REVISITING SUPERVISED VISION TRANSFORMERS: AN OPTIMIZED RECIPE

Anonymous authors
Paper under double-blind review

ABSTRACT

We revisit supervised Vision Transformers training and study two components: the optimizer and the classifier. We combine gradient sampling (GS), with AdaBelief optimizer and show that gradient perturbation using running gradient statistics improves downstream performance and fine-tuning convergence. We further propose a lightweight cross-attention classifier that trains end-to-end with the backbone using multi-scale training. We show that multi-scale training with cross-attention improves accuracy, efficiency and robustness. Our models transfers well to object detection and beat MAE-pretrained detectors using less epochs. We further combine our model with modern architectural components, and our final recipe reaches 86.5% in ImageNet-1K using ViT-L and 55.8 mAP in COCO using ViT-B.

1 INTRODUCTION

The transformer (Vaswani et al., 2017) is a promising architecture and it has been widely adopted in many domains such as natural language processing (NLP), computer vision, medicine and others. In computer vision, the vision transformer (Dosovitskiy et al., 2020) has pushed the performance of many fundamental tasks like image classification (Alabdulmohsin et al., 2023), object detection (Carion et al., 2020), segmentation (Kirillov et al., 2023), depth estimation (Yang et al., 2024), image generation (Peebles & Xie, 2023), 3D scene understanding (Wang et al., 2025) and others and it has unlocked a myriad of applications. In the past years, many improvements have been made in the transformer architecture, including the use of better positional embeddings (Heo et al., 2024; Chu et al., 2021; Beyer et al., 2022; Liu et al., 2021b), convolutional stem layers (Xiao et al., 2021; Wang et al., 2021; Yuan et al., 2021a), hybrid convolutional-transformer architectures (Graham et al., 2021; d'Ascoli et al., 2021; Chen et al., 2021), multi-scale transformers (Liu et al., 2021b; Yuan et al., 2021b; Beyer et al., 2023; Fan et al., 2021), better training strategies (Touvron et al., 2022b; Steiner et al., 2021; Touvron et al., 2022a), data scaling (Alabdulmohsin et al., 2023; Zhai et al., 2022) and better feature aggregation techniques (Psomas et al., 2023; Touvron et al., 2021b; Zhai et al., 2022; Psomas et al., 2025; Touvron et al., 2021a). In this work, we revisit the vision transformer architecture from multiple axes, including architecture, training recipe and optimizer and propose a new design that further pushes its performance.

Our design proposes two components, an optimizer that combines gradient sampling, and a new efficient cross-attention classifier. The gradient sampling optimizer, GS for short, updates the network's parameters with random gaussian gradients whose mean and variance are estimated by an exponential moving average during training. GS injects larger gaussian noise to parameter-updates with high historical gradient variance and increases convergence. Intuitively, the parameters with high gradient variance are likely to be inside a local minimum and GS pushes them away by injecting proportionally larger noise. This idea is similar to Sharpness Aware Minimisation (Foret et al., 2020), however the latter injects noise in the model's parameters during the loss computation and not during gradient updates. Dropout (Hinton et al., 2012) and Dropath (Huang et al., 2016) also alter the network's parameters by randomly deactivating neurons or layers, however these affect directly the loss and not the gradient.

Further, we propose the cross attention classifier namely CA. It uses learnable class tokens that attend to the spatial tokens from the ViT backbone and it is trained end-to-end with cross-entropy. To make CA stable and competitive, we use a small head dimension, a cosine scorer and a small atten-

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.

tion gate that increase the performance and stability. Our module costs additional 0.7M parameters and 0.2 GFLOPS compared to the baseline. Although cross-attention has been explored in the past (Touvron et al., 2021a; Psomas et al., 2023; Li et al., 2021), the dominant designs use a cls-token or global average pooling. In our work, we show that cross-attention is not competitive under single resolution training but when we switch training with multiple resolutions, cross attention becomes better than the cls-token or global average pooling because it allows the class tokens to pool multi-resolution features instead of averaging the representations. Thus, we argue that cross-attention is more effective, because on the one hand multi-scale resolution effectively lowers the training time, by sampling lower resolutions, and on the other cross attention pools features more effectively and gives a better supervision signal to the backbone.

We conduct extensive ablations of our components and we show that our GS scales well in full ImageNet21K training, improves the finetuning convergence using 10 classification tasks and it increases the average performance using LLaVA-Qwen2.5-3B-Siglip model in 15 VQA benchmarks. Also, we show that the CA achieves better classification and object detection performance than a linear classifier or cls-token, it is robust across various test resolutions, distributions shifts and object detection. Finally, we combine our proposed components and several other recent architectural breakthroughs into a highly optimized recipe that trains the ViT-L to 85.5 percentage points (pp), in the standard 400 epoch training setting and 86.5 pp with 384 resolution finetuning. We list our contributions as follows:

- We study the vision transformer under multiple axes including optimizer, classifier and multi-scale training.
- We show that cross attention is more robust under multi-scale training and inference than other classification designs.
- Our ViT-L model reaches up to 86.5 top-1 accuracy on ImageNet1K pretrained from scratch and our ViT-B detector achieves mAP 55.8% using supervised learning.

We hope that our recipe can be adopted and foster further research in vision transformers.

2 Related Work

ViT. The vision transformer (ViT) architecture treats the image as a sequence of patches which means that it lacks the locality bias of CNNs and makes supervised training hard in small data regimes like Imagenet1K (Dosovitskiy et al., 2020). To this end, many early works proposed recipes in an effort to close the gap against CNNs (Touvron et al., 2022b; Steiner et al., 2021; Touvron et al., 2022a; Beyer et al., 2022; Touvron et al., 2021b; Xiao et al., 2021; Touvron et al., 2023). Most recent advancements include the use of mixed rotary embeddings that replace the absolute embeddings, dynamic normalization Zhu et al. (2025) and optimized ViT recipes (Wang et al., 2026; Fuller et al., 2025). Our work follows this direction and proposes an efficient training recipe for the ViT model that transfers well to object detection.

Linear attention / SSM. Two other recent lines of work focus on linear attention (Fan et al., 2025; Li et al., 2024d; Han et al., 2024) and state-space-models (Liu et al., 2024c; Xiao et al., 2025; Hatamizadeh & Kautz, 2025) which replace the self-attention layer with an efficient kernel approximation or a recurrent model respectively. These are promising architectures, however most of them report results under 224 image resolution, this is common for the ImageNet1K benchmark, however, the self-attention module is cheap under such small train resolutions e.g. it accounts only for 3% of the total computation inside the standard ViT-L model and self-attention is highly optimized using flash-attention (Shah et al., 2024; Dao et al., 2022). For these reason, we adopt the pure ViT design in our recipe because the self-attention might be a more expressive and universal module than linear or SSM models.

Registers Despite the nice properties of self-attention, vision transformers have several drawbacks, including attention artifacts, also known as attention sinks. Recent work shows that registers (Darcet et al., 2024) which are learnable tokens can absorb attention artifacts and enhance the performance. Jumbo-ViT (Fuller et al., 2025) optimizes the dimension of the learnable registers and strikes a good accuracy-compute trade-off. Compared to them, our ViT models use extra class tokens via the cross-attention classifier, however these class tokens interact with the backbone patch tokens only

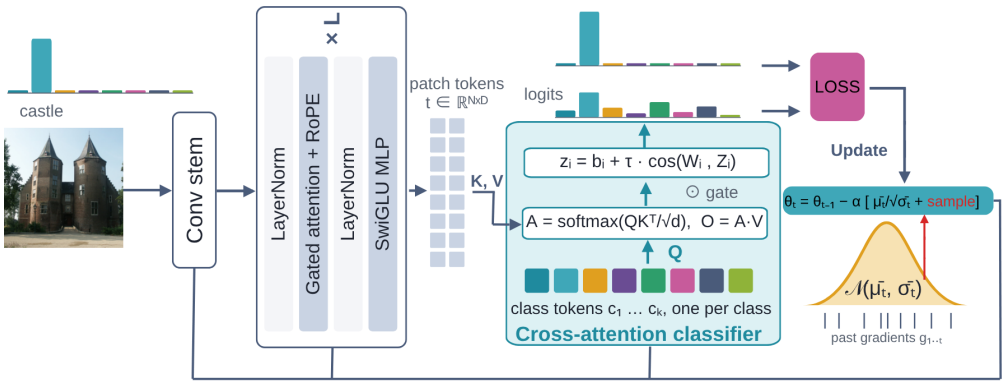


Figure 1: Our proposed pipeline combines strong architectural components, a cross-attention classifier and a gradient sampling optimizer.

in the penultimate layer, not across all layers and they are used for classification, thus they can be discarded in downstream tasks and they do not increase the backbone capacity.

