Self-supervised learning in computer vision

Ishan Misra

Facebook AI Research

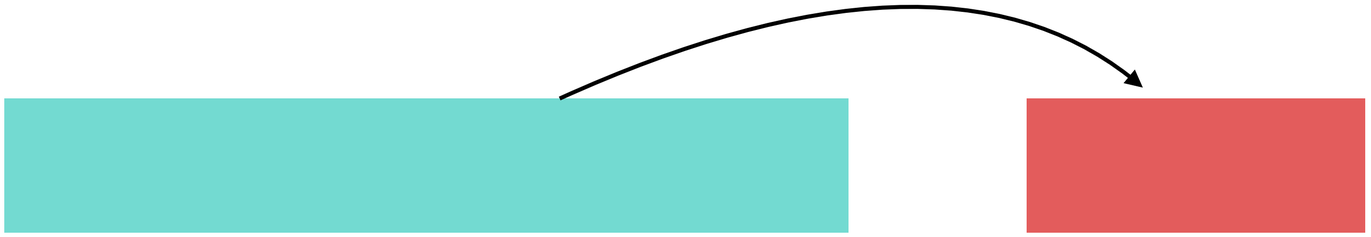
The promise of "alternative" supervision

- Getting "real" labels is difficult and expensive
  - ImageNet with 14M images took 22 human years.

- Obtain labels using a "semi-automatic" process
  - Hashtags
  - GPS locations
  - Using the data itself: "self"-supervised

What is "self" supervision?

- Obtain "labels" from the data itself by using a "semi-automatic" process
- Predict part of the data from other parts



Observed data

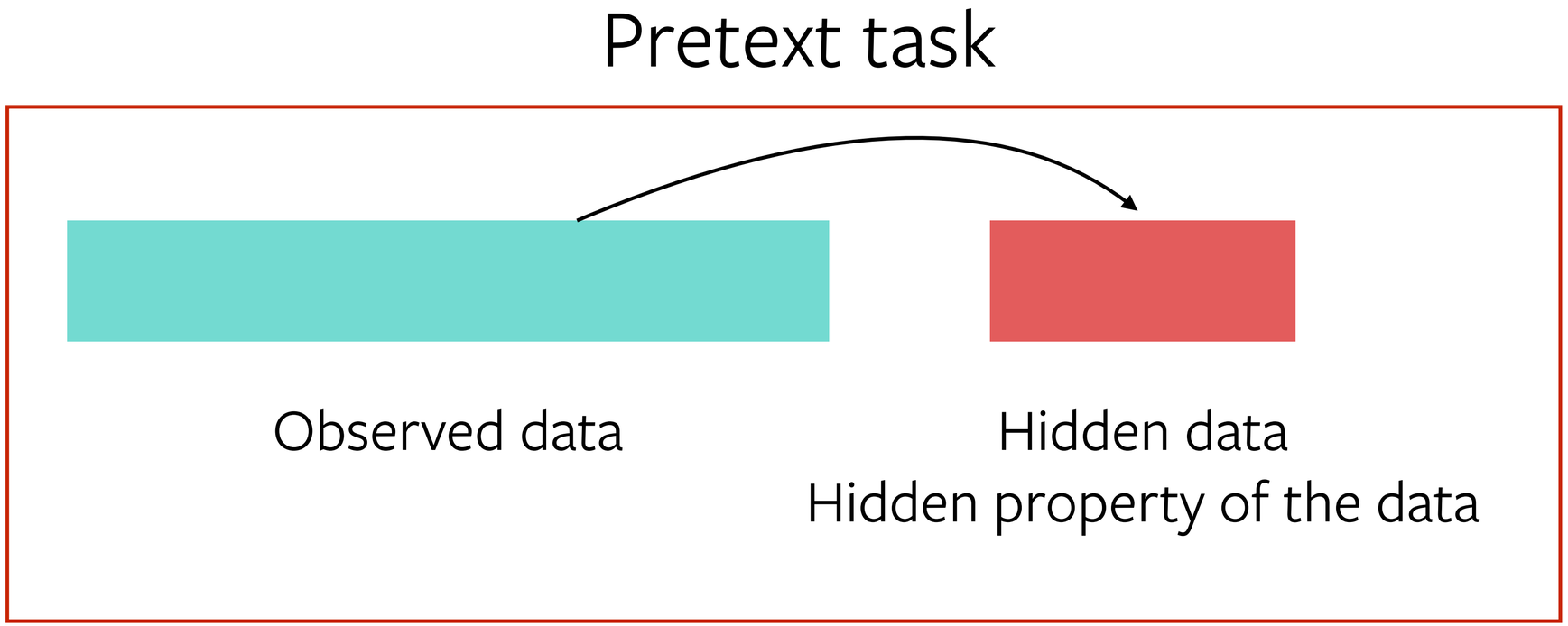
Hidden data

Hidden property of the data

In the context of Computer Vision

Pretext task

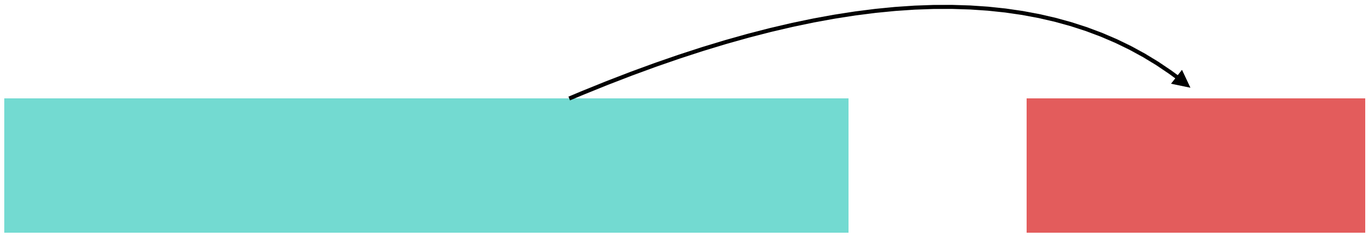
- Self-supervised task used for learning representations
- Often, not the "real" task (like image classification) we care about



Pretext task - Doersch et al., 2015, Unsupervised visual representation learning by context prediction

Pretext task

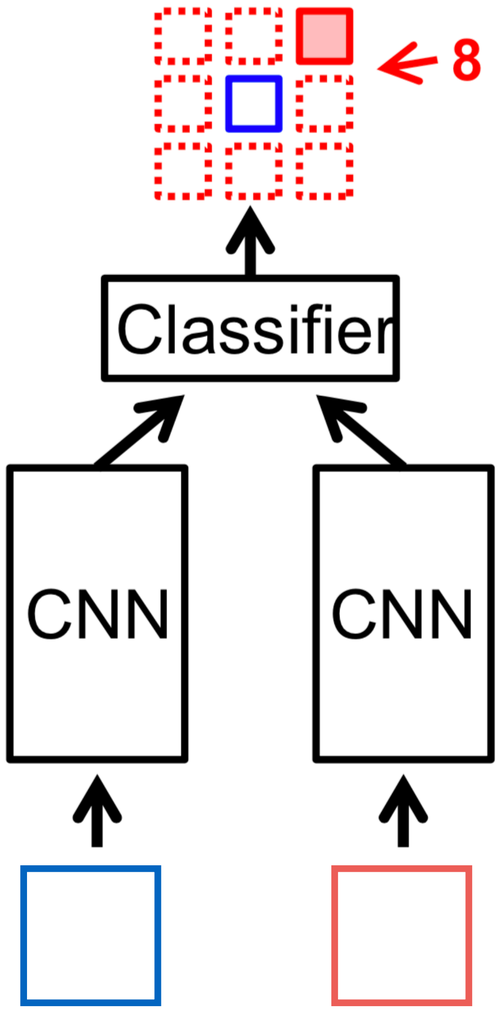
- Using images
- Using video
- Using video and sound

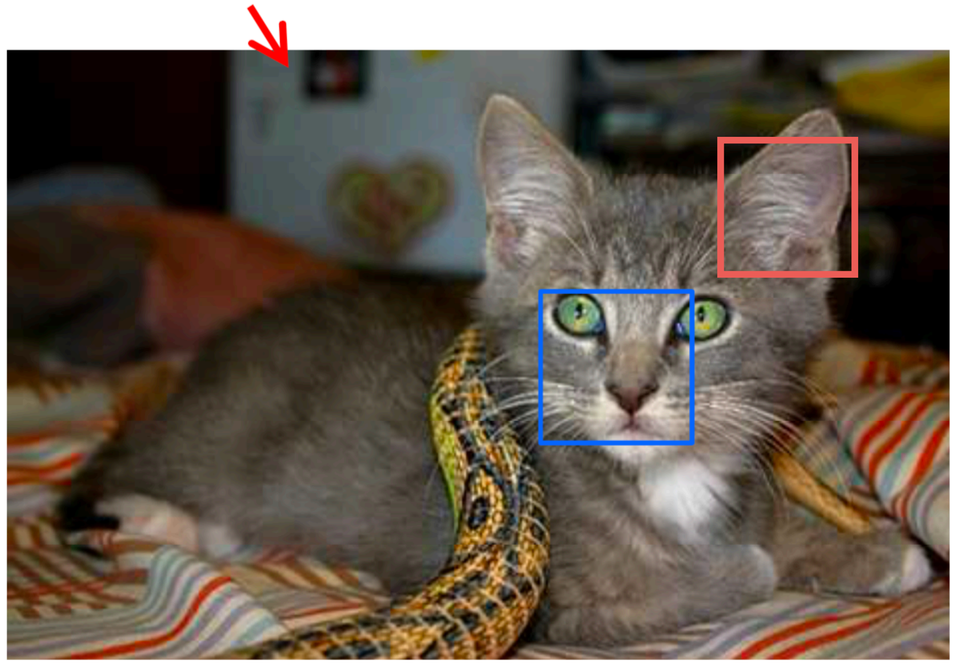


Observed data

Hidden data
Hidden property of the data

Relative Position of patches



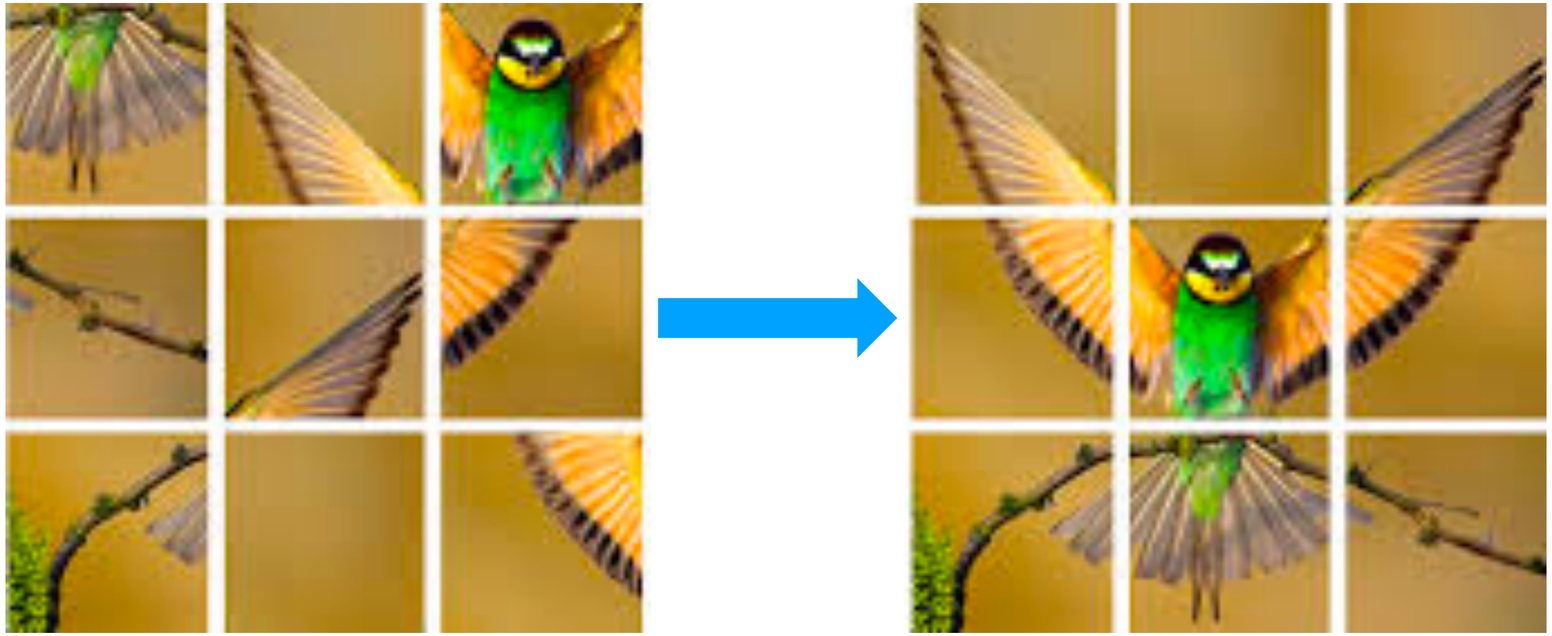


Randomly Sample Patch
Sample Second Patch

Input: Two patches
Output: 8-way classification

Doersch et al., 2015, Unsupervised visual representation learning by context prediction

Jigsaw



Jigsaw puzzles
(Noorozi & Favaro, 2016)

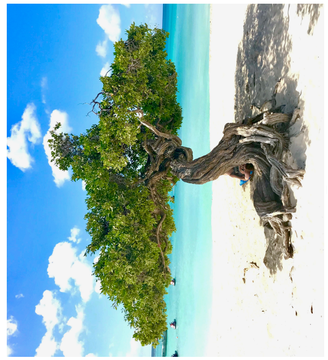
Input: nine patches
Permute using one of N permutations

Output: N-way classification

Set N << 9!

