Self-supervised learning for computer vision: SimCLR, MoCo, SwAV & Co.

Michael M. Pieler
ML Engineer
contextflow.com

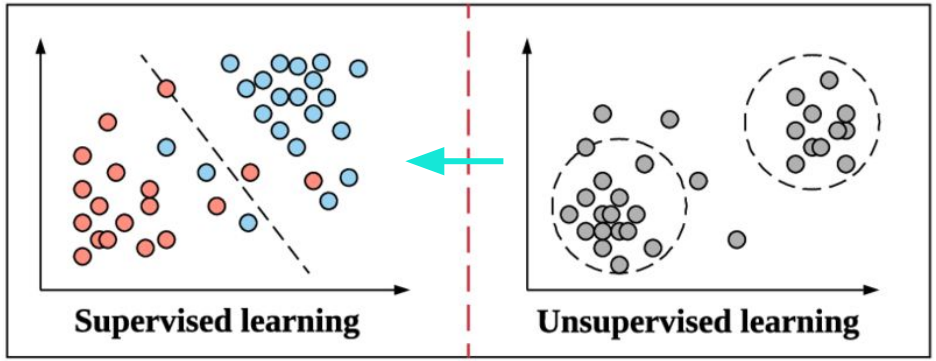
Supervised vs. self-supervised learning?

Supervised learning:

“Traditional setup”: we have input data and the associated labels.

Self-supervised learning:

We use unlabeled data with special learning objectives.



Self-Supervised Representation Learning

Image: Examples of Supervised Learning (Linear Regression) and Unsupervised... | Download Scientific Diagram

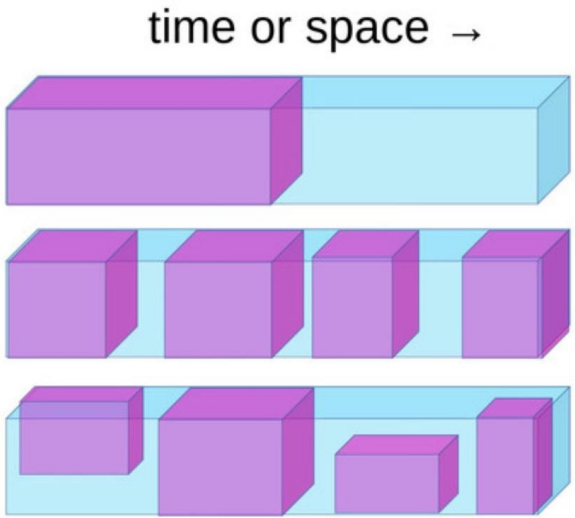
“How much information for learning” theory?

- **“Pure” Reinforcement Learning (cherry)**
  - The machine predicts a scalar reward given once in a while.
  - **A few bits for some samples**
- **Supervised Learning (icing)**
  - The machine predicts a category or a few numbers for each input
  - Predicting human-supplied data
  - **10→10,000 bits per sample**
- **Self-Supervised Learning (cake génoise)**
  - The machine predicts any part of its input for any observed part.
  - Predicts future frames in videos
  - **Millions of bits per sample**

Slide from Yann LeCun from https://www.gwern.net/newsletter/2020/05#baking-the-cake

Traditional self-supervised learning setup

- Predict any part of the input from any other part.
- Predict the future from the past.
- Predict the invisible from the visible.
- Predict any occluded, masked, or corrupted part from all available parts.



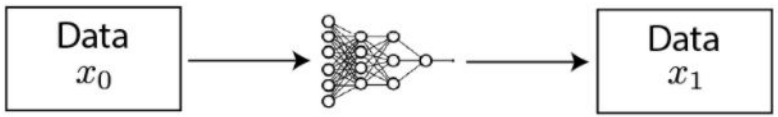
- Pretend there is a part of the input you don't know and predict that.
- Reconstruction = SSL when any part could be known or unknown

There are a lot of different approaches. We will focus on: SimCLR, MoCo and SwAV

Slide from Y. LeCun ICLR 2020; detailed overview: Self-Supervised Representation Learning