Multi-resolution training Multi-resolution ViTs have previously been explored by ResFormer (Tian et al., 2023) and NaViT(Dehghani et al., 2023). ResFormer improves resolution generalization through scale-consistency training, whereas NaViT supports native variable resolutions and aspect ratios through sequence packing at large pretraining scale. Our focus is complementary: rather than introducing multi-resolution training itself, we study its interaction with the classification head under standard ImageNet-1K supervised training. We find that multi-scale training improves GAP by 0.3 pp but improves CA more i.e. 0.8 pp.

Classification head design The dominant classification designs in ViTs use a cls-token Dosovitskiy et al. (2020) or a global average pooling followed by a linear classifier (Beyer et al., 2022). Attentive pooling has also been explored in the past for classification (Touvron et al., 2021a; Psomas et al., 2023; Li et al., 2021), finetuning of self-supervised methods (Psomas et al., 2025), for multi-label classification (Liu et al., 2021a), and object discovery (Locatello et al., 2020). Our Cross Attention classifier follows this direction and further introduces attention gating and cosine normalization. Compared to previous works, we show that cross attention is more effective under multi-scale resolution training because it allows the model to learn better multi-scale representations as we show in our experiments. We propose to combine cross attention with multi-scale because it saves training time and achieves better results.

Optimizer Recent work shows that the backbone and the optimizer are coupled, and a good pair for one architecture is not a good pair for another (Li et al., 2024e; Kunstner et al., 2024). For example, CNN models are trained primarily with SGD, whereas vision transformers use AdamW Loshchilov & Hutter (2017). Some works in large language models utilise Muon Jordan et al. (2024), however, Muon is difficult to scale in practice and its performance saturates Liu et al. (2025). In our work, we jointly study the ViT and optimizer and we build upon the AdaBelief (Zhuang et al., 2020) because this optimizer has shown good potential. Adabelief, reduces the optimization step when the gradient disagrees with the momentum. However, when the step size is reduced, it might slow down the optimization and as a remedy we propose to sample random gradients adaptively. Intuitively, our sampling method allows the network to explore solutions in the parameter space, in regions with high uncertainty, and we show empirically that this increases the convergence to downstream tasks and the overall performance. Compared to SAM (Foret et al., 2020), our gradient sampling perturbs the gradient update and not the weights of the model and it requires a single forward pass.

3 METHOD

Our method consists of the gradient sampling optimizer and the cross attention classifier and our pipeline is summarized in Fig 1.

3.1 OPTIMIZER

Adaptive optimizers, have shown great success especially when training vision transformers (Touvron et al., 2022b; He et al., 2022; Siméoni et al., 2025; Bolya et al., 2025). We have developed a new optimizer that combines gradient sampling, with AdaBelief optimizer (Zhuang et al., 2020) using decoupled weight decay (Loshchilov & Hutter, 2017). Our new gradient-sampling optimizer, *GS*, is defined as follows, let θ_0 be the initial network's weights, w the weight decay and g_t the gradient with respect to θ of the objective function that we want to optimize at timestep t, α the learning rate, $\mu_0 = 0$, $\sigma_0 = 0$ the exponential moving averages and β_1, β_2 exponential moving average coefficients. Then the gradient sampling update is:

Algorithm 1 *GS* optimizer

1: $\mu_t = \beta_1 \cdot \mu_{t-1} + (1 - \beta_1) \cdot g_t$
2: $\sigma_t = \beta_2 \cdot \sigma_{t-1} + (1 - \beta_2) \cdot (\mu_t - g_t)^2$
3: $\bar{\mu}_t = \mu_t / (1 - \beta_1^t)$
4: $\bar{\sigma}_t = \sigma_t / (1 - \beta_2^t)$
5: $\theta_t = \theta_{t-1} - \alpha \cdot \left(\frac{\bar{\mu}_t}{\sqrt{\bar{\sigma}_t} + \epsilon} + w\theta_{t-1} + \mathbb{N}(\bar{\mu}_t, \bar{\sigma}_t) \right)$

GS adaptively reweights the learning rate based on the exponential moving averages of the mean $\bar{\mu}_t$ and variance $\bar{\sigma}_t$. Additionally, it adds a random gaussian noise for every parameter θ, independently, that is proportional to the historic variance and centered around the historic mean. When the variance is small, then the generated gradient is similar to the historic gradient and it preserves the optimization trend. When the variance is large, it means that the network has reached an uncertain point that could be a harmful optimization outlier because the gradient is dissimilar to the historic mean. In such cases, GS reduces the adaptive step size and it adds a random gradient to escape this point. This is similar to AdaBelief (Zhuang et al., 2020), which also reduces the adaptive step when the gradient is dissimilar to the historic mean. However, the reduction in the step-size might halt the optimization and slow the training. Instead, our method samples a random gradient using the historic mean and variance and accelerates the optimization as we show in our experiments.

Connection to AdaBelief. Ignoring decoupled weight decay for clarity, the proposed update can be rewritten as

$$\theta_t = \theta_{t-1} - \alpha \left(\frac{\bar{\mu}_t}{\sqrt{\bar{\sigma}_t} + \epsilon} + \mathbb{N}(\bar{\mu}_t, \bar{\sigma}_t) \right). \tag{1}$$

Using the reparameterization

$$\mathbb{N}(\bar{\mu}_t, \bar{\sigma}_t) = \bar{\mu}_t + \xi_t, \qquad \xi_t \sim \mathbb{N}(0, \bar{\sigma}_t), \tag{2}$$

the update becomes

$$\theta_t = \theta_{t-1} - \alpha(\frac{\bar{\mu}_t}{\sqrt{\bar{\sigma}_t} + \epsilon}) - \alpha(\bar{\mu}_t + \xi_t). \tag{3}$$

The first term in equation 3 is exactly the AdaBelief-style adaptive direction and our proposed method adds an additional drift term $\bar{\mu}_t$ together with a zero-mean Gaussian perturbation ξ_t. The ξ_t parameter represents the exploration term and it does not grow with time, i.e. $E(\xi_t) = 0$. The extra drift remains controlled because if we assume the same condition as AdaBelief i.e. $\|g_t\| \leq G$ for all t, then the momentum recursion

$$\mu_t = \beta_1 \mu_{t-1} + (1 - \beta_1) g_t \tag{4}$$

implies

$$\|\mu_t\| \leq G(1 - \beta_1^t), \qquad \|\bar{\mu}_t\| \leq G. \tag{5}$$

In words, the additional drift introduced by our method is bounded and does not grow with time.

Visual Explanation. In Fig. 2, we show how our proposed GS optimizer operates using a toy 2D example where the y-axis is the cost function and x-axis the parameters. There are three regions of interest R1, R2 and R3 and for each a different Gaussian is used for sampling. In R1 and R3 regions the gradients, (green arrows) are pointing to the same direction thus, the historic variance $\bar{\sigma}_t$ is small and the generated gradient will be close to the historic mean $\bar{\mu}_t$, which is the

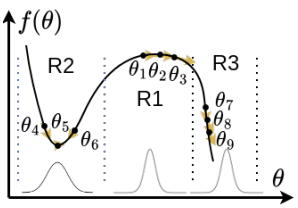


Figure 2: Gradient sampling

trend of the optimization. When the gradients misalign like in region R2, then the variance becomes large and the generated gradient is more noisy, i.e it deviates from the historic mean. In region R2 Adabelief lowers the learning rate because it considers this region a minima, in comparison to that, our GS perturbs the gradients which might help escape local minima or discover flatter minima.