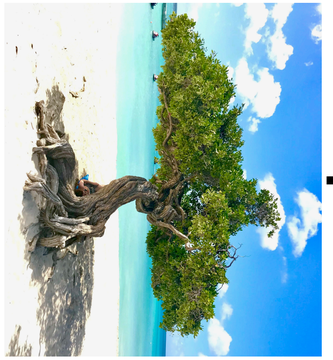
Predicting Rotations

0^0



90^0

180^0



270^0

Input: image rotated by [0, 90, 180, 270]

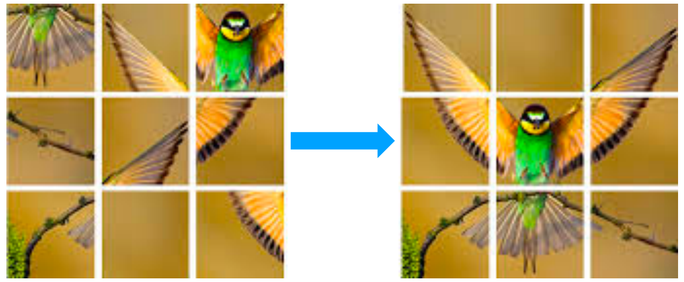
Output: 4-way classification

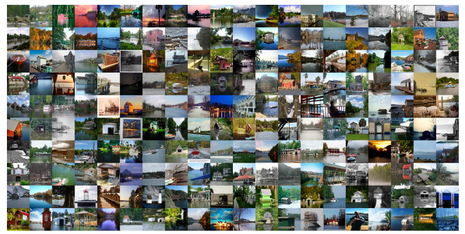
Gidaris et al., 2018, Predicting Image Rotations

What is missing from "pretext" tasks?
Or in general "proxy" tasks

The hope of generalization

- We really **hope** that the pre-training task and the transfer task are "aligned"



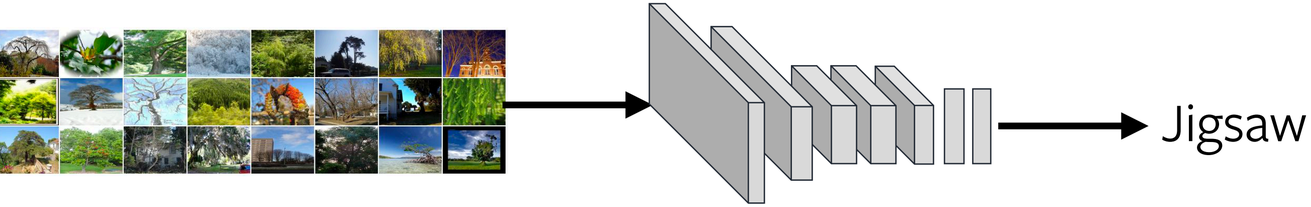


Pre-training

Self-supervised

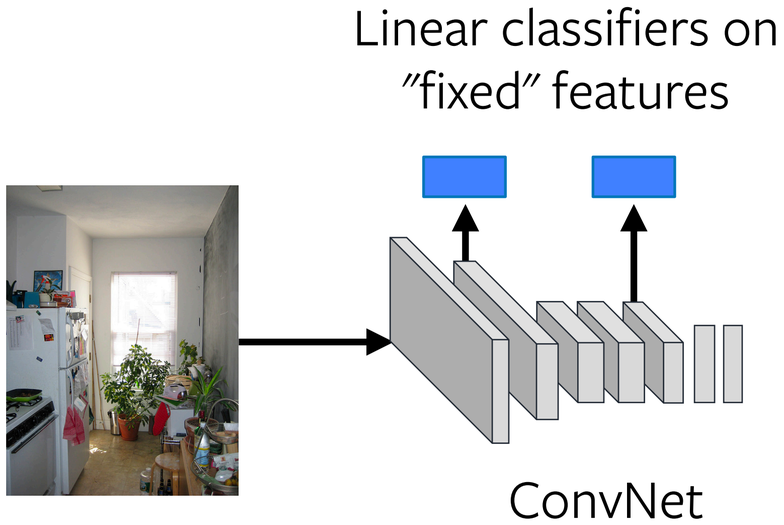
Transfer Tasks

The hope of generalization ... ?



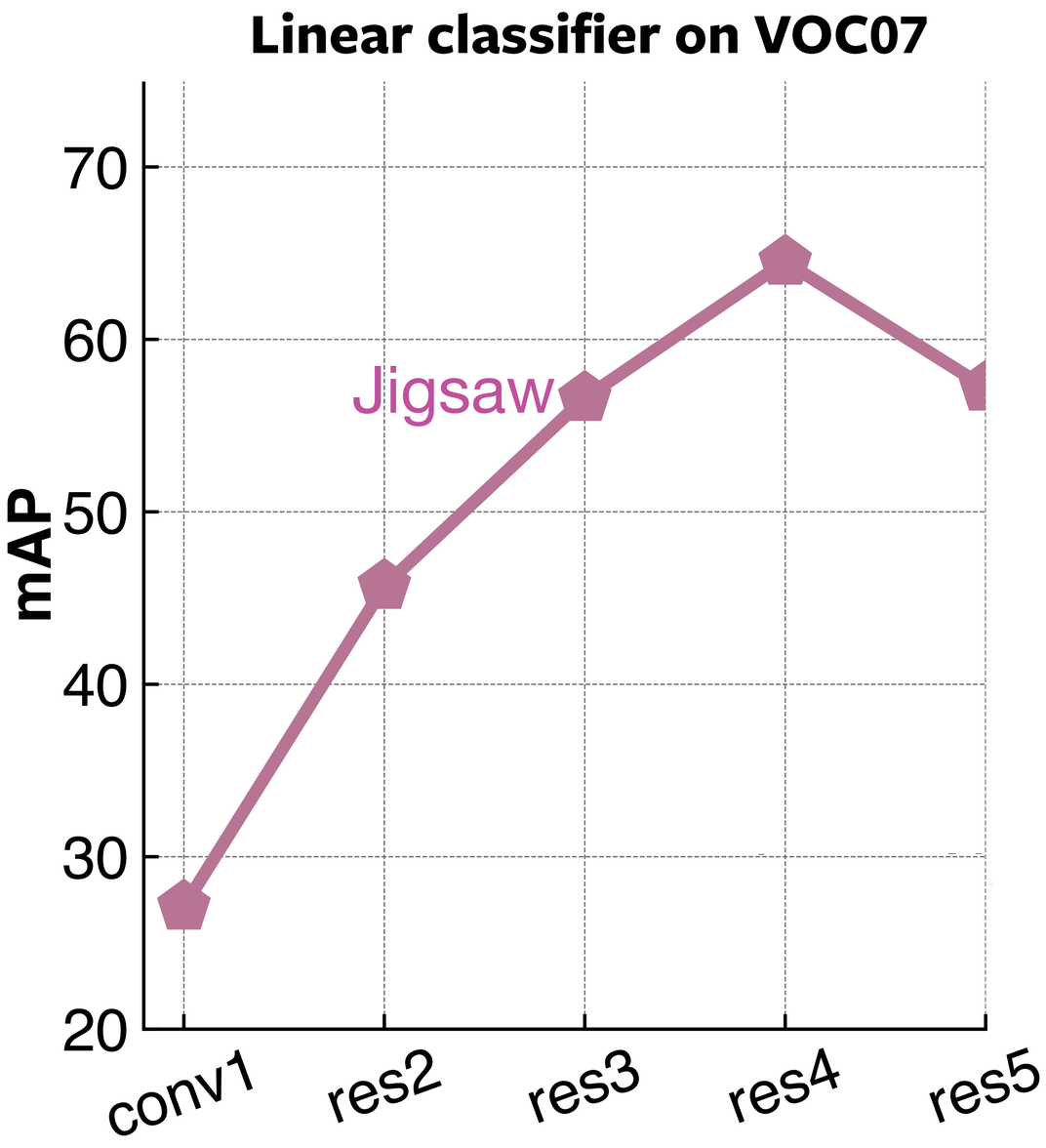
Pre-training

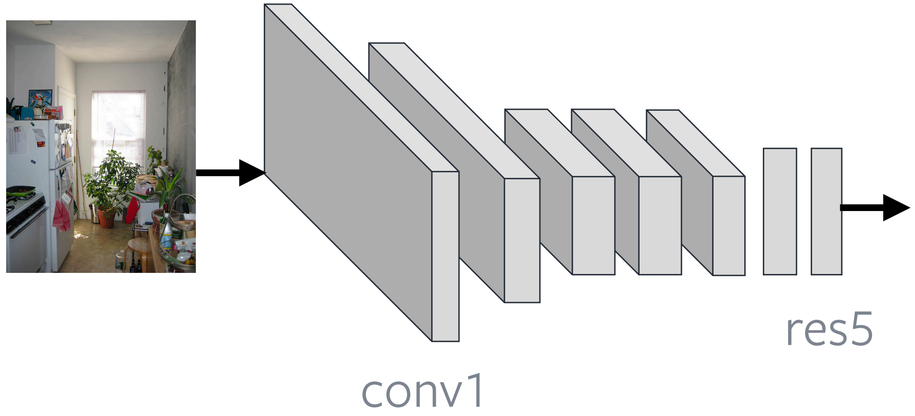
Weak or self-supervised



Transfer

Higher layers do not generalize ...





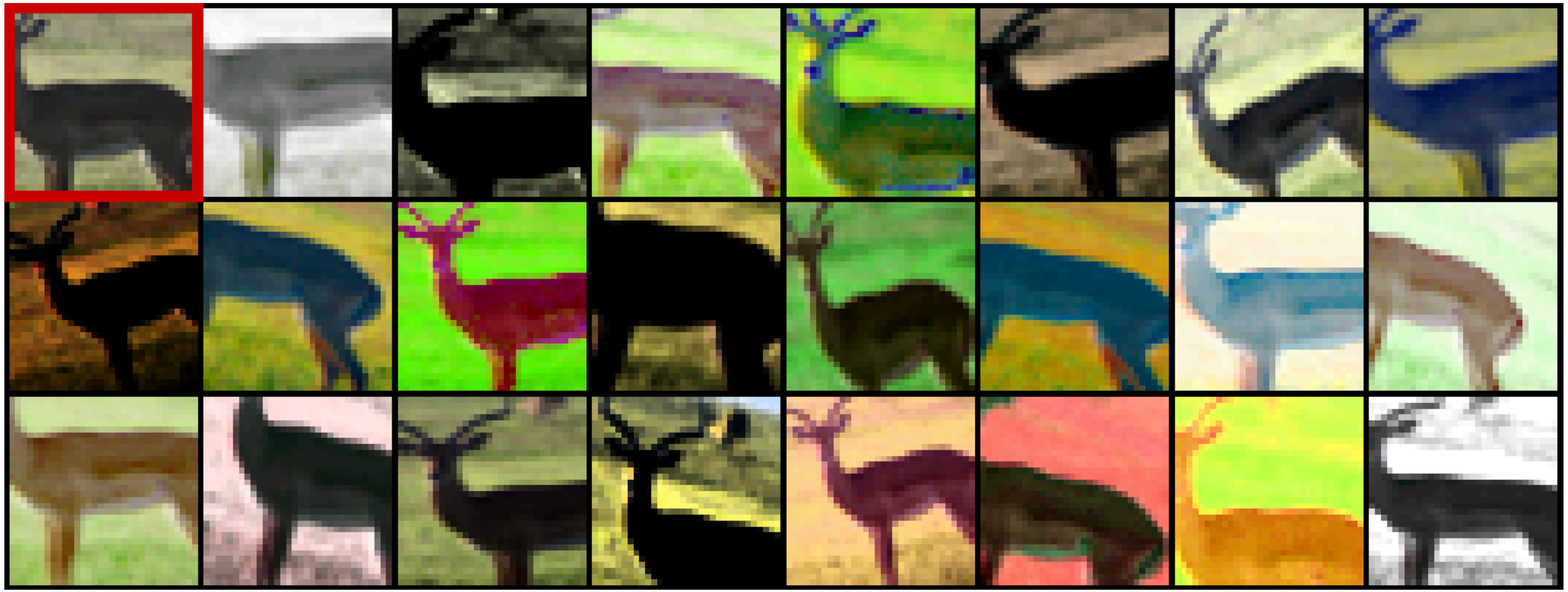
mAP = mean Average Precision
(Higher is better)

Pre-trained features should ...

- Represent how images relate to one another

- Be robust to "nuisance factors" -- Invariance
- e.g., exact location of objects, lighting, exact color

Popular & Common principle for most methods

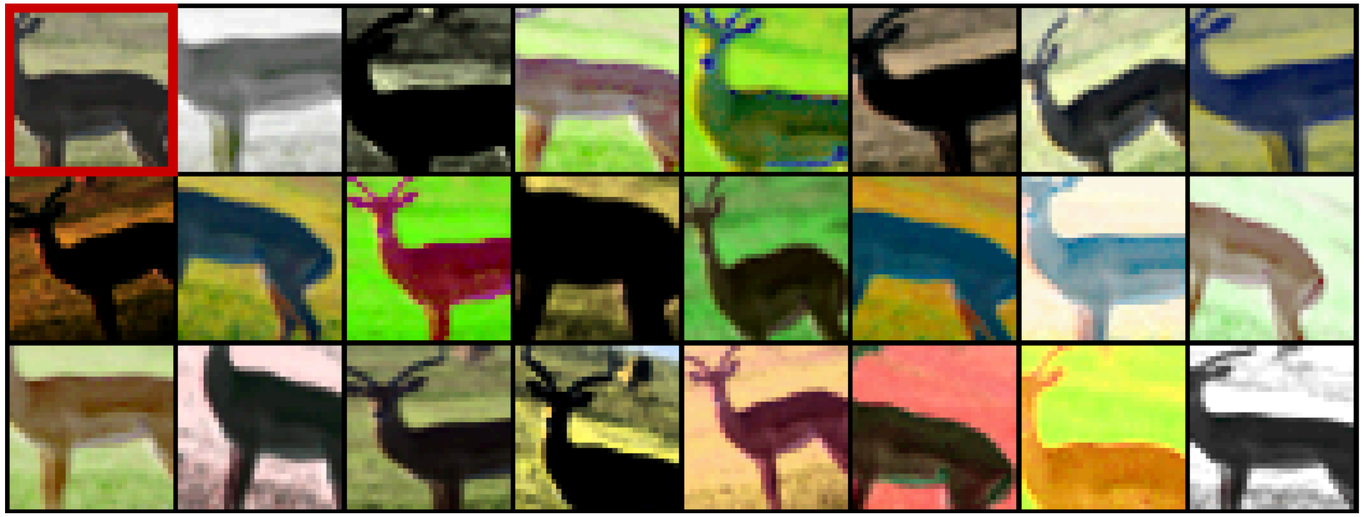


Learn features such that:

$$f_\theta(I) = f_\theta(\text{augment}(I))$$

Figure from Dosovitskiy et al., 2014

Why is it useful?



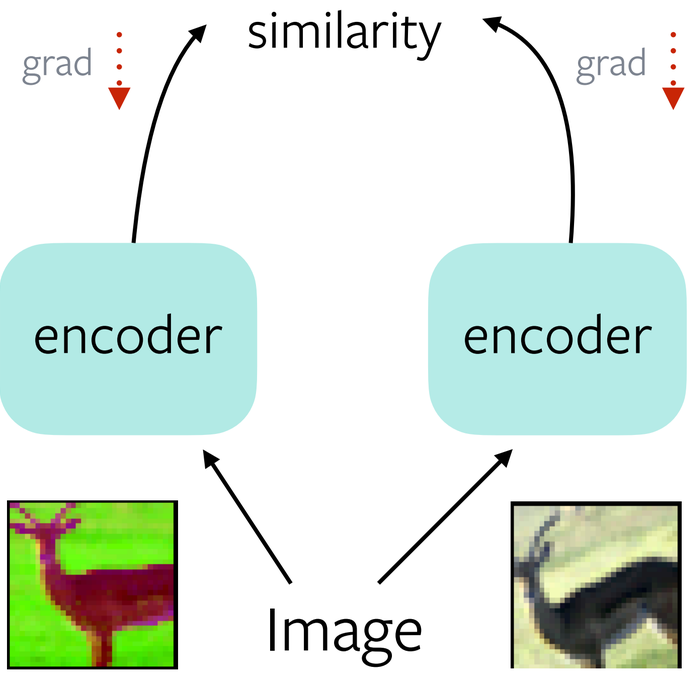
Learn features such that:

$$f_\theta(I) = f_\theta(\text{augment}(I))$$

Learned features are invariant to "nuisance factors"
or data augmentation

Can it work?

