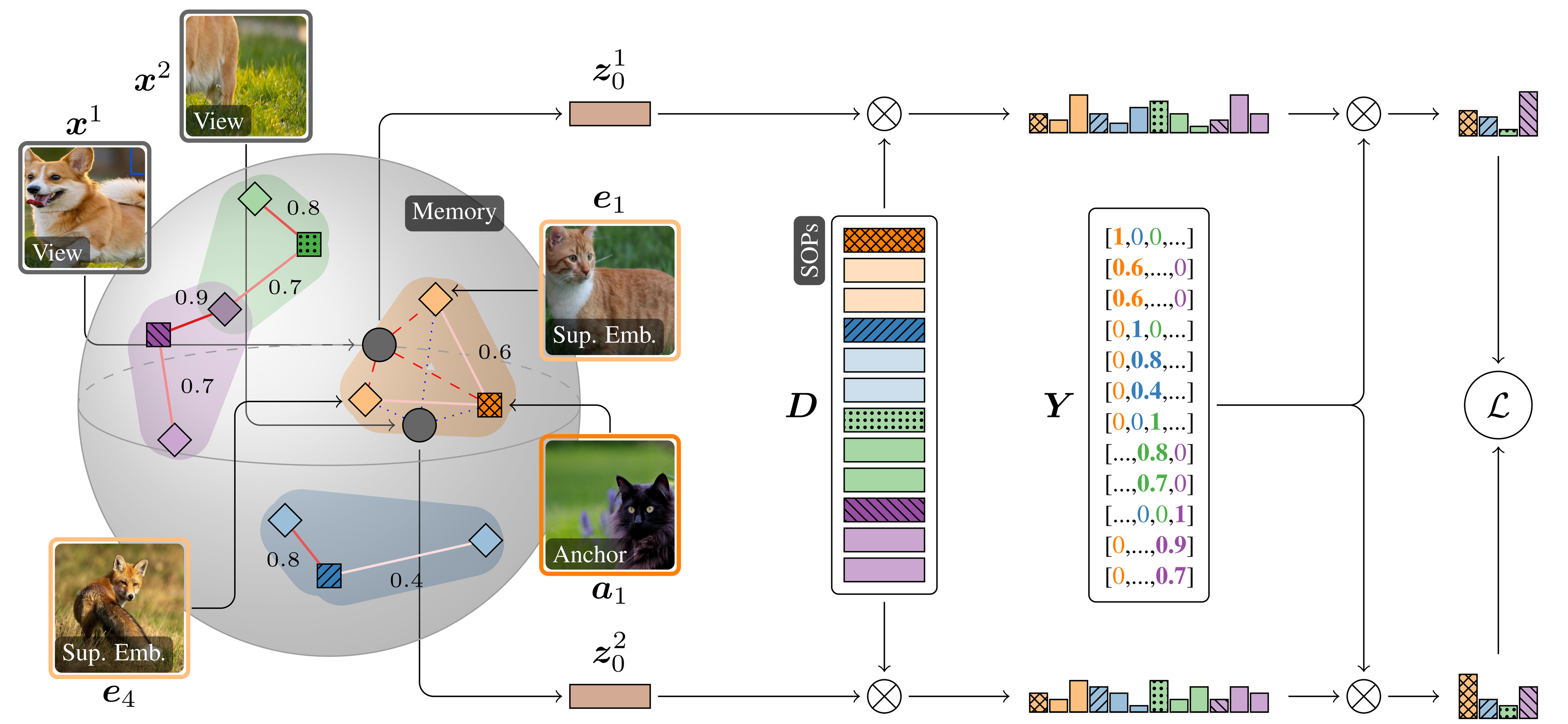
Self-Organizing Visual Prototypes for Non-Parametric Representation Learning

Thalles Silva ❁ Helio Pedrini ❁ Adín Ramírez Rivera ❃
❁ University of Campinas ❃ University of Oslo

METHODOLOGY



WHY REPLACE CLASSIC PROTOTYPES IN SSL?

