Attention, Please!
Revisiting Attentive Probing
Through the Lens of Efficiency

Bill Psomas*, Dionysis Christopoulos*, Eirini Baltzi, Ioannis Kakogeorgiou, Tilemachos Aravanis,
Nikos Komodakis, Konstantinos Karantzalos, Yannis Avrithis, Giorgos Tolias

How do we train vision models today?

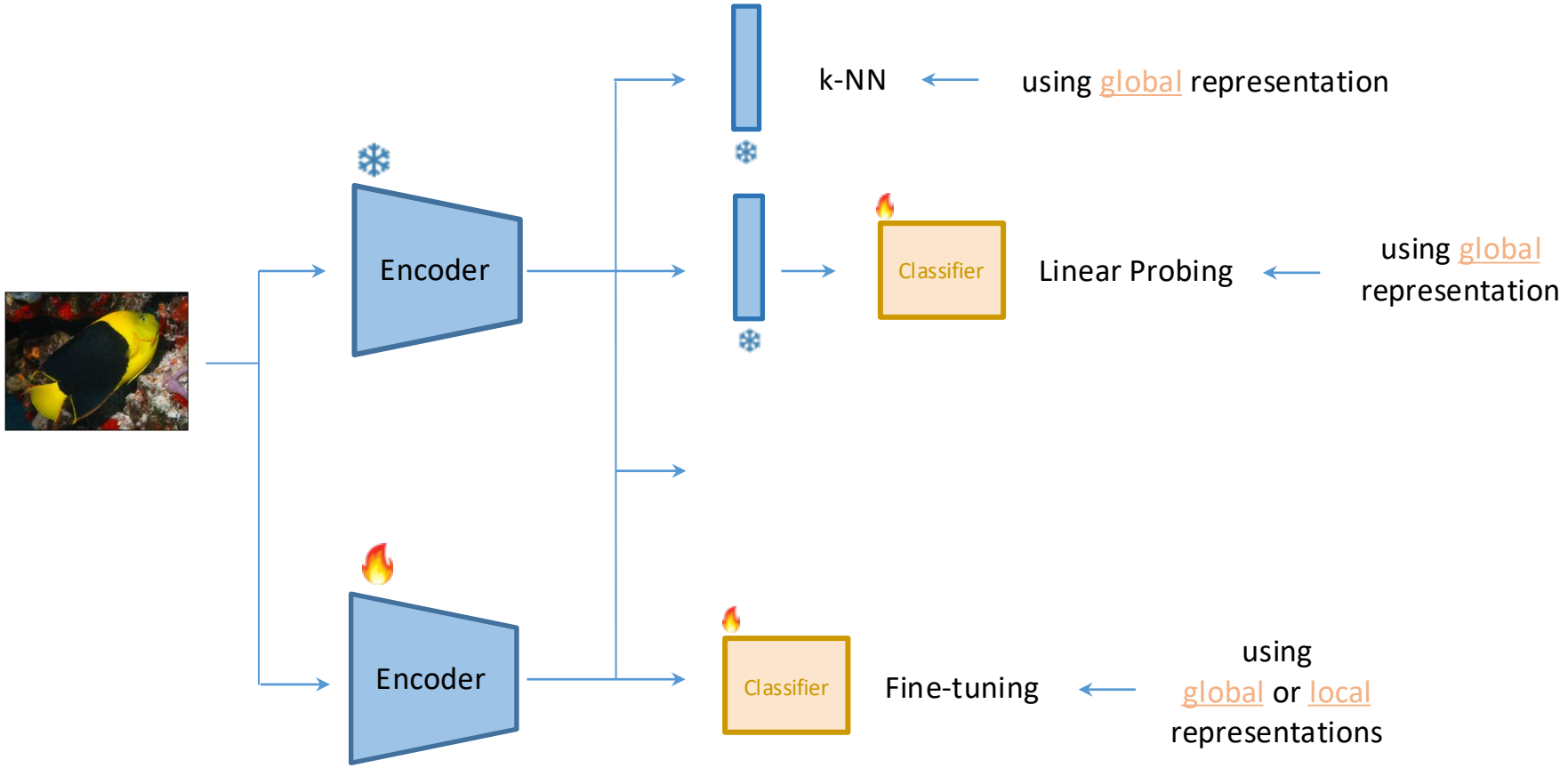
SSL / Joint Embedding (JEPA)
Objective: invariance across augmentations
Representation: strong **global**

Vision-Language (VLMs)
Objective: align image + text embeddings
Representation: strong **global**

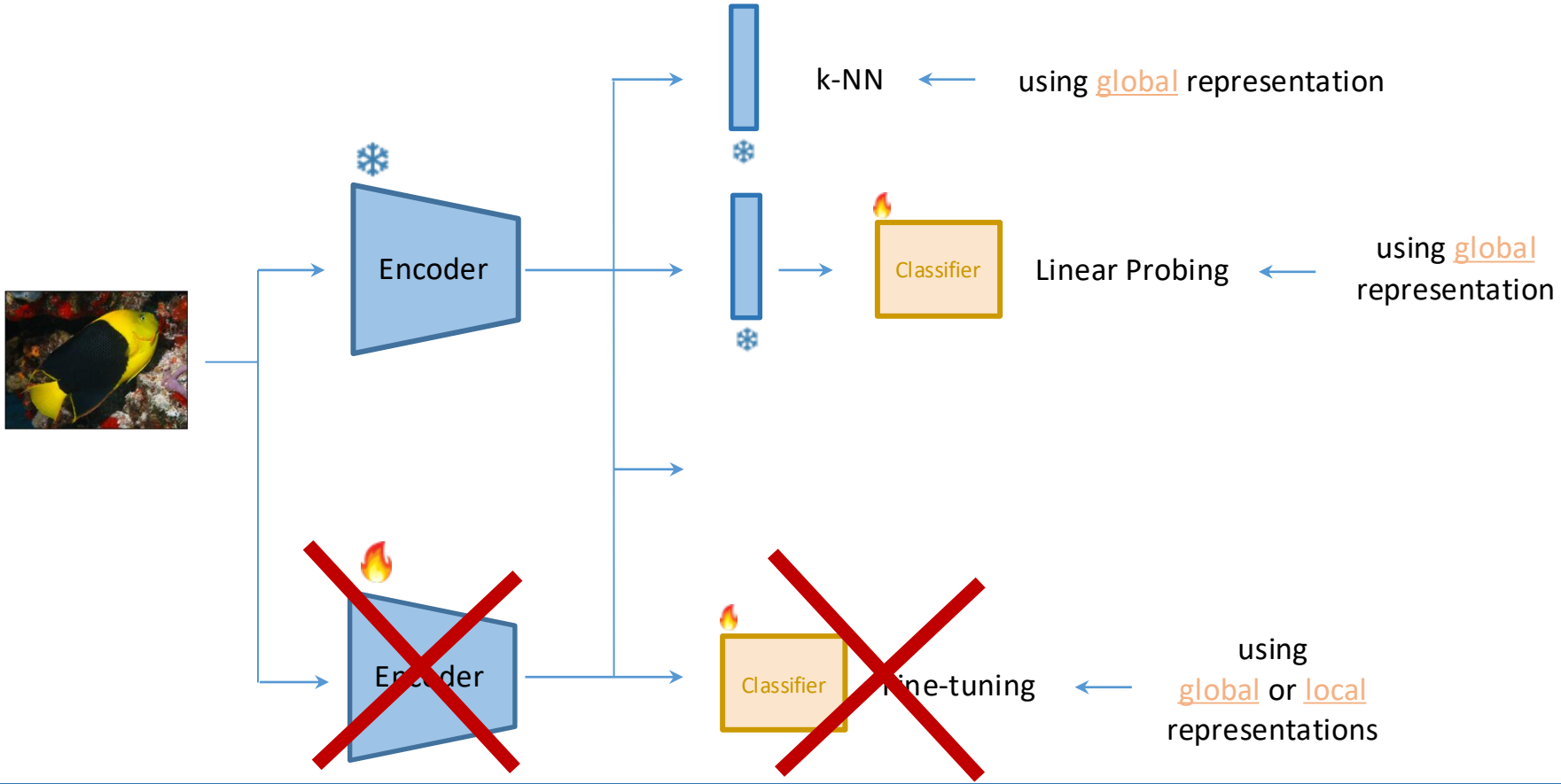
SSL / Masked Image Modeling (MIM)
Objective: reconstruct / predict masked patches
Representation: strong **local**