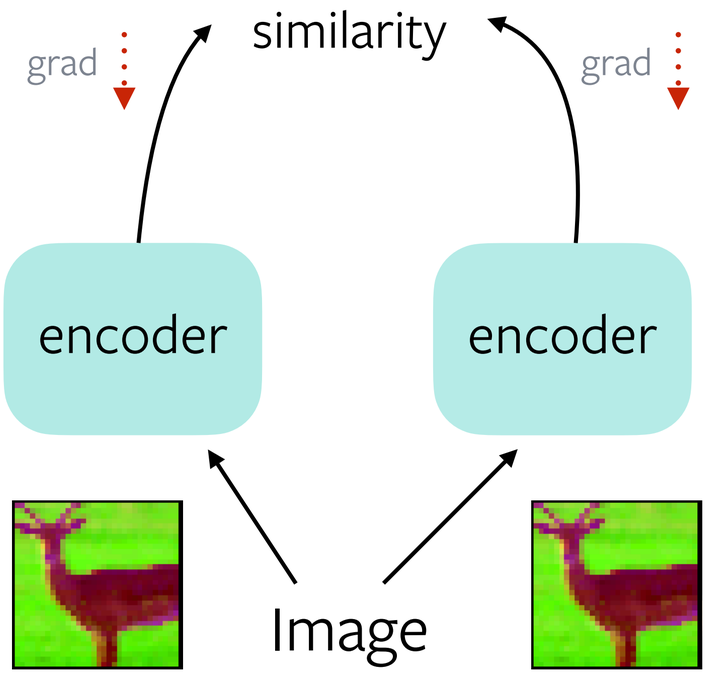
$$f_\theta(I) = f_\theta(\text{augment}(I))$$



Trivial Solutions

$$f_\theta(I) = f_\theta(\text{augment}(I))$$

$$f_\theta(I) = \text{constant}$$



Satisfies the invariance property, but not useful

Categorization of recent self-supervised methods

Many ways to avoid trivial solutions

Similarity Maximization Objective

- Contrastive learning
  - MoCo, PIRL, SimCLR
- Clustering
  - DeepCluster, SeLA, SwAV
- Distillation
  - BYOL, SimSiam

Redundancy Reduction Objective

- Redundancy Reduction
  - Barlow Twins

Many ways to avoid trivial solutions

Similarity Maximization Objective

- Contrastive learning
  - MoCo, PIRL, SimCLR
- Clustering
  - DeepCluster, SeLA, SwAV
- Distillation
  - BYOL, SimSiam

Redundancy Reduction Objective

- Redundancy Reduction
  - Barlow Twins

Pretraining

- ImageNet without labels (1.3M images)
- ResNet-50 initialized randomly

Evaluation using Transfer Learning

Transfer to downstream task

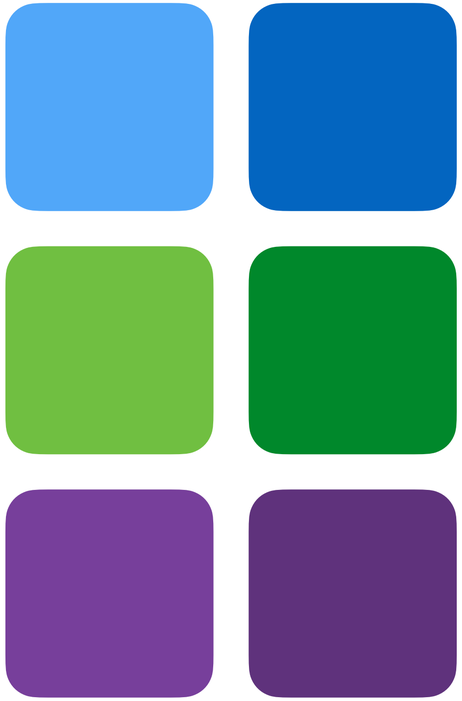
- Train a linear classifier on frozen features
- Full finetuning of the network

Pretext-Invariant Representation Learning (PIRL)

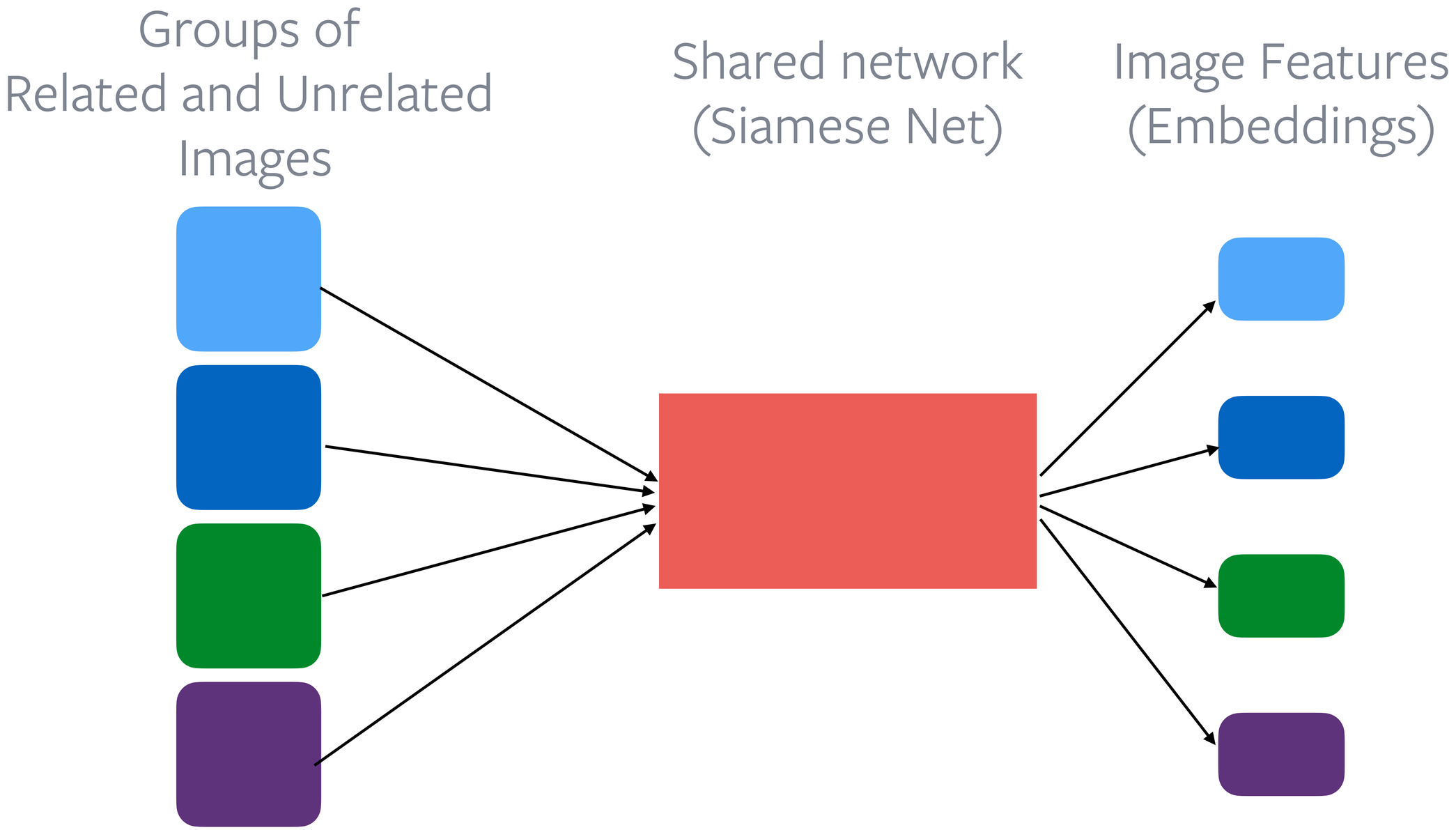
Ishan Misra, Laurens van der Maaten

Contrastive Learning

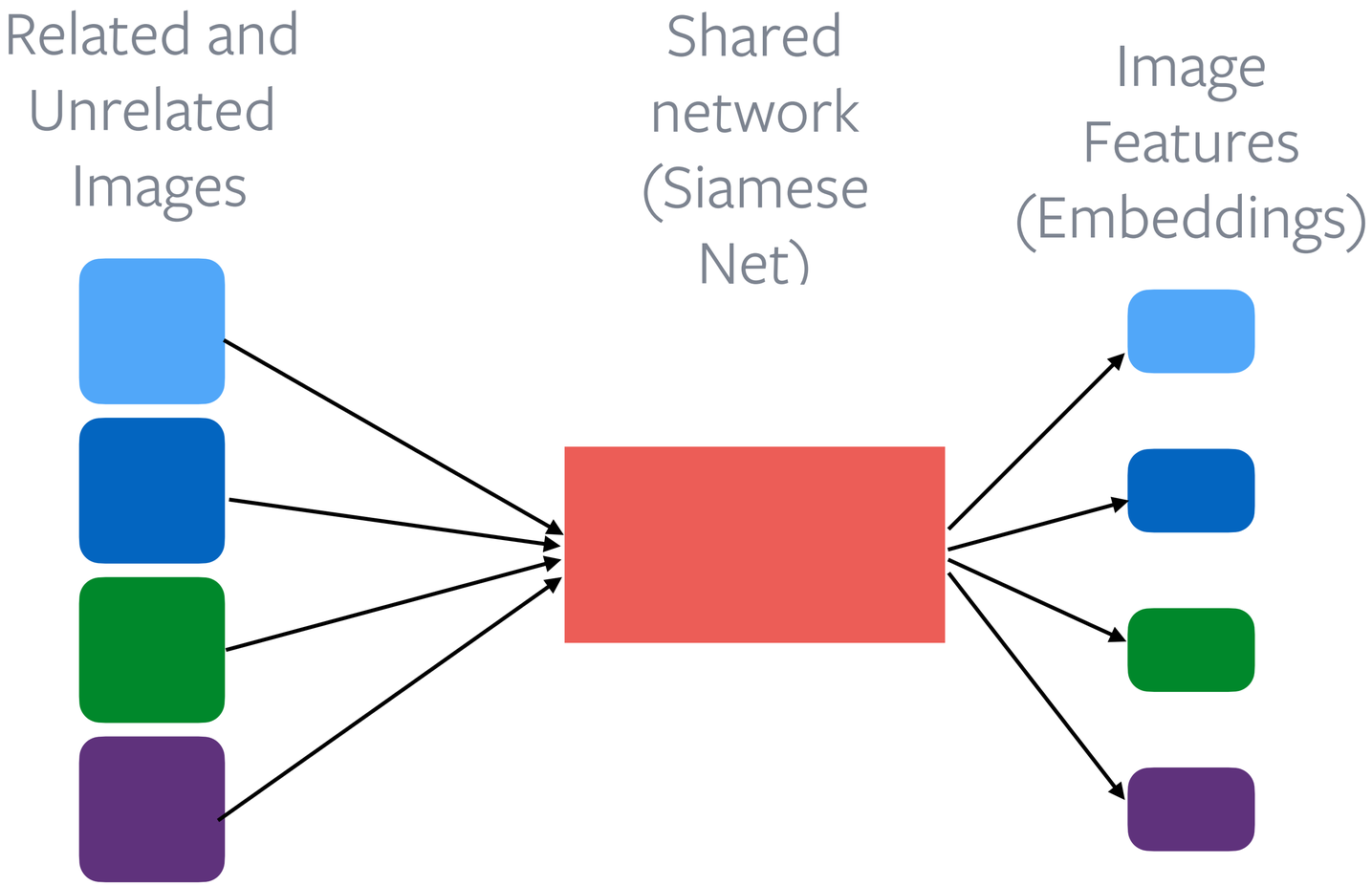
Groups of
Related and Unrelated
Images



Contrastive Learning



Contrastive Learning

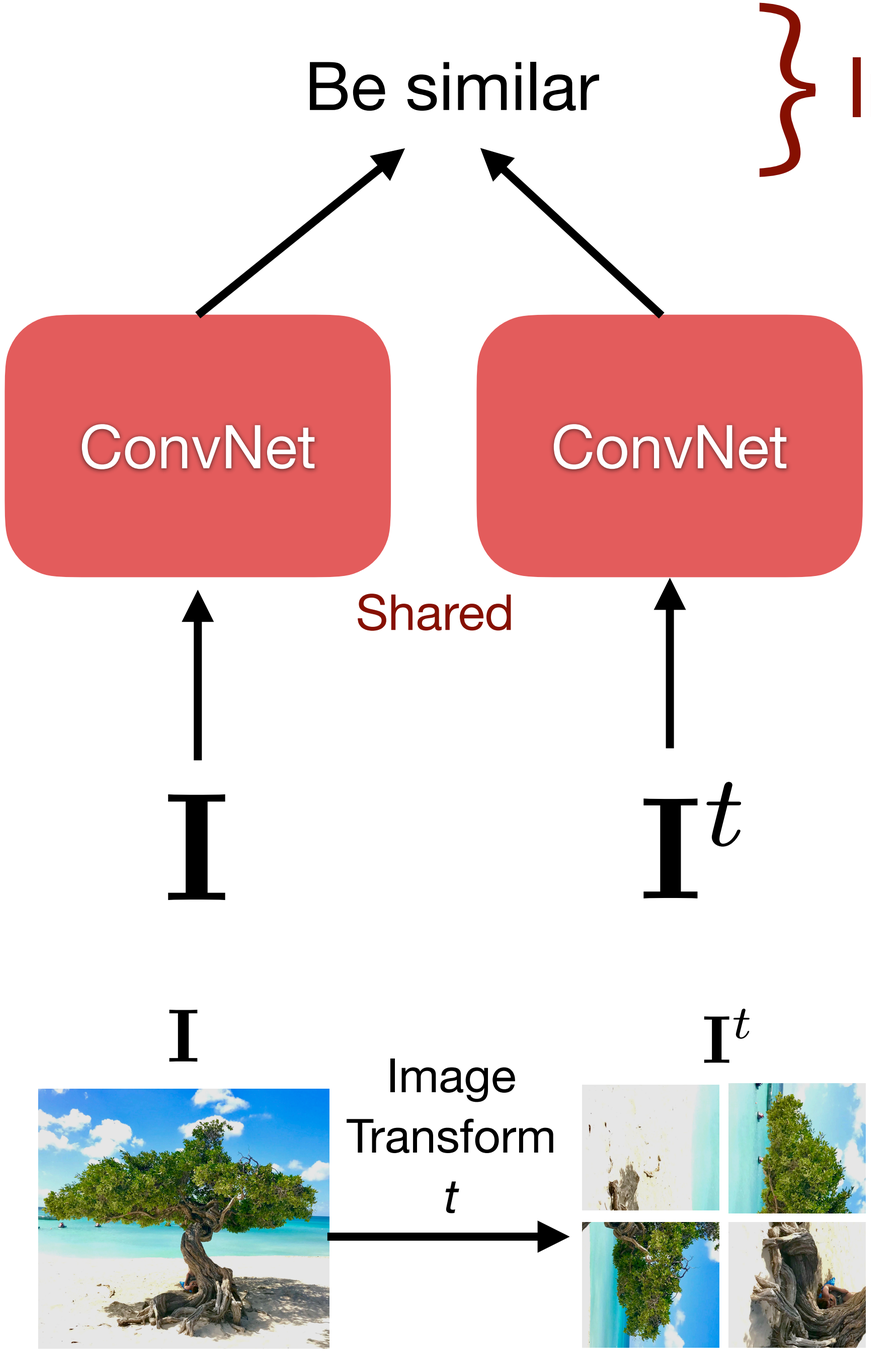


Loss Function

Embeddings from related images should be closer than embeddings from unrelated images