Pain-point in Prototypical SSL		What Self-Organising Prototypes give instead
✗ Single vector must encode *all* features of a region in space (cluster).	$\Longrightarrow$	✓ Each prototype = anchor + multiple support embeddings
✗ Needs *over-clustering* $K \gg C$.	$\Longrightarrow$	✓ No extra clusters: random anchors already cover the space
✗ Parametric centroids drift or die → centering, sharpening, Sinkhorn,...	$\Longrightarrow$	✓ Purely non-parametric supports no drift, no collapse, no add-on tricks

SOP = richer & adaptive prototypes, space-filling without the engineering overhead.

SOP: HOW A PROTOTYPE SELF-ASSEMBLES (every iteration)

1. Sample Anchors:
Randomly select K anchor embeddings $\{a_i\}_{i=1}^K$ from memory E.
2. Build Self-Organizing Prototypes (SOPs):
for each anchor a_i **do**
  Find k nearest neighbors $\mathcal{N}_k(a_i)$ in E.
  Define SOP: $\mathcal{S}_i = \{a_i\} \cup \mathcal{N}_k(a_i)$.
end for
3. Compute Support Contributions:
for each SOP $\mathcal{S}_i$ **do**
  for each support $e_j \in \mathcal{S}_i$ **do**
    Compute similarity $\langle e_j, a_i \rangle$ as contribution score $Y_{j,i}$.
  end for
end for
4. Assign Views to SOPs:
For each view u, compute similarity $P(u)$ to all SOPs.
5. Optimize:
Use cross-entropy loss between view assignments.

What this buys us

- SOPs are *data-centred*, rebuilt every step ⇒ no drift.
- 1 SOP ⇒ Multiple supports ⇒ richer regional description.
- New anchors every iteration ⇒ full coverage, no over-clustering
- One mechanism, two scales (global & local)

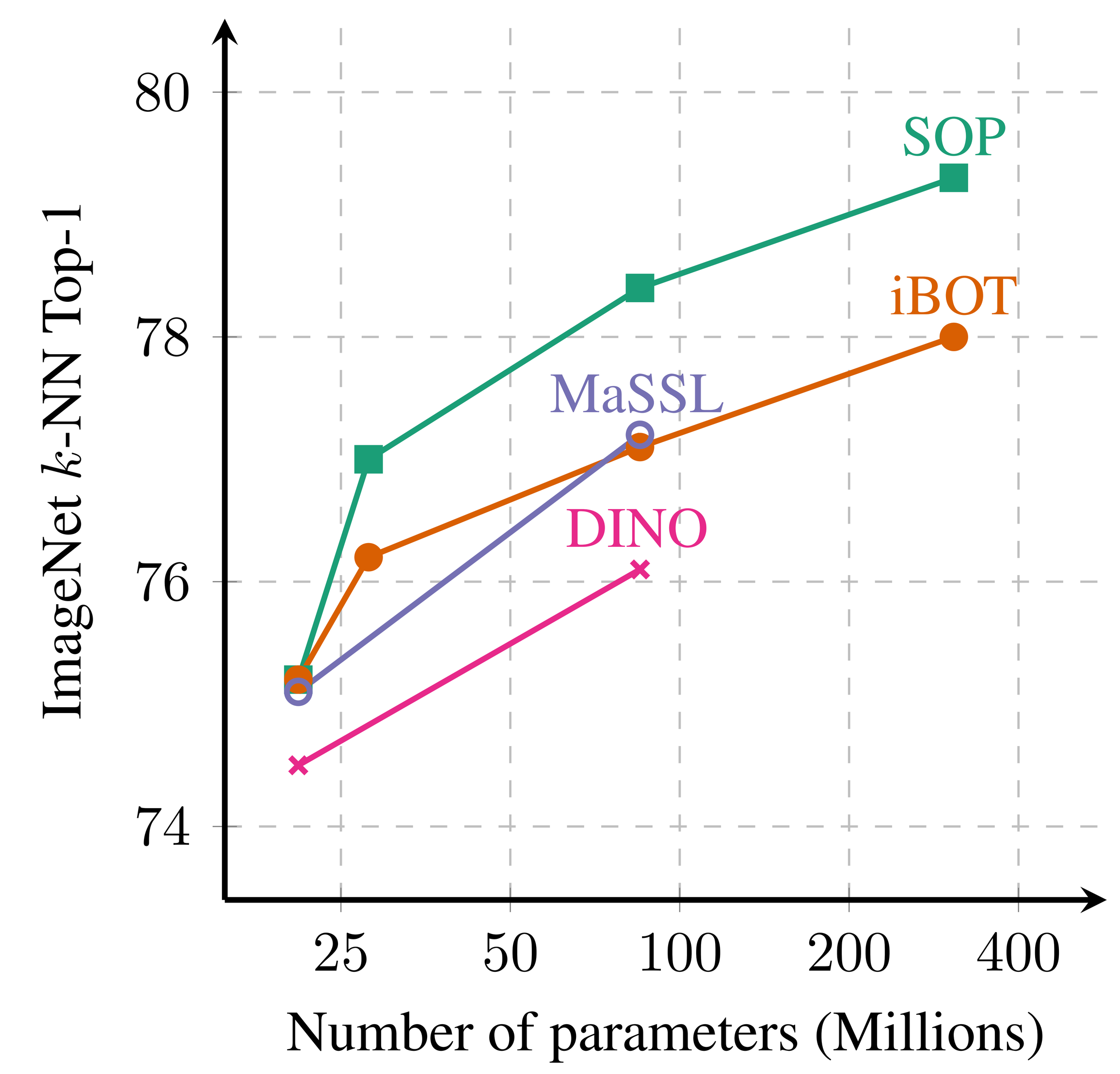
View's embeddings
Self-Organizing Prototypes