Generative & contrastive self-supervised learning

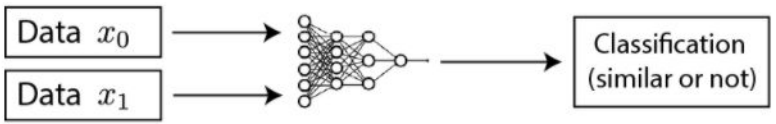
Generative / Predictive



Loss measured in the output space

Examples: Colorization, Auto-Encoders

Contrastive

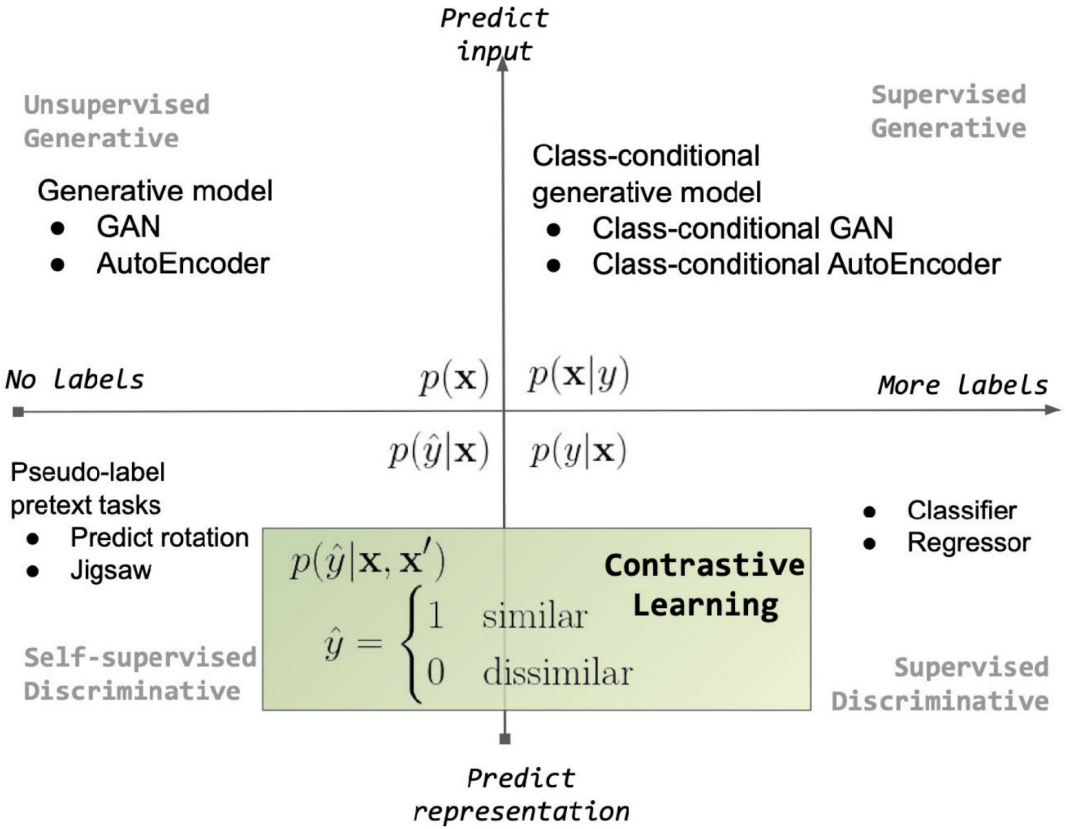


Loss measured in the representation space

Examples: TCN, CPC, Deep-InfoMax

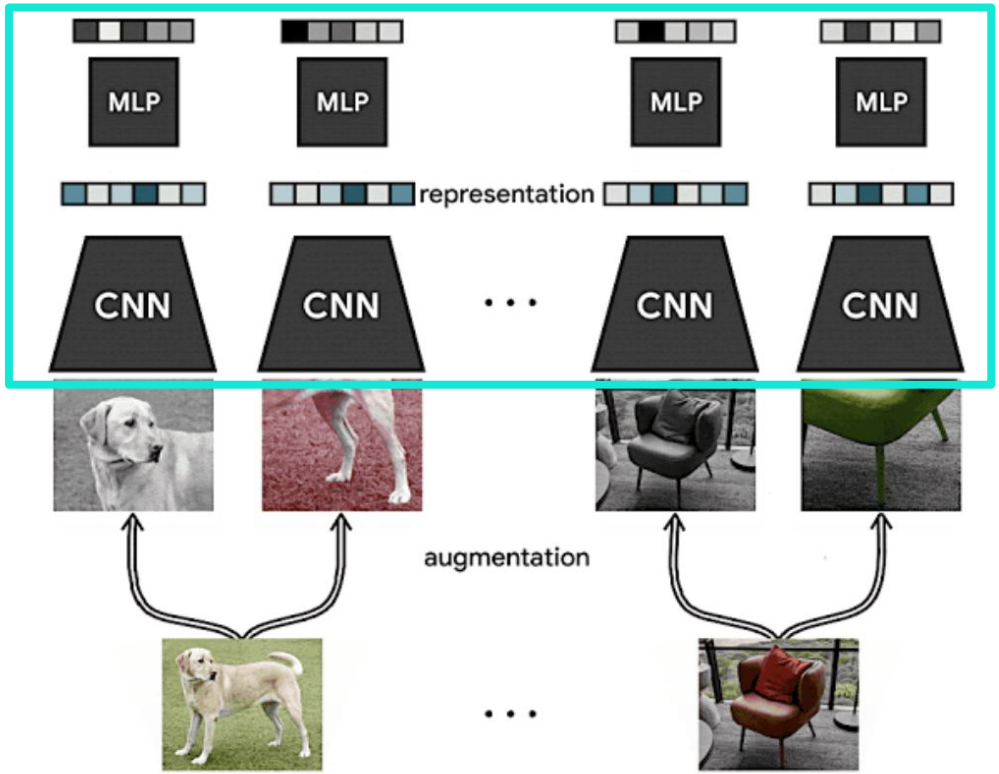
Contrastive Self-Supervised Learning

Contrastive self-supervised learning



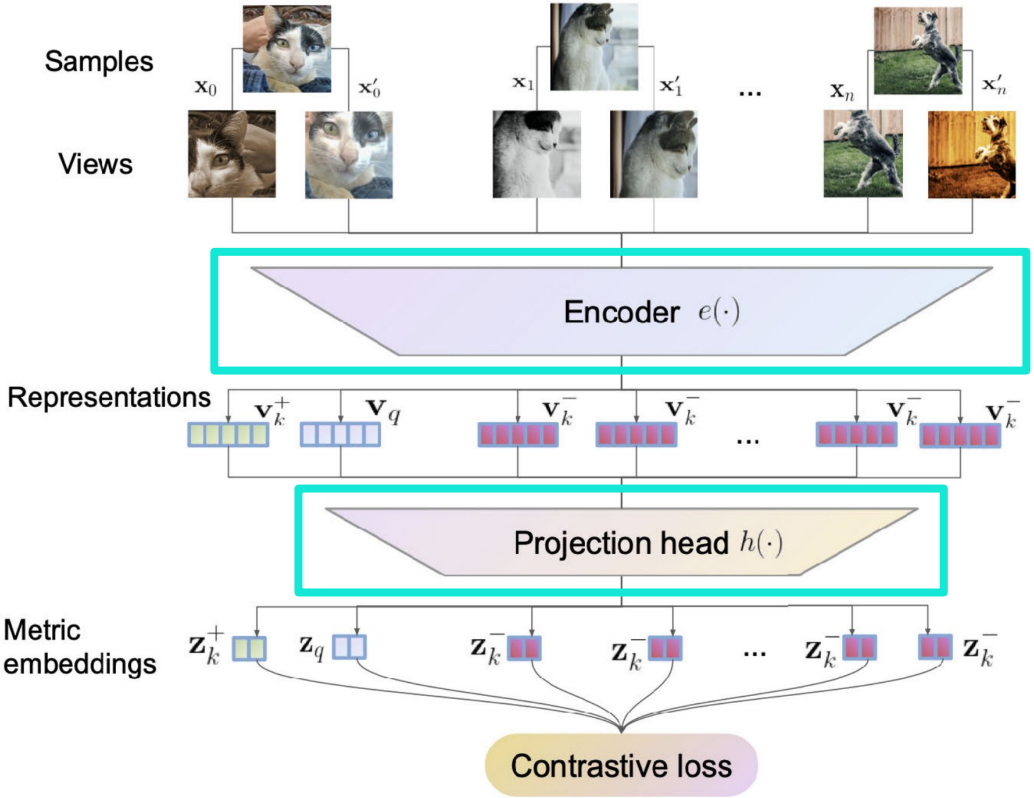
Contrastive Representation Learning: A Framework and Review