3.2 Cross Attention Classifier

Let $t \in \mathbb{R}^{N \times D}$ denote the spatial tokens from the last transformer layer, where N is the number of tokens and D is the hidden dimension. CA uses cross-attention with learnable class tokens $c \in \mathbb{R}^{k \times D}$ to produce a classification vector $z \in \mathbb{R}^k$. Below we show the equations using one attention head for brevity, in practice we use multiple heads. Using projection matrices $W_q, W_k, W_v, W_g \in \mathbb{R}^{D \times D_h}$, we compute

$$
\begin{aligned}
&Q = cW_q, \quad K = tW_k, \quad V = tW_v, \quad A = \operatorname{softmax}\left(\frac{QK^\top}{\sqrt{D_h}}\right), \quad O = AV, \\
&g = \operatorname{sigmoid}\left[\left(\frac{1}{N}\sum_{n=1}^{N} t_n\right) W_g\right], \quad Z = g \odot O, \quad z_i = b_i + \tau \, \frac{\langle W_i, Z_i \rangle}{\|W_i\| \, \|Z_i\|}.
\end{aligned}
\tag{6}
$$

The attention matrix $A \in (0,1)^{k \times N}$ connects the class-tokens to the patch tokens and it shows how each class correlates with the spatial tokens, akin to a class-conditioned spatial attention map. The matrices $Z, W \in \mathbb{R}^{k \times D_h}$, correspond to a classification vector and a cosine scorer respectively, g is an elementwise-attention gate, similar to the squeeze gate in SEnets (Hu et al., 2018), $b \in \mathbb{R}^k$ is a class bias and $\tau \in \mathbb{R}$ a temperature parameter.

We found that element-wise gating g improves the performance and the use of a cosine scorer W stabilizes the training. We also ablated the head dimension D_h and the number of attention heads, finding that more heads with a small D_h work well.

Multi-scale training. Multi-scale training uses various image resolutions which is advantageous because it lowers the computing resources by sampling low image train resolutions using the uniform $\mathbf{U}()$ distribution, it regularises the training and increases the performance. We have found the CA benefits greatly from multi-scale training because it can capture the most salient features across various training resolutions via the cross attention layer. During testing, CA shows better performance than the average pooling for both higher and lower test-time resolutions as we show in our experiments.

4 Experimental Setup.

We perform experiments using primarily ViT models on a variety of benchmarks including classification, detection and vision question-answering (VQA) tasks. We ablate ViT components on various benchmarks, particularly focusing on ImageNet1K classification training from scratch using a default setting (baseline) that is inspired by Deit3 (Touvron et al., 2022b). This setting is a ViT-B model with convolutional stemming, fixed sinusoidal embeddings, softmax attention, layer-scale, drop-path 0.3, global pooling, linear classifier, pretrained from scratch for 400 epochs using AdamW, learning rate 0.003, weight decay 0.05, beta$_1$ 0.9, beta$_2$ 0.99, batch-size 2048, mixup-alpha 0.8, cutmix-alpha 1.0, 3-Augment, color-jittering 0.3 and repeated augmentations on ImageNet1K. We use a plethora of datasets summarised in Table 1 for training and evaluating the components of our ViT recipe. Unless specified all models use the default baseline. We give more details in the Appendix.

5 Results

Ablations on the gradient sampling optimiser We compare our optimizer against AdamW on ImageNet21K pretraining and downstream finetuning in 8 large classic vision datasets summarised in Table 1. We use the same optimizer-hyperparameters, and we train ViT-Small/Base models and for 90 epochs and finetune for 100 epochs using a single optimizer for both phases. As Table 2

Table 1: Dataset details.

Dataset	Train size	Test size	#Classes	Type
ImageNet21K-p (Ridnik et al., 2021)	12,358,688	561,050	11,221	Object Classification
ImageNet (Deng et al., 2009)	1,281,167	50,000	1000	Object Classification
ImageNetV2 (Recht et al., 2019)	-	10,000	1000	Object Classification
iNaturalist2017 (Van Horn et al., 2018)	579,184	95,986	5,089	Species Classification
iNaturalist2018 (Van Horn et al., 2018)	437,513	24,426	8,142	Species Classification
iNaturalist2019 (Van Horn et al., 2018)	237,943	30300	1010	Species Classification
iNaturalist2021 (Van Horn et al., 2018)	2,686,843	500,000	10,000	Species Classification
CIFAR10 (Krizhevsky et al., 2009)	50,000	10,000	10	Object Classification
CIFAR100 (Krizhevsky et al., 2009)	50,000	10,000	100	Object Classification
Places365 (Zhou et al., 2017)	1,803,459	36,499	365	Scene Classification
LLaVA-1.5-PT (Liu et al., 2023a)	558000	N/A	N/A	Image Caption
OpenLLaVA-Next (Chen & Xing, 2024)	1000000	N/A	N/A	VQA
COCO (Lin et al., 2014)	118287	5000	80	Object Detection

Table 2: Comparative results on 10 popular image classification datasets.

Method	I21K	I1K	I1KV2	iNat2017	iNat2018	iNat2019	iNat2021	Places365	C10	C100	Average
ViT-S AdamW	44.2	83.0	73.2	71.1	74.9	77.2	81.8	**55.6**	**98.8**	90.5	75.0
ViT-S GS	**44.4**	83.0	**73.6**	71.1	**75.2**	**77.3**	**82.0**	55.3	98.7	**91.0**	**75.2**
ViT-B AdamW	46.7	84.6	75.0	73.5	79.3	80.1	85.0	**56.2**	98.9	91.6	77.1
ViT-B GS	**46.8**	**84.8**	**75.5**	**73.7**	79.3	**80.6**	85.0	56.0	98.9	**92.3**	**77.3**

shows, our optimizer increases the macro-average performance on ViT-S and ViT-B but the accuracy gains are small. The reason is that both our optimiser and AdamW converge to a similar solution, however, ours has a faster convergence in the first epochs as we show in the Figure 3 for all downstream datasets. This faster convergence may occur because the random gradients during pretraining make the model more robust to downstream distribution shifts.

Experiments on ImageNet1k. We do a comprehensive empirical study of our optimizer in ImageNet1K. We test our optimizer against both Squeeze-Excite (Hu et al., 2018) Resnet50 and ViT models, against other optimizers like AdamW (Loshchilov & Hutter, 2017), Muon (Liu et al., 2025), AdaBelief(Zhuang et al., 2020) and SAM (Foret et al., 2020) and we also ablate the choices of the sampling distributions and sampling counts in Table 3. The results show that (a) our optimizer increases the performance of all ViT models, (b) using 1 random sample is enough (0 sample falls back to Adabelief), (c) our optimizer outperforms all other optimizers and (d) the use of a Gaussian sampling distribution is better than the use of Logistic or the Gumbel.