Generative (Diffusion / AR)
Objective: generate images by modelling data distribution
Representation: strong **local**

How do we evaluate vision models (for global tasks) today?



How do we evaluate vision models (for global tasks) today?



Are these evaluation protocols suitable
for pre-training families optimizing local representations?

Are these evaluation protocols suitable for pre-training families optimizing local representations?

Method	Architecture	k-NN	Linear Probing	
MAE (MIM) *optimizing **local**	ViT-L/16	58.2	76.0	
DINO (JEPA) *optimizing **global**	ViT-B/16	**76.1**	**77.3**	

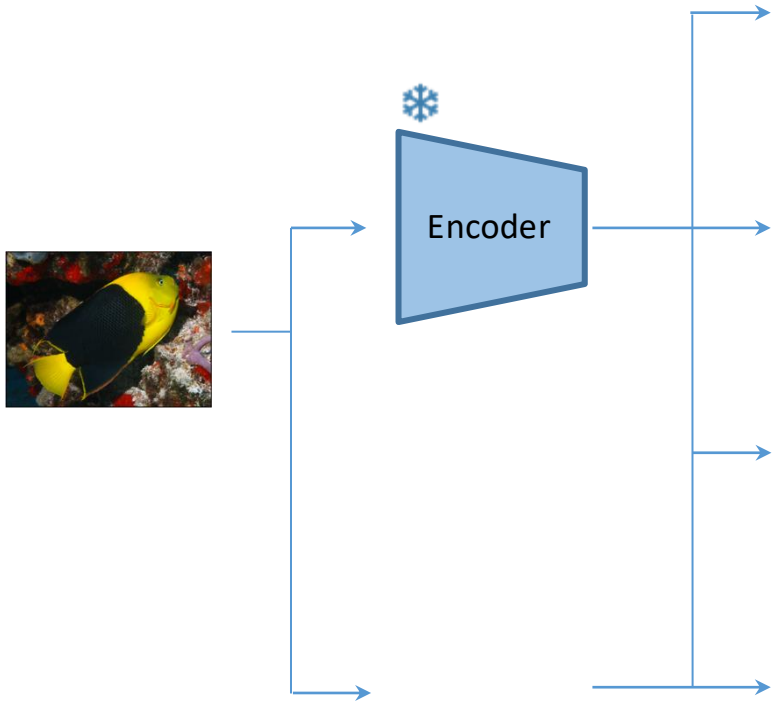
*ImageNet-1k validation set results.

Are these evaluation protocols suitable for pre-training families optimizing local representations?

Method	Architecture	k-NN	Linear Probing	Attentive Probing (ours)
MAE (MIM) *optimizing local	ViT-L/16	58.2	76.0	**79.3**
DINO (JEPA) *optimizing global	ViT-B/16	**76.1**	**77.3**	77.8

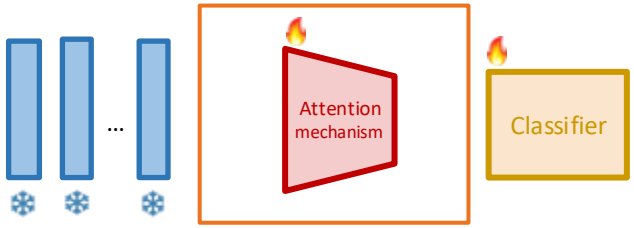
*ImageNet-1k validation set results.

Attentive probing



✅ Using local representations.

✅ Aggregating them through attention.



Attentive Probing

Attentive probing: Issues

- ✗ Attentive probes vary significantly.
- ✗ Often suffer from over-parametrization.
- ✗ Lack of framework for comparison.

Attentive probing: Our contributions

- ❌ Attentive probes vary significantly.
- ❌ Often suffer from over-parametrization.
- ❌ Lack of framework for comparison.

- ✅ Systematic benchmark and analysis of attentive probing methods.
- ✅ New method achieving best performance vs. parameter-efficiency trade-off.

Attentive probing: Our contributions

- ❌ Attentive probes vary significantly.
- ❌ Often suffer from over-parametrization.
- ❌ Lack of framework for comparison.

- ✅ Systematic benchmark and analysis of attentive probing methods.
- ✅ New method achieving best performance vs. parameter-efficiency trade-off.

Attentive probing: Our contributions

- ❌ Attentive probes vary significantly.
- ❌ Often suffer from over-parametrization.
- ❌ Lack of framework for comparison.

- ✅ Systematic benchmark and analysis of attentive probing methods.
- ✅ New method achieving best performance vs. parameter-efficiency trade-off.

METHOD	QUERY SOURCE	KEY TRANSFORM	VALUE TRANSFORM	ATTENTION	POOLING
MHCA	$\boldsymbol{q}_j \in \mathbb{R}^{d_a}$	$K_j = W_{K_j} X$	$V_j = W_{V_j} X$	$\boldsymbol{a}_j = \sigma_m(K_j^\top \boldsymbol{q}_j)$	$\boldsymbol{y}_j = V_j \boldsymbol{a}_j$
AbMILP	$\boldsymbol{q} \in \mathbb{R}^{D_i}$	$K = X$	$V = X$	$\boldsymbol{a} = \sigma_m(K^\top \boldsymbol{q})$	$\boldsymbol{y} = V \boldsymbol{a}$
AIM	$\boldsymbol{q}_j \in \mathbb{R}^{d_a}$	$K_j = W_{K_j} \mathrm{BN}(X)$	$V_j = W_{V_j} \mathrm{BN}(X)$	$\boldsymbol{a}_j = \sigma_m(K_j^\top \boldsymbol{q}_j)$	$\boldsymbol{y}_j = V_j \boldsymbol{a}_j$
DELF	$\boldsymbol{q} \in \mathbb{R}^{D_i}$	$K = \mathrm{ReLU}(W X)$	$V = W X$	$\boldsymbol{a} = \sigma_m(K^\top \boldsymbol{q})$	$\boldsymbol{y} = V \boldsymbol{a}$
SimPool	$\boldsymbol{q} = W_Q \boldsymbol{u} \in \mathbb{R}^{D_i}, \boldsymbol{u} = \frac{1}{N} X^\top \mathbf{1}$	$K = W_K \mathrm{LN}(X)$	$V = \mathrm{LN}(X)$	$\boldsymbol{a} = \sigma_m(K^\top \boldsymbol{q})$	$\boldsymbol{y} = V \boldsymbol{a}$
V-JEPA	$\boldsymbol{q}_j = W_{Q_j} \boldsymbol{u} \in \mathbb{R}^{d_a}, \boldsymbol{u} \in \mathbb{R}^{D_i}$	$K_j = W_{K_j} \mathrm{LN}(X)$	$V_j = W_{V_j} \mathrm{LN}(X)$	$\boldsymbol{a}_j = \sigma_m(K_j^\top \boldsymbol{q}_j)$	$\boldsymbol{y} = \phi(W_P V_j \boldsymbol{a}_j)$
EP (ours)	$\boldsymbol{q}_j \in \mathbb{R}^{D_i}$	$K = X$	$V_j = W_{V_j} X$	$\boldsymbol{a}_j = \sigma_m(K^\top \boldsymbol{q}_j)$	$\boldsymbol{y}_j = V_j \boldsymbol{a}_j$

