gradient energy beyond the first 12 singular directions is numerically zero. More importantly, the two gradient routes exhibit different structures. When we numerically record the two gradient directions, the Value (V) route shows a rather uniform direction, differing only by their non-negative attention weights. It therefore acts largely as a shared mode across patches: over 38.7k ImageNet-val images, it accounts for 71% of the gradient norm, with each patch gradient having an average cosine similarity of 0.90 to the image-wide mean resulting in a gradient distribution such as Fig. 1 (b), (c). The discriminative patch gradients at $L-1$ block is not a sufficient condition for learning better patches but it denotes the corrective source signal that gets propagated back to the upstream blocks, therefore holding significant importance.

In the next section, we introduce Cross-aggregation (CAg) that adds a third distinct and significant route for the patch gradients, together with Local Mixing (LMix) which makes up our proposed architecture for improved patch semantics. A final note on the gradient rank is that it does not reflect the actual rank of the patch gradients which can be lower. Therefore, in Sec. A, we use effective rank (Roy & Vetterli, 2007) to estimate the actual rank of patch gradients induced from CLS. The full description of the gradient analysis (on top of the above) is provided in Sec. A.

4 CROSS-AGGREGATION AND LOCAL MIXING TRANSFORMER (CALIT)

We describe the proposed CALiT architecture in two parts. First, we introduce *Cross-Aggregation* (CAg) that adds another patch→CLS forward pathway by aggregating patch heads into CLS in the expanded FFN space. Second, *Local Mixing* applies depth-wise convolution (Li et al., 2021) in parallel over the expanded FFN representations of patch tokens. Overall these two sub-modules add biased information exchange in the privileged space of each transformer block. The complete architecture is illustrated in Fig. 2.

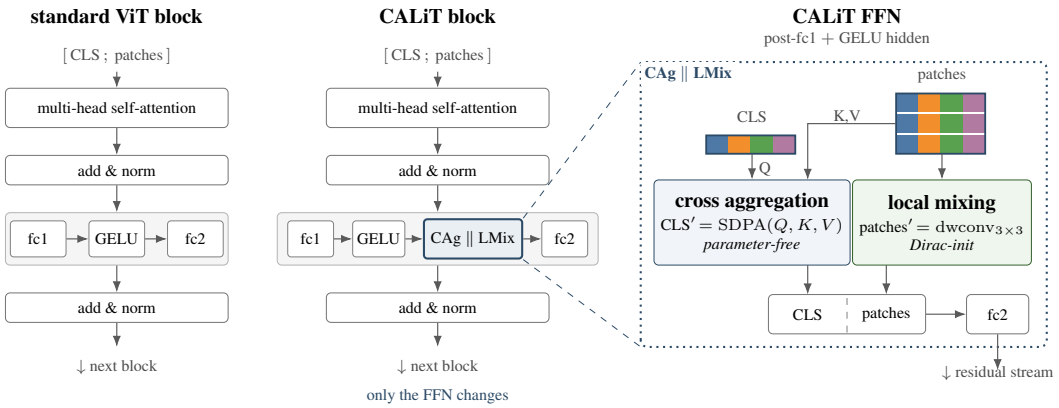

Figure 2: **CALiT as a small FFN change.** *Left:* A standard ViT block. The feed-forward network (`fc1`→GELU→`fc2`) is applied identically to the CLS and patch tokens. *Middle:* A CALiT block. CALiT block is identical except for the FFN. The post-GELU hidden in the FFN passes through the light-weight parallel sub-modules. The resulting $4D$-dim CLS and patch embeddings then pass through `fc2` without any other change. *Right:* zoom-in of the CALiT-specific sub-modules in FFN. The post-`fc1`/GELU hidden is read as separate heads (colors).

4.1 Cross Aggregation

Cross Aggregation (CAg), is the parameter-free CLS-side component which enforces a strong inductive bias on the CLS token. Within a single block, let $\mathbf{y} = \mathrm{SA}(\mathbf{x}) \in \mathbb{R}^{(1+N)\times D}$ be the input to the FFN and let $\mathbf{h} = \sigma(\mathbf{W}_1\mathbf{y}) \in \mathbb{R}^{(1+N)\times 4D}$ be the post-GELU hidden representation. Intuitively, we introduce a one way multi-headed attention across the two token types, *i.e.*, each CLS hidden against the patch heads in fixed pools, called slots. To that end, we split each hidden into S equal *slots* of width $4D/S$, which we treat as a set of concept vectors. Stacking them gives a CLS slot matrix $\mathbf{C} \in \mathbb{R}^{S\times 4D/S}$ (the S slots of $\mathbf{h}_c$) and a patch slot matrix $\mathbf{P} \in \mathbb{R}^{NS\times 4D/S}$ (the S slots of each of the N patches). Cross aggregation updates every CLS slot with a similarity-weighted average of all patch slots, using standard multi-head attention in which $\mathbf{C}$ supplies the queries and $\mathbf{P}$ both the keys and the values:

$$\mathbf{C}' = \mathrm{MHA}_H(\mathbf{C}, \mathbf{P}, \mathbf{P}), \qquad \mathrm{head}_a = \mathrm{softmax}\Big(\frac{\mathbf{C}_a\mathbf{P}_a^\top}{\sqrt{d_h}}\Big)\mathbf{P}_a, \tag{3}$$

where $\mathbf{C}_a, \mathbf{P}_a$ are the a-th of H column blocks of width $d_h = 4D/(SH)$. $\mathbf{C}'$ is reshaped back into the new CLS hidden $\mathbf{h}'_c$ and the patch hiddens pass through unchanged to LMix. H equals the number of self-attention heads. We choose $S = 4$ for ViT-S ($d_h = 64$) and $S = 8$ for ViT-B ($d_h = 32$) for stable training.

Key Observations. i) Parameter-free inductive bias: Eq. (3) defines parameter-free similarity-weighted pooling of patch features in the expanded FFN space. Here, slots are used to limit the pool size of the keys and values for the stability of CAg. (ii) Same basis: Both $\mathbf{h}_c$ and $\mathbf{h_p}$ live in the same projected basis defined by $\mathbf{W}_1$. (iii) Direct gradient path: the operation introduces a direct edge from $\mathbf{c}'$ to every patch through the softmax in Eq. (3), so a loss on $\mathbf{c}_L$ flows backward to every patch in addition to the SA path. (iv) The added compute is H softmaxes of size $S \times NS$ per block, which at $N = 196$ (ViT-S/16 at 224^2) is negligible compared to SA's $(1+N)^2$ softmax and the linear layers in SA and FFN.