Contrastive self-supervised learning



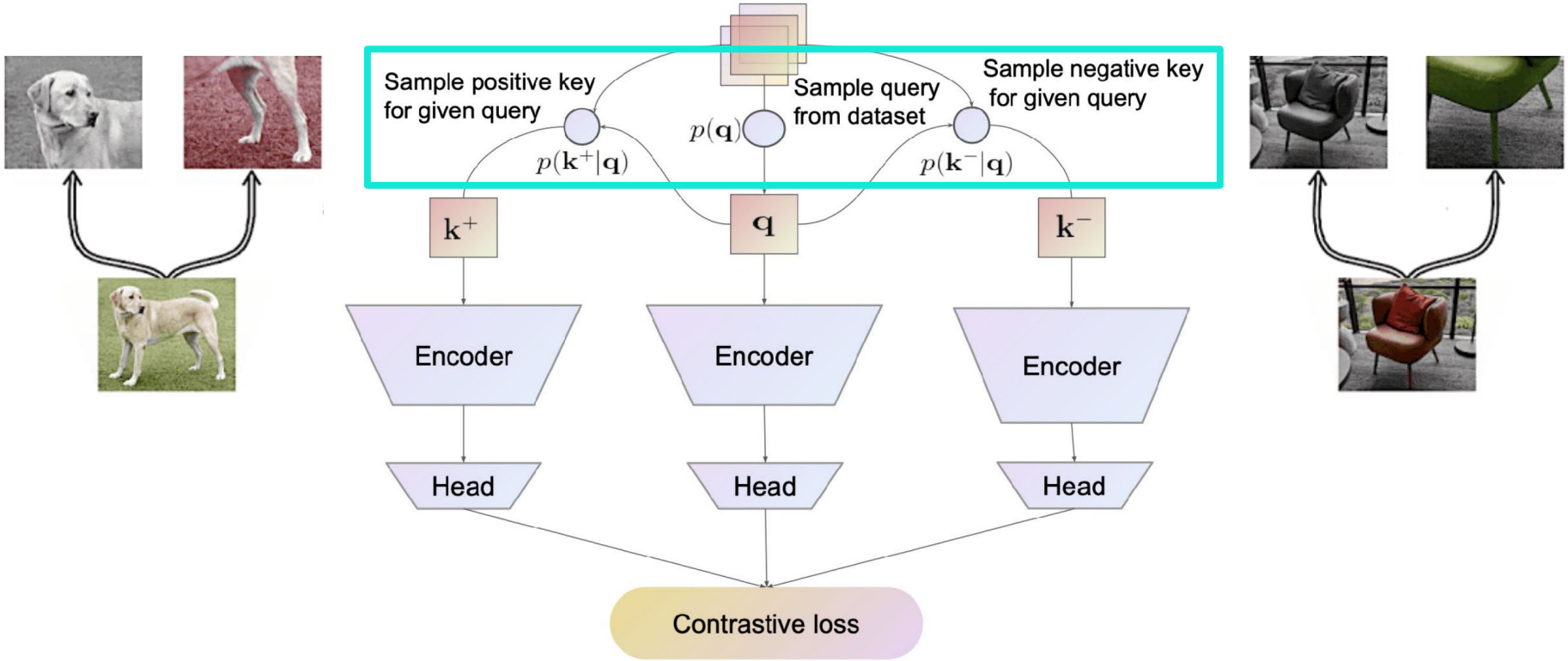
Advancing Self-Supervised and Semi-Supervised Learning with SimCLR

Contrastive self-supervised learning



Contrastive Representation Learning: A Framework and Review

Contrastive self-supervised learning



Contrastive Representation Learning: A Framework and Review & Advancing Self-Supervised and Semi-Supervised Learning with SimCLR

The InfoNCE loss

Information noise contrastive loss: cross-entropy loss for an N-way softmax classifier.

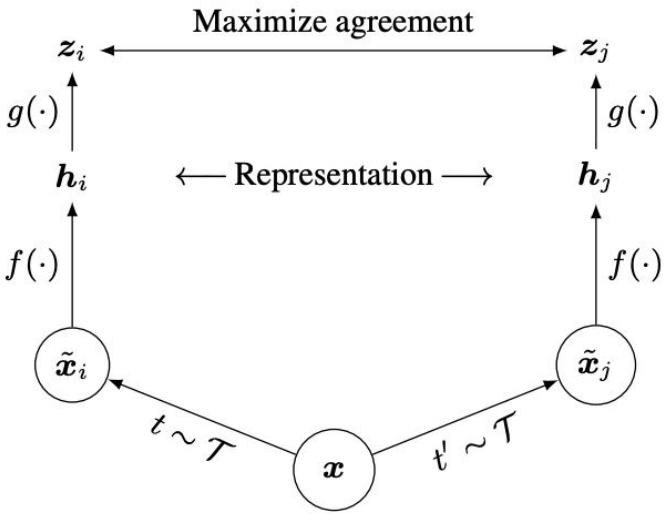
$$\text{score}(f(x), f(x^+)) >> \text{score}(f(x), f(x^-))$$

$$\mathcal{L}_N = -\mathbb{E}_X \left[\log \frac{\exp\left(f(x)^T f\left(x^+\right)\right)}{\exp\left(f(x)^T f\left(x^+\right)\right) + \sum_{j=1}^{N-1} \exp\left(f(x)^T f\left(x_j\right)\right)} \right]$$

Scoring function is the dot product.
Maximizes the mutual information between f(x) and f(x+).

Contrastive Self-Supervised Learning

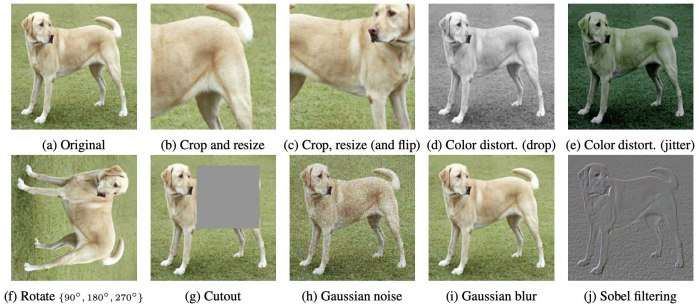
SimCLR - overview

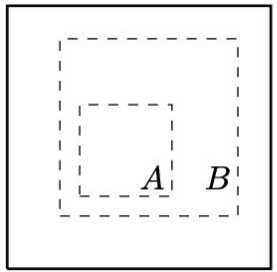


SimCLR v1 https://arxiv.org/abs/2002.05709,
SimCLR v2 https://arxiv.org/abs/2006.10029,
Advancing Self-Supervised and Semi-Supervised Learning with SimCLR

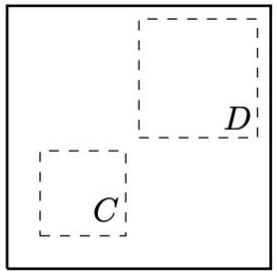
SimCLR - key findings

- Composition of multiple data augmentation operations is crucial and benefits from stronger data augmentation than supervised learning.
- Introducing a learnable nonlinear transformation between the representation and the contrastive loss substantially improves the quality of the learned representations.
- Benefits from normalized embeddings and an appropriately adjusted temperature parameter.
  (→ Features are embedded on a unit hypersphere.)
- Benefits from larger batch sizes and longer training compared to its supervised counterpart.