Attentive probing: Our contributions

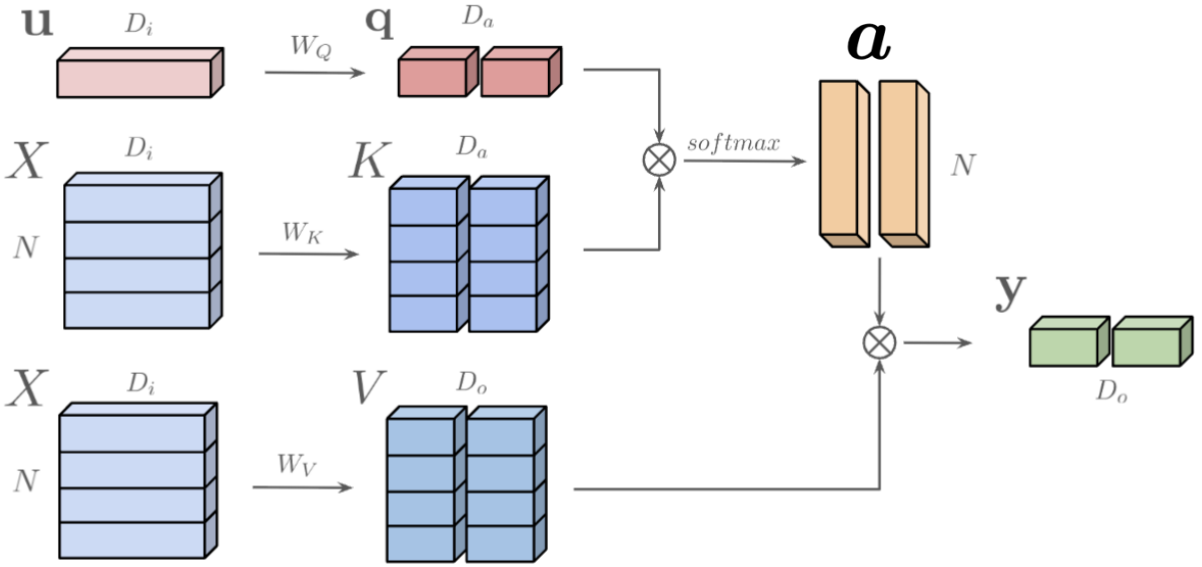
- ❌ Attentive probes vary significantly.
- ❌ Often suffer from over-parametrization.
- ❌ Lack of framework for comparison.

- ✅ Systematic benchmark and analysis of attentive probing methods.
- ✅ New method achieving best performance vs. parameter-efficiency trade-off.

METHOD	QUERY SOURCE	KEY TRANSFORM	VALUE TRANSFORM	ATTENTION	POOLING
MHCA	$\boldsymbol{q}_j \in \mathbb{R}^{d_a}$	$K_j = W_{K_j} X$	$V_j = W_{V_j} X$	$\boldsymbol{a}_j = \sigma_m(K_j^\top \boldsymbol{q}_j)$	$\boldsymbol{y}_j = V_j \boldsymbol{a}_j$
AbMILP	$\boldsymbol{q} \in \mathbb{R}^{D_i}$	$K = X$	$V = X$	$\boldsymbol{a} = \sigma_m(K^\top \boldsymbol{q})$	$\boldsymbol{y} = V \boldsymbol{a}$
AIM	$\boldsymbol{q}_j \in \mathbb{R}^{d_a}$	$K_j = W_{K_j}\,\mathrm{BN}(X)$	$V_j = W_{V_j}\,\mathrm{BN}(X)$	$\boldsymbol{a}_j = \sigma_m(K_j^\top \boldsymbol{q}_j)$	$\boldsymbol{y}_j = V_j \boldsymbol{a}_j$
DELF	$\boldsymbol{q} \in \mathbb{R}^{D_i}$	$K = \mathrm{ReLU}(W X)$	$V = W X$	$\boldsymbol{a} = \sigma_m(K^\top \boldsymbol{q})$	$\boldsymbol{y} = V \boldsymbol{a}$
SimPool	$\boldsymbol{q} = W_Q \boldsymbol{u} \in \mathbb{R}^{D_i}, \boldsymbol{u} = \frac{1}{N} X^\top \mathbf{1}$	$K = W_K\,\mathrm{LN}(X)$	$V = \mathrm{LN}(X)$	$\boldsymbol{a} = \sigma_m(K^\top \boldsymbol{q})$	$\boldsymbol{y} = V \boldsymbol{a}$
V-JEPA	$\boldsymbol{q}_j = W_{Q_j} \boldsymbol{u} \in \mathbb{R}^{d_a}, \boldsymbol{u} \in \mathbb{R}^{D_i}$	$K_j = W_{K_j}\,\mathrm{LN}(X)$	$V_j = W_{V_j}\,\mathrm{LN}(X)$	$\boldsymbol{a}_j = \sigma_m(K_j^\top \boldsymbol{q}_j)$	$\boldsymbol{y} = \phi(W_P V_j \boldsymbol{a}_j)$
EP (ours)	$\boldsymbol{q}_j \in \mathbb{R}^{D_i}$	$K = X$	$V_j = W_{V_j} X$	$\boldsymbol{a}_j = \sigma_m(K^\top \boldsymbol{q}_j)$	$\boldsymbol{y}_j = V_j \boldsymbol{a}_j$

Vanilla Multi-Head Cross Attention (MHCA)