Experiments with Vision Language Model. We further test the performance of gradient sampling using vision language models (VLMs), following the LLAVA series paradigm (Li et al., 2024c; Liu et al., 2023b; 2024a; Li et al., 2024a;b). In detail, we pretrain the Qwen2.5 (Qwen et al., 2025) with the SigLIP-so400M (Zhai et al., 2023) vision encoder, using the LLaVA-1.5-PT dataset that has 558k image-text pairs (Liu et al., 2023a), keeping all parameters frozen except for the connector MLP. Next, we perform instruction tuning where we fully finetune the models using the OpenLLaVA-Next dataset (Chen & Xing, 2024). To evaluate the performance, we use the LMM-evaluation kit (Zhang et al., 2024) that includes many vision-language benchmarks such as MME (Fu et al., 2023), AI2D (Kembhavi et al., 2016), DocVQA (Mathew et al., 2021), GQA (Hudson &

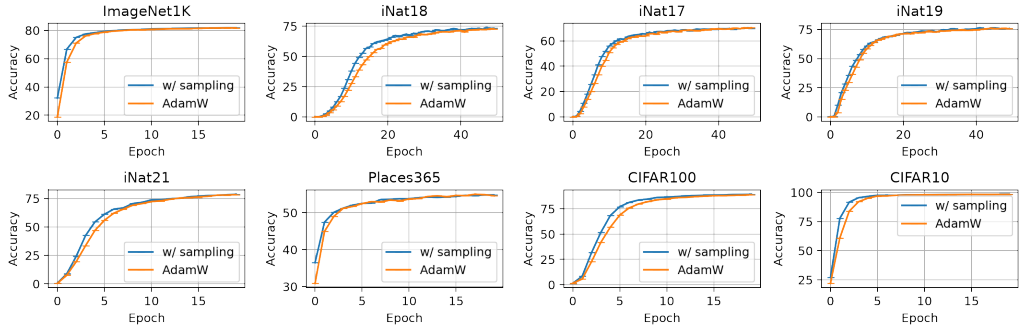


Figure 3: Comparison of AdamW versus our optimizer in validation-accuracy curves for 8 downstream datasets. Our optimizer clearly converges faster in all cases, sometimes achieving higher total accuracy.

Table 3: Ablations of our Gradient Sampling optimizer.

Architecture	top-1
SE-R50 AdamW	81.5
SE-R50 + GS	**81.6**
ViT-B AdamW	83.7
ViT-B + GS	**83.9**
ViT-L AdamW	84.3
ViT-L + GS	**84.4**

(a) Results on I1K.

Sample count	top-1
0	83.7
1	**83.9**
2	**83.9**
4	83.8

(b) Ablation on I1K using ViT-B.

Method	top-1
AdamW	83.7
Muon	83.6
AdaBeliefW	83.7
SAM	81.2
GS	**83.9**

(c) SOTA comparison using ViT-B and I1K.

Distribution	top-1
No sampling	44.2
Logistic	44.3
Gumbel	44.0
Gaussian	**44.4**

(d) Sampling distributions on I21K using ViT-S.

Table 4: Results on the visual instruction following task, using a Qwen2.5-SigLIP model.

Method	mme	ai2d	doc^{vqa}	gqa	mmb^{cc}	mmb^{cn}	mmb^{en}	mmb^{ru}	mmmu	ocrb	ok^{vqa}	pope	sqa^{I}	vqa^{T}	vis^{wiz}	avg
DeepSeek-Janus7B	1567	-	-	59.1	-	-	79.2	-	41.0	-	-	87.4	-	-	-	-
LLaVA-1.5-7B	1511	-	-	62.0	-	58.3	64.3	-	-	-	-	86.0	71.6	61.3	50.0	-
LLaVA-Next-7B	1519	66.6	74.4	64.2	-	60.6	67.4	-	35.8	-	-	86.5	70.1	64.9	57.6	-
VILA-7B	1533	-	-	62.3	-	61.7	70.3	-	-	-	-	85.5	68.3	64.4	57.8	-
InternVL-Chat-7B	1525	-	-	62.9	-	-	-	-	-	-	-	87.1	-	57.0	52.5	-
Qwen2.5-1.5B	1445	65.0	46.3	59.9	38.6	61.9	68.0	40.7	28.0	39.4	51.2	87.3	72.2	52.9	51.0	55.6
w/GS	1432	65.5	49.0	60.3	39.2	61.9	68.4	42.3	29	41.8	51.3	85.6	74.3	53.9	51.4	**56.4**
Qwen2.5-3B	1549	70.3	52.0	61.8	40.8	71.9	73.5	46.1	30.7	40.7	52.0	86.7	71.0	56.2	58.7	59.3
w/ GS	1529	71.2	53.4	62.2	42.5	69.9	73.1	48.2	28.8	42.6	50.0	87.6	75.7	56.7	61.7	**60.0**

Manning, 2019), MMB (Liu et al., 2024b), OCRB (Liu et al., 2024d), MMMU (Yue et al., 2024), OKVQA (Marino et al., 2019), POPE (Li et al., 2023), SQA (Lu et al., 2022), TextVQA (Singh et al., 2019) and VisWiz (Gurari et al., 2018). As Table 4 shows our optimizer improves the performance of the VLM by 0.8pp macro-average on Qwen2.5 1.5B and by 0.6pp macro-average on Qwen2.5 3B compared to AdamW.

Ablations on the ViT architecture In an effort to push the performance we use many recent architectural components and we ablate them in Table 5. We start with the baseline model that achieves 83.5pp. We add multi-scale training and the performance increases to 83.7 reducing the training time by 22%. Further, we add our optimizer and the performance increases to 83.9. Using ROPE mixed-embeddings (Heo et al., 2024) and positional embeddings (Dosovitskiy et al., 2020) we reach 84.1pp.[1] When we switch to an eval-crop ratio of 1.0, it adds 0.1pp and using SwiGLU activation (Shazeer, 2020) the model reaches 84.3. Using gated element-wise attention (Qiu et al., 2026), the performance climbs to 84.6. Using CA classifier, we further boost the ViT performance to 84.8, adding negligible extra parameters and FLOPS to the model. Next we use co-training (Touvron et al., 2023) that adds 0.1pp. Using logit calibration Saerens et al. (2002) we increase the performance by another 0.1pp without increasing FLOPS. We call this model optimised ViT, oViT.

Ablations on the Cross Attention Classifier We perform detailed ablations of the Cross Attention classifier, including the head dimension and the effects of multi-scale training and multi-scale inference in Table 6. CA improves the performance of the (v5) Rope-ViT-B model, across all test resolutions, as we show in Table 6-a, particularly at 512 where it surpasses the linear classifier by 0.8pp. In Table 6-b we show that using a head dimension of 16 is the most cost-effective as it improves the performance of (v6) and uses less parameters. In Table 6-c, we show that multiscale training (MS) greatly benefits the CA classifier as it increases the performance by 0.8pp and the relative improvement is larger than using the linear classifier with MS. Also, we note that under single-scale resolution, CA achieves smaller performance than GAP, this shows that the CA and MS have a synergetic effect. In Table 6-d we show that the CA-pretrained backbone increases the object detection performance by 0.3pp in AP, indicating that the multiscale representations are superior for object detection. In Table 6-e, we show that CA is better than both GAP and cls-token using the (v0.5) architecture. Finally, in Table 6-f we show that multiscale training saves 20% training time and the CA classifier needs 4 extra H200 GPU hours to train (that is 30 extra minutes using a 8GPU node).

Robustness We perform experiments using various ImageNet distribution shifts like ImageNet-V2 Recht et al. (2019), ImageNet-R (Hendrycks et al., 2021a), ImageNet-A (Hendrycks et al., 2021b), ImageNet-Sketch (Wang et al., 2019) and ImageNet-C[2] Hendrycks & Dietterich (2019) to understand the robustness of the CA classifier compared to the linear classifier. Further, we freeze the

[1] We interpolate the positional embeddings during MS training.

[2] For Imagenet-C we report the macro average of the most severely corrupted sets as in (Fuller et al., 2025).

Table 5: Ablation of improvements using ViT-B trained from scratch on ImageNet1K using 224 resolution.