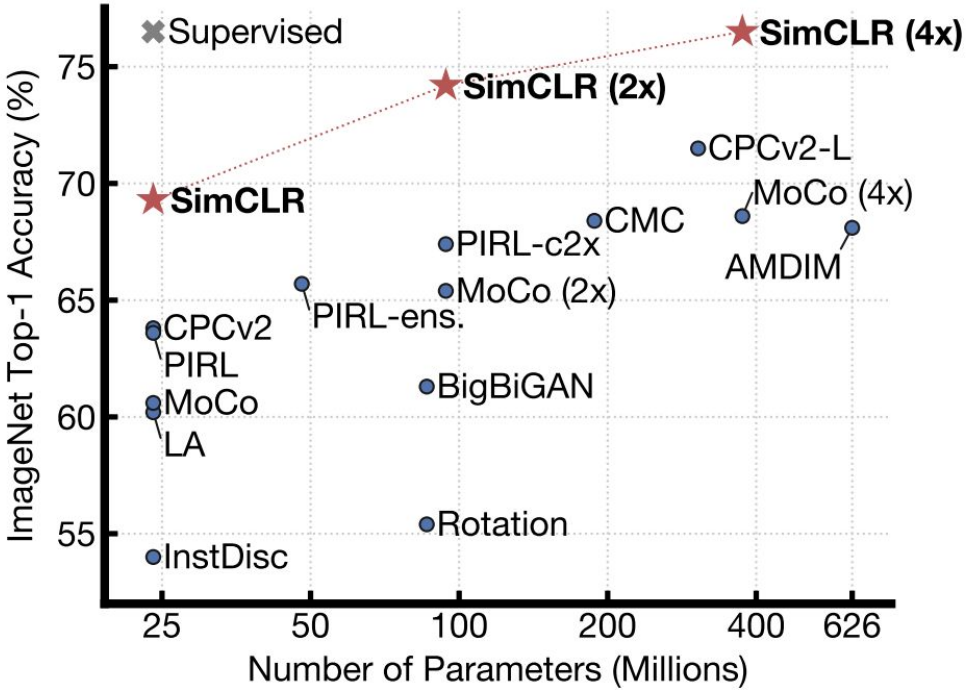
(a) Global and local views.



(b) Adjacent views.

SimCLR v1 https://arxiv.org/abs/2002.05709,
SimCLR v2 https://arxiv.org/abs/2006.10029,
Advancing Self-Supervised and Semi-Supervised Learning with SimCLR

SimCLR - results



SimCLR v1 https://arxiv.org/abs/2002.05709, SimCLR v2 https://arxiv.org/abs/2006.10029, Advancing Self-Supervised and Semi-Supervised Learning with SimCLR

Features on the unit hypersphere?

Features are (usually) L2 normalized to embed them on a unit hypersphere:

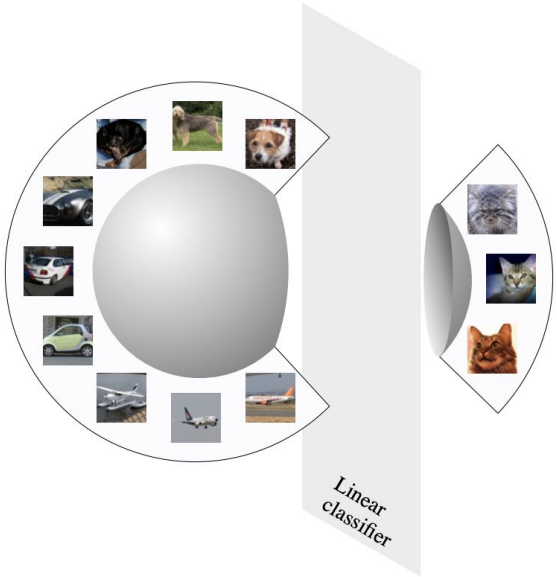
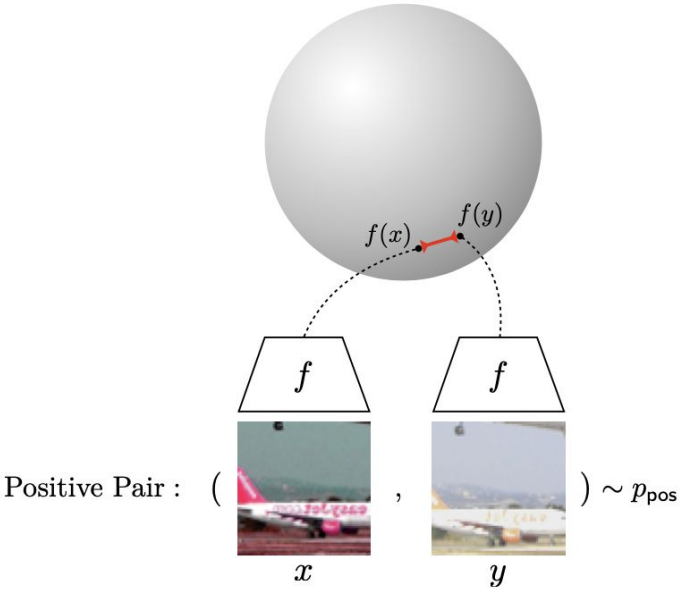


Figure 2: **Hypersphere:** When classes are well-clustered (forming spherical caps), they are linearly separable. The same does not hold for Euclidean spaces.

https://arxiv.org/abs/2005.10242

Alignment & uniformity?

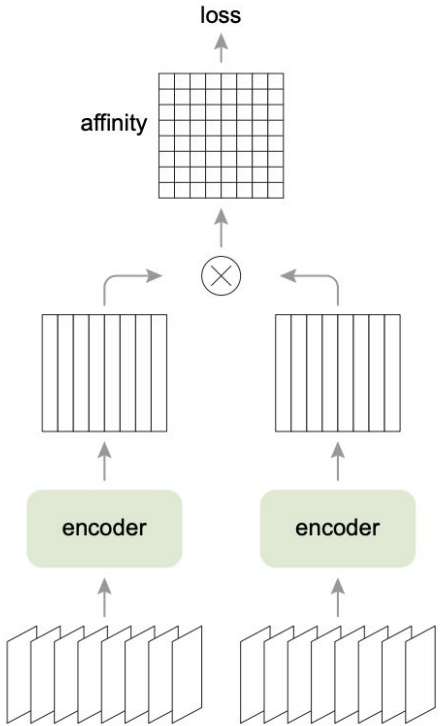


Alignment: Similar samples have similar features.
(Figure inspired by Tian et al. (2019).)