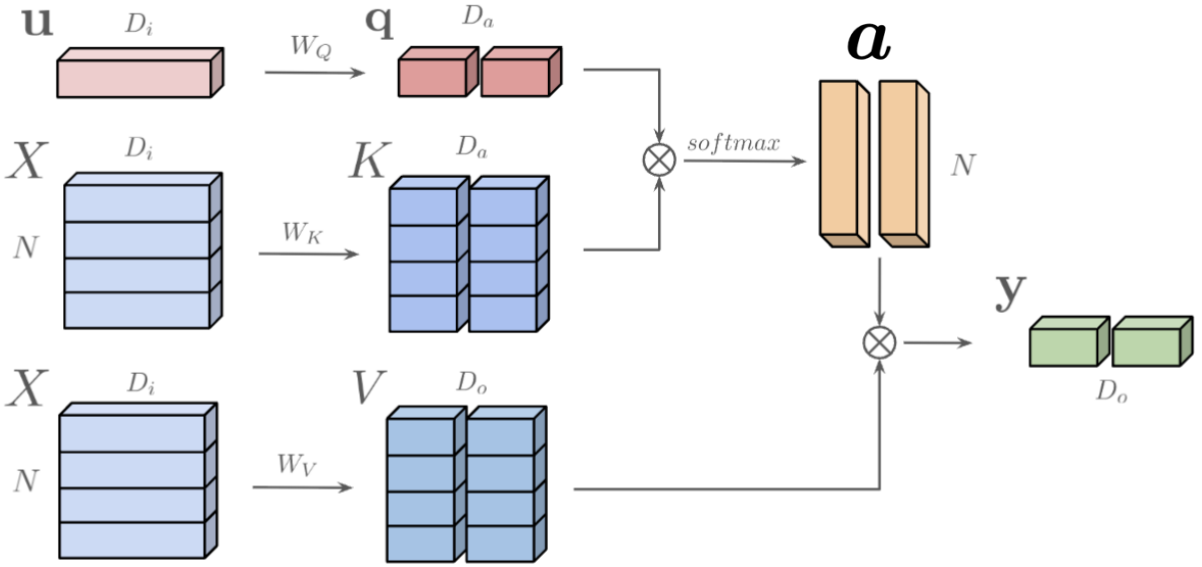
- Input vector $\mathtt{u}$, input features X, linear projections W_Q, W_K, and W_V, *multiple heads.*



- Given input features X, the goal is to generate an output image-level feature y.

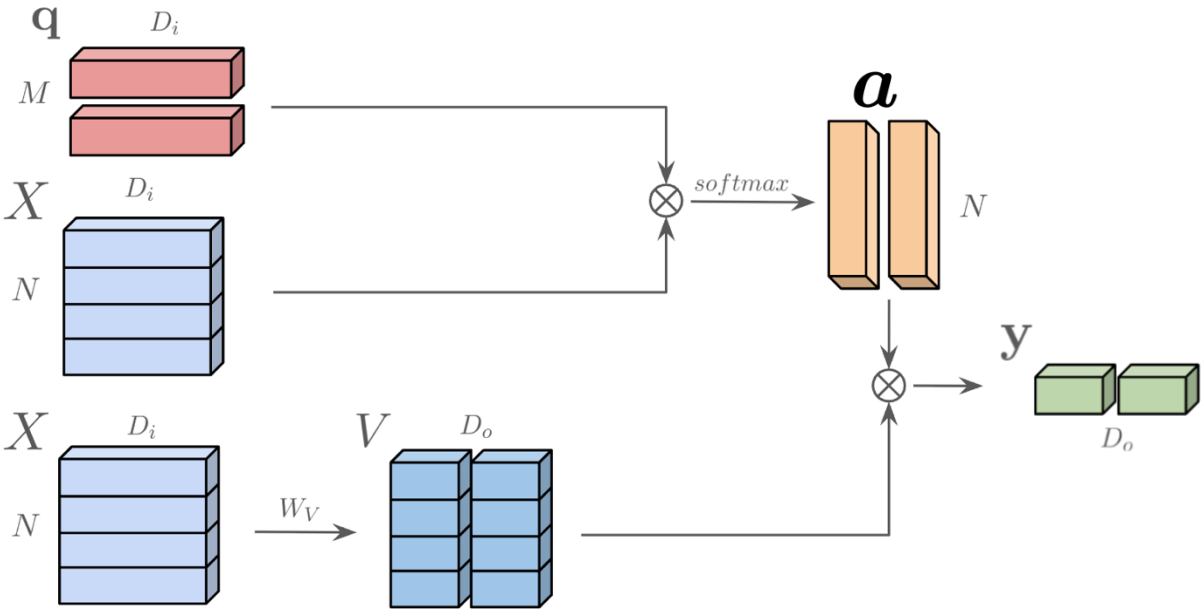
Vanilla Multi-Head Cross Attention (MHCA)

Our question: Can we simplify this architecture?



EP: Efficient Probing (removing W_Q, replacing W_K)

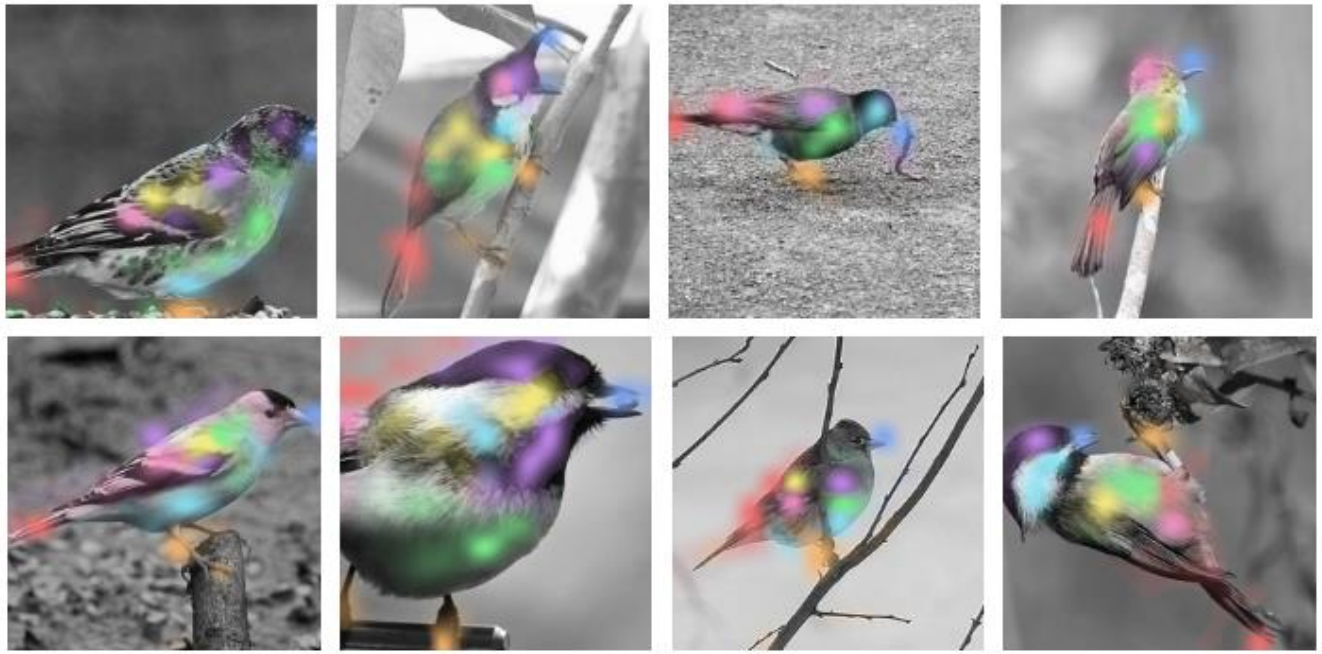
- Learn M query vectors in the full-dimensional space D_i directly.