$$P(u) = \mathrm{softmax}(u\, D^\top)\, Y$$

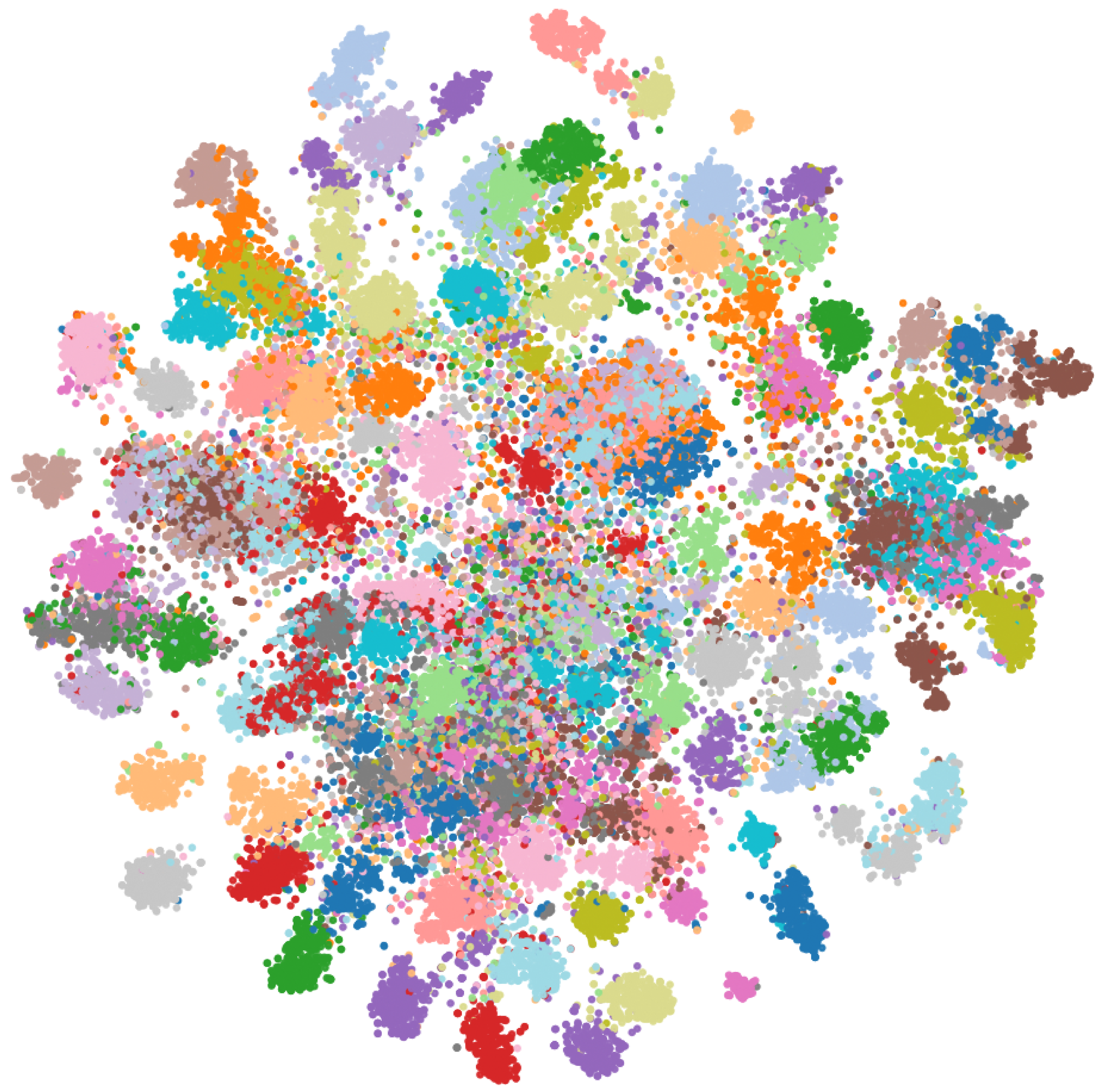
Soft-contributions

$$L = -\sum_x P(z_0^1)^\top \log P(z_0^2)$$

k-NN Downstream Performance

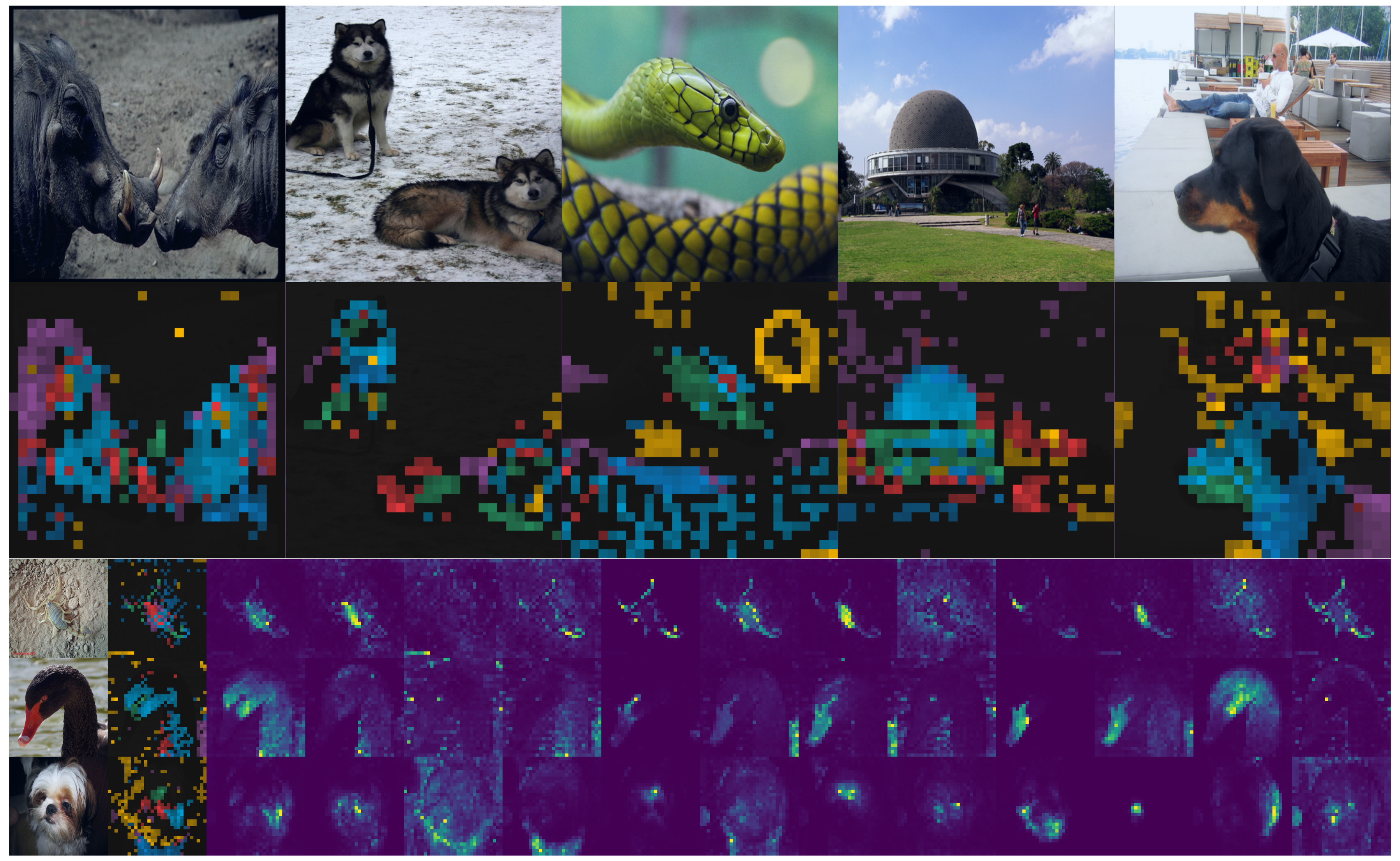


Learned Embedding Space - CIFAR-100



Forty-Second International Conference on Machine Learning (ICML)
Acknowledgements: Sigma2 NN8104K, RCN 309439, & CAPES 001.

Self-Attention Maps



Transfer learning (k-NN) / Video obj. segmen. (DAVIS 2017)

Method	Arch	Ep.	k-NN	Lin.	1%	10%	100%
EsViT	Swin-T/14	300	77.0	78.7			
iBOT	Swin-T/14	300	76.2	79.3			
SOP	Swin-T/14	300	**77.2**	**79.4**			
iBOT	ViT-S/16	800	75.2	77.9	61.9	75.1	82.3