4.2 Patch-local mixing (LMix) in FFN

The local mixing (LMix) uses a 3×3 depthwise convolution (DWConv) on the patch slices $\mathbf{h}_p$, in the corresponding post-activation 2D feature map $\mathbb{R}^{4D\times\sqrt{N}\times\sqrt{N}}$. Concretely,

$$\mathbf{h}'_p = \mathrm{Reshape}\big(\mathrm{DWConv}_{3\times3}(\mathrm{Reshape}^{-1}(\mathbf{h}_p))\big). \tag{4}$$

The dwconv operates per-channel: each of the $4D$ concept channels has its own 3×3 kernel. Importantly, we apply the convolution exclusively to the patch tokens. Per block this adds $9 \times 4D$

parameters. For ViT-S ($D = 384$) this is 13.8k parameters per block, or roughly 0.17 M total, a $\sim 0.8\%$ overhead over backbone parameters. The DWConv operations are identical to those in LocalViT (Li et al., 2021) except that we do not introduce new nonlinearities, so that the patch and CLS features share the same linear basis. The initialization of DWConv is discussed in Sec. B.1.

5 EXPERIMENTS

5.1 ARCHITECTURES AND SETUP

We evaluate ViT-S/16 ($D = 384$, 12 blocks, 6 self-attention heads) and ViT-B/16 ($D = 768$, 12 blocks, 12 self-attention heads), with patch size 16 and CAg head size 64 and 32 respectively. Lower head size (32) and layer scale parameters are used in B scale for stability of training. CALiT and CAT are always trained with 300 epochs schedule. This is mainly motivated by a more informed CLS-gradient with CAg (Sec. 3, Sec. A), which helps shorten the schedule. A shorter schedule in itself is a tremendous advantage for pre-training. In supervised ImageNet-1k training, we use AdamW, batch size 2048, effective learning rate 2×10^{-3}, cosine decay, 5-epoch warmup, RandAugment, mixup, cutmix, label smoothing 0.1, and drop-path 0.1. Public checkpoints used for comparison use higher or same epoch schedules. For CALiT-iBOT -S/B, we train for 300 epochs with a global batch size of 1024. CALiT adds $+0.75\%$ parameters / $+1.44\%$ FLOPs at S and $+0.40\%$ / $+1.03\%$ at B. We implemented all our pre-training on either 1 or 2 NVIDIA H200, with gradient accumulation. Linear probes were conducted in NVIDIA A100-40g. We exclusively used 4 datasets: ADE20k (Zhou et al., 2017), Pascal VOC (Everingham et al., 2010), Cityscapes (Cordts et al., 2016) and COCO-Stuff (Caesar et al., 2018) for the main results with linear probes.

Dense linear probe protocols. We use linear probe provided semantic quality as the principle metric for pretraining advantage following DINOv2 (Oquab et al., 2023). The metric is computed using mean intersection over union (mIoU) using the ground-truth labels following standard protocols (Everingham et al., 2010). We use frozen backbones and a DINOv2-style BNHead (BatchNorm followed by 1×1 convolution), aspect-preserving resizing and random crops (512^2 for ADE20K/VOC/COCO-Stuff, 1024^2 for Cityscapes), AdamW (lr 10^{-3}, weight decay 10^{-4}), polynomial decay with 1500 warmup iterations, and batch size 16. Target schedules are 40k iterations, except for COCO which uses 200k iterations, with sliding-window inference. This protocol is identical to DINOv2 and is followed by other architectural/pre-training works (Marouani et al., 2026). Fig. 1 left, summarizes the semantic quality gains in various pre-training and vision language finetuning experiment.

5.2 SUPERVISED IN-1K: CLASSIFICATION AND SEMANTIC QUALITY

We use the DeiT-I (Touvron et al., 2021a) protocol for ViT-S experiments and use their publicly available checkpoint to conduct our own linear probe experiments. We do not conduct our own probes for ViT-B and instead use the DeiT-III (Touvron et al., 2022) baseline numbers reported in (Marouani et al., 2026). As predicted by the rank deficiency in gradients, supervised learning benefits the most from the architecture of CALiT in Tab. 1, with the highest gain of $+24$ mIoU between CALiT-S and ViT-S in Cityscapes (Cordts et al., 2016). Moreover, the CAT-S numbers reveal that the gains come largely from CAg, the main contribution of this work.

Additionally, we use LocalViT-S (Li et al., 2021), trained in Deit-I setting, to ablate CAg in CALiT. Note that LocalViT-S uses the same approach for local convolusions in FFN, except for an additional Swish (Ramachandran et al., 2017) activation after the convolutions. As reported in Sec. 1, the depthwise 3×3 convolutions of LocalViT reduces the overall semantic quality of patch tokens in 3 out 4 datasets. The gains only remain in Cityscapes where convolutions add an advantage from resolution-robustness. The 4-way comparison in ViT-S supervised learning empirically establishes the importance of CAg on patch semantics.

5.3 SELF-SUPERVISED LEARNING: DINO, IBOT AND DINOV2

The self-supervised learning pre-training includes three different methods including DINO (Caron et al., 2021), iBOT (Zhou et al., 2022) and DINOv2 (Oquab et al., 2023). Since DINOv2 pre-training is not possible for compute and data constraints, we borrow the tested reproducible configuration