- W_V lets D_o to be a smaller dimensional space than D_i, *e.g.*, $D_o = D_i/8$.

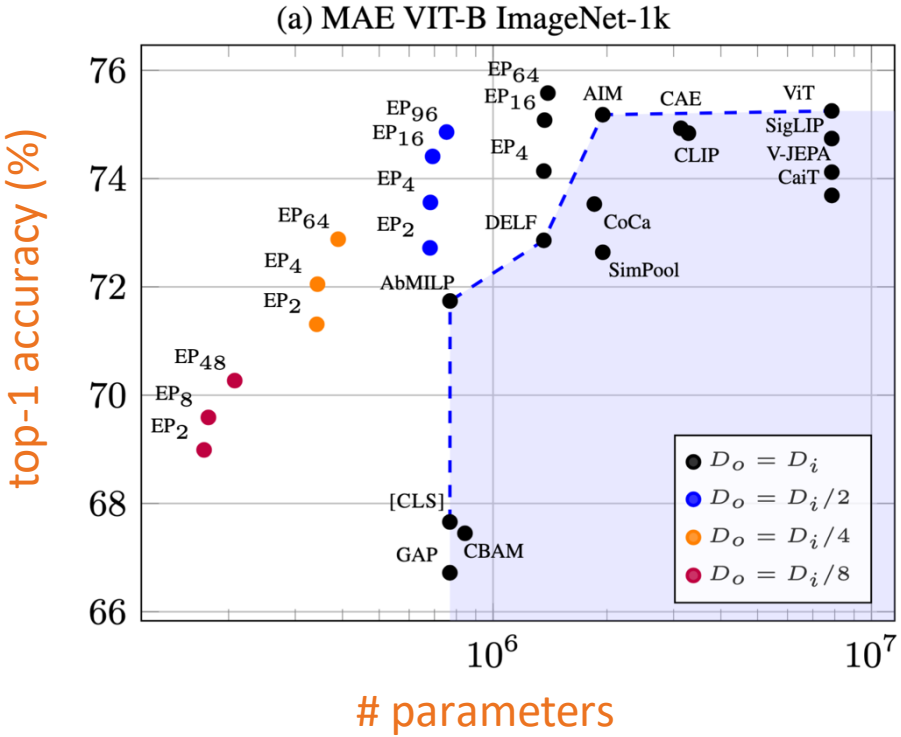
Attention maps of EP with 8 queries (EP_8)

- Each query captures complementary regions.
- Semantic correspondences emerge (e.g. tails, beaks, feet).



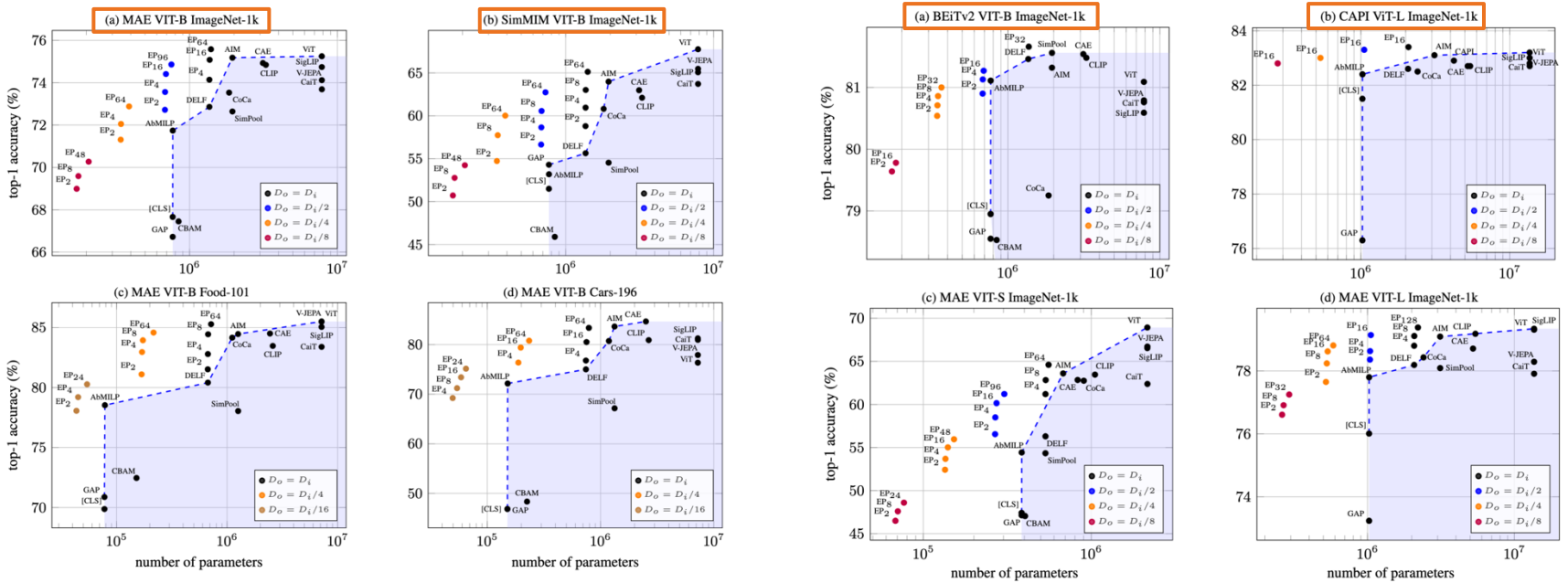
Accuracy vs. # parameters

- Recall: W_V lets D_o to be a smaller dimensional space than D_i, *e.g.*, $D_o = D_i/8$.