Hadsell et al., 2005, DrLim

PIRL



Invariant to Pretext transform

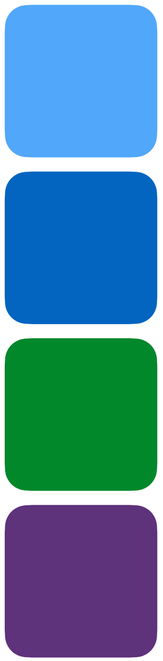
$$L_{\text{contrastive}}(\mathbf{v}_{\mathbf{I}}, \mathbf{v}_{\mathbf{I}^t})$$

Invariance to

- Data Augmentations
- Multiple views created by pretext task (Jigsaw/Rotation)

Contrastive Learning in PIRL

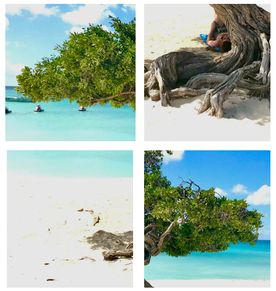
Dataset



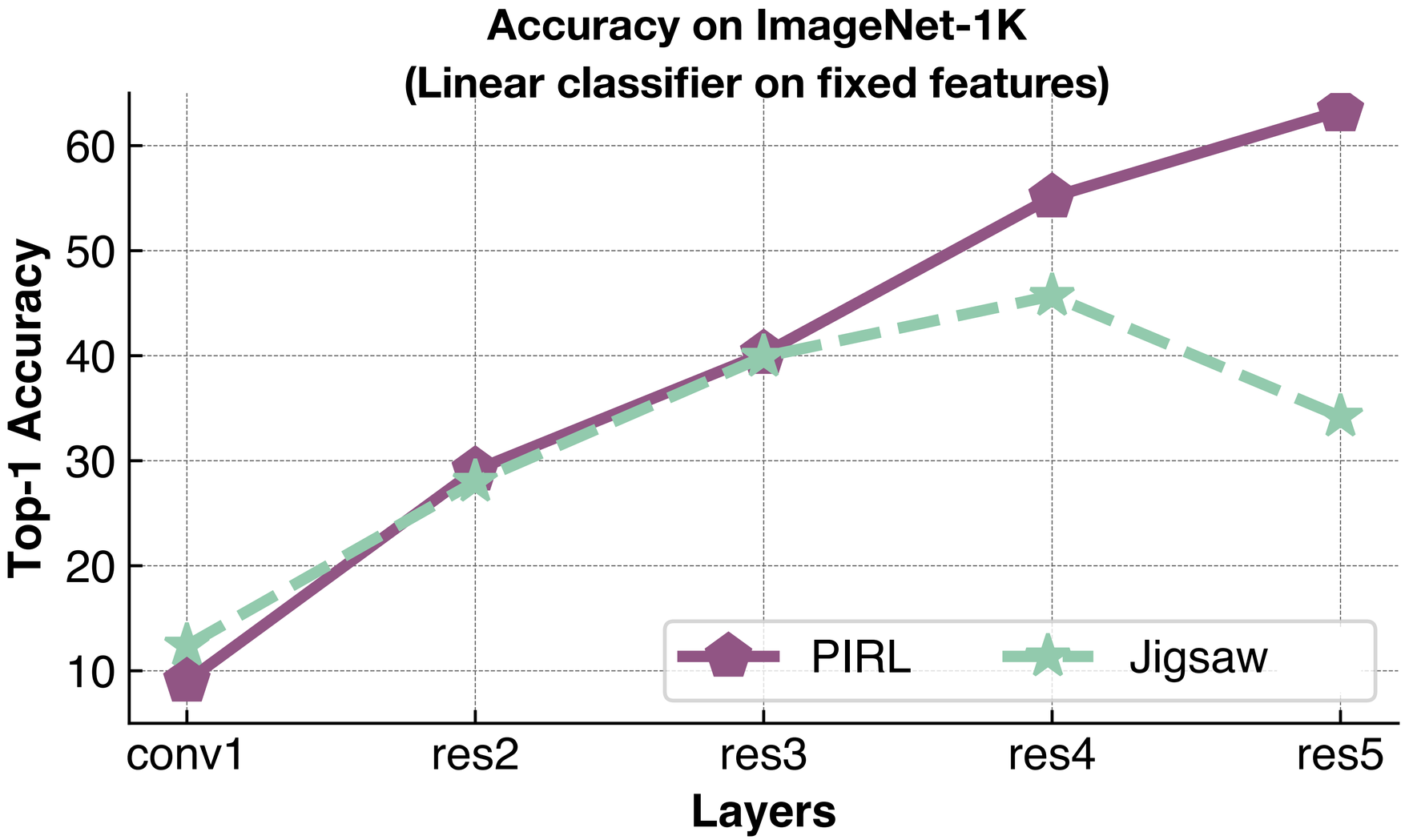
Loss Function

$\mathbf{I}$

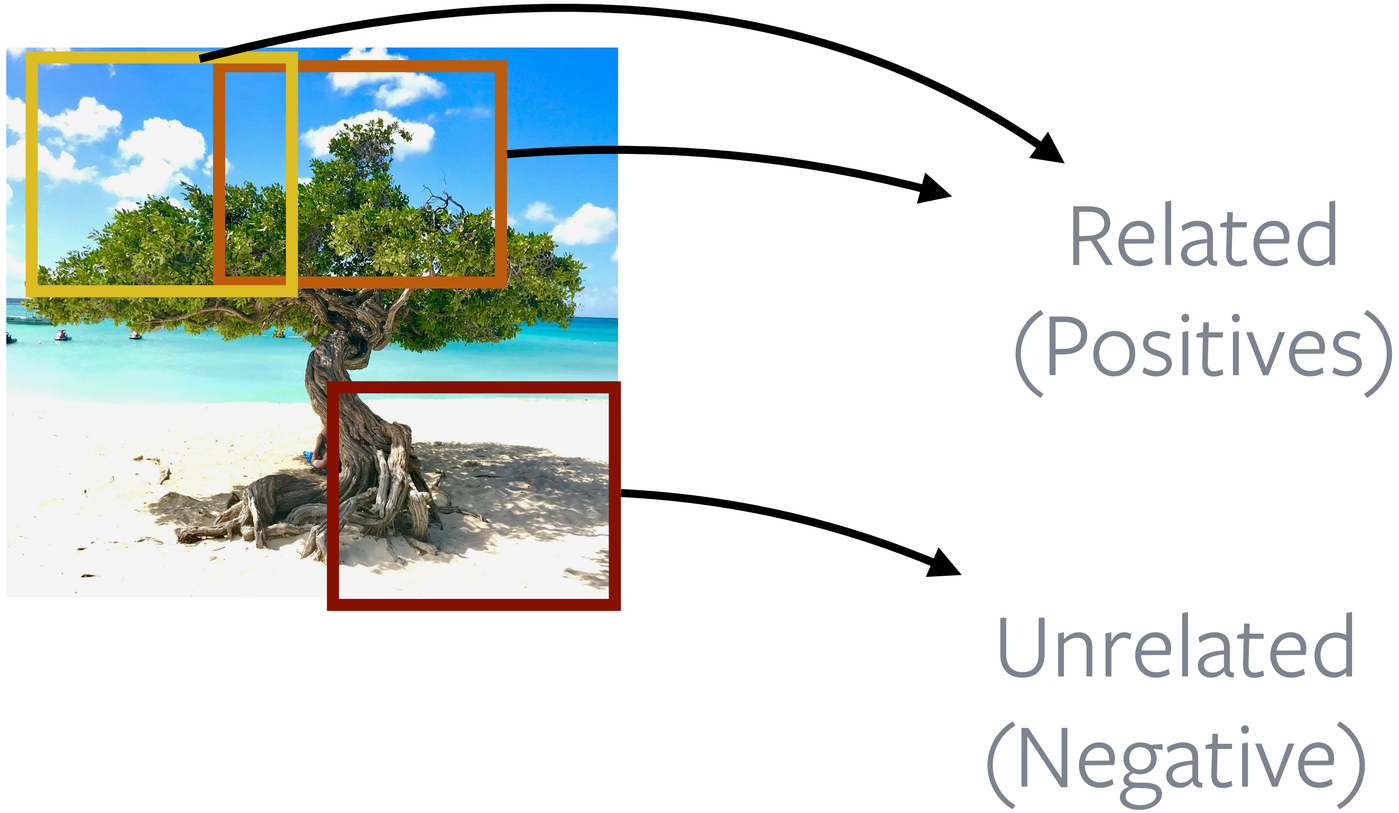
$\mathbf{I}^t$



Semantic Features?

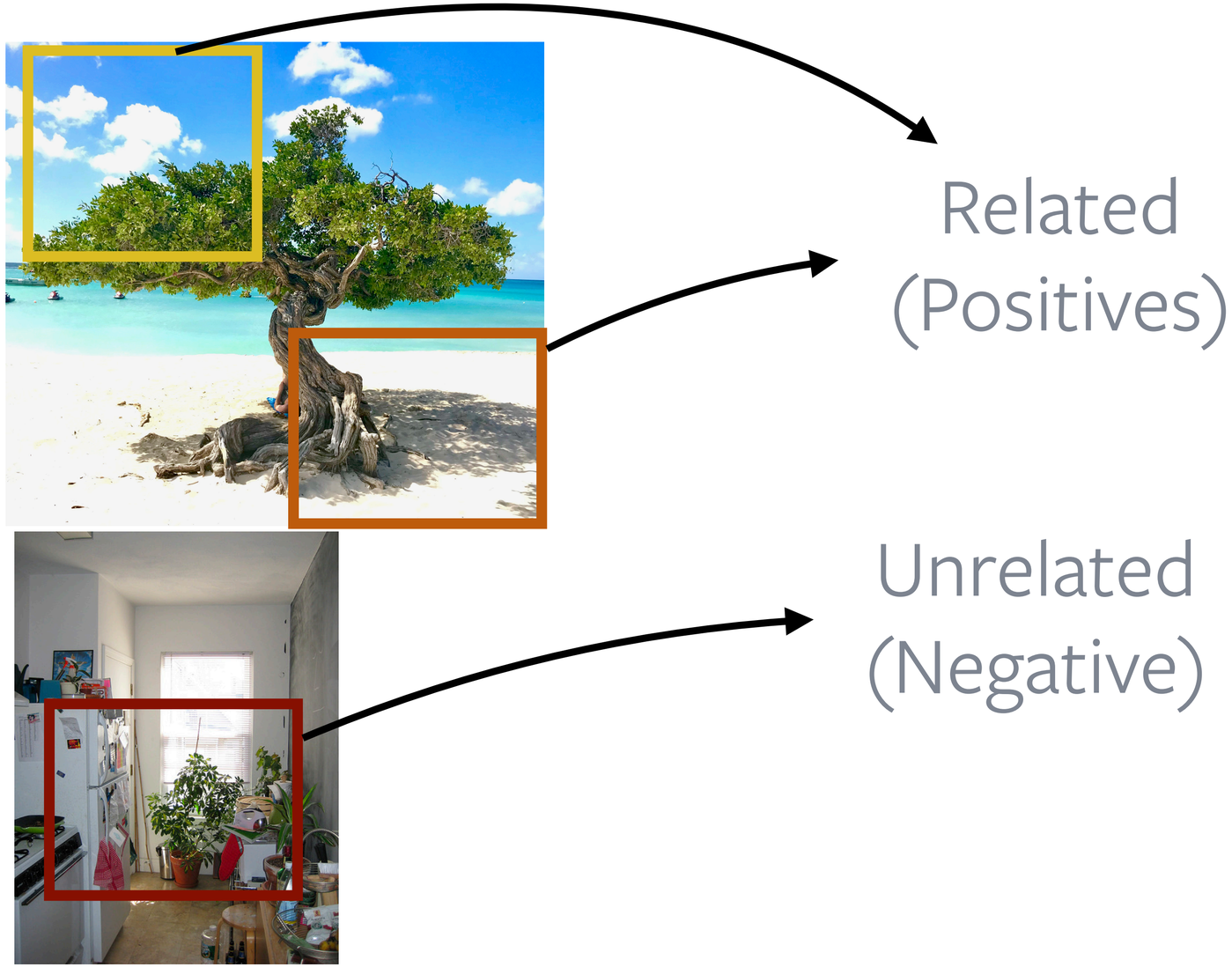


Nearby patches vs. distant patches of an Image



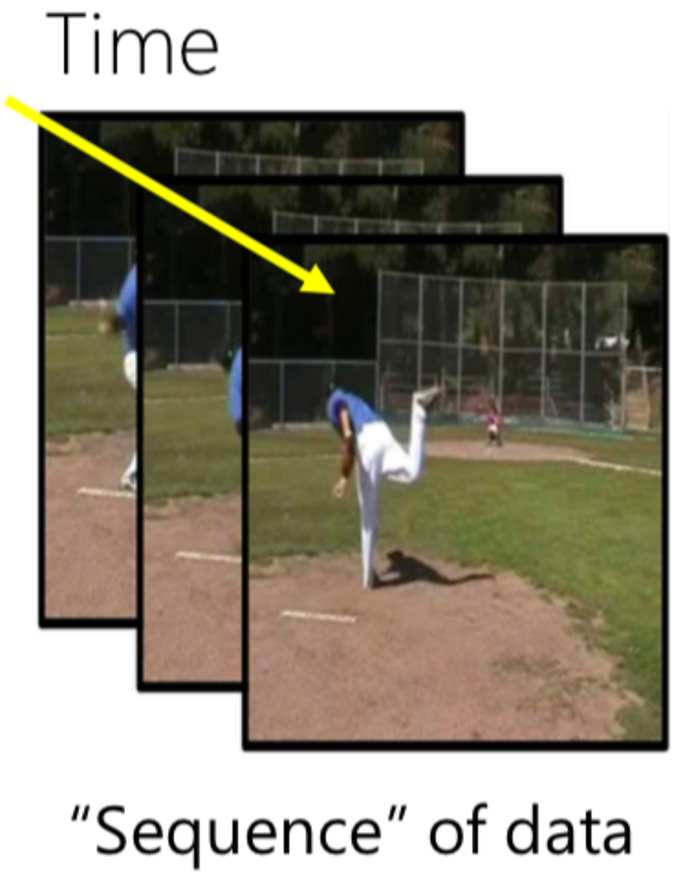
van der Oord et al., 2018,
Henaff et al., 2019
Contrastive Predictive Coding

Patches of an image vs. patches of other images



Wu et al., 2018, Instance Discrimination
He et al., 2019, MoCo
Misra & van der Maaten, 2019, PIRL
Chen et al., 2020, SimCLR