Model	Pretrain (ep.)	IN-1k top-1	ADE20K	VOC	Cityscapes	COCO-Stuff
ViT-S/16 — *supervised IN-1k*						
DeiT-S/16†	sup. 300	79.85	24.48	60.51	27.53	29.22
LocalViT-S†	sup. 300	**80.83**	20.23	49.85	40.01	22.41
CAT-S	sup. 300	79.96	29.62	67.68	46.10	34.79
CALiT-S (ours)	sup. 300	80.57	**31.65**	**71.50**	**51.56**	**36.68**
architectural Δ		*+0.72*	*+7.17*	*+10.99*	*+24.03*	*+7.46*
ViT-S/16 — *self-supervised IN-1k, iBOT*						
iBOT ViT-S/16	iBOT 800	**77.9** (lp)	29.78	66.67	49.09	34.71
CALiT-S (ours)	iBOT **300**	76.3 (lp)	**32.81**	**73.78**	**55.96**	**37.97**
architectural Δ		*-1.6*	*+3.03*	*+7.11*	*+6.87*	*+3.26*
ViT-S/16 — *self-supervised IN-1k, DINO*						
DINO ViT-S/16	DINO 800	**77.0** (lp)	23.79	50.41	45.89	26.08
CALiT-S (ours)	DINO **300**	75.9 (lp)	**27.45**	**61.47**	**49.36**	**31.48**
architectural Δ		*-1.1*	*+3.66*	*+11.06*	*+3.47*	*+5.40*
ViT-B/16 — *supervised IN-1k*						
DeiT-III-B‡	sup. 400	81.8	25.4	61.7	48.9	N/A
CALiT-B (ours)	sup. **300**	**82.20**	**32.23**	**71.88**	**53.09**	**37.31**
architectural Δ		*+0.40*	*+6.83*	*+10.18*	*+4.19*	N/A
ViT-B/16 — *self-supervised IN-1k, iBOT*						
iBOT ViT-B/16	iBOT 400	**79.5** (lp)	36.29	72.99	54.89	38.90
CALiT-B (ours)	iBOT **300**	79.3 (lp)	**37.25**	**75.82**	**59.66**	**41.08**
architectural Δ		*-0.2*	*+0.96*	*+2.83*	*+4.77*	*+2.18*
ViT-B/16 — *self-supervised IN-22K, DINOv2*						
DINOv2 ViT-B/16§	DINOv2 600k	**80.4** (lp)	38.3	76.6	58.4	N/A
CALiT-B (ours)	DINOv2 600k	77.6 (lp)	**38.73**	**77.37**	**64.44**	**41.51**
architectural Δ		*-2.8*	*+0.43*	*+0.77*	*+6.04*	N/A
ViT-B/16 — *iBOT IN-1k, then vision-language fine-tuning (LeVLJEPA, DataComp-12M)*						
iBOT ViT-B/16	iBOT 400 + VL 5k	**44.78** (zs)	36.93	75.52	54.54	39.63
CALiT-B (ours)	iBOT 300 + VL 5k	44.65 (zs)	**38.57**	**77.93**	**60.14**	**41.01**
architectural Δ		*-0.13*	*+1.64*	*+2.42*	*+5.60*	*+1.38*

Table 1: **Main results (mean IoU) under the DINOv2-style linear probe.** We conduct linear probes in the DINOv2-style for all approaches and report the mIoU. For the baselines DeiT-III-B (Touvron et al., 2022) and DINOv2 ViT-B/16 mIoU we use the numbers reported by Marouani et al. (Marouani et al., 2026), which uses the same probing configuration as DINOv2. For consistency, we choose 300 epochs for CALiT as the standard pretraining protocol in ImageNet1k, same or lower than the baseline. Top-1 classification in Imagenet1k for self-supervised models are measured using linear probe (denoted by lp) according to baseline's approach and for LeVLJEPA we use zero shot (denoted by zs). Table is divided into 8 different sections, each reporting respective mIoU and Δ as CALiT - ViT for each method. Bold marks the highest value per architecture within each block. The supervised ViT-S Cityscapes lift ($+24.03$) is the largest single number in the table. CALiT loses to ViT in 5 out of 7 ImageNet classification top-1 probes. The 4-method ablation in the first section shows that the highest patch semantic gains in CALiT comes from CAg.

from (Marouani et al., 2026). Similar to the DeiT-III linear probe, we directly borrow the DINOv2 semantic mIoU from (Marouani et al., 2026). Note that the publicly available DINOv2 model was trained on a far larger scale (data and model) and is not comparable here. Unlike the supervised case, DINOv2 pre-training consists of patch-based losses. Nonetheless, the same conclusions of supervised learning hold in DINOv2 CALiT-B. The results show a large gain in Cityscapes, where the obtained CALiT-B mIoU are comparable to the DINOv2-L baseline reported in (Marouani et al., 2026).

DINO consists of no patch-based losses. Therefore, we only test the approach in CALiT-S vs. ViT-S. CALiT-S here uses a short 300 epochs schedule, different from the public 800 epochs schedule, as object gradients are more readily available in CALiT making long schedules unnecessary. Interestingly, for the DINO pre-training, the biggest improvement is seen in the standard Pascal VOC dataset with $+7$ boost in mIoU. The iBOT semantics comparison is very similar to DINOv2, as both rely mainly on the patch MIM loss for pre-training. Both CALiT-S and CALiT-B show significant improvement in semantics with iBOT pre-training, with their largest gains again in Cityscapes.

5.4 Vision Language Encoder Fine-tuning: LeVLJEPA

Unlike the previous experiments, we conduct a short fine-tuning experiment to show CALiT's application in vision-language encoder training. The primary supervision in a vision language

Model	Pretrain (ep.)	Cityscapes (val) all	Cityscapes (val) thin	ACDC zero-shot all	ACDC zero-shot thin	ACDC trained all	ACDC trained thin
ViT-S/16 — *supervised IN-1k*							
DeiT-S/16†	sup. 300	27.41	12.21	15.41	6.48	21.62	10.33
CAT-S (Cross Aggregation only)	sup. 300	46.10	29.90	25.44	12.22	37.42	20.01
CALiT-S (ours)	sup. 300	**51.05**	**40.06**	**28.34**	**18.77**	**40.73**	**26.42**
architectural Δ		*+23.64*	*+27.85*	*+12.93*	*+12.29*	*+19.10*	*+16.09*
ViT-S/16 — *self-supervised IN-1k, iBOT*							
iBOT ViT-S/16	iBOT 800	48.73	31.01	32.84	17.54	40.25	21.41
CALiT-S (ours)	iBOT **300**	**55.80**	**43.81**	**37.97**	**25.25**	**48.34**	**31.24**
architectural Δ		*+7.07*	*+12.80*	*+5.13*	*+7.71*	*+8.09*	*+9.83*
ViT-B/16 — *self-supervised IN-1k, iBOT*							
iBOT ViT-B/16	iBOT 400	54.75	37.15	36.28	22.30	47.00	26.77
CALiT-B (ours)¶	iBOT **300**	**58.46**	**47.66**	**38.45**	**26.15**	**53.03**	**37.91**
architectural Δ		*+3.71*	*+10.51*	*+2.17*	*+3.85*	*+6.04*	*+11.13*

Table 2: **Thin-structure segmentation and robustness (mean IoU).** We test three different pre-training families for 7 different small classes (fence, pole, traffic light, traffic sign, rider, motorcycle, bicycle) of Cityscapes, considered as thin classes. We report the val mIoU under the Table 1 linear probe. For ACDC zero-shot, we use the Cityscapes trained linear head to obtain the mIoU in the validation set. The ACDC trained uses 1600 training split to train a linear head from scratch, using a shorter 4k iterations. We use the validation set of 400 for the mIoU, same as in Cityscapes. We obtain very high mIoU gains with CALiT in both trained versions.