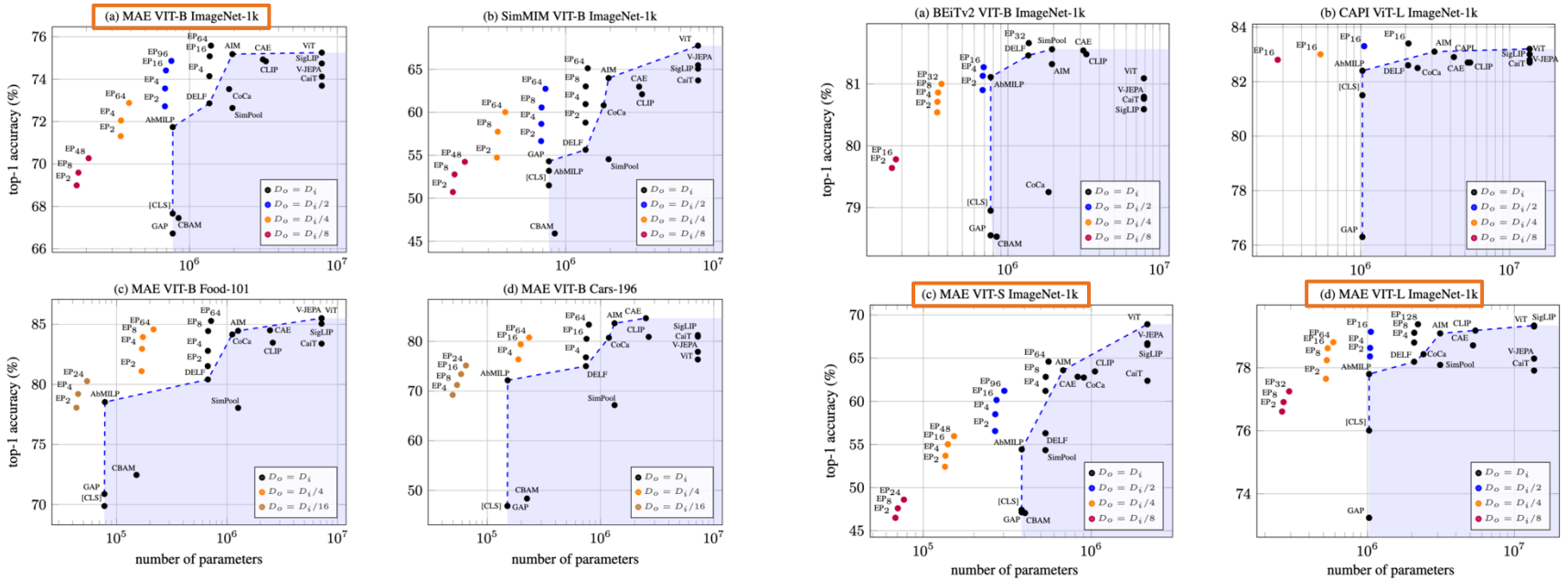
Accuracy vs. # parameters

- Across pre-training methods.



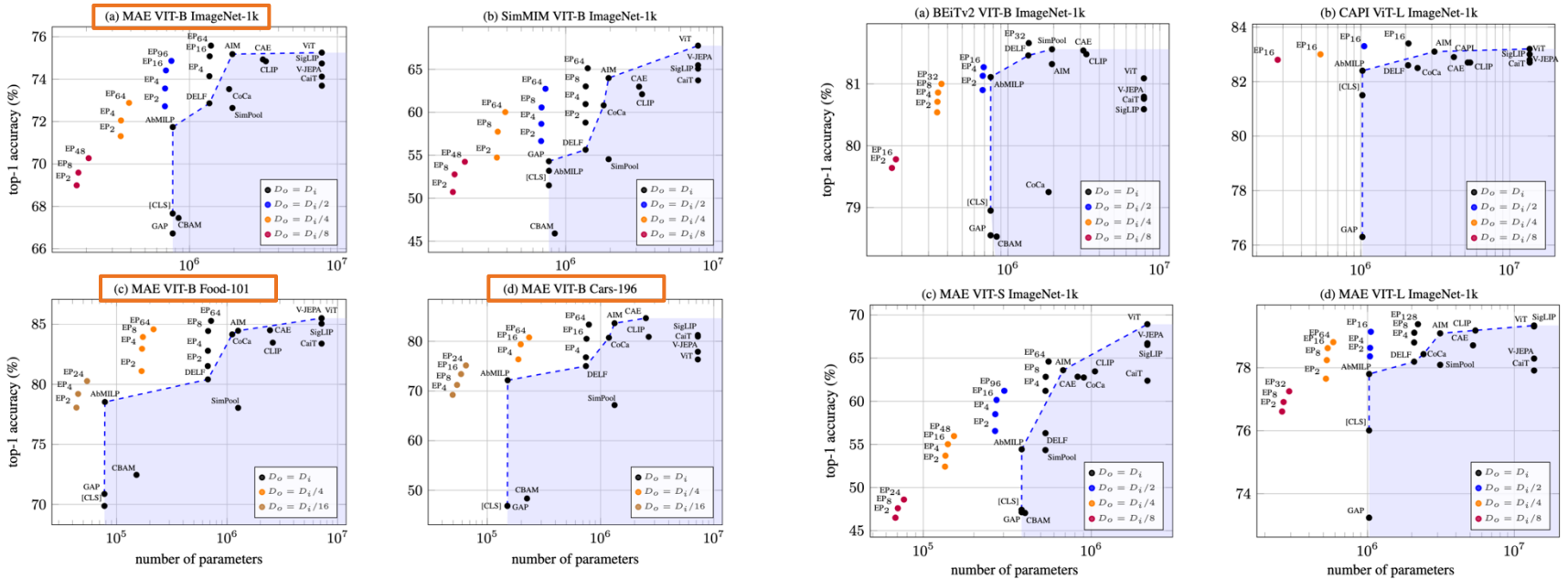
Accuracy vs. # parameters

- Across backbones of varying size.



Accuracy vs. # parameters

- Across datasets.



Comparison of pre-training paradigms

- In terms of different evaluation protocols.

	Method	Arch	Pre-training	k-NN	LP	# Par.	EP	# Par.	Gain	FT†	# Par.
MIM	MAE (He et al., 2022)	ViT-S/16	IN-1K	26.7	47.4	0.4M	64.6	0.5M	+17.2	80.6	22M
		ViT-B/16		46.1	67.7	0.8M	75.6	1.4M	+7.9	83.6	87M
		ViT-L/16		58.2	76.0	1.0M	79.3	2.1M	+3.3	85.9	304M
	BEiTv2 (Peng et al., 2022)	ViT-B/16	IN-1K	74.8	79.0	0.8M	81.7	1.4M	+2.7	85.0	87M
	SimMIM (Xie et al., 2022)	ViT-B/16	IN-1K	15.1	51.5	0.8M	65.1	1.4M	+13.6	83.8	87M
	CAPI (Darcet et al., 2025)	ViT-L/14	IN-1K	76.7	81.5	1.0M	83.6	2.1M	+2.1	×	304M
JEA	BYOL (Grill et al., 2020)	RN-50	IN-1K	64.8	74.3	2.0M	75.1	6.3M	+0.8	77.7	26M
	DINO (Caron et al., 2021)	ViT-B/16	IN-1K	76.1	77.3	0.8M	77.8	1.4M	+0.5	82.8	87M
HYBRID	iBOT (Zhou et al., 2021)	ViT-B/16	IN-1K	77.0	78.7	0.8M	79.2	1.4M	+0.5	84.0	87M
	DINOv2 (Oquab et al., 2024)	ViT-B/14	LVD-142M	81.8	83.2	0.8M	84.0	1.4M	+0.8	×	87M
		ViT-L/14		83.5	85.2	1.0M	85.6	2.1M	+0.4	×	304M
	Franca (Venkataramanan et al., 2025)	ViT-L/14	IN-21k	82.2	83.8	1.0M	84.3	2.1M	+0.5	×	304M
	DINOv3 (Siméoni et al., 2025)	ViT-B/16	LVD-1689M	83.0	84.0	0.8M	84.4	1.4M	+0.4	×	87M
		ViT-L/16		85.3	86.6	1.0M	87.1	2.1M	+0.5	×	304M
VLM	CLIP (Radford et al., 2021)	ViT-L/14	WIT	77.2	82.3	0.8M	83.4	2.1M	+1.1	×	305M
	SigLIP (Zhai et al., 2023)	ViT-L/16	WebLI	83.7	84.1‡	1.0M	86.1	2.1M	+2.0	×	305M
	SigLIP2 (Tschannen et al., 2025)	ViT-L/16	WebLI	84.4	85.2‡	1.0M	87.0	2.1M	+1.8	×	305M
GEN	DiT (Peebles & Xie, 2023)	DiT-XL/2	IN-1K	8.3	32.7‡	1.2M	57.0	2.5M	+24.3	×	676M
	AIMv2 (Fini et al., 2025)	ViT-L/14	custom*	80.8	84.8‡	1.0M	85.9	2.1M	+1.1	×	304M

Note. custom*: DFN-2B (Fang et al., 2023), COYO (Byeon et al., 2022), HQITP (Fini et al., 2025). Default EP = EP_{32}.