Components	#Params (M)	FLOPS (G)	top-1
(v0) Baseline	82.8	17.1	83.5
(v0.5) Baseline (MS)	82.8	17.1	83.7
(v1) Gradient Sampling	82.8	17.1	83.9
(v2) ROPE	83.0	17.1	84.1
(v3) eval-crop ratio: 1.0	83.0	17.1	84.2
(v4) SwiGLU	83.0	17.1	84.3
(v5) Gated Attention	89.5	18.2	84.6
(v6) Attentive Classifier	90.2	18.4	84.8
(v7) cotraining submodels	90.2	18.4	84.9
(v8) calibration = oViT	90.2	18.4	**85.0**

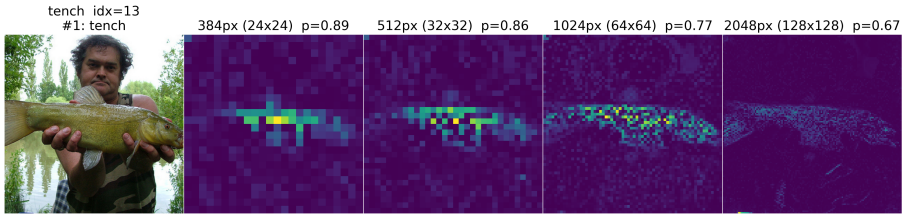


Figure 4: Multiscale robustness: CA captures many fine grained details of the object such as the tail of the fish, even in very high resolutions i.e. 2048x2048, even though it was trained at 384x384.

(v5) and (v6) backbones and fit new CA and linear classifiers respectively to disentangle the influence of representation and classifier. We show in Table 7 that our CA classifier improves the performance by 1.3pp on average compared to the linear classifier and it is more robust under image corruptions, i.e. it achieves 3.4pp higher average top-1 accuracy in ImageNet-C. Also, we note that the CA refit to the linear-pretrained backbone, i.e. (v5) $\rightarrow$ (v6), has smaller performance than linear refit to the CA-pretrained backbone i.e. (v6) $\rightarrow$ (v5). This shows that the CA classifier and backbone co-adapt during training and they need to be trained jointly.

Comparison against the SOTA We compare our final design, i.e. oViT against strong transformer models like Deit 3 (Touvron et al., 2022b), ConvNext (Liu et al., 2022) and the recent ViT-5 model Wang et al. (2026), which is a recent strong recipe paper in Table 8 using eval-crop 1.0 for fairness. Our oViT-B achieves 85.0 top-1 outperforming ViT-5, by 0.8pp using half epochs. Our oViT-L model achieves 85.5pp outperforming the Deit3-L by 0.6pp and half epochs. When we finetune the models to 384 resolution, our oViT-B achieves 85.9, surpassing the ViT-5 by 0.5pp, and using the oViT-L we achieve 86.5 top-1 surpassing Deit-3 by $0,7$pp, which is a large margin. We also note that multi-scale train saves approximately 20% train-time and there is no need for finetuning step making our recipe efficient and effective.

Object detection In the past, supervised-pretrained ViT models were unable to compete against Swin (Liu et al., 2021b) or CNNs and the remedy was to pretrain the ViT with MAE (Li et al., 2022; He et al., 2022). This works but MAE pretrain requires 1600 epochs for a competitive ViT-B model He et al. (2022). We show that using a small run, i.e. 400 epoch ImageNet1K supervised pretraining

Table 6: Ablations of our Cross Attention Classifier.

Res	128	192	224	256	288	336	384	512
(v5)	80.3	83.8	84.6	84.8	84.9	84.6	84.0	81.9
(v6)	**80.4**	**84.0**	**84.8**	**85.0**	**85.2**	**85.1**	**84.7**	**82.7**

(a)Evaluation across various test resolutions.

Head Dim	Top-1
8	84.1
16	**84.8**
32	84.8
64	84.5

(b)Head dimension using (v6).

Method	top-1
(v5) no MS	84.3
(v6) no MS	84.0
(v5)	84.6
(v6)	**84.8**

(c)Multi scale train.

Method	Backbone	AP	AP50	AP75
Dino-4scale-3x	ViT-B	54.3	73.0	59.4
Dino-4scale-3x	ViT-B+CA	**54.6**	73.3	59.6

(d)Results in COCO using DINO and 3x schedule.

Method	Top-1
cls-token	83.4
GAP	83.7
CA	**84.0**

(e)Classification designs using (v0.5).

Method	H200-GPU hours
No MS	177.6
(v5)	139.0
(v6)	143.7

(f)Training time.

Table 7: Robustness comparison under various ImageNet distribution shifts.

Model	ImageNet1K	ImageNet1K-V2	ImageNetR	ImageNetA	ImageNet-Sketch	ImageNet-C	Average
(v5)	84.6	74.5	**53.8**	43.3	39.6	40.3	56.0
(v6)	**84.8**	**74.8**	53.3	**46.1**	**41.2**	**43.7**	**57.3**
(v5) $\rightarrow$ (v6)	82.9	71.4	44.4	17.1	34.1	39.1	48.2
(v6) $\rightarrow$ (v5)	**84.3**	**74.2**	**52.6**	**44.8**	**39.1**	**41.7**	**56.1**

Table 8: SOTA Comparison in ImageNet1K, $\uparrow$ shows finetuning the 224 model.

Model	Train Resolution	Finetune resolution	#Params (M)	FLOPS (G)	Train epochs	Finetuning epochs	top-1
ConvNext B	224	n/a	89	15.4	300	0	83.8
Deit-3 B	192	224	87	17.6	800	20	83.8
ViT-5 B	192	224	87	17.9	800	20	84.2
oViT B	$\mathbf{U}(8, 14) \cdot 16$	n/a	90.2	18.4	400	0	**85.0**
ConvNext B	n/a	$\uparrow$ 384	89	45.0	n/a	30	85.1
Deit-3 B	n/a	$\uparrow$ 384	87	55.0	n/a	20	85.0
ViT-5 B	n/a	$\uparrow$ 384	87	55.0	n/a	20	85.4
oViT B	n/a	$\uparrow$ 384	90.2	59.8	n/a	20	**85.9**
ConvNext L	224	n/a	198	34.4	300	0	84.3
Deit-3 L	192	224	304	61.6	800	20	84.9
ViT-5 L	192	224	304	62.8	400	20	84.9
oViT L	$\mathbf{U}(8, 14) \cdot 16$	n/a	323	66.7	400	0	**85.5**
ConvNext L	n/a	$\uparrow$ 384	198	101.0	n/a	30	85.5
Deit-3 L	n/a	$\uparrow$ 384	304	191.2	n/a	20	85.8
ViT-5 L	n/a	$\uparrow$ 384	305	192.5	n/a	20	86.0
oViT L	n/a	$\uparrow$ 384	323	205.9	n/a	20	**86.5**

is good enough to produce competitive results in COCO. In detail, we adopt Dino (Zhang et al., 2022) 3x, 4 scales with strides $4, 8, 16, 32$ using Detr-style (Zhu et al., 2020) augmentations and the mmdetection framework (Chen et al., 2019). As the results suggest in Table 9, using the same Dino detector, oViT trains with less epochs and compute (more discussion in Appendix) and achieves 0.8pp better AP than ViTDet-B. Also, oViT-B achieves better AP than strong backbones like Swin Liu et al. (2021b) and InternImage (Wang et al., 2023).

Visualisations We show the attention maps $A \in \mathbf{R}^{1\times N}$, of the top-1 prediction in the cross-attention layer as the resolution N increases from 384 to 2048 resolution in Figure 4. CA is robust even in extremely larger test-time resolutions and captures fine-grained object details like the *tail* of the fish, highlighting CA's multiscale feature extraction ability. However, we also note that as the resolution increases the classification confidence of the model decreases, which can flip the class decision to a wrong prediction.

6 Conclusion

In this work, we propose a strong ViT recipe that includes an optimizer and an efficient cross-attention classifier. We have tested these components using various datasets and architectures and have shown that the optimizer shows faster convergence than AdamW and that the CA classifier is more robust in multi-scale training and inference, and domain shifts. We show that Cross-attention, is superior when it is combined with multi-resolution training and using it with other refinements achieves strong results in ImageNet1K e.g. up to 86.5 top-1. Our models transfers well object detection tasks and train faster and better than MAE backbones achieving up to 55.8 mAP. We hope that our recipe can be useful for practitioners and the community and it can catalyze further research in vision transformers.