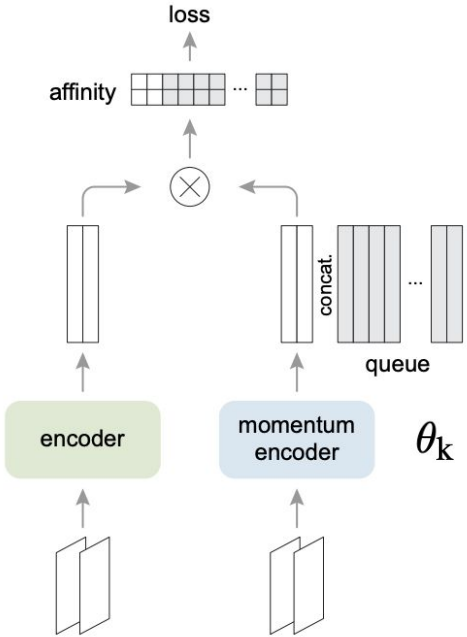
Uniformity: Preserve maximal information.

https://arxiv.org/abs/2005.10242

Momentum contrast (MoCo) - overview



(a) end-to-end



(b) Momentum Contrast

$$\theta_{\mathrm{k}} \leftarrow m\theta_{\mathrm{k}} + (1-m)\theta_{\mathrm{q}}.$$

MoCo v1 https://arxiv.org/abs/1911.05722, MoCo v2 https://arxiv.org/abs/2003.04297

MoCo - results

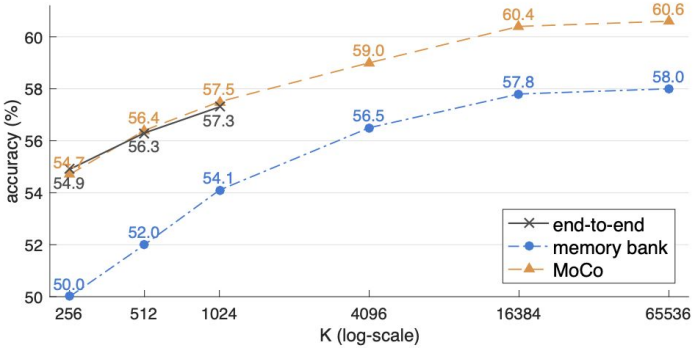


Figure 3. **Comparison of three contrastive loss mechanisms** under the ImageNet linear classification protocol. We adopt the same pretext task (Sec. 3.3) and only vary the contrastive loss mechanism (Figure 2). The number of negatives is K in memory bank and MoCo, and is $K-1$ in end-to-end (offset by one because the positive key is in the same mini-batch). The network is ResNet-50.

case	unsup. pre-train MLP	aug+	cos	epochs	batch	ImageNet acc.
MoCo v1 [6]				200	256	60.6
SimCLR [2]	✓	✓	✓	200	256	61.9
SimCLR [2]	✓	✓	✓	200	8192	66.6
MoCo v2	✓	✓	✓	200	256	**67.5**
*results of **longer** unsupervised training follow:*						
SimCLR [2]	✓	✓	✓	1000	4096	69.3
MoCo v2	✓	✓	✓	800	256	**71.1**

Table 2. **MoCo *vs.* SimCLR**: ImageNet linear classifier accuracy (**ResNet-50, 1-crop 224×224**), trained on features from unsupervised pre-training. "aug+" in SimCLR includes blur and stronger color distortion. SimCLR ablations are from Fig. 9 in [2] (we thank the authors for providing the numerical results).

mechanism	batch	memory / GPU	time / 200-ep.
MoCo	256	**5.0G**	**53 hrs**
end-to-end	256	7.4G	65 hrs
end-to-end	4096	93.0G†	n/a

Table 3. **Memory and time cost** in 8 V100 16G GPUs, implemented in PyTorch. †: based on our estimation.

MoCo v1 https://arxiv.org/abs/1911.05722,
MoCo v2 https://arxiv.org/abs/2003.04297

Debiased contrastive learning

Number of (false) negative samples?

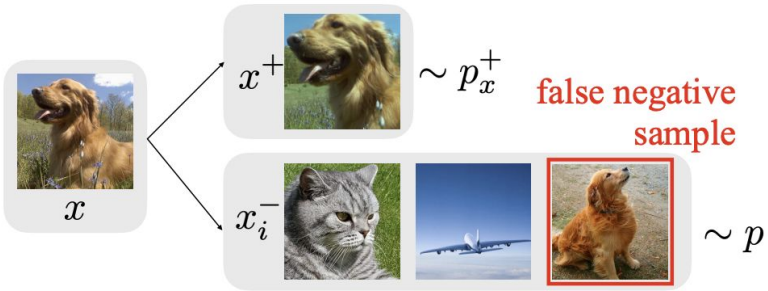


Figure 1: **"Sampling bias":** The common practice of drawing negative examples x_i^- from the data distribution $p(x)$ may result in x_i^- that are actually similar to x.

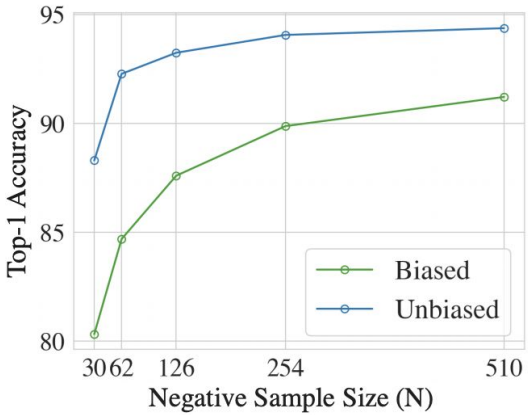
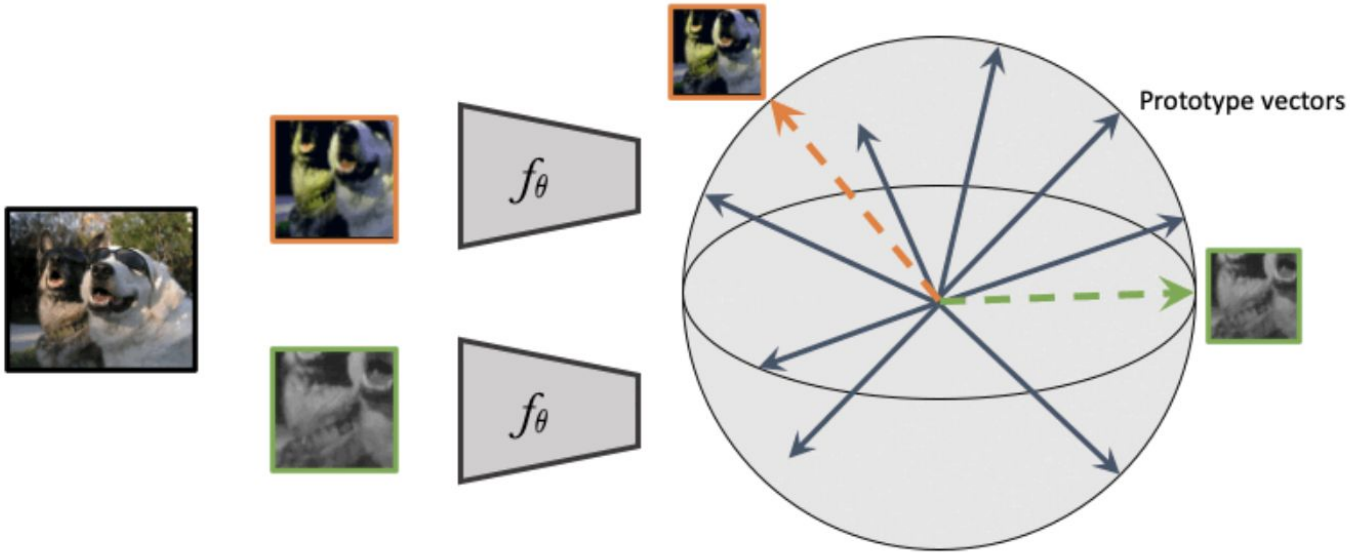