Comparison of pre-training paradigms

- In terms of different evaluation protocols.

	Method	Arch	Pre-training	k-NN	LP	# Par.	EP	# Par.	Gain	FT†	# Par.
MIM	MAE (He et al., 2022)	ViT-S/16	IN-1K	26.7	47.4	0.4M	64.6	0.5M	+17.2	80.6	22M
		ViT-B/16		46.1	67.7	0.8M	75.6	1.4M	+7.9	83.6	87M
		ViT-L/16		58.2	76.0	1.0M	79.3	2.1M	+3.3	85.9	304M
	BEiTv2 (Peng et al., 2022)	ViT-B/16	IN-1K	74.8	79.0	0.8M	81.7	1.4M	+2.7	85.0	87M
	SimMIM (Xie et al., 2022)	ViT-B/16	IN-1K	15.1	51.5	0.8M	65.1	1.4M	+13.6	83.8	87M
	CAPI (Darcet et al., 2025)	ViT-L/14	IN-1K	76.7	81.5	1.0M	83.6	2.1M	+2.1	×	304M
JEA	BYOL (Grill et al., 2020)	RN-50	IN-1K	64.8	74.3	2.0M	75.1	6.3M	+0.8	77.7	26M
	DINO (Caron et al., 2021)	ViT-B/16	IN-1K	76.1	77.3	0.8M	77.8	1.4M	+0.5	82.8	87M
HYBRID	iBOT (Zhou et al., 2021)	ViT-B/16	IN-1K	77.0	78.7	0.8M	79.2	1.4M	+0.5	84.0	87M
	DINOv2 (Oquab et al., 2024)	ViT-B/14	LVD-142M	81.8	83.2	0.8M	84.0	1.4M	+0.8	×	87M
		ViT-L/14		83.5	85.2	1.0M	85.6	2.1M	+0.4	×	304M
	Franca (Venkataramanan et al., 2025)	ViT-L/14	IN-21k	82.2	83.8	1.0M	84.3	2.1M	+0.5	×	304M
	DINOv3 (Siméoni et al., 2025)	ViT-B/16	LVD-1689M	83.0	84.0	0.8M	84.4	1.4M	+0.4	×	87M
		ViT-L/16		85.3	86.6	1.0M	87.1	2.1M	+0.5	×	304M
VLM	CLIP (Radford et al., 2021)	ViT-L/14	WIT	77.2	82.3	0.8M	83.4	2.1M	+1.1	×	305M
	SigLIP (Zhai et al., 2023)	ViT-L/16	WebLI	83.7	84.1‡	1.0M	86.1	2.1M	+2.0	×	305M
	SigLIP2 (Tschannen et al., 2025)	ViT-L/16	WebLI	84.4	85.2‡	1.0M	87.0	2.1M	+1.8	×	305M
GEN	DiT (Peebles & Xie, 2023)	DiT-XL/2	IN-1K	8.3	32.7‡	1.2M	57.0	2.5M	+24.3	×	676M
	AIMv2 (Fini et al., 2025)	ViT-L/14	custom*	80.8	84.8‡	1.0M	85.9	2.1M	+1.1	×	304M

Note. custom*: DFN-2B (Fang et al., 2023), COYO (Byeon et al., 2022), HQITP (Fini et al., 2025). Default EP = EP_{32}.

- largest gains.

- largest gains.

Comparison of pre-training paradigms

- In terms of different evaluation protocols.