encoder training comes from the CLS token, making CAg even more important here. We choose LeVLJEPA (Kuhn et al., 2026) for its simplicity as well as its choice of ViT-B, a relatively small scale baseline. However, finetuning with CALiT poses a significant challenge, as it requires an initialization with a pre-trained CALiT model. This is not possible as the majority of DataComp-L (Gadre et al., 2023) links are already invalid. We therefore choose to initialize CALiT-B and ViT-B with their respective iBOT IN1k model, for lack of better pairs and use DataComp12M (Vasu et al., 2024) with just 8.7 million images as finetuning data. We use a training budget of 5k iterations (warmup 1k), batch size 2048, *i.e.*, 1.2 epochs, with a learning rate of 1e-4. On the text side, we use the same pre-trained frozen text encoder and projector obtained from the publicly available checkpoint (Kuhn et al., 2026). Tab. 1 bottom section shows the results, which are surprising in two ways. First, it shows leVLJEPA loss despite using only CLS improves the iBOT model convincingly for CALiT in all datasets. The highest improvement over the initialization is in Pascal VOC, for both models, where single clean object classes exist. The highest improvement over the baseline finetuned encoder is again in Cityscapes where thin classes heavily influence the overall metric.

5.5 Thin Classes Semantics: Cityscapes and ACDC

CALiT significantly improves thin class semantics. For consistency between Cityscapes and the related ACDC dataset (Sakaridis et al., 2025) we define thin classes as a set of fixed 7 classes, *e.g.*, bicycle, pole, etc. The ACDC dataset consists of urban driving images with harder adversarial conditions such as fog, rain and snow. Tab. 2 lists the results in thin classes highlighting the high gains on iBOT-S/B as well as supervised ViT-S comparisons. Results show very high gains in Cityscapes thin classes, moderate to high in ACDC zero shot using Cityscapes trained linear head, and again very high gains in ACDC trained linear probes. We visualize the impact of CALiT in Fig. 3 using a Cityscapes image pair. Both the linear head segmentation and PCA backbone feature visualization on different ImageNet1k trained models show distinct improvement with CALiT for small structures.

While CALiT and the ablated models show large improvements in thin classes, classification often shows a small loss. This is expected as we add an additional inductive bias to the CLS token, to summarize itself from the patches' privileged space in the FFN. The ablation in Tab. 3 shows that this constraint can be relaxed, *e.g.*, by adding on only half the dimensions. The relaxation is enough to induce similar improvement in the patch semantics.

5.6 Ablations

In Tab. 1, we showed the effect of our main contribution CAg and CAg+LMix. However, CAg itself can be implemented in many different ways. We tested 4 viable approaches and report the semantic

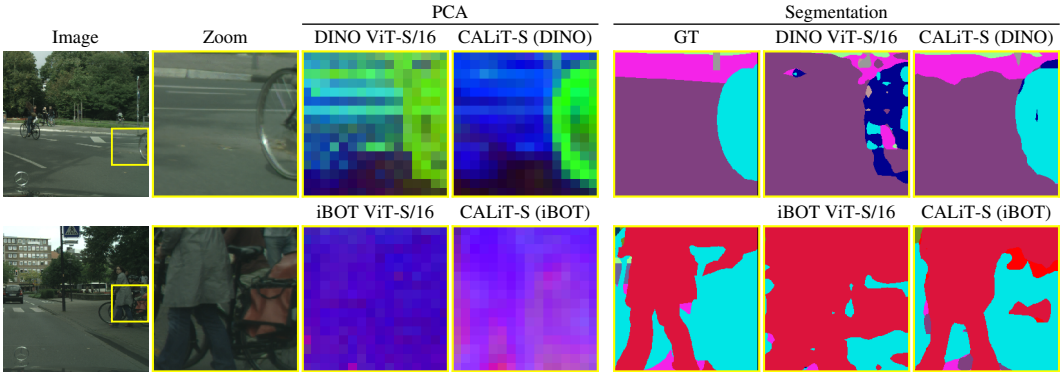

Figure 3: **CALiT's effect on thin structures for DINO and iBOT in S/16.** Each row shows a Cityscapes val image, a zoomed region (yellow box), the region's patch-feature PCA (top three components of the frozen last-layer features, as RGB) and its segmentation (ground truth and each model's linear-probe prediction). Baselines fail on the bicycle and person, CALiT recovers it.

Model	IN-1k top-1	ADE20K	VOC	Cityscapes
CALiT-S (reference)	80.57	**31.65**	**71.50**	**51.56**
single-variable changes to CALiT-S (same recipe):				
CAg in the last block only	79.54	25.99	60.78	43.86
half-width CAg	*80.64*	*29.83*	*67.29*	*50.76*
half-width CAg, no LMix	80.16	29.17	67.24	47.72
mean-pool CAg (no attention)	**80.73**	29.54	66.47	48.19

Table 3: **Single-variable ablations of CALiT-S** (supervised IN-1k, linear probes, protocol of Tab. 1). Each row changes one component of CALiT-S under the same recipe. Placing CAg only in the last block loses most of the dense gain, so per-block placement matters. Applying CAg to half of the FFN width is enough to recover most of the gains. Implementing CAg as a uniform mean loses some patch quality, often worse than half CAg. Bold denotes the best and *italic* denotes the second best.

linear probes in Tab. 3. First, we add CAg in the last block only. As expected, this performs marginally better than DeiT-S except in Cityscapes, thanks to LMix. Although the last block placement can improve the gradients in that block, patches in the subsequent blocks still receive lower quality corrective signals without CAg in every block. Using only half the heads for CAg in the CLS token, with normalization in the remaining heads performs well despite the asymmetric operations in the token. Finally both removing LMix in the half-width CAg and using mean pool attention removes similar patch quality.

6 Conclusion

We analyzed the impact of a CLS-based loss on a ViT encoder on its last block gradients using mathematical and experimental tools. We showed that plain ViTs trained under CLS-only supervision under-refine patch features, at least in part from the poorly informative patch gradients. We therefore, proposed an additional information exchange from patches to the CLS token in the privileged expanded space of the FFN, called *Cross Aggregation* (CAg). This added additional corrective signals back to the patches. Furthermore, combining the local mixing with depth-wise convolutions in the analogous patch hidden, we proposed CALiT (*Cross Aggregation and Local Mixing*). We compared linear probes on several pre-training approaches as well as on vision-language finetuning. These experiments showed a consistent gain in semantics, and high gains specifically for thin classes, empirically demonstrating better patch discrimination.