Figure 2: **Sampling bias leads to performance drop:** Results on CIFAR-10 for drawing x_i^- from $p(x)$ (biased) and from data with different labels, i.e., truly semantically different data (unbiased).

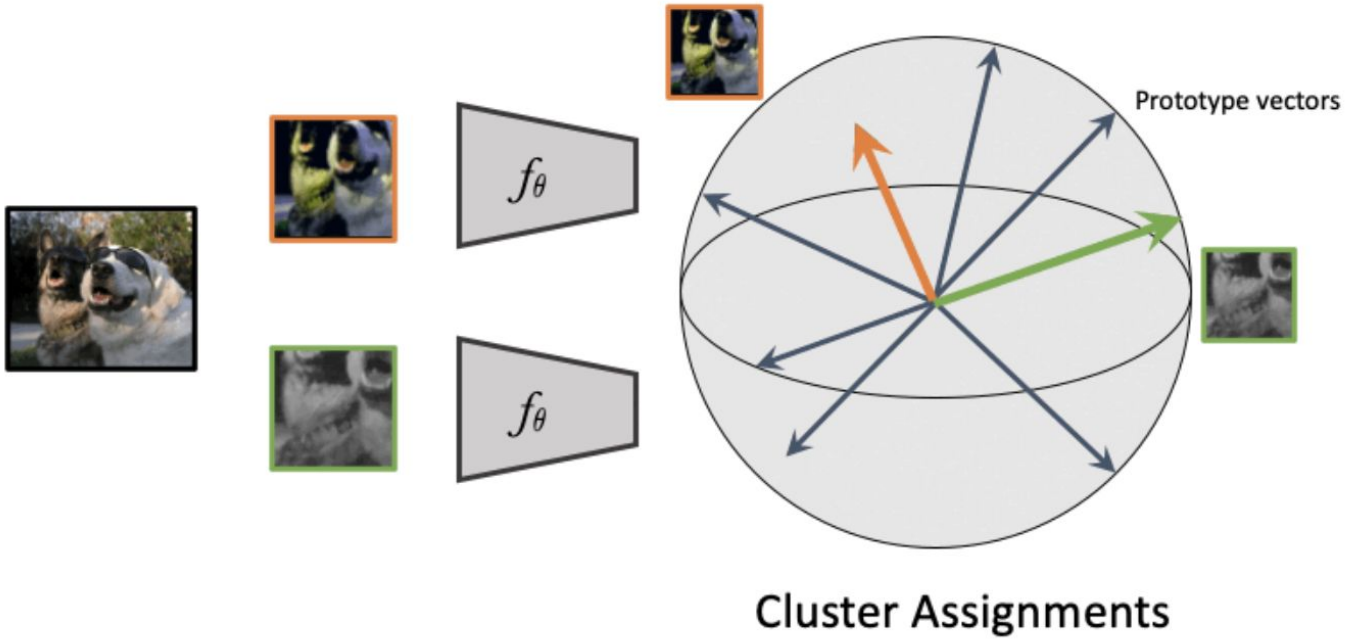
https://arxiv.org/abs/2007.00224

Swapping Assignments between Views (SwAV)



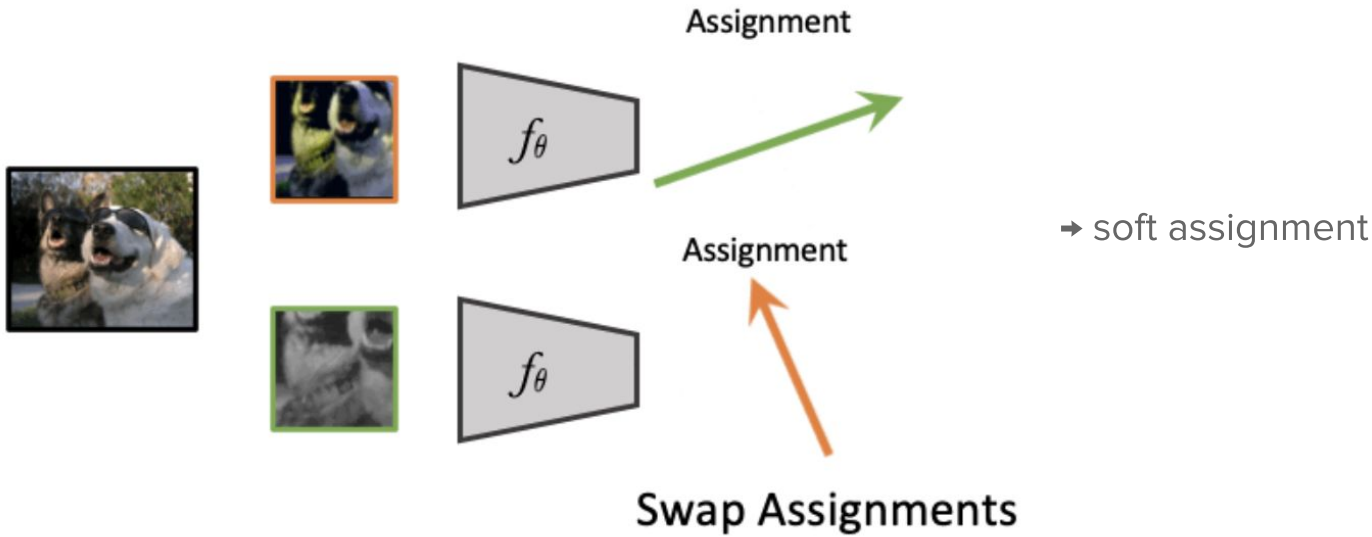
https://arxiv.org/abs/2006.09882 & https://github.com/facebookresearch/swav

