Effective Self-supervised Pre-training for Low-compute Networks without Distillation

Fuwen Tan[1], Fatemeh Saleh[2], Brais Martinez[1]

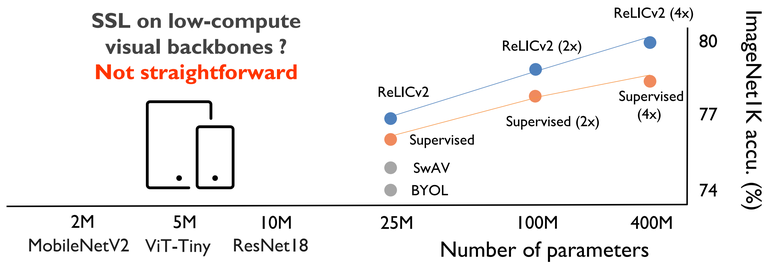
[1]Samsung AI Center, Cambridge, [2]Microsoft Research, Cambridge,

SAIC-Cambridge Microsoft

ICLR 2023

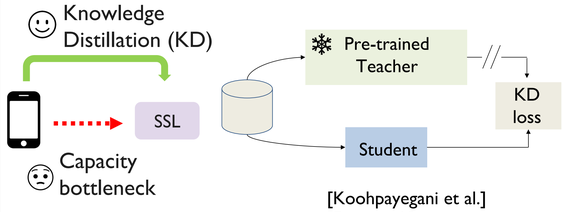
The Power of Self-supervised Learning (SSL)

SSL on low-compute visual backbones ?
Not straightforward



Previous Research

KD w different losses



[Koohpayegani et al.]
[Fang et al.]
[Xu et al.]
[Gao et al.]

Pros
- Re-use teachers
- Easier to optimize

Cons
- New data
  - Teacher pre-training
- New on-device tasks
  - Continual Learning
  - Federated Learning

Contributions

- Revisit SSL in low-compute pre-training, showing that lightweight networks can learn high-quality visual representations using self-supervised signals alone, without knowledge distillation.
- Demonstrate that SSL low-compute pre-training benefits from a weaker self-supervised target aligning views in similar spatial scales and contexts.
- Training recipes enhance various SSL methods (e.g., MoCo-v2, SwAV, DINO) across low-size networks, including CNNs (e.g., MobileNetV2, ResNet18, ResNet34) and ViT-Ti, outperforming state-of-the-art distillation-based approaches.

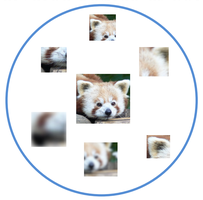
Diagnosing SSL for Lightweight Networks

SSL

Previous consensus:
Capacity bottleneck

Our finding:
Model complexity vs regularization strength

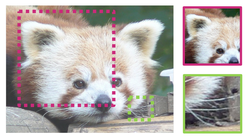
Aligning views of diverse scales & contexts



Large networks ☺

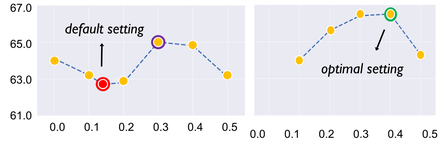
Small networks ☹

Views with different "crop scales"



Global view B: $S_g \sim 1.0$ *of the image area*

Local view B: $0.05 \sim S_l$ *of the image area*

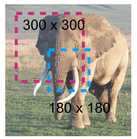


S_g, with S_l set to 0.14 (*default value*)

S_l, with S_g set to 0.3 (*optimal value*)

Views with different "pixel scales"

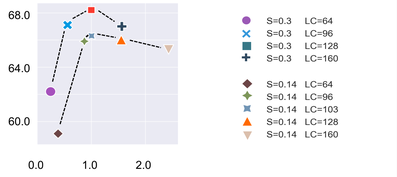
$$Pixel\ scale = \frac{target\ res.}{source\ res.}$$



$$\frac{224 \times 224}{300 \times 300} \approx 0.56$$

$$\frac{96 \times 96}{180 \times 180} \approx 0.28$$

The best performance achieves when the views have similar pixel scales (i.e. $mean(PS_l/PS_g) \approx 1.0$).



$mean(PS_l/PS_g)$
PS_l, PS_g: pixel scale of global & local views

$S = S_g = S_l$
LC: target res. of the local views

Rebalance the global and local losses

L_l: *loss between a pair of global-local views*

L_g: *loss between two global views*

P_{gg}: # of *global-global view pairs*

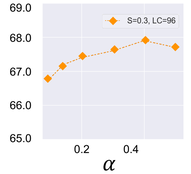
P_{gl}: # of *global-local view pairs*

Default formulation

$$L = \frac{L_g + L_l}{P_{gg} + P_{gl}}$$

Our formulation

$$L = \alpha \cdot \frac{L_g}{P_{gg}} + (1 - \alpha) \cdot \frac{L_l}{P_{gl}}$$



Improving Representative SSL approaches

Representative SSL w. MobileNetV2 [Sandler et al.] as the backbone		Linear evaluation on ImageNet-1K Top-1 (%)	Top-5 (%)
MoCo-v2 [Chen et al.] w. local views	Baseline	60.6	83.3
	Ours	61.6 (+1.0)	84.2 (+0.9)
SwAV [Caron et al.]	Baseline	65.2	85.6
	Ours	67.3 (+2.1)	87.2 (+1.6)
DINO [Caron et al.]	Baseline	66.2	86.4
	Ours	68.3 (+2.1)	87.8 (+1.4)
Supervised		71.9	90.3

Improving Representative Visual Backbones

The improvement is larger for smaller networks

Improving Downstream Applications

Backbone	Method	Mask R-CNN FPN 1x on COCO Object Det.	Instance Seg.	Semi-supervised Learning on ImageNet-1K 1% label	10% label
MobileNetV2	Supervised	33.1	29.8	-	-
	DINO baseline	30.9	28.1	47.9	61.3
	DINO + Ours	32.1 (+1.2)	29.1 (+1.0)	50.6 (+2.7)	63.5 (+2.2)
ResNet18	Supervised	34.5	31.6	-	-
	DINO baseline	32.7	30.6	44.5	59.2
	DINO + Ours	34.1 (+1.4)	31.8 (+1.2)	49.8 (+5.3)	63.0 (+3.8)
ResNet34	Supervised	38.7	35.0	-	-
	DINO baseline	37.6	34.6	52.4	65.4
	DINO + Ours	38.6 (+1.0)	35.5 (+0.9)	55.2 (+2.8)	67.2 (+1.8)

Comparing to SOTA, all with KD

Method	Linear evaluation on ImageNet-1K MobileNetV2 Top-1	Top-5	ResNet18 Top-1	Top-5	ResNet34 Top-1	Top-5
Supervised	71.9	90.3	69.8	89.1	73.3	91.4
CompRess	65.8	-	62.6	-	-	-
SimReg	**69.1**	-	65.1	-	-	-
SEED	-	-	63.0	84.9	65.7	86.8
DisCo	-	-	65.2	86.8	67.6	88.6
BINGO	-	-	65.5	87.0	68.9	89.0
Ours	68.8	87.8	**66.8**	**87.3**	**70.8**	**90.0**