Frames of a video

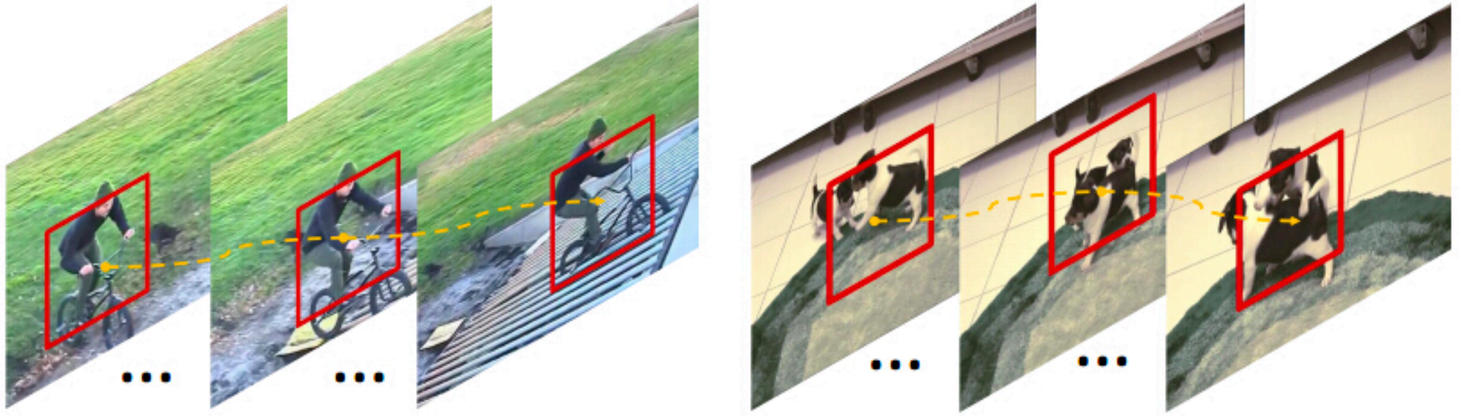


Hadsell et al., 2005, DrLim
van der Oord et al., 2018, CPC

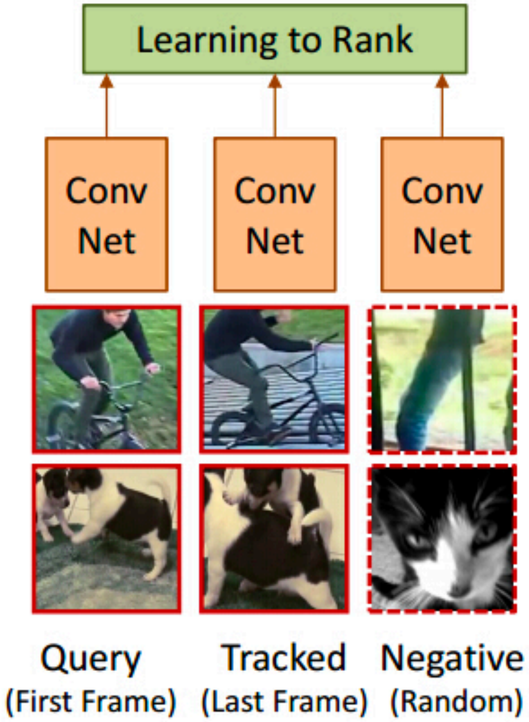
Video & Audio

AVID+CMA - Morgado et al., 2020
GDT - Patrick et al., 2020

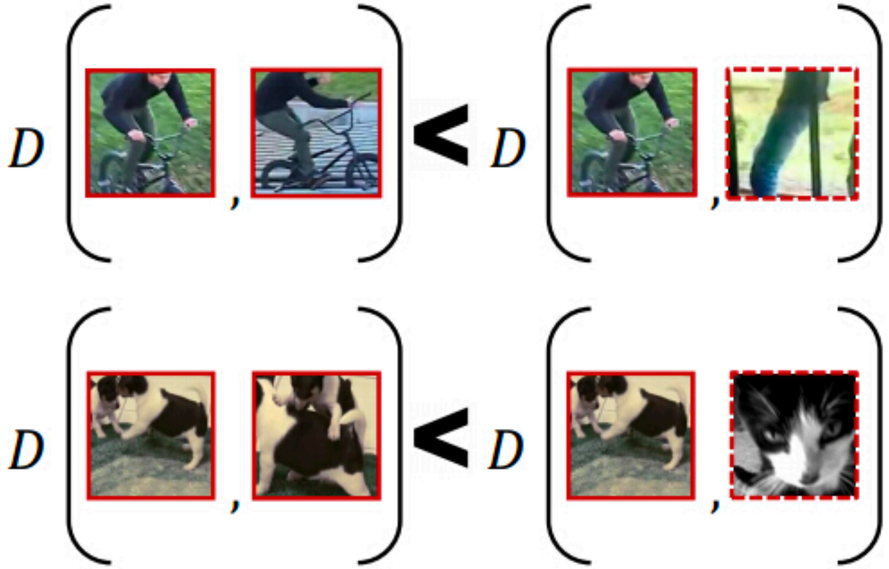
Tracking Objects



(a) Unsupervised Tracking in Videos



(b) Siamese-triplet Network



D: Distance in deep feature space

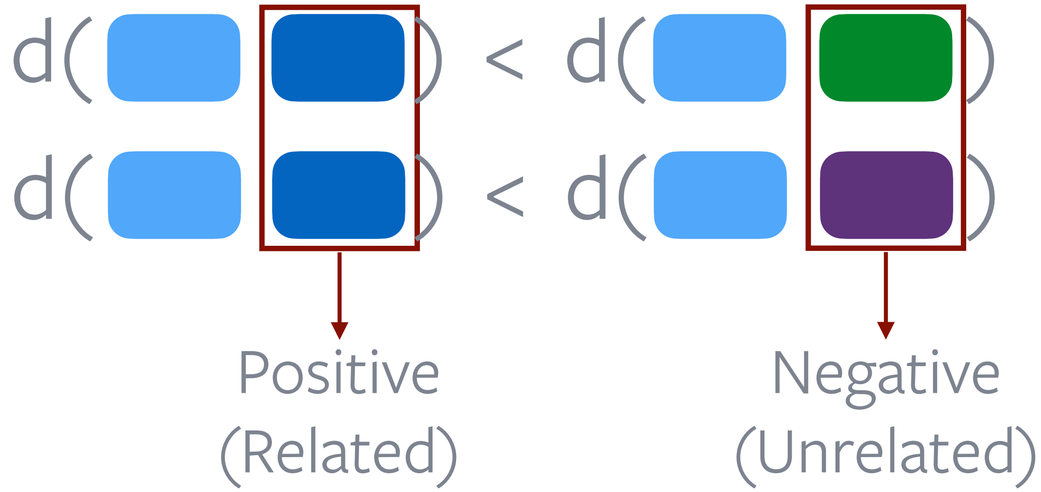
(c) Ranking Objective

Wang & Gupta, 2015, Unsupervised Learning of Visual Representations using Videos

Good negatives are necessary

Loss Function

Embeddings from related images should be closer than embeddings from unrelated images

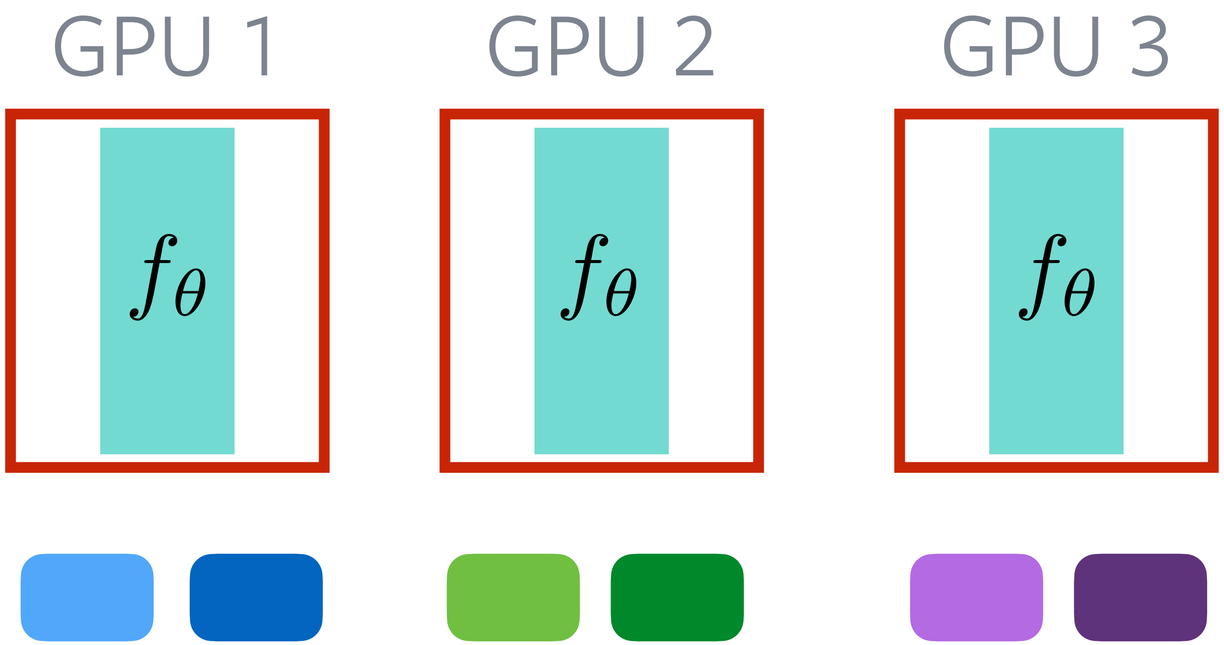


Good negatives are *very* important in contrastive learning

Hadsell et al., 2005, DrLim

SimCLR

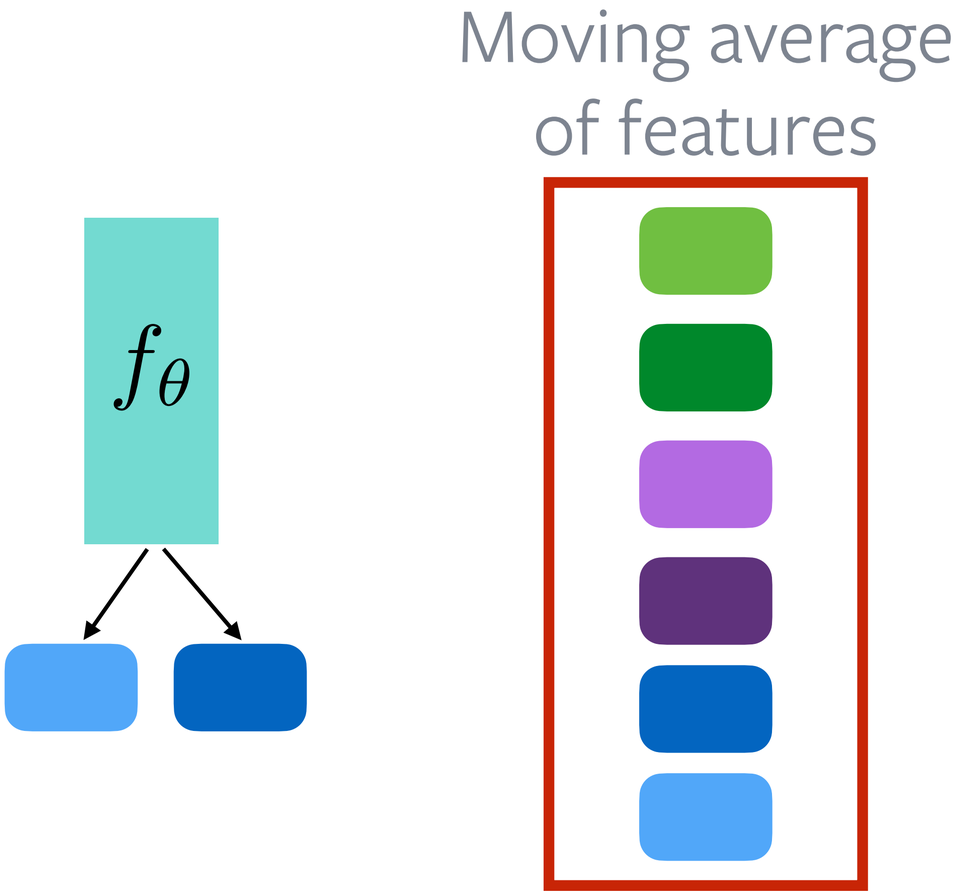
- Large batch size - e.g. in SimCLR
- Pros - Simple to implement
- Cons - Large batch size



A simple framework for contrastive learning - Chen et al., 2020

Memory Bank

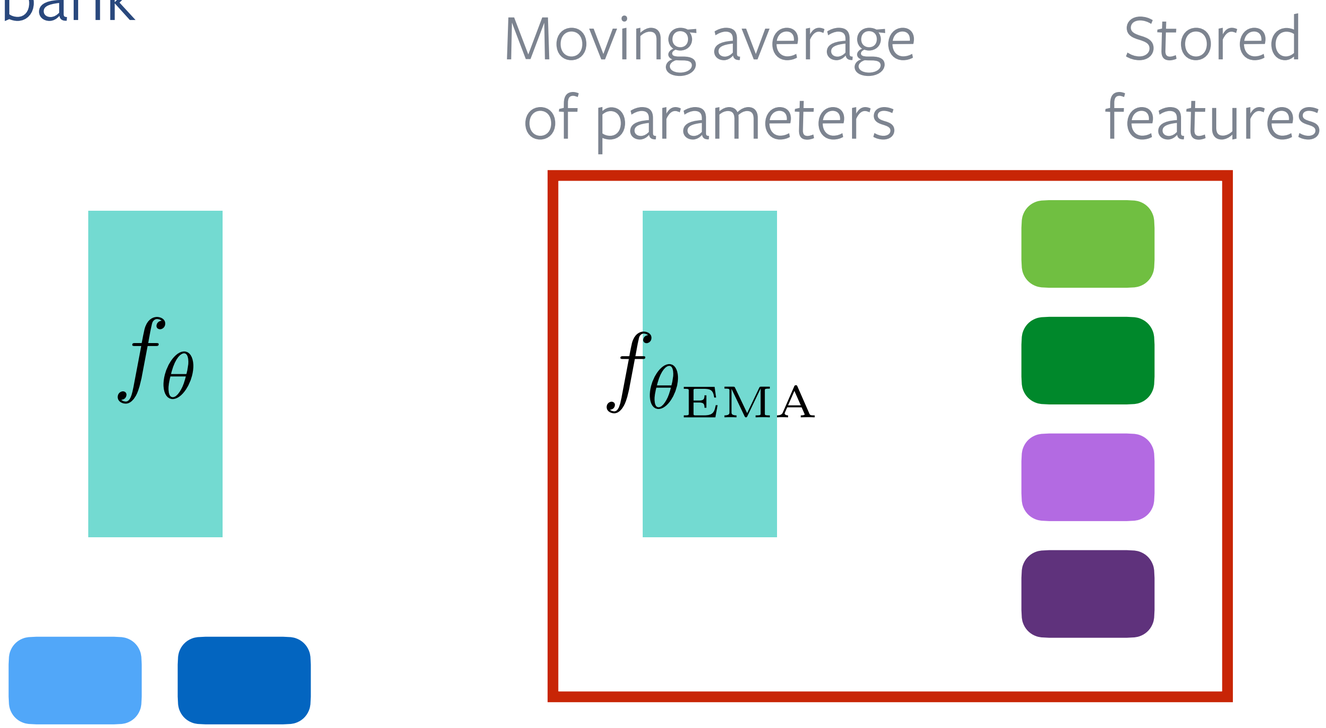
- Maintain a "memory bank" -- momentum of activations
- Pros - compute efficient
- Cons - Needs large memory, not "online"