Limitations and Future Work. For computational reasons, we tested our approach only in S and B scales, with no hyperparameter tuning. Although, data and model size scaling is relatively consistent in ViT, additional tuning such as partial CAg may be needed in larger scales. Nonetheless, we showed with a theoretical interpretation that biased information exchange can be beneficial in the expanded space where tokens undergo the most significant transformation.

AI USE STATEMENT

We used AI (Claude, Gemini, GPT) at various stages of the work. The primary use was in coding and over 95% of the python codes were written using AI, and verified by authors. This included several plotting codes. Ideation was done by the authors. In the initial investigation, AI was used as a search tool to find relevant work and cross-check similarities to ensure novelty. This is relevant to the citations in the literature review, although all of the bib texts were obtained from google scholar or relevant websites. Additionally we used GPT to verify and ensure correctness of Eq. (9), despite its simplicity and obviousness. Other minor uses include AI review and grammar corrections.

REPRODUCIBILITY STATEMENT

We provide all details on the architectural parameters in the main text. Other details regarding pre-training if missing should use the standard configuration for that training, *e.g.*, DINO, iBOT, etc. Efforts were made to reuse existing literature's numbers from trusted sources and to avoid pre-training models when unnecessary. Additional information, including schedules and batch sizes are provided in the main text. We conducted linear probes twice in all cases and used their mean to ensure reproducibility. Furthermore, in order to reduce noise, we used paired data training for each comparison. While we were unable to scale up the models for compute reasons, we verified our results by creating a small synthetic dataset and a small 4-layered vision transformer. All codes, data and checkpoints will be shared.

REFERENCES

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *CVPR*, 2023.

Thomas Bachlechner, Bodhisattwa Prasad Majumder, Henry Mao, Garrison Cottrell, and Julian McAuley. ReZero is all you need: Fast convergence at large depth. In *UAI*, 2021.

Randall Balestriero and Yann LeCun. LeJEPA: Provable and scalable self-supervised learning without the heuristics. *arXiv preprint arXiv:2511.08544*, 2025.

Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1209–1218, 2018.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.

Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. CrossViT: Cross-attention multi-scale vision transformer for image classification. In *ICCV*, 2021.

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 *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8493–8502, 2022.

Zihang Dai, Hanxiao Liu, Quoc V. Le, and Mingxing Tan. CoAtNet: Marrying convolution and attention for all data sizes. In *NeurIPS*, 2021.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *ICLR*, 2024.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019.

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. In *ICLR*, 2021.

Amil Dravid, Yasaman Bahri, Alexei A Efros, and Yossi Gandelsman. Neuron populations exhibit divergent selectivity with scale. *arXiv preprint arXiv:2606.03990*, 2026.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *arXiv preprint arXiv:2209.10652*, 2022.

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

Anthony Fuller, Yousef Yassin, Daniel G. Kyrollos, Evan Shelhamer, and James R. Green. Simpler fast vision transformers with a jumbo CLS token. In *ICML*, 2025.

Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. *Advances in Neural Information Processing Systems*, 36: 27092–27112, 2023.

Jianyuan Guo, Kai Han, Han Wu, Chang Xu, Yehui Tang, Chunjing Xu, and Yunhe Wang. CMT: Convolutional neural networks meet vision transformers. In *CVPR*, 2022.

Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. Finding neurons in a haystack: Case studies with sparse probing. *arXiv preprint arXiv:2305.01610*, 2023.

Ali Hassani, Steven Walton, Nikhil Shah, Abulikemu Abuduweili, Jiachen Li, and Humphrey Shi. Escaping the big data paradigm with compact transformers. 2021. URL `https://arxiv.org/abs/2104.05704`.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Qibin Hou, Cheng-Ze Lu, Ming-Ming Cheng, and Jiashi Feng. Conv2Former: A simple transformer-style ConvNet for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Weizhe Hua, Zihang Dai, Hanxiao Liu, and Quoc V. Le. Transformer quality in linear time. In *ICML*, 2022.

Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, et al. Perceiver IO: A general architecture for structured inputs & outputs. In *ICLR*, 2022.

Lukas Kuhn, Giuseppe Serra, Randall Balestriero, and Florian Buettner. Levljepa: End-to-end vision-language pretraining without negatives. *arXiv preprint arXiv:2607.00784*, 2026.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *ICML*, 2019.

Yawei Li, Kai Zhang, Jiezhang Cao, Radu Timofte, and Luc Van Gool. LocalViT: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*, 2021.

Yanqing Liu, Xianhang Li, Letian Zhang, Zirui Wang, Zeyu Zheng, Yuyin Zhou, and Cihang Xie. Openvision 2: A family of generative pretrained visual encoders for multimodal learning. *arXiv preprint arXiv:2509.01644*, 2025.

Yun Liu, Yu-Huan Wu, Guolei Sun, Le Zhang, Ajad Chhatkuli, and Luc Van Gool. Vision transformers with hierarchical attention. *Machine Intelligence Research*, 2024. arXiv:2106.03180.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. In *CVPR*, 2022.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *NeurIPS*, 2020.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. Leworld-model: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026.

Alexis Marouani, Oriane Siméoni, Hervé Jégou, Piotr Bojanowski, and Huy V. Vo. Revisiting [cls] and patch token interaction in vision transformers. In *ICLR*, 2026.

Thiziri Nait Saada, Alireza Naderi, and Jared Tanner. Mind the gap: a spectral analysis of rank collapse and signal propagation in attention layers. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 45561–45587. PMLR, 13–19 Jul 2025.

Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Jungwoo Park, Young Jin Ahn, Kee-Eung Kim, and Jaewoo Kang. Monet: Mixture of monosemantic experts for transformers. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 22271–22302, 2025.

Prajit Ramachandran, Barret Zoph, and Quoc V Le. Swish: a self-gated activation function. *arXiv preprint arXiv:1710.05941*, 7(1):5, 2017.

Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. ImageNet-21K pretraining for the masses. In *NeurIPS Datasets and Benchmarks Track*, 2021.

Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *15th European Signal Processing Conference (EUSIPCO)*, pp. 606–610, Poznań, Poland, 2007.