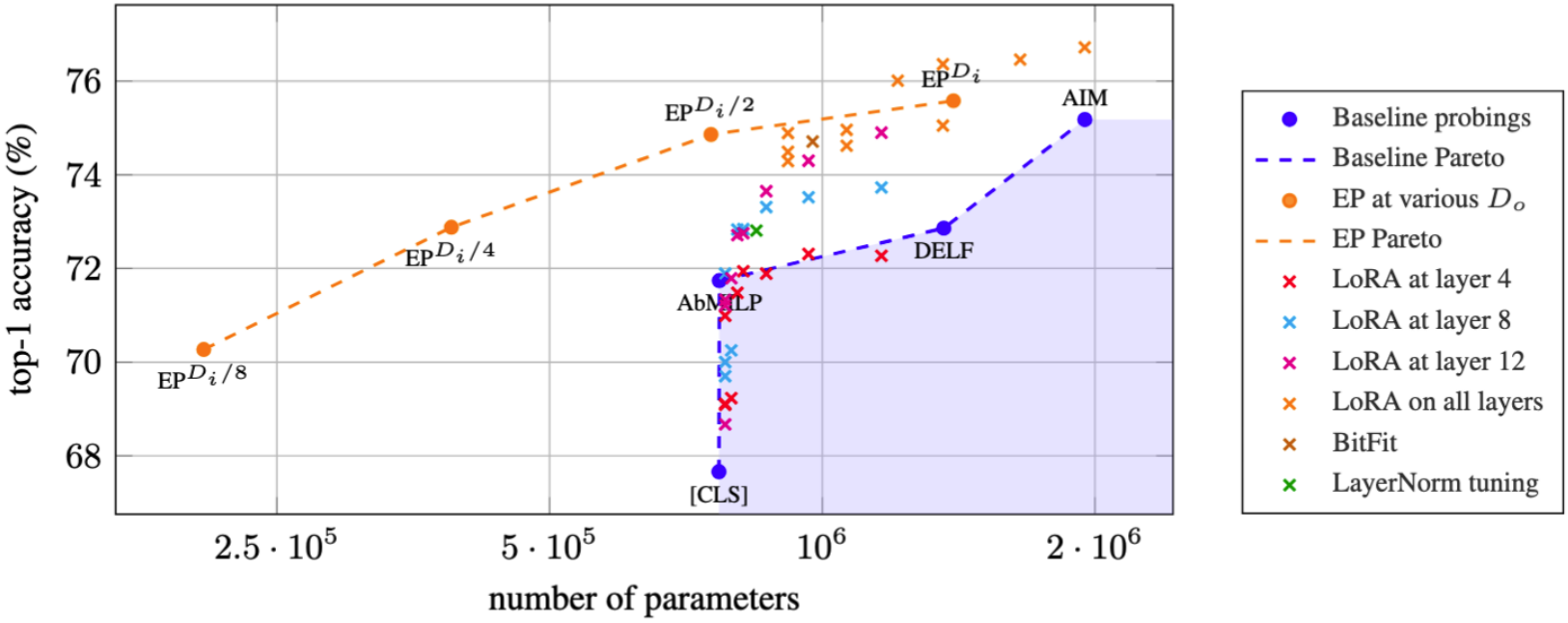
	Method	Arch	Pre-training	k-NN	LP	# Par.	EP	# Par.	Gain	FT†	# Par.
MIM	MAE (He et al., 2022)	ViT-S/16	IN-1K	26.7	47.4	0.4M	64.6	0.5M	+17.2	80.6	22M
		ViT-B/16		46.1	67.7	0.8M	75.6	1.4M	+7.9	83.6	87M
		ViT-L/16		58.2	76.0	1.0M	79.3	2.1M	+3.3	85.9	304M
	BEiTv2 (Peng et al., 2022)	ViT-B/16	IN-1K	74.8	79.0	0.8M	81.7	1.4M	+2.7	85.0	87M
	SimMIM (Xie et al., 2022)	ViT-B/16	IN-1K	15.1	51.5	0.8M	65.1	1.4M	+13.6	83.8	87M
	CAPI (Darcet et al., 2025)	ViT-L/14	IN-1K	76.7	81.5	1.0M	83.6	2.1M	+2.1	×	304M
JEA	BYOL (Grill et al., 2020)	RN-50	IN-1K	64.8	74.3	2.0M	75.1	6.3M	+0.8	77.7	26M
	DINO (Caron et al., 2021)	ViT-B/16	IN-1K	76.1	77.3	0.8M	77.8	1.4M	+0.5	82.8	87M
HYBRID	iBOT (Zhou et al., 2021)	ViT-B/16	IN-1K	77.0	78.7	0.8M	79.2	1.4M	+0.5	84.0	87M
	DINOv2 (Oquab et al., 2024)	ViT-B/14	LVD-142M	81.8	83.2	0.8M	84.0	1.4M	+0.8	×	87M
		ViT-L/14		83.5	85.2	1.0M	85.6	2.1M	+0.4	×	304M
	Franca (Venkataramanan et al., 2025)	ViT-L/14	IN-21k	82.2	83.8	1.0M	84.3	2.1M	+0.5	×	304M
	DINOv3 (Siméoni et al., 2025)	ViT-B/16	LVD-1689M	83.0	84.0	0.8M	84.4	1.4M	+0.4	×	87M
		ViT-L/16		85.3	86.6	1.0M	87.1	2.1M	+0.5	×	304M
VLM	CLIP (Radford et al., 2021)	ViT-L/14	WIT	77.2	82.3	0.8M	83.4	2.1M	+1.1	×	305M
	SigLIP (Zhai et al., 2023)	ViT-L/16	WebLI	83.7	84.1‡	1.0M	86.1	2.1M	+2.0	×	305M
	SigLIP2 (Tschannen et al., 2025)	ViT-L/16	WebLI	84.4	85.2‡	1.0M	87.0	2.1M	+1.8	×	305M
GEN	DiT (Peebles & Xie, 2023)	DiT-XL/2	IN-1K	8.3	32.7‡	1.2M	57.0	2.5M	+24.3	×	676M
	AIMv2 (Fini et al., 2025)	ViT-L/14	custom*	80.8	84.8‡	1.0M	85.9	2.1M	+1.1	×	304M

Note. custom*: DFN-2B (Fang et al., 2023), COYO (Byeon et al., 2022), HQITP (Fini et al., 2025). Default EP = EP_{32}.

- moderate gains.

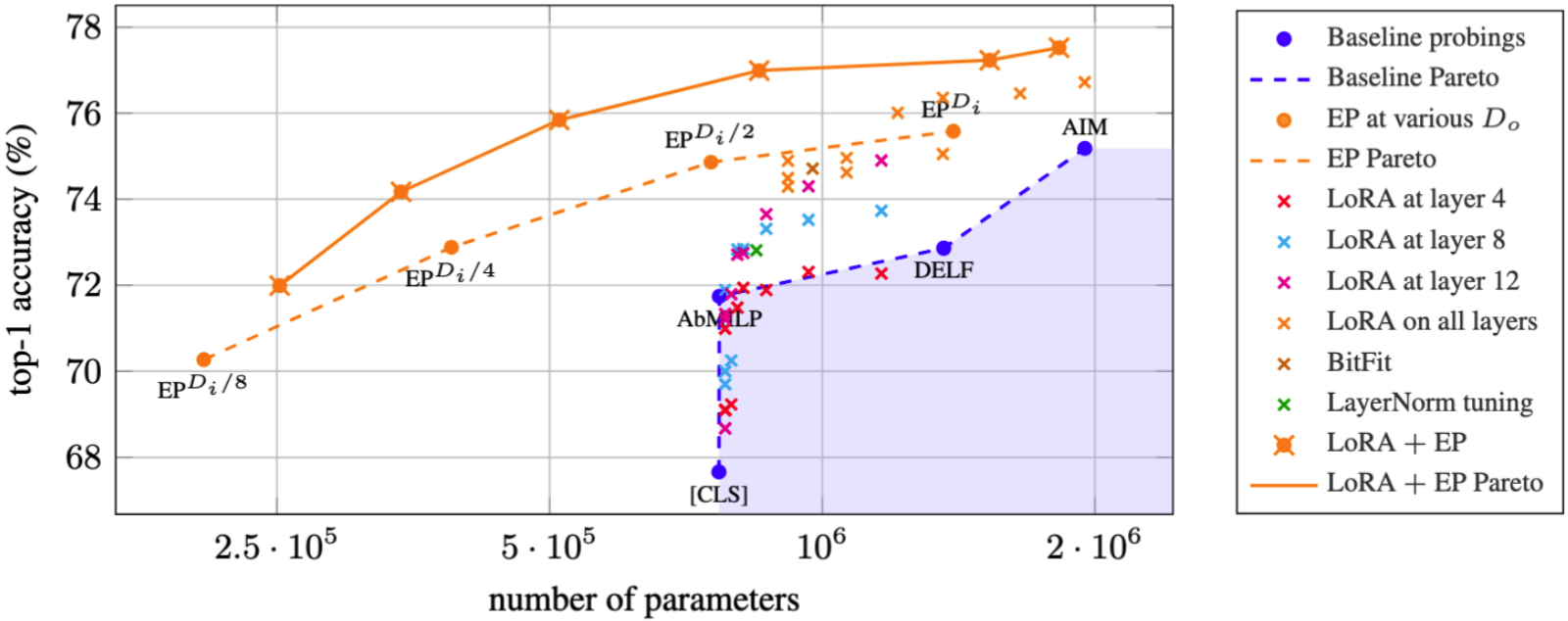
Comparison against PEFT methods

- Common PEFT methods: LoRA, BitFit, Layernorm tuning.
- EP >> LoRA on a single layer, BitFit, and Layernorm tuning.
- EP on par with LoRA on all layers.



But can they be complementary?

- Combine the most parameter-efficient LoRA with EP for different D_o.
- Resulting LoRA + EP: new dominant region in accuracy-parameter plane.
- Improving over both pure LoRA and pure EP.



Take-home messages

- Efficient Probing (EP):
  - Is plug and play.
  - Is compatible with all pre-training paradigms.
  - Unlocks the potential of families optimizing local representations.
  - Is complementary with PEFT methods.