Non parametric Instance Discrimination - Wu et al., 2018

MoCo

- Maintain "momentum" network - MoCo
- Pros - online
- Cons - extra memory for parameters/stored features, extra fwd pass compared to memory bank



Momentum Contrast - He et al., 2019

Many ways to avoid trivial solutions

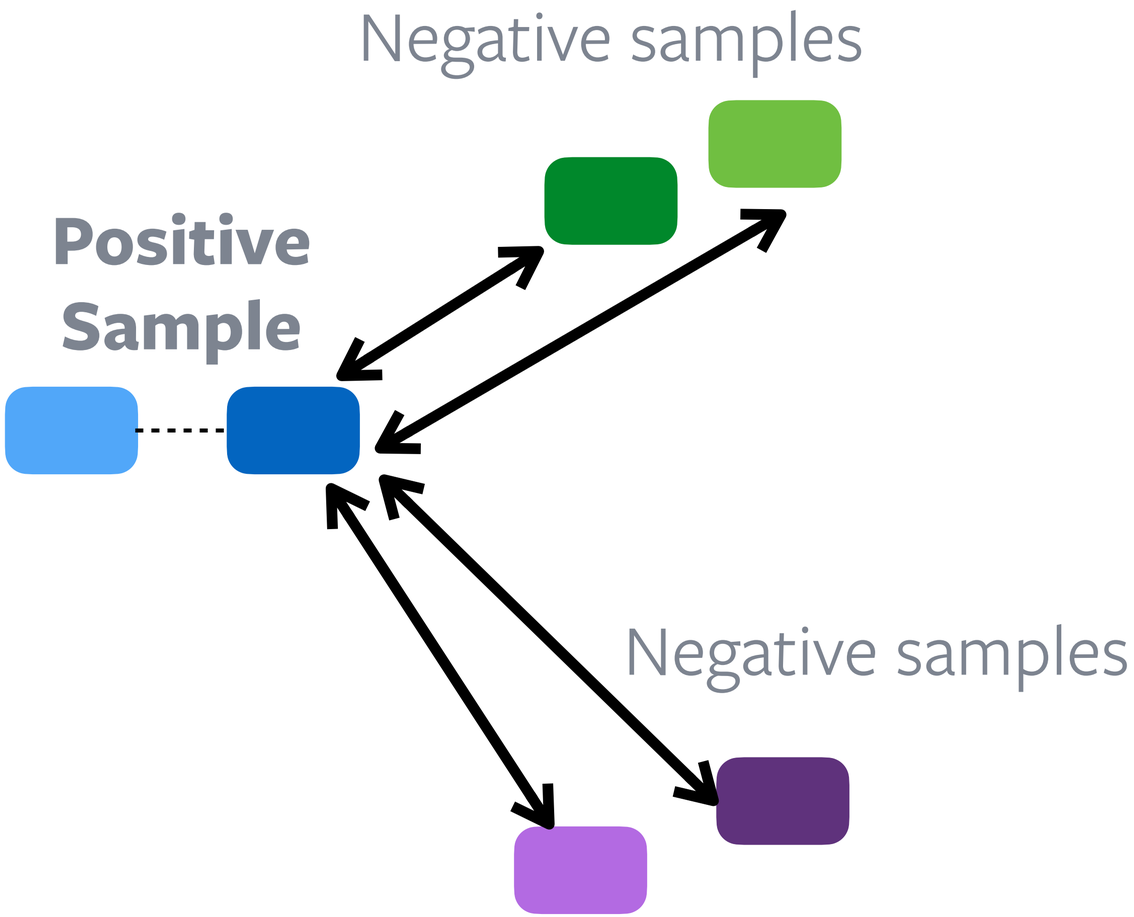
Similarity Maximization Objective

- Contrastive learning
  - MoCo, PIRL, SimCLR
- Clustering
  - DeepCluster, SeLA, SwAV
- Distillation
  - BYOL, SimSiam

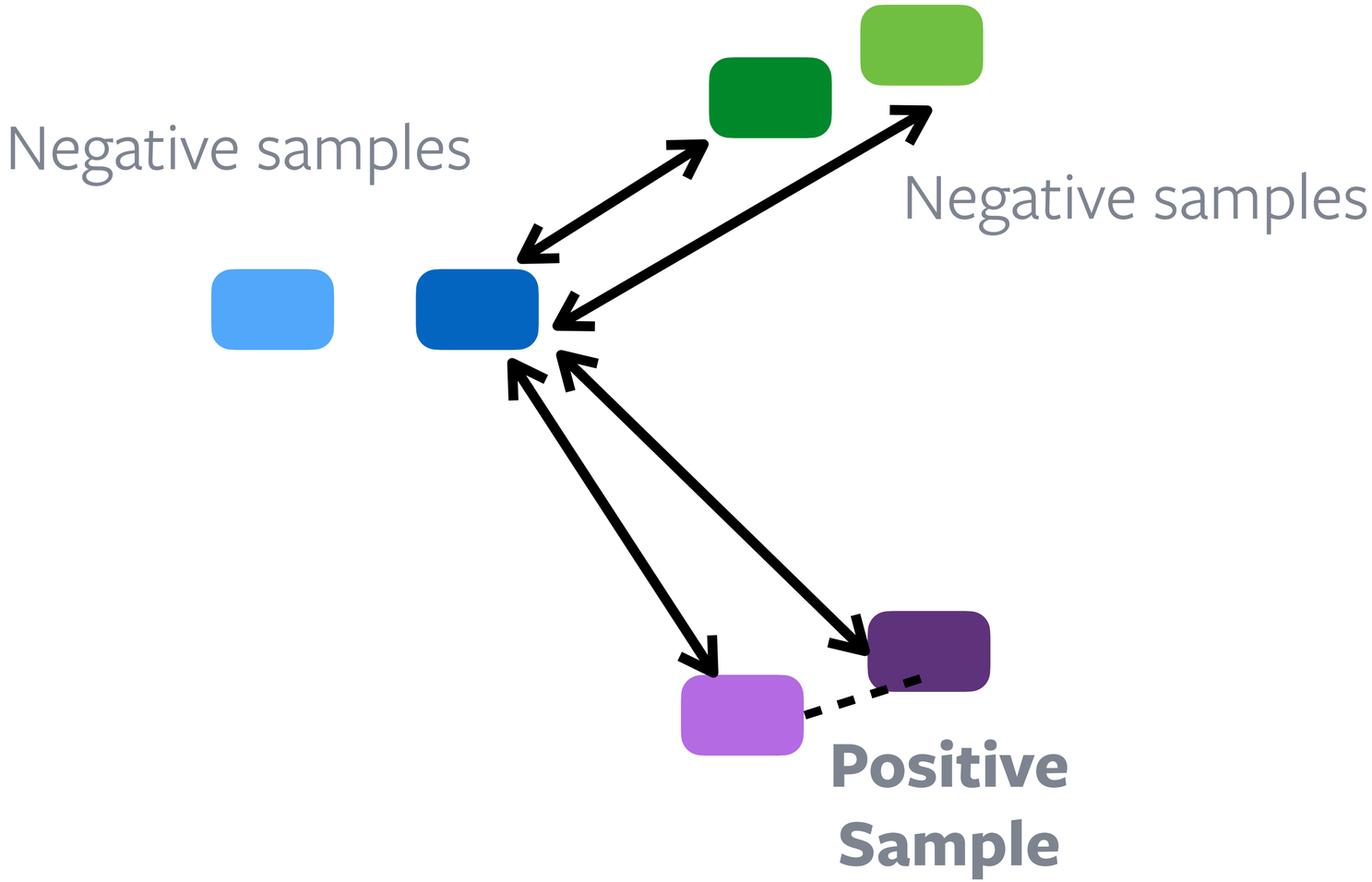
Redundancy Reduction Objective

- Redundancy Reduction
  - Barlow Twins

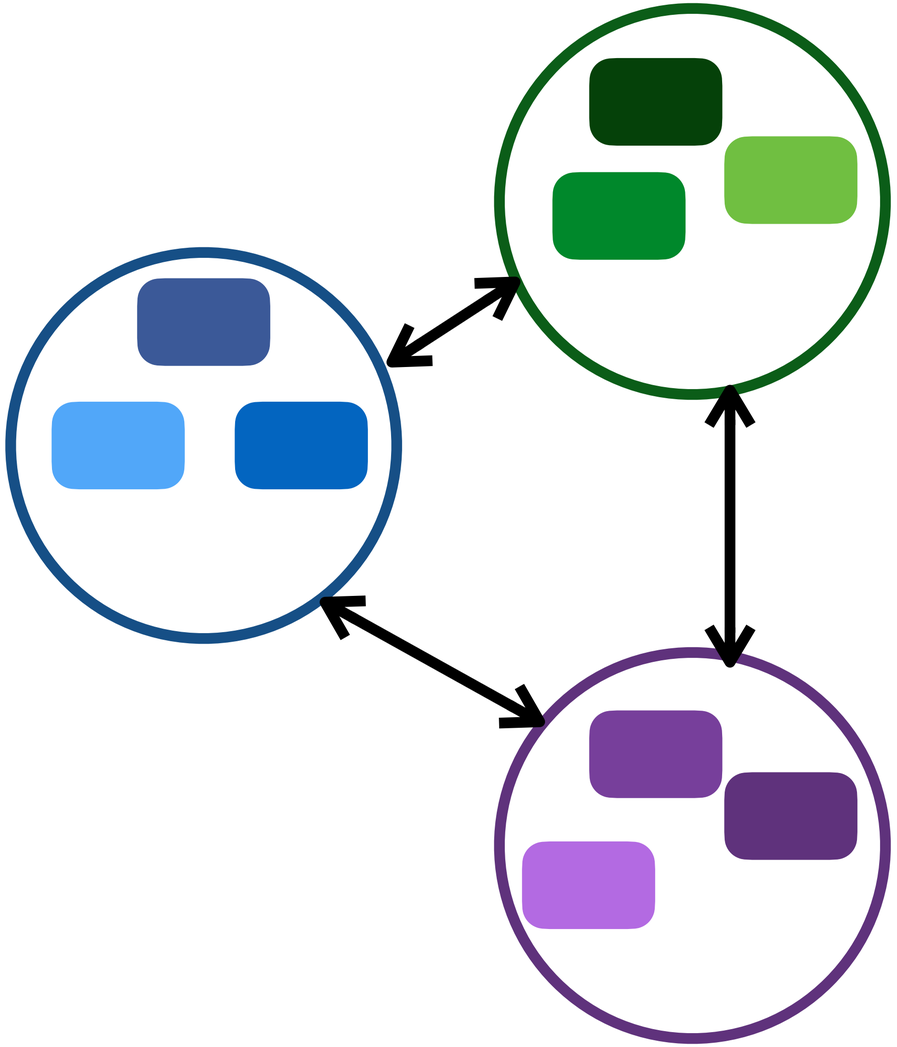
Contrastive learning -- what does it do?



Contrastive learning -- what does it do?

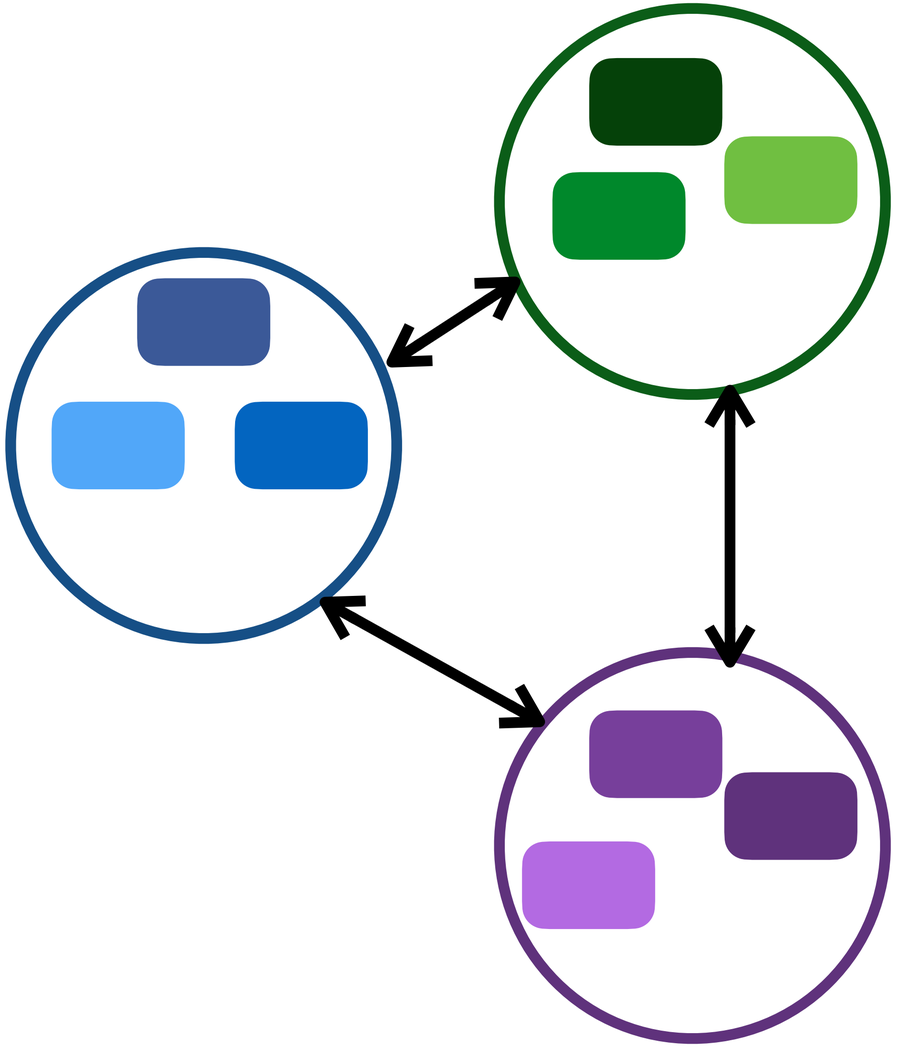


Contrastive Learning => Groups in feature space



Creates groups
in the feature space

Clustering creates groups too



Creates groups
in the feature space

So does **clustering**?!