Michael S. Ryoo, AJ Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. Token-Learner: What can 8 learned tokens do for images and videos? In *NeurIPS*, 2021.

Christos Sakaridis, Haoran Wang, Ke Li, Arpit Jadon, Wim Abbeloos, Daniel Olmeda Reino, Luc Van Gool, Dengxin Dai, et al. Acdc: The adverse conditions dataset with correspondences for robust semantic driving scene perception. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Carmelo Scribano, Mohammad Mahdi, Nedyalko Prisadnikov, Yuqian Fu, Giorgia Franchini, Danda Pani Paudel, Marko Bertogna, and Luc Van Gool. Accelerating vision foundation models with drop-in depthwise convolution, 2026. URL `https://arxiv.org/abs/2605.22132`.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, et al. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025.

Xavier Suau, Luca Zappella, and Nicholas Apostoloff. Finding experts in transformer models. *arXiv preprint arXiv:2005.07647*, 2020.

Xavier Suau, Luca Zappella, and Nicholas Apostoloff. Self-conditioning pre-trained language models. *arXiv preprint arXiv:2110.02802*, 2021.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021a.

Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *ICCV*, 2021b.

Hugo Touvron, Matthieu Cord, and Hervé Jégou. DeiT III: Revenge of the ViT. In *ECCV*, 2022.

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.

Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. MaxViT: Multi-axis vision transformer. In *ECCV*, 2022.

Pavan Kumar Anasosalu Vasu, Hadi Pouransari, Fartash Faghri, Raviteja Vemulapalli, and Oncel Tuzel. Mobileclip: Fast image-text models through multi-modal reinforced training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024.

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.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *ICCV*, 2021.

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

Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. CvT: Introducing convolutions to vision transformers. In *ICCV*, 2021.

Xun Wu, Shaohan Huang, Wenhui Wang, and Furu Wei. Multi-head mixture-of-experts. In *NeurIPS*, 2024.

Chenglin Yang, Siyuan Qiao, Qihang Yu, Xiaoding Yuan, Yukun Zhu, Alan Yuille, Hartwig Adam, and Liang-Chieh Chen. MOAT: Alternating mobile convolution and attention brings strong vision models. In *ICLR*, 2023.

Yansong Yang, Bo Tu, and Zhe Lin. UMoE: Unifying attention and FFN with shared experts. *arXiv preprint arXiv:2505.07260*, 2025.

Weihao Yu, Chenyang Si, Pan Zhou, Mi Luo, Yichen Zhou, Xinchao Wang, Jiashi Feng, and Shuicheng Yan. MetaFormer baselines for vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

Seokju Yun, Dongheon Lee, and Youngmin Ro. MetaMixer is all you need. In *NeurIPS*, 2024.

Sergey Zagoruyko and Nikos Komodakis. DiracNets: Training very deep neural networks without skip-connections. In *ICLR Workshop*, 2018.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, 2023.

Hongyi Zhang, Yann N. Dauphin, and Tengyu Ma. Fixup initialization: Residual learning without normalization. In *ICLR*, 2019.

Minshen Zhang, Xiang Hu, Jianguo Li, Wei Wu, and Kewei Tu. FlashMHF: Flash multi-head feed-forward network. *arXiv preprint arXiv:2512.06989*, 2025.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ADE20K dataset. In *CVPR*, 2017.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. iBOT: Image BERT pre-training with online tokenizer. In *ICLR*, 2022.

APPENDIX

A GRADIENT-FLOW ANALYSIS

A.1 EXPERIMENTS

The rank limited CLS-gradients established the main motivation for CALiT and specifically the CAg module. Here we describe the experiments conducted for the analysis. We choose 3 different baselines: ViT-S sup., iBOT-S, iBOT-B, encompassing supervised, self-supervised as well as S and B scales to compare against their CALiT counterparts.

Of the 50k ImageNet validation set, we select all 38.7k images with ground-truth bounding box that contained at least one image patch completely outside. We then compute the CLS loss based patch gradient (9) at the final block using the differentiation of the loss $\ell(\mathbf{c}_L)$ w.r.t. the post-LN input patches to the final block, and finally measure the numerical values. For good patch semantics, one expects to find the patch gradients to be different for the foreground object and the background. To that end, we record the mean gradient direction inside the ground-truth object bounding box as the representative object gradient and compute the cosine similarity for all patch gradients. We then measure how well the gradients agree with the representative vector, inside the bounding box and in the background.

In addition to the object directional agreement of gradients, we compute a norm-weighted contrast, wherein we weight each cosine value of Fig. 1 (b) with that patch's norm. This is depicted as the third bar for each model in Fig. 4 and Fig. 5. In Fig. 4, we compare ViT-S, CAT-S and CALiT-S supervised in 100 and 300 epochs on the same qualitative gradient agreement and quantitative metrics thus defined. For iBOT, we do not re-train the baseline or CAT-S. We therefore, plot the metrics only for the final models on the CLS-based loss gradients in ViT vs. CALiT after the final epoch.

Three observations hold in every setting. (i) ViT's patch gradients agree with the object direction both inside and outside the box, and its effective rank stays at or below the $2n_h$ bound (12 for S, 24 for B). (ii) CALiT's effective rank is 2.6–3.6$\times$ ViT's, and its norm-weighted contrast 1.6–2.6$\times$, higher on 77–91% of images. (iii) In the supervised models the effective-rank gap is present from epoch 100 and stays constant, while the contrast gap widens with training. Cross aggregation alone (CAT) already lifts the effective rank and local mixing adds contrast.

A.2 RANK OF CLS-TO-PATCH GRADIENTS

A.3 SETUP AND ASSUMPTIONS

Consider a single sample containing a CLS token z_0 and N patch tokens $z_1, \ldots, z_N \in \mathbb{R}^d$ in ViT, represented as column vectors. These tokens are the inputs to the linear QKV projections. For a pre-LayerNorm attention block, they are therefore the outputs of the pre-attention LayerNorm.