Swapping Assignments between Views (SwAV)



https://arxiv.org/abs/2006.09882 & https://github.com/facebookresearch/swav

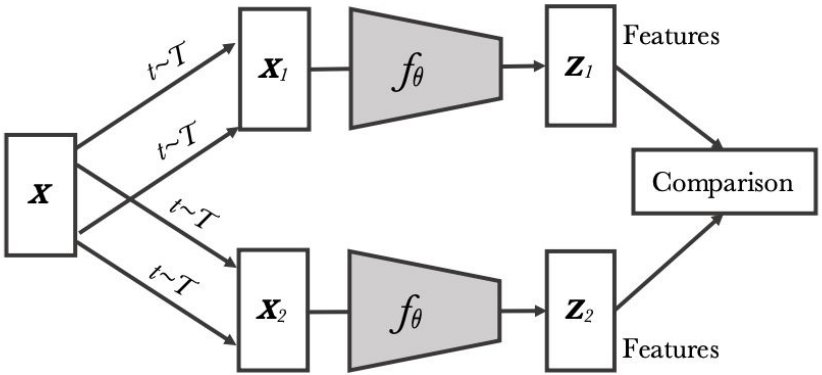
Swapping Assignments between Views (SwAV)



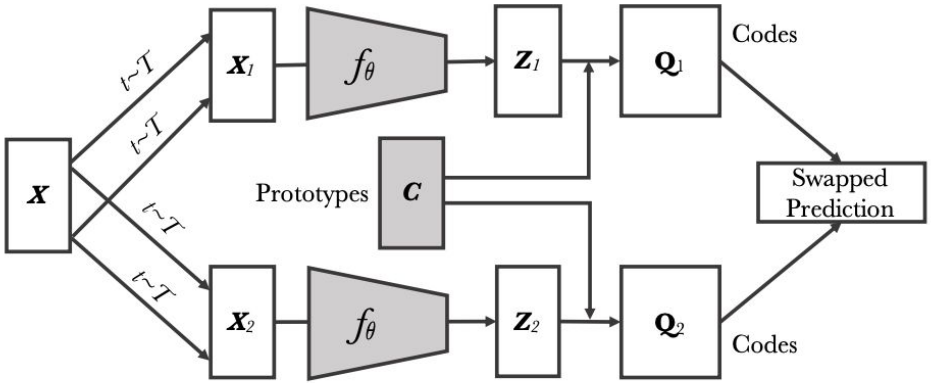
➔ soft assignment

https://arxiv.org/abs/2006.09882 & https://github.com/facebookresearch/swav

Swapping Assignments between Views (SwAV)



Contrastive instance learning



Swapping Assignments between Views (Ours)

https://arxiv.org/abs/2006.09882 & https://github.com/facebookresearch/swav

Swapping Assignments between Views (SwAV)

https://arxiv.org/abs/2006.09882 & https://github.com/facebookresearch/swav

SwAV: Multi-crop

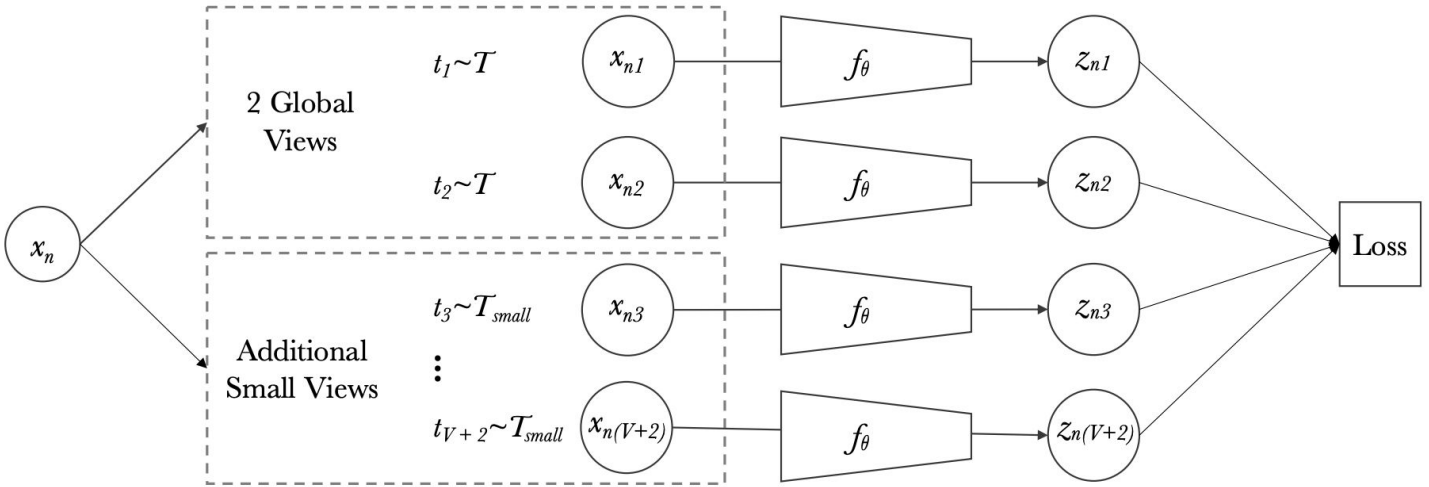
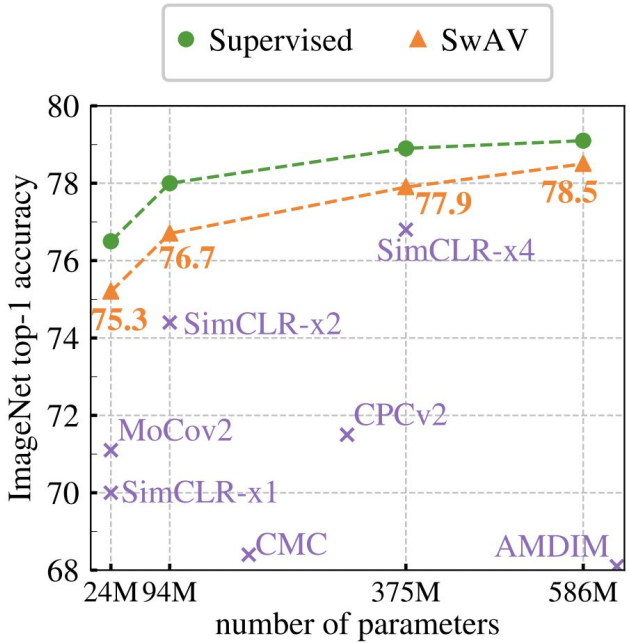


Figure 5: **Multi-crop**: the image x_n is transformed into $V + 2$ views: two global views and V small resolution zoomed views.

https://arxiv.org/abs/2006.09882 & https://github.com/facebookresearch/swav

SwAV: Results



Method	Arch.	Param.	Top1
Supervised	R50	24	76.5
Colorization [63]	R50	24	39.6
Jigsaw [44]	R50	24	45.7
NPID [56]	R50	24	54.0
BigBiGAN [15]	R50	24	56.6
LA [66]	R50	24	58.8
NPID++ [42]	R50	24	59.0
MoCo [24]	R50	24	60.6
SeLa [2]	R50	24	61.5
PIRL [42]	R50	24	63.6
CPC v2 [27]	R50	24	63.8
PCL [35]	R50	24	65.9
SimCLR [10]	R50	24	70.0
MoCov2 [11]	R50	24	71.1
SwAV	R50	24	**75.3**

https://arxiv.org/abs/2006.09882 & https://github.com/facebookresearch/swav

The future: (Image) Transformers?