Swapping Assignments between Views (SwAV)

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, Armand Joulin

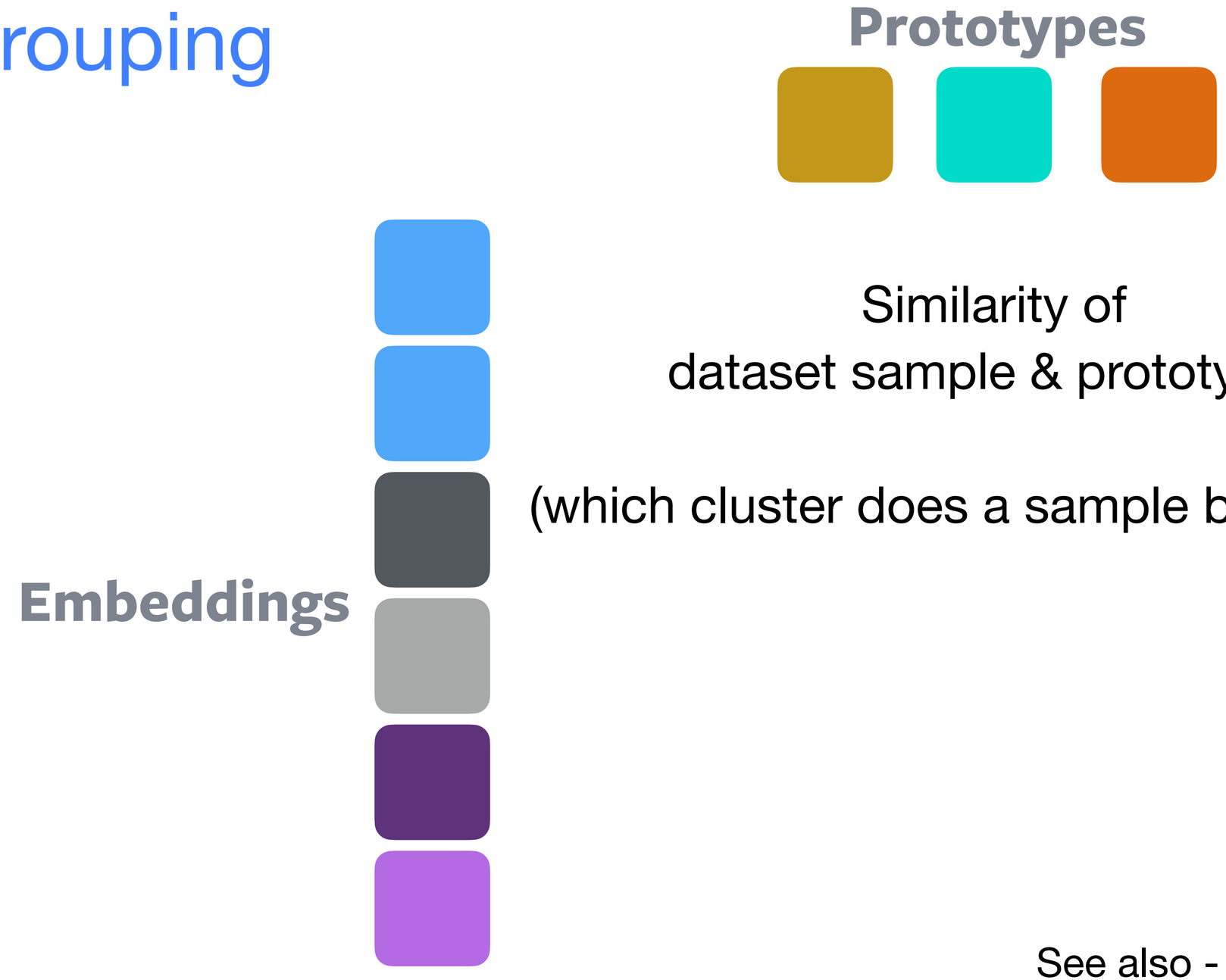
Key Idea

- What we want $$f_\theta(I) = f_\theta(\text{augment}(I))$$

- How we do it - I and augment(I) must belong to the same "group" or cluster

- Prevent trivial solutions by controlling the clustering process

Grouping

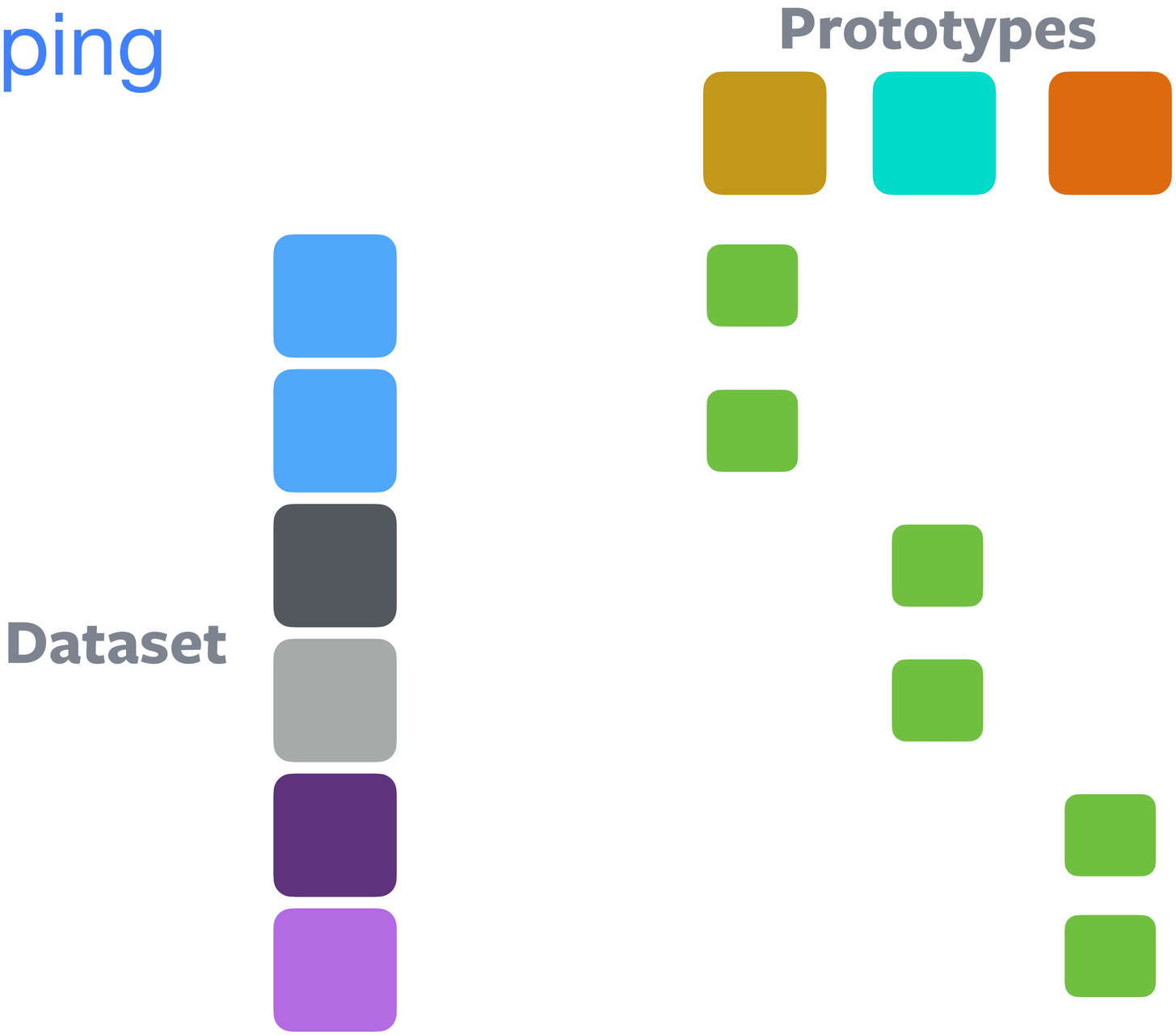


Similarity of
dataset sample & prototypes

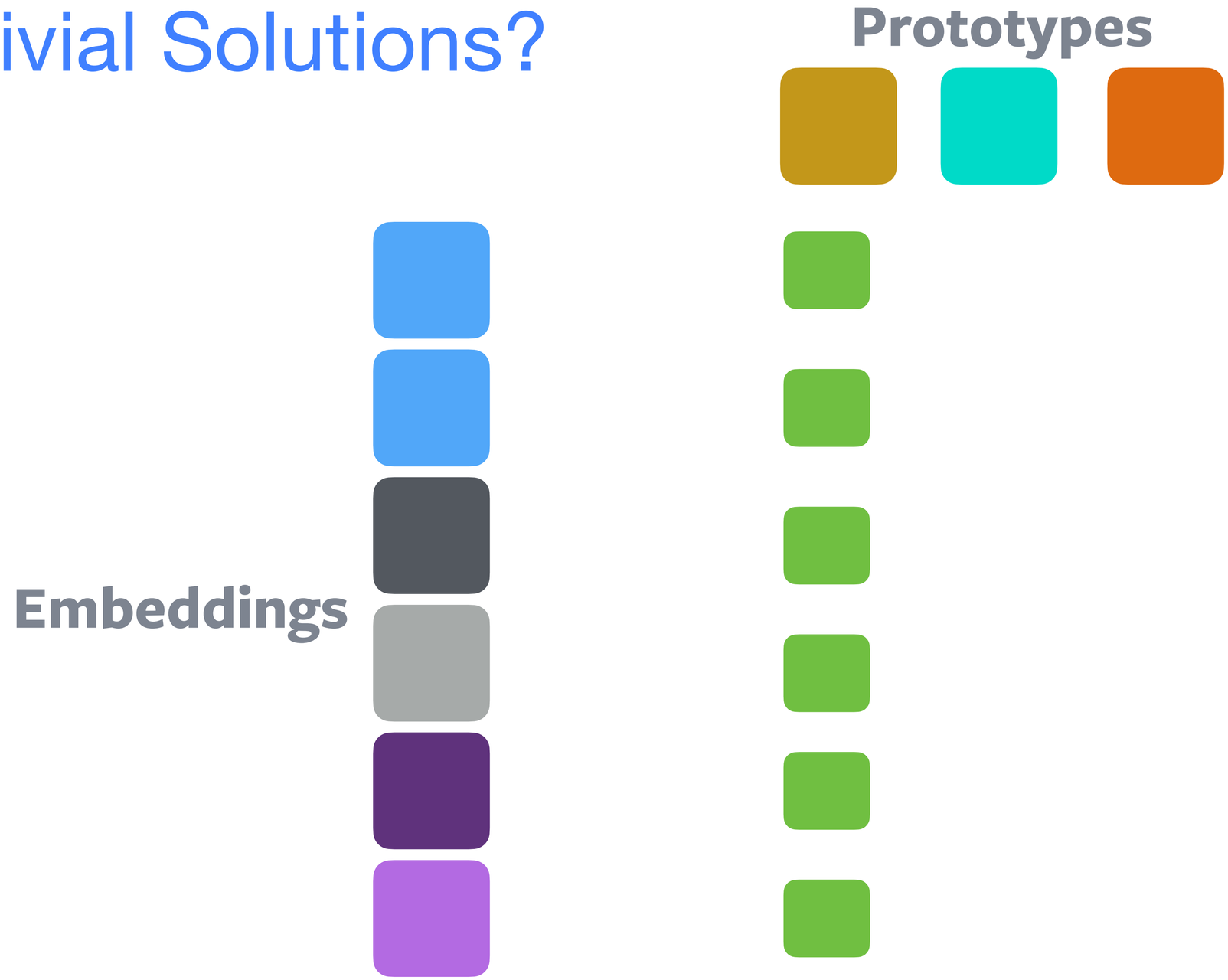
(which cluster does a sample belong to?)

See also - *SeLa* by Asano et al., 2019

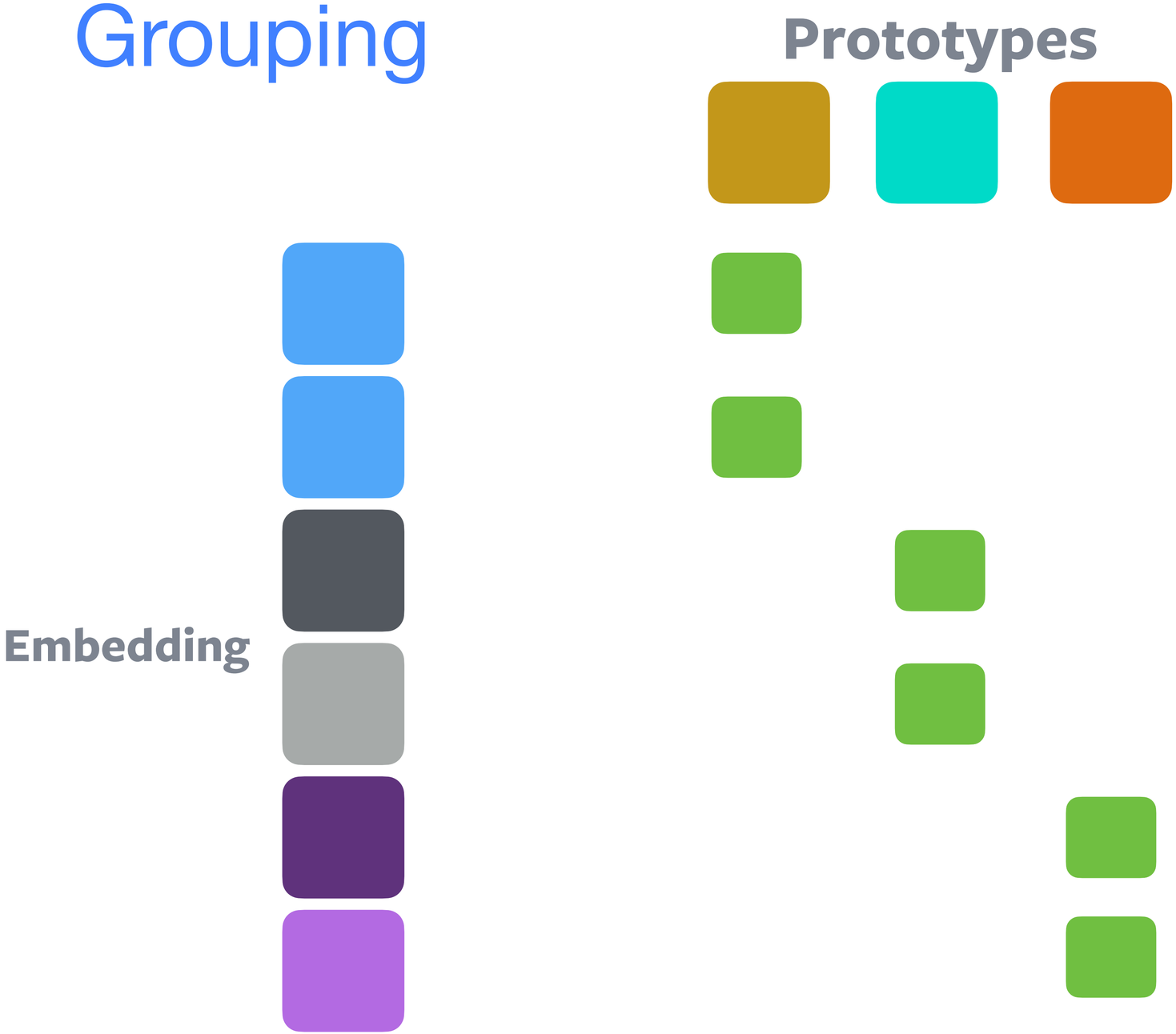
Grouping



Trivial Solutions?