Table 9: COCO val2017 object detection with the DINO-4scale detector. Baseline results are from the detrex model zoo (Ren et al., 2023). †Uses IN-22K data.

Backbone	Pretraining	Epochs	AP^{box}
ViTDet-B (Li et al., 2022)	IN-1K, MAE 1600 epochs	50	55.0
Swin-B (Liu et al., 2021b)	IN-1K, sup. 300 epochs	12	53.4
ConvNext-B (Liu et al., 2022)	IN-22K 90 epochs	12	55.1
InternImage-B (Wang et al., 2023)	IN-1K, sup. 300 epochs	12	54.7
oViT-B (ours)	IN-1K, sup. 400 epochs	36	**55.8**

7 AI STATEMENT

The research ideas, code, experiments and manuscript were produced by the authors, we used AI agents only to speed up running experiments, to check the code for bugs, and to check the manuscript for typos, grammar and consistency. All AI-assisted changes were reviewed and verified by the authors.

REFERENCES

Ibrahim M Alabdulmohsin, Xiaohua Zhai, Alexander Kolesnikov, and Lucas Beyer. Getting vit in shape: Scaling laws for compute-optimal model design. *Advances in Neural Information Processing Systems*, 36:16406–16425, 2023.

Lucas Beyer, Xiaohua Zhai, and Alexander Kolesnikov. Better plain vit baselines for imagenet-1k. *arXiv preprint arXiv:2205.01580*, 2022.

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*, pp. 14496–14506, 2023.

Daniel Bolya, Po-Yao Huang, Peize Sun, Jang Hyun Cho, Andrea Madotto, Chen Wei, Tengyu Ma, Jiale Zhi, Jathushan Rajasegaran, Hanoona Rasheed, et al. Perception encoder: The best visual embeddings are not at the output of the network. *arXiv preprint arXiv:2504.13181*, 2025.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020.

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

Lin Chen and Long Xing. Open-llava-next: An open-source implementation of llava-next series for facilitating the large multi-modal model community. `https://github.com/xiaoachen98/Open-LLaVA-NeXT`, 2024.

Zhengsu Chen, Lingxi Xie, Jianwei Niu, Xuefeng Liu, Longhui Wei, and Qi Tian. Visformer: The vision-friendly transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 589–598, 2021.

Xiangxiang Chu, Zhi Tian, Bo Zhang, Xinlong Wang, and Chunhua Shen. Conditional positional encodings for vision transformers. *arXiv preprint arXiv:2102.10882*, 2021.

Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. *Advances in neural information processing systems*, 33:18613–18624, 2020.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in neural information processing systems*, 35:16344–16359, 2022.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International Conference on Learning Representations*, 2024.

Mostafa Dehghani, Basil Mustafa, Josip Djolonga, Jonathan Heek, Matthias Minderer, Mathilde Caron, Andreas Steiner, Joan Puigcerver, Robert Geirhos, Ibrahim M Alabdulmohsin, et al. Patch n'pack: Navit, a vision transformer for any aspect ratio and resolution. *Advances in Neural Information Processing Systems*, 36:2252–2274, 2023.

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, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Stéphane d'Ascoli, Hugo Touvron, Matthew L Leavitt, Ari S Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. In *International conference on machine learning*, pp. 2286–2296. PMLR, 2021.

Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6824–6835, 2021.

Qihang Fan, Huaibo Huang, and Ran He. Breaking the low-rank dilemma of linear attention. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25271–25280. IEEE, 2025.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

Anthony Fuller, Yousef Yassin, Daniel G Kyrollos, Evan Shelhamer, and James R Green. Thicker and quicker: A jumbo token for fast plain vision transformers. *arXiv preprint arXiv:2502.15021*, 2025.

Benjamin Graham, Alaaeldin El-Nouby, Hugo Touvron, Pierre Stock, Armand Joulin, Hervé Jégou, and Matthijs Douze. Levit: a vision transformer in convnet's clothing for faster inference. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 12259–12269, 2021.

Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3608–3617, 2018.

Dongchen Han, Ziyi Wang, Zhuofan Xia, Yizeng Han, Yifan Pu, Chunjiang Ge, Jun Song, Shiji Song, Bo Zheng, and Gao Huang. Demystify mamba in vision: A linear attention perspective. *Advances in neural information processing systems*, 37:127181–127203, 2024.

Ali Hatamizadeh and Jan Kautz. Mambavision: A hybrid mamba-transformer vision backbone. In *2025 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 25261–25270. IEEE, 2025.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8320–8329. IEEE, 2021a.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15262–15271, 2021b.

Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. Rotary position embedding for vision transformer. In *European Conference on Computer Vision*, pp. 289–305. Springer, 2024.

Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141, 2018.

Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *European conference on computer vision*, pp. 646–661. Springer, 2016.

Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6700–6709, 2019.

Keller Jordan, Yuchen Jin, Vlado Boza, Jiacheng You, Franz Cesista, Laker Newhouse, and Jeremy Bernstein. Muon: An optimizer for hidden layers in neural networks, 2024. URL https://kellerjordan.github.io/posts/muon/.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pp. 235–251. Springer, 2016.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

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

Frederik Kunstner, Alan Milligan, Robin Yadav, Mark Schmidt, and Alberto Bietti. Heavy-tailed class imbalance and why adam outperforms gradient descent on language models. *Advances in Neural Information Processing Systems*, 37:30106–30148, 2024.

Bo Li, Hao Zhang, Kaichen Zhang, Dong Guo, Yuanhan Zhang, Renrui Zhang, Feng Li, Ziwei Liu, and Chunyuan Li. Llava-next: What else influences visual instruction tuning beyond data?, 2024a. URL https://llava-vl.github.io/blog/2024-05-25-llava-next-ablations/.

Bo Li, Kaichen Zhang, Hao Zhang, Dong Guo, Renrui Zhang, Feng Li, Yuanhan Zhang, Ziwei Liu, and Chunyuan Li. Llava-next: Stronger llms supercharge multimodal capabilities in the wild, 2024b. URL https://llava-vl.github.io/blog/2024-05-10-llava-next-stronger-llms/.

Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024c.

Liangzhi Li, Bowen Wang, Manisha Verma, Yuta Nakashima, Ryo Kawasaki, and Hajime Nagahara. Scouter: Slot attention-based classifier for explainable image recognition. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1026–1035. IEEE, 2021.

Lujun Li, Zimian Wei, Peijie Dong, Wenhan Luo, Wei Xue, Qifeng Liu, and Yike Guo. Attnzero: efficient attention discovery for vision transformers. In *European Conference on Computer Vision*, pp. 20–37. Springer, 2024d.

Siyuan Li, Juanxi Tian, Zedong Wang, Luyuan Zhang, Zicheng Liu, Weiyang Jin, Yang Liu, Baigui Sun, and Stan Z Li. Unveiling the backbone-optimizer coupling bias in visual representation learning. *arXiv preprint arXiv:2410.06373*, 2024e.

Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *European conference on computer vision*, pp. 280–296. Springer, 2022.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Zachary C. Lipton, Yu-Xiang Wang, and Alexander J. Smola. Detecting and correcting for label shift with black box predictors. In *International Conference on Machine Learning (ICML)*, pp. 3122–3130. PMLR, 2018.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023b.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024a. URL https://llava-vl.github.io/blog/2024-01-30-llava-next/.

Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, et al. Muon is scalable for LLM training. *arXiv preprint arXiv:2502.16982*, 2025.

Shilong Liu, Lei Zhang, Xiao Yang, Hang Su, and Jun Zhu. Query2label: A simple transformer way to multi-label classification. *arXiv preprint arXiv:2107.10834*, 2021a.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pp. 216–233. Springer, 2024b.

Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. Vmamba: Visual state space model. *Advances in neural information processing systems*, 37:103031–103063, 2024c.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 67(12), December 2024d. ISSN 1869-1919. doi: 10.1007/s11432-024-4235-6. URL http://dx.doi.org/10.1007/s11432-024-4235-6.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021b.

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

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *Advances in neural information processing systems*, 33:11525–11538, 2020.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pp. 3195–3204, 2019.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 2200–2209, 2021.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Bill Psomas, Ioannis Kakogeorgiou, Konstantinos Karantzalos, and Yannis Avrithis. Keep it simpool: Who said supervised transformers suffer from attention deficit? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5350–5360, 2023.

Bill Psomas, Dionysis Christopoulos, Eirini Baltzi, Ioannis Kakogeorgiou, Tilemachos Aravanis, Nikos Komodakis, Konstantinos Karantzalos, Yannis Avrithis, and Giorgos Tolias. Attention, please! revisiting attentive probing through the lens of efficiency. *arXiv preprint arXiv:2506.10178*, 2025.

Zihan Qiu, Zekun Wang, Bo Zheng, Zeyu Huang, Kaiyue Wen, Songlin Yang, Rui Men, Le Yu, Fei Huang, Suozhi Huang, et al. Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free. *Advances in Neural Information Processing Systems*, 38:100092–100118, 2026.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL `https://arxiv.org/abs/2412.15115`.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pp. 5389–5400. PMLR, 2019.

Tianhe Ren, Shilong Liu, Feng Li, Hao Zhang, Ailing Zeng, Jie Yang, Xingyu Liao, Ding Jia, Hongyang Li, He Cao, et al. detrex: Benchmarking detection transformers. *arXiv preprint arXiv:2306.07265*, 2023.

Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. Imagenet-21k pretraining for the masses. *arXiv preprint arXiv:2104.10972*, 2021.

Marco Saerens, Patrice Latinne, and Christine Decaestecker. Adjusting the outputs of a classifier to new a priori probabilities: A simple procedure. *Neural Computation*, 14(1):21–41, 2002.

Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. *Advances in Neural Information Processing Systems*, 37:68658–68685, 2024.

Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8317–8326, 2019.

Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv preprint arXiv:2106.10270*, 2021.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

Rui Tian, Zuxuan Wu, Qi Dai, Han Hu, Yu Qiao, and Yu-Gang Jiang. Resformer: Scaling vits with multi-resolution training. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22721–22731. IEEE, 2023.

Hugo Touvron, Matthieu Cord, Alaaeldin El-Nouby, Piotr Bojanowski, Armand Joulin, Gabriel Synnaeve, and Hervé Jégou. Augmenting convolutional networks with attention-based aggregation. *arXiv preprint arXiv:2112.13692*, 2021a.

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*, pp. 32–42, 2021b.

Hugo Touvron, Matthieu Cord, Alaaeldin El-Nouby, Jakob Verbeek, and Hervé Jégou. Three things everyone should know about vision transformers. In *European Conference on Computer Vision*, pp. 497–515. Springer, 2022a.

Hugo Touvron, Matthieu Cord, and Hervé Jégou. Deit iii: Revenge of the vit. In *European conference on computer vision*, pp. 516–533. Springer, 2022b.

Hugo Touvron, Matthieu Cord, Maxime Oquab, Piotr Bojanowski, Jakob Verbeek, and Hervé Jégou. Co-training 2l submodels for visual recognition. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11701–11710. IEEE, 2023.

Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8769–8778, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Feng Wang, Sucheng Ren, Tiezheng Zhang, Predrag Neskovic, Anand Bhattad, Cihang Xie, and Alan Yuille. ViT-5: Vision transformers for the mid-2020s. *arXiv preprint arXiv:2602.08071*, 2026.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pp. 10506–10518, 2019.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5294–5306, 2025.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 568–578, 2021.

Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14408–14419. IEEE, 2023.

Chaodong Xiao, Minghan Li, Deyu Meng, Lei Zhang, et al. Spatial-mamba: Effective visual state space models via structure-aware state fusion. In *International Conference on Learning Representations*, volume 2025, pp. 73777–73795, 2025.

Tete Xiao, Mannat Singh, Eric Mintun, Trevor Darrell, Piotr Dollar, and Ross Girshick. Early convolutions help transformers see better. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 30392–30400. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/ff1418e8cc993fe8abcfe3ce2003e5c5-Paper.pdf.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37:21875–21911, 2024.

Kun Yuan, Shaopeng Guo, Ziwei Liu, Aojun Zhou, Fengwei Yu, and Wei Wu. Incorporating convolution designs into visual transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 579–588, 2021a.

Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zi-Hang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 558–567, 2021b.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9556–9567, 2024.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6023–6032, 2019.

Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12104–12113, 2022.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11975–11986, 2023.

Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, and Ziwei Liu. Lmms-eval: Reality check on the evaluation of large multimodal models, 2024. URL https://arxiv.org/abs/2407.12772.

Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017.

Jiachen Zhu, Xinlei Chen, Kaiming He, Yann LeCun, and Zhuang Liu. Transformers without normalization. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14901–14911. IEEE, 2025.

Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.

Juntang Zhuang, Tommy Tang, Yifan Ding, Sekhar C Tatikonda, Nicha Dvornek, Xenophon Papademetris, and James Duncan. Adabelief optimizer: Adapting stepsizes by the belief in observed gradients. *Advances in neural information processing systems*, 33:18795–18806, 2020.

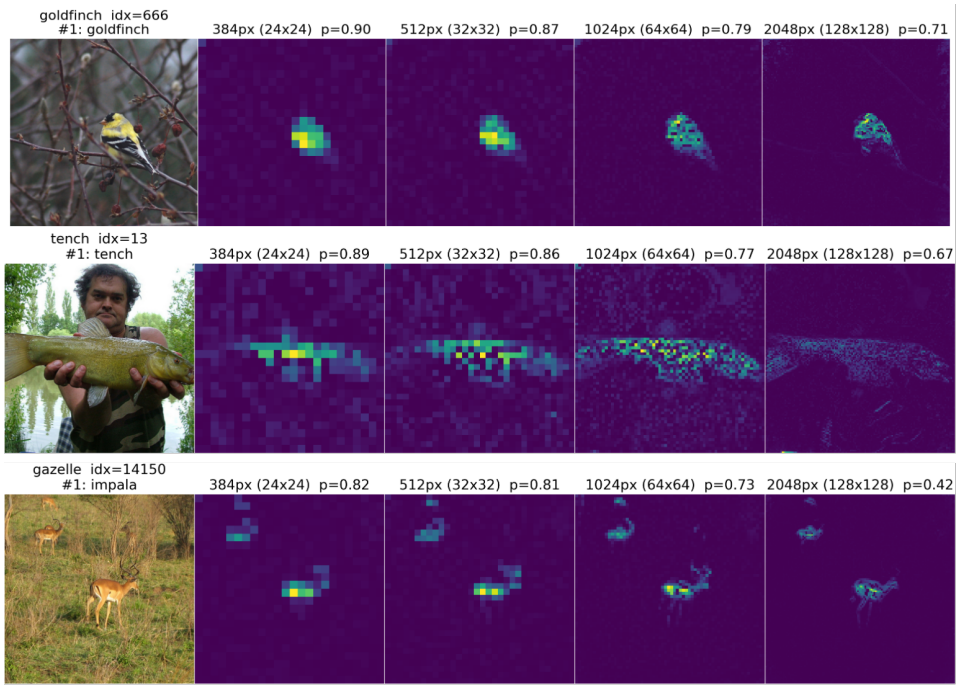


Figure 5: Visualization of the cross-attention maps across multi-scale image resolutions.