MaSSL	ViT-S/16	800	75.1	77.8			
SOP	ViT-S/16	800	**75.3**	77.9	**62.1**	75.1	82.3
iBOT	ViT-B/16	400	77.1	79.5	68.45	78.1	84.0
MaSSL	ViT-B/16	400	77.2	79.6			
SOP	ViT-B/16	400	**78.21**	**79.9**	**69.53**	**78.4**	**84.2**
iBOT	ViT-L/16	250	78.0	81.0			84.8
I-JEPA	ViT-H/14	300		79.3			
SOP	ViT-L/16	250	**79.2**	**81.2**			**84.9**

Method	Data	Arch.	$(\mathcal{J}\&\mathcal{F})_m$	$\mathcal{J}_m$	$\mathcal{F}_m$
Sup.					
IN-1K	IN-1K	ViT-S/8	66.0	63.9	68.1
STM	I/D/Y	RN50	81.8	79.2	84.3
Self-Sup.					
CT	VLOG	RN50	48.7	46.4	50.0
MAST	YT-VOS	RN18	65.5	63.3	67.6
STC	Kinetics	RN18	67.6	64.8	70.2
DINO	IN-1K	ViT-S/16	61.8	60.2	63.4
	IN-1K	ViT-B/16	62.3	60.7	63.9
iBOT	IN-1K	ViT-S/16	61.8	60.4	63.2
	IN-1K	ViT-B/16	62.7	61.7	63.7
SOP	IN-1K	ViT-B/16	**63.3**	**61.7**	**65.0**

Image retrieval / Robustness to background changes

			$\mathcal{R}$Ox		$\mathcal{R}$Par	
Method	Arch.	Epo.	M	H	M	H
Sup.	RN101	100	49.8	18.5	74.0	52.1
DINO	ViT-B/16	400	37.4	13.7	63.5	35.6
iBOT	ViT-B/16	400	36.8	14.3	64.1	36.6
MaSSL	ViT-B/16	400	39.3	14.1	65.8	38.1
SOP	ViT-B/16	400	**42.7**	**17.5**	**67.3**	**41.3**

	Background Changes							Clean
	OF	MS	MR	MN	NF	OBB	OBT	IN-9
iBOT	91.9	89.7	81.9	79.7	54.7	17.6	20.4	96.8
MaSSL	91.0	90.2	83.0	80.4	53.4	15.8	23.7	97.6
SOP	**93.3**	**91.4**	**85.6**	**83.1**	**55.8**	**19.9**	22.8	97.1