We study the patch activation-gradient matrix

$$G_Z = \begin{bmatrix} (\nabla_{z_1}\mathcal{L})^\top \\ \vdots \\ (\nabla_{z_N}\mathcal{L})^\top \end{bmatrix} \in \mathbb{R}^{N\times d}. \tag{5}$$

We assume standard linear QKV projections and scaled dot-product softmax attention, without attention dropout, query/key normalization, or token-dependent key transformations. At the last block

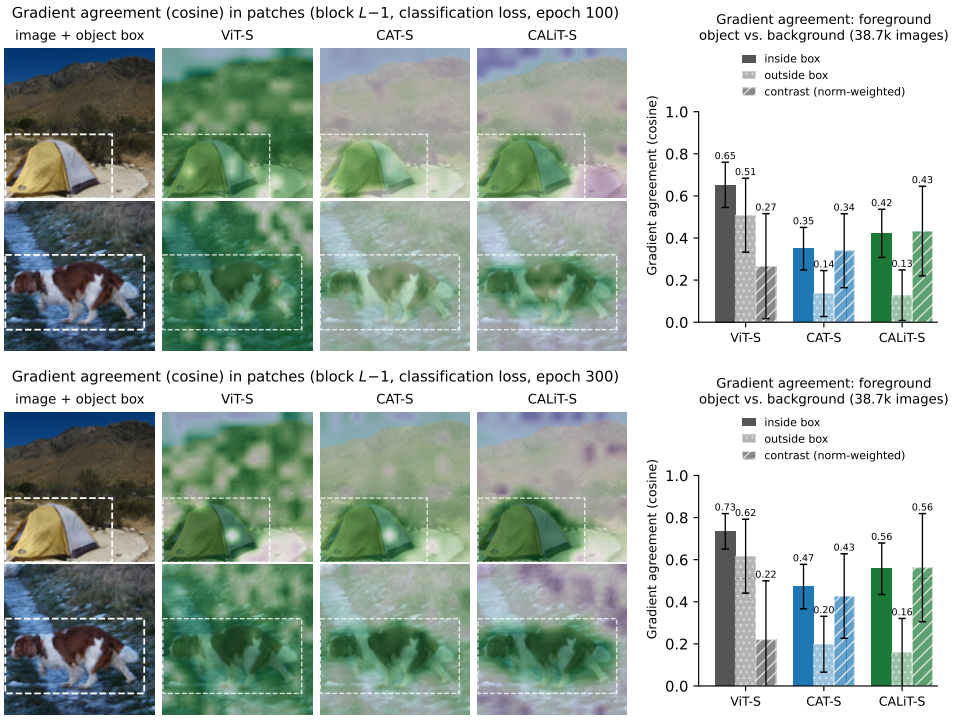

Figure 4: Gradient agreement of the supervised ViT-S/16 models at epoch 100 (top) and 300 (bottom); epoch 200 is Fig. 1. Layout as Fig. 1 b–c, with the norm-weighted contrast added.

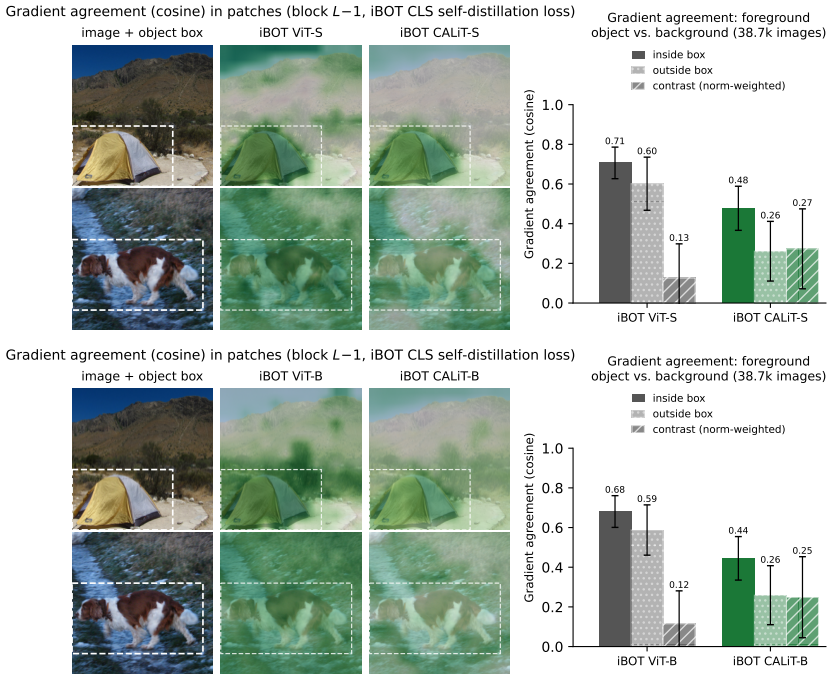

Figure 5: Gradient agreement under iBOT's CLS self-distillation loss, ViT-S/16 (top) and ViT-B/16 (bottom).

	Model	In	Out	Contrast	Contrast (norm-wtd.)	Eff. rank	Win (%)
supervised IN-1k ViT-S/16, cross-entropy							
epoch 100	ViT-S	0.653	0.508	0.144	0.266	11.0	—
	CAT-S	0.349	0.136	0.213	0.340	37.9	67
	CALiT-S	0.422	0.128	0.294	0.433	39.4	77
epoch 200	ViT-S	0.705	0.565	0.139	0.267	11.7	—
	CAT-S	0.406	0.146	0.260	0.409	38.4	77
	CALiT-S	0.501	0.144	0.357	0.510	38.9	85
epoch 300	ViT-S	0.735	0.617	0.118	0.220	12.1	—
	CAT-S	0.472	0.198	0.274	0.427	39.8	83
	CALiT-S	0.557	0.158	0.398	0.562	37.0	91
iBOT IN-1k, CLS self-distillation loss							
ViT-S/16	iBOT ViT-S	0.706	0.602	0.105	0.126	13.0	—
	iBOT CALiT-S	0.478	0.261	0.216	0.273	44.8	83
ViT-B/16	iBOT ViT-B	0.681	0.588	0.093	0.115	20.3	—
	iBOT CALiT-B	0.445	0.259	0.186	0.249	53.1	82

Table 4: **Patch-gradient statistics at the input of the last block, all models.** Means over the 38,651 ImageNet-val images with a usable object box. *In / Out*: mean cosine between each patch gradient and the mean in-box gradient direction, over patches inside / outside the box; *Contrast*: In − Out; *norm-wtd.*: the same with each patch weighted by its gradient norm. *Eff. rank*: entropy-effective rank of the $196 \times D$ patch-gradient matrix. *Win*: share of images on which the model's norm-weighted contrast exceeds the ViT of the same block.

input, the loss depends on this attention layer only through its CLS output: patch outputs through the FFN have no downstream path to the loss.

For one head, we define:

$$q = W_Q z_0, \qquad k_j = W_K z_j, \qquad v_j = W_V z_j, \tag{6}$$

where $W_Q, W_K \in \mathbb{R}^{d_k \times d}$ and $W_V \in \mathbb{R}^{d_v \times d}$.