A Appendix

A.1 Cross-attention head Geometry

When using the CA, we apply the same number of heads as the backbone, that is 12 for a ViT-B and 16 for ViT-L, etc, and we lower the head dimension for all heads to make it efficient. Further, we also use Layernorm for the k vector. The attention gate is applied elementwise as we found that it improves the performance compared to head-wise gating. We do not use a W_o out-projection after the SDPA, we use directly the cosine scorer in the concatenated heads and a temperature parameter 16.

A.2 Multi-scale training

When we train with multi-scale training, we uniformly sample image resolutions from the set of $\{128, 144, 160, 176, 192, 208, 224\}$ resolutions at the start of a new epoch and resample in the next epoch keeping the total batch-size fixed. We did not test with increasing the sampling frequency to sub-epoch level. In Figure 5 we show more cross-attention maps, our CA captures many details from the objects. In Fig.6 we show how CA compares against the linear classifier using ViT-B. As the figure shows CA produces better results for higher resolutions.

A.3 Test-time calibration

In many applications, it is often that the train data distribution might be differ from the test-data distribution causing significant accuracy degradation. One way to solve this problem, is by calibrating the model's own prediction to match the test-distribution (Lipton et al., 2018), however, this requires prior knowledge of the aggregate downstream distribution which is unrealistic. We circumvent this problem, by calculating the model's bias using the train-distribution. In detail, we propagate a uniform train-image distribution inside the model, and calculate the average per-class prediction. Using the average per-class predictions we calculate the model bias $\mathbb{B}_y$ as follows:

$$\mathbb{B}_y = -\kappa \log(\frac{1}{N}\sum_{i=1}^{N} softmax(f_\theta(x_i))) \tag{7}$$

where $f_\theta(x) \in \mathbb{R}^y$, is the classifier that maps the image x to a logit vector of dimension y and κ a weight hyper-parameter. During testing, we inject the bias inside the softmax to debias the model's predictions. Intuitively, $\mathbb{B}_y$ captures the model's intrinsic bias because it is the delta between the controlled uniform train distribution and the predicted distribution.

A.4 Experimental details

In Table 10, we show more details about the pretraining and finetuning of our OViT recipe.

Table 10: ImageNet1K oViT config.

config	ImageNet-1k@224	ImageNet-1k ↑ 384
optimizer	GS	GS
learning rate	3e-3	1e-5
weight decay	0.05	0.1
optimizer momentum	β_1, β_2=0.9, 0.99	β_1, β_2=0.9, 0.99
batch size	2048	512
learning rate schedule	cosine decay	cosine decay
warmup epochs	20	5
training epochs	400	20
augmentation	ThreeAug Touvron et al. (2022b)	rand-augment (Cubuk et al., 2020) (9, 0.5)
label smoothing Szegedy et al. (2016)	0.0	0.1
mixup Zhang et al. (2017)	0.8	0.8
cutmix Yun et al. (2019)	1.0	1.0
drop path Huang et al. (2016)	0.3(B), 0.5 (L)	0.35(B), 0.6 (L)
Cross attention head dimension	16	16
Loss	BCE	CE
Color jitter	0.3	0.3
Repeated augmentations	4	0

B Training Compute: Supervised ViT-B versus MAE

We compare the total training cost of our supervised ViT-B in configuration (v6) of Table 5 (Gradient Sampling, RoPE, SwiGLU, gated attention and the attentive classifier; 400 epochs with multi-scale training, MS) against MAE ViT-B (He et al., 2022) with its best-performing schedule: 1600 epochs of masked pre-training followed by 100 epochs of supervised fine-tuning. MAE's accuracy requires both stages, so both are counted. Our model has a single supervised stage.

Counting protocol. We measure the per-image forward cost by instantiating each model and counting the multiply–accumulate operations of all linear layers, convolutions and attention products with PyTorch's `FlopCounterMode`. Elementwise operations (normalisation, activations, RoPE, softmax) are excluded. Following common practice, one multiply–accumulate is reported as one FLOP. Both models are counted with the same protocol. For MAE ViT-B fine-tuning it gives 17.6 GFLOPs, which matches the commonly reported value. For our (v6) model at 224 it gives 19.2 GFLOPs, of which 0.7 G are attention products. Training compute per image is taken as three times the forward cost (forward plus backward), and one epoch is 1,281,167 images. MAE pre-training encodes only the 25% visible patches, so its encoder costs 4.4 GFLOPs. Its 8-block, 512-wide decoder processes all 196 positions and costs 5.4 GFLOPs, giving 9.8 GFLOPs per image in total. MS training draws the input resolution per epoch, so we weight the per-resolution cost by the schedule the run actually used (Table 11). This gives an average of 12.0 GFLOPs per training image.

Table 11: Per-resolution forward cost of our (v6) ViT-B and the number of epochs the MS schedule spent at each resolution in the reported 400-epoch run. The final epochs are fixed at 224.

Resolution	128	144	160	176	192	208	224	Avg. / Total
Forward GFLOPs	6.22	7.85	9.69	11.74	14.00	16.47	19.17	12.01
Epochs	64	53	59	68	48	46	62	400

Results. Our (v6) ViT-B reaches 1.2 points higher top-1 than MAE ViT-B with 3.6× less training compute (18.5 vs. 66.8×10^{18} FLOPs). MAE's pre-training stage alone costs 3.2× our full training. The two methods have comparable parameter counts at inference: 90.2 M for ours and 86.4 M for

Table 12: Total training cost on ImageNet-1K. GFLOPs are per image, forward pass. Training compute is $3\times$ forward over all epochs. GPU-hours are wall clock on $8\times$ NVIDIA H200 and include in-training evaluation. †Measured for 100 pre-training epochs with our reimplementation of MAE and scaled linearly to 1600. ‡He et al. (2022).

Method	Stage	#Params (M)	GFLOPs / image	Train compute (10^{18} FLOPs)	GPU-hours	Top-1
MAE ViT-B	pre-train, 1600 ep	111.7	9.8	60.0	276.8†	–
	fine-tune, 100 ep	86.4	17.6	6.8	21.0	–
	total			**66.8**	**297.8**	83.6‡
Ours, (v6)	supervised + MS, 400 ep	90.2	12.0 (19.2 at 224)	**18.5**	**165.1**	**84.8**
MAE relative to ours				3.6×	1.8×	−1.2

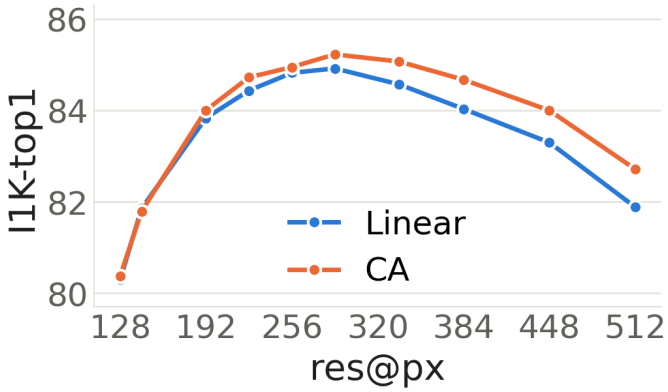


Figure 6: Cross attention is more robust than the linear classifier especially for higher resolutions.

the fine-tuned MAE ViT-B. MAE's pre-training additionally uses a 26.0 M-parameter decoder that is discarded afterwards.

FLOPs versus GPU-hours. In wall clock the gap narrows to $1.8\times$ (165.1 vs. 297.8 GPU-hours), because the two pipelines achieve different hardware throughput. MAE pre-training sustains about 120 TFLOP/s per GPU (counting 2 FLOPs per multiply–accumulate) and MAE fine-tuning about 180 TFLOP/s. Both use bf16 and large per-GPU batches, and pre-training encodes only a quarter of the tokens. Our supervised run sustains about 60 TFLOP/s: it uses fp16 autocast, runs small matrix multiplications at the low MS resolutions, and evaluates after every epoch. GPU-hours therefore reflect these two implementations, whereas the FLOP comparison is implementation-independent.