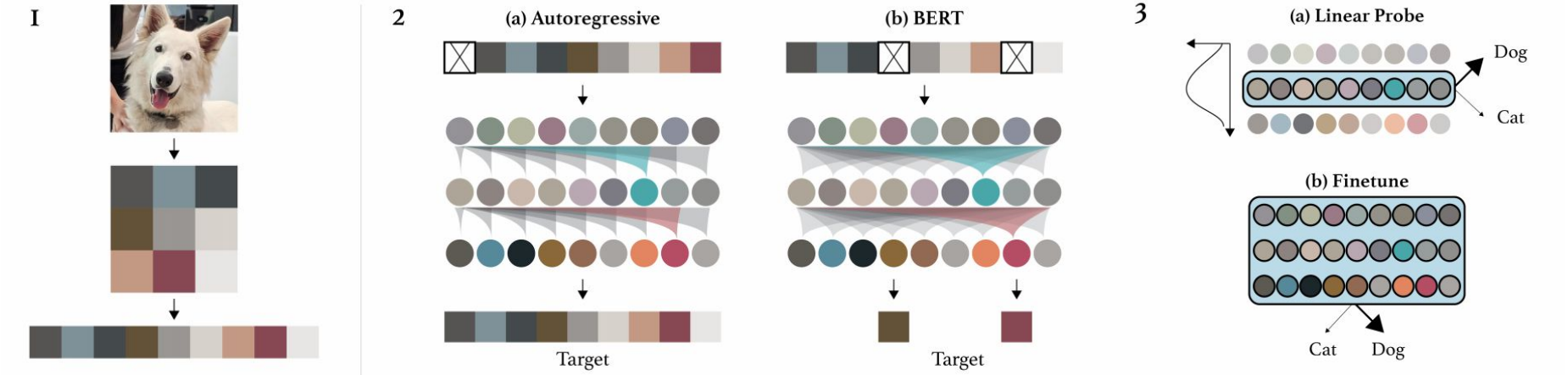
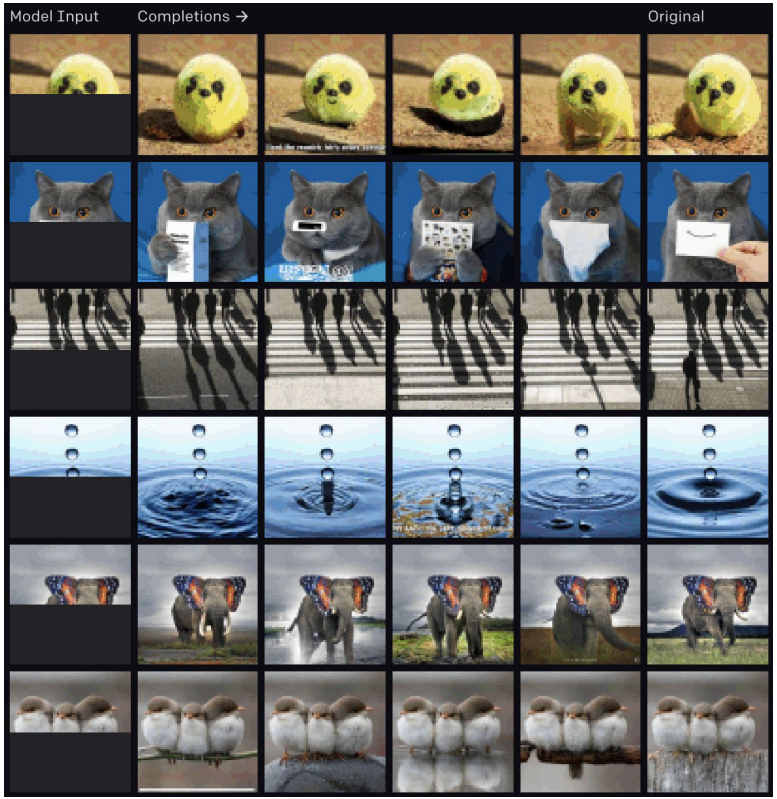


Image GPT, Generative Pretraining from Pixels

Image Transformers



METHOD	INPUT RESOLUTION	FEATURES	PARAMETERS	ACCURACY
Rotation[53]	original	8192	86M	55.4
iGPT-L	32x32	1536	1362M	60.3
BigBiGAN[37]	original	16384	86M	61.3
iGPT-L	48x48	1536	1362M	65.2
AMDIM[13]	original	8192	626M	68.1
MoCo[24]	original	8192	375M	68.6
iGPT-XL	64x64	3072	6801M	68.7
SimCLR[12]	original	2048	24M	69.3
CPC v2[25]	original	4096	303M	71.5
iGPT-XL	64x64	3072 x 5	6801M	72.0
SimCLR	original	8192	375M	**76.5**

A comparison of linear probe accuracies between our models and state-of-the-art self-supervised models. We achieve competitive performance while training at much lower input resolutions, though our method requires more parameters and compute.

Image GPT, Generative Pretraining from Pixels

Thank you for your attention! Questions?

Sources:
Self-Supervised Representation Learning
Contrastive Self-Supervised Learning
Contrastive Representation Learning: A Framework and Review
Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere
SimCLR v1 A Simple Framework for Contrastive Learning of Visual Representations,
SimCLR v2 Big Self-Supervised Models are Strong Semi-Supervised Learners,
Advancing Self-Supervised and Semi-Supervised Learning with SimCLR
MoCo v1 Momentum Contrast for Unsupervised Visual Representation Learning
MoCo v2 Improved Baselines with Momentum Contrastive Learning
Debiased Contrastive Learning
SwAV Unsupervised Learning of Visual Features by Contrasting Cluster Assignments
facebookresearch/swav: PyTorch implementation of SwAV https//arxiv.org/abs/2006.09882
Image GPT
Generative Pretraining from Pixels
https://www.gwern.net/newsletter/2020/05#baking-the-cake

Additional material:
VDLM #32:
https://github.com/vdlm/meetups/blob/master/Meetups/Meetup_32/slides/Self-Supervised%20Learning%20-%20Bonitz.pdf
Understanding View Selection for Contrastive Learning
Approximating Wasserstein distances with PyTorch