The CLS attention logits, weights, and output are

$$s_j = \frac{q^\top k_j}{\sqrt{d_k}}, \qquad a_j = \frac{\exp(s_j)}{\sum_{t=0}^{N} \exp(s_t)}, \qquad o = \sum_{j=0}^{N} a_j v_j. \tag{7}$$

All attention sums include the CLS token. Let

$$\delta = \nabla_o \mathcal{L} \in \mathbb{R}^{d_v} \tag{8}$$

denote the upstream gradient evaluated at the current sample. Below we rewrite the observation in Sec. 3 as a proposition and prove it.

A.4 Single-head gradient decomposition

Proposition A.1 (Single-head rank bound)**.** *Under the above assumptions, the gradient of a non-CLS patch token satisfies*

$$\nabla_{z_i} \mathcal{L} = \underbrace{a_i W_V^\top \delta}_{\textit{value route}} + \underbrace{a_i \delta^\top (v_i - o) \frac{W_K^\top q}{\sqrt{d_k}}}_{\textit{key route}}. \tag{9}$$

Consequently,

$$\operatorname{rank}(G_V) \le 1, \qquad \operatorname{rank}(G_K) \le 1, \tag{10}$$

and

$$\boxed{\operatorname{rank}(G_Z) \le \min\{N, d, 2\}.} \tag{11}$$

Proof. For a patch index $i \ge 1$, the CLS query $q = W_Q z_0$ does not depend on z_i and hence, the patch-query contribution is zero. Therefore, only the Value and Key routes remain.

Value route. Holding the attention weights fixed while differentiating the value branch gives

$$\nabla_{z_i}\mathcal{L}|_V = a_i W_V^\top \delta. \tag{12}$$

This is a pathwise derivative, not an assumption that the attention weights are frozen during training.

Key route. The chain rule gives

$$\nabla_{z_i}\mathcal{L}|_K = a_i \delta^\top (v_i - o)\frac{W_K^\top q}{\sqrt{d_k}}. \tag{13}$$

Outer-product factorization. We now define the shared directions

$$u = W_V^\top \delta, \qquad w = \frac{W_K^\top q}{\sqrt{d_k}}, \tag{14}$$

and the patch-dependent coefficients

$$\beta_i = a_i \delta^\top (v_i - o). \tag{15}$$

Collecting only patch coefficients,

$$a = (a_1, \ldots, a_N)^\top, \qquad \beta = (\beta_1, \ldots, \beta_N)^\top, \tag{16}$$

we have

$$G_V = a u^\top, \qquad G_K = \beta w^\top, \tag{17}$$

and therefore

$$G_Z = a u^\top + \beta w^\top = [a \quad \beta]\,[u \quad w]^\top. \tag{18}$$

Each outer product has rank at most one. Together, Eq. 18 gives a rank of 2 establishing the result stated in Eq. (11). □

As it is obvious and trivial, we do not provide proof for the increased rank with CAg. Instead we provide measured ranks which demonstrates the increased information flow from CLS to patches during the backward pass.

B CALiT Block: Pseudocode

The pseudocode below summarizes cross aggregation and local mixing. CA denotes the parameter-free aggregation defined in Sec. 4.1. Batch dimensions, LayerScale, and stochastic depth are omitted for clarity.

CALiT block

Input / Output: $\mathbf{x} \mapsto \mathbf{z}$. $\mathbf{x}, \mathbf{z} \in \mathbb{R}^{(1+N)\times D}$, patch grid $\sqrt{N} \times \sqrt{N}$.
$\mathbf{W}_1, \mathbf{W}_2$: FFN linear layers. S: slots. H: heads (as in SA). $[\cdot\,;\,\cdot]$: Token concatenation.

Feature expansion and cross aggregation

1 $\mathbf{y} \leftarrow \mathbf{x} + \mathrm{SA}(\mathrm{LN}(\mathbf{x}))$
2 $\mathbf{h} \leftarrow \mathrm{GELU}(\mathbf{W}_1\,\mathrm{LN}(\mathbf{y}))$
3 $(\mathbf{h}_c, \mathbf{h}_p) \leftarrow \mathrm{Split}(\mathbf{h})$
4 $\mathbf{C} \leftarrow \mathrm{Reshape}(\mathbf{h}_c) \in \mathbb{R}^{S\times 4D/S}$
$\mathbf{P} \leftarrow \mathrm{Reshape}(\mathbf{h}_p) \in \mathbb{R}^{NS\times 4D/S}$
5 $\mathbf{C}' \leftarrow \mathrm{MHA}_H(\mathbf{C}, \mathbf{P}, \mathbf{P})$
6 $\mathbf{h}'_c \leftarrow \mathrm{Reshape}^{-1}(\mathbf{C}')$

Split separates the CLS row from the N patch rows.
MHA has no projection weights (Eq. (3)).

Local mixing and output projection

7 $\mathbf{G} \leftarrow \mathrm{Reshape}^{-1}(\mathbf{h}_p) \in \mathbb{R}^{4D\times\sqrt{N}\times\sqrt{N}}$
8 $\mathbf{h}'_p \leftarrow \mathrm{Reshape}(\mathrm{DWConv}_{3\times3}(\mathbf{G}))$
9 $\mathbf{z} \leftarrow \mathbf{y} + \mathbf{W}_2[\mathbf{h}'_c; \mathbf{h}'_p]$
10 **return z**

CAg (line 5) reads the un-mixed patch hiddens $\mathbf{h}_p$.
$S = 4$ for ViT-S, $S = 8$ for ViT-B.

Backbone parameters are equal to a standard ViT for CAT; CALiT adds only the dwconv kernels.

B.1 DWConv Identity initialization.

We initialize the dwconv as a Dirac kernel: center weight $= 1$, and the eight neighbor weights $= 0$. At step 0 in model optimization, $\mathrm{DWConv}(\mathbf{h}_p) = \mathbf{h}_p$, and the forward is mathematically equivalent to that of CAT. This is the same parametrization introduced by DiracNets (Zagoruyko & Komodakis, 2018) for residual-free very-deep training, applied here to a small per-channel kernel rather than a full conv. Similar identity at initialization, is also used by ReZero (Bachlechner et al., 2021) and LayerScale (Touvron et al., 2021b), although they scale the *output* of a randomly-initialised operator via a learnable scalar/diagonal whereas Dirac init constrains directly the *operator's weights*.