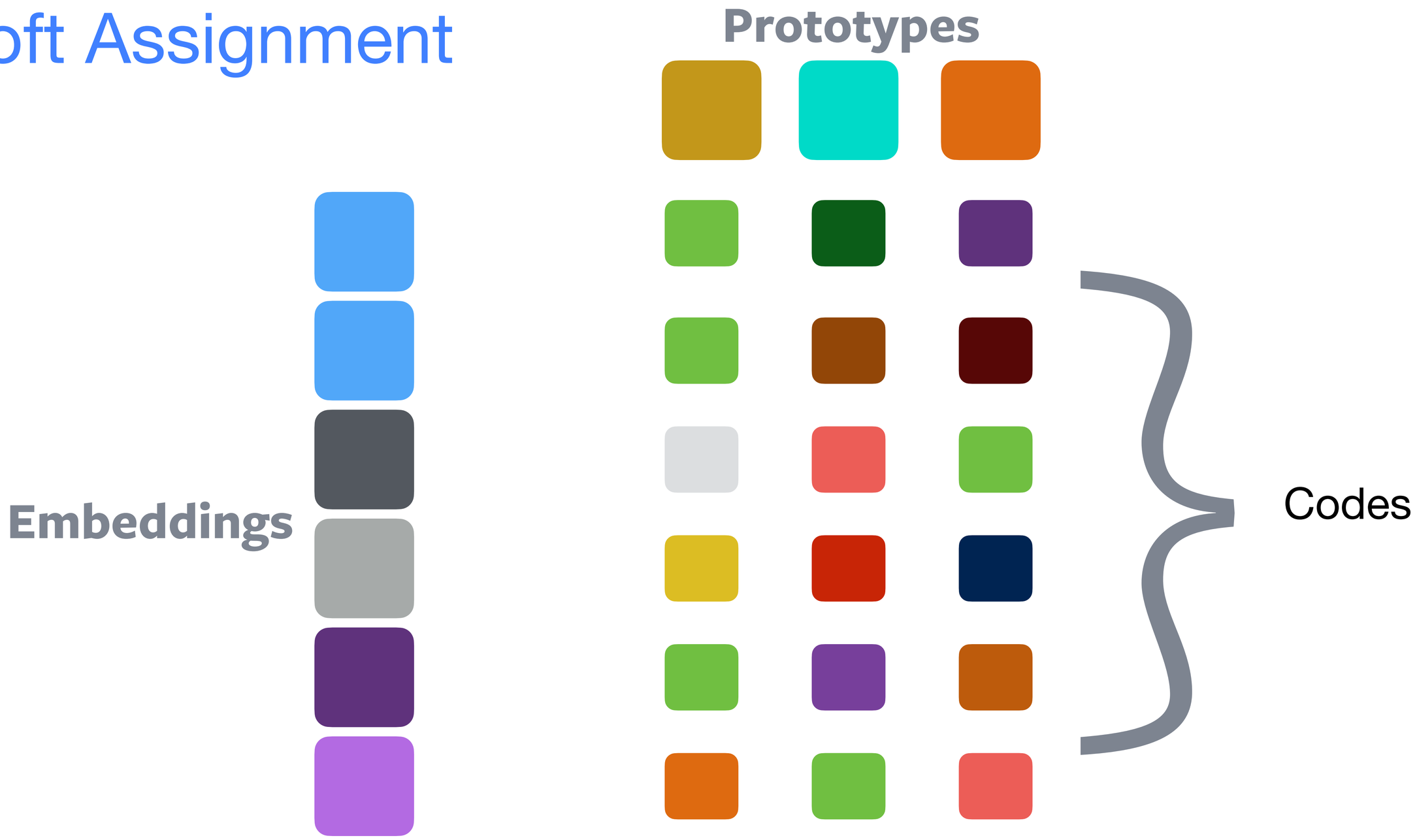
Grouping

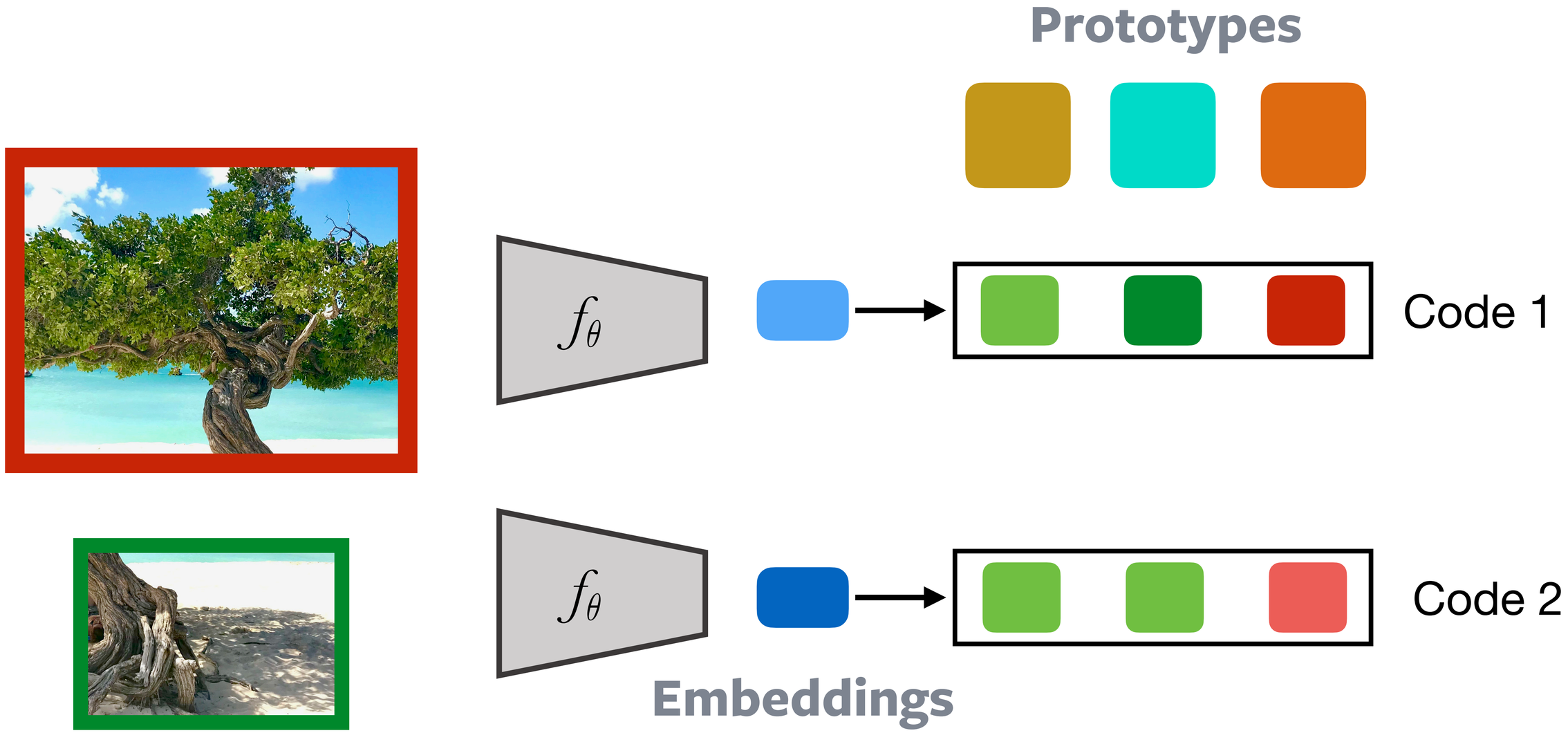


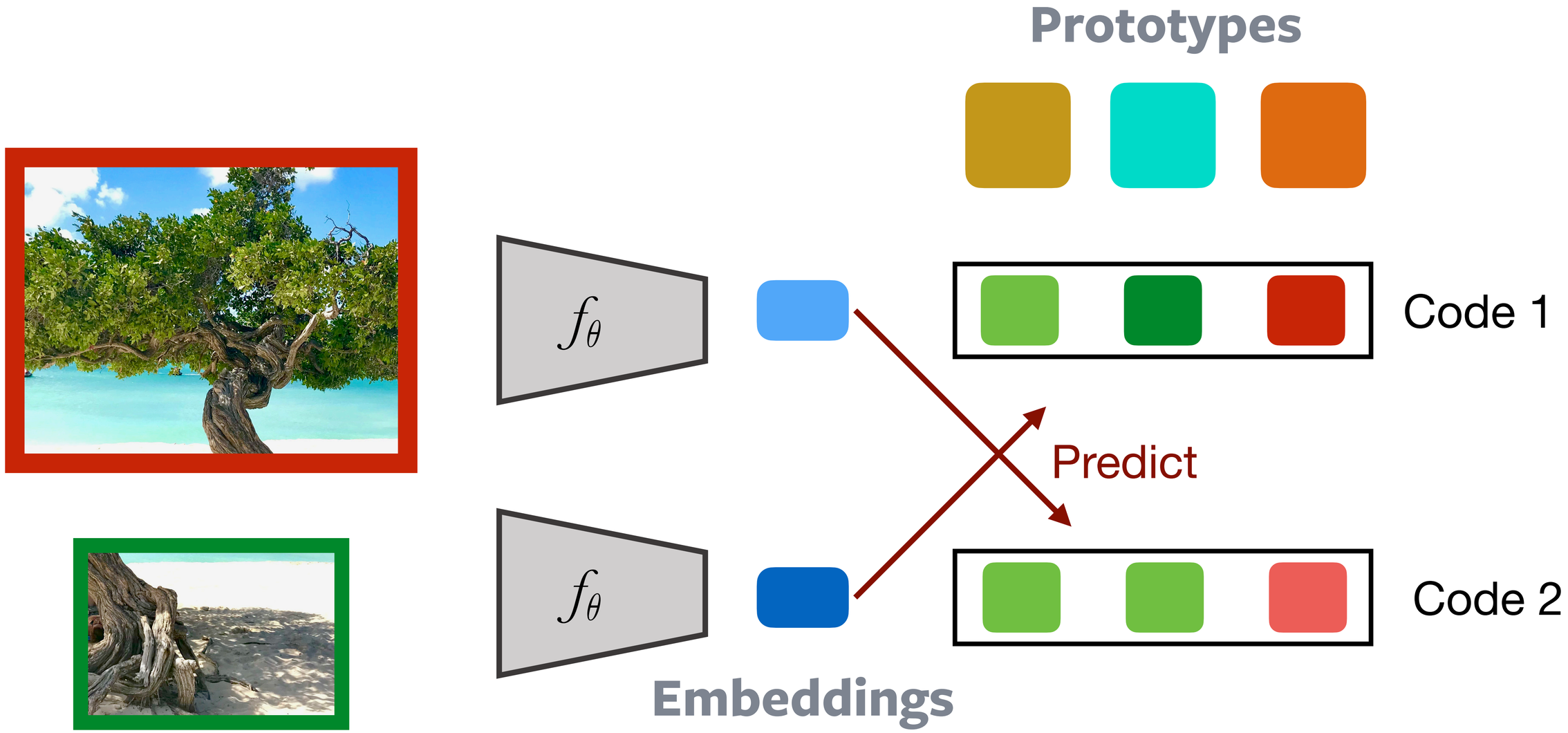
Equipartition constraint --
Given N samples and K prototypes,
each prototype is most similar to N/K samples

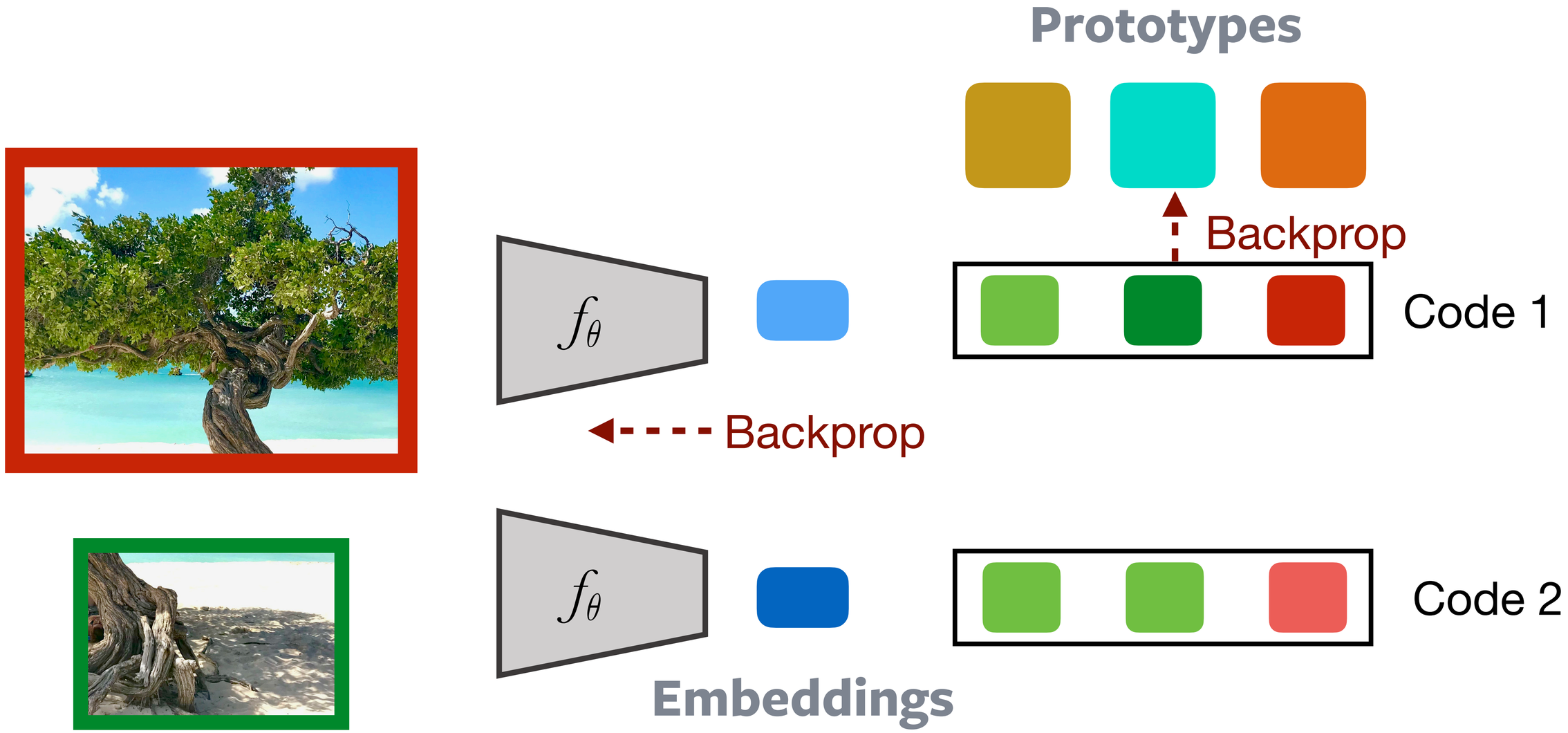
Implemented using
Optimal Transport (Sinkhorn-Knopp)

Soft Assignment




Prototypes
f_θ
Code 1
f_θ
Code 2
Embeddings


Prototypes
f_θ
f_θ
Code 1
Predict
Code 2
Embeddings



Not contrastive!

Key Results

	Linear Classifier (Fixed Features)			Detection (Fine-tuned)	
	ImageNet	**Places**	**iNaturalist**	**VOC07+12**	**COCO**
Supervised	76.5	53.2	46.7	81.3	40.8
Prior self-supervised	71.1 (-5.4)	52.1	38.9	82.5	42.0
SwAV	75.3 (-1.2)	**56.7**	**48.6**	**82.6**	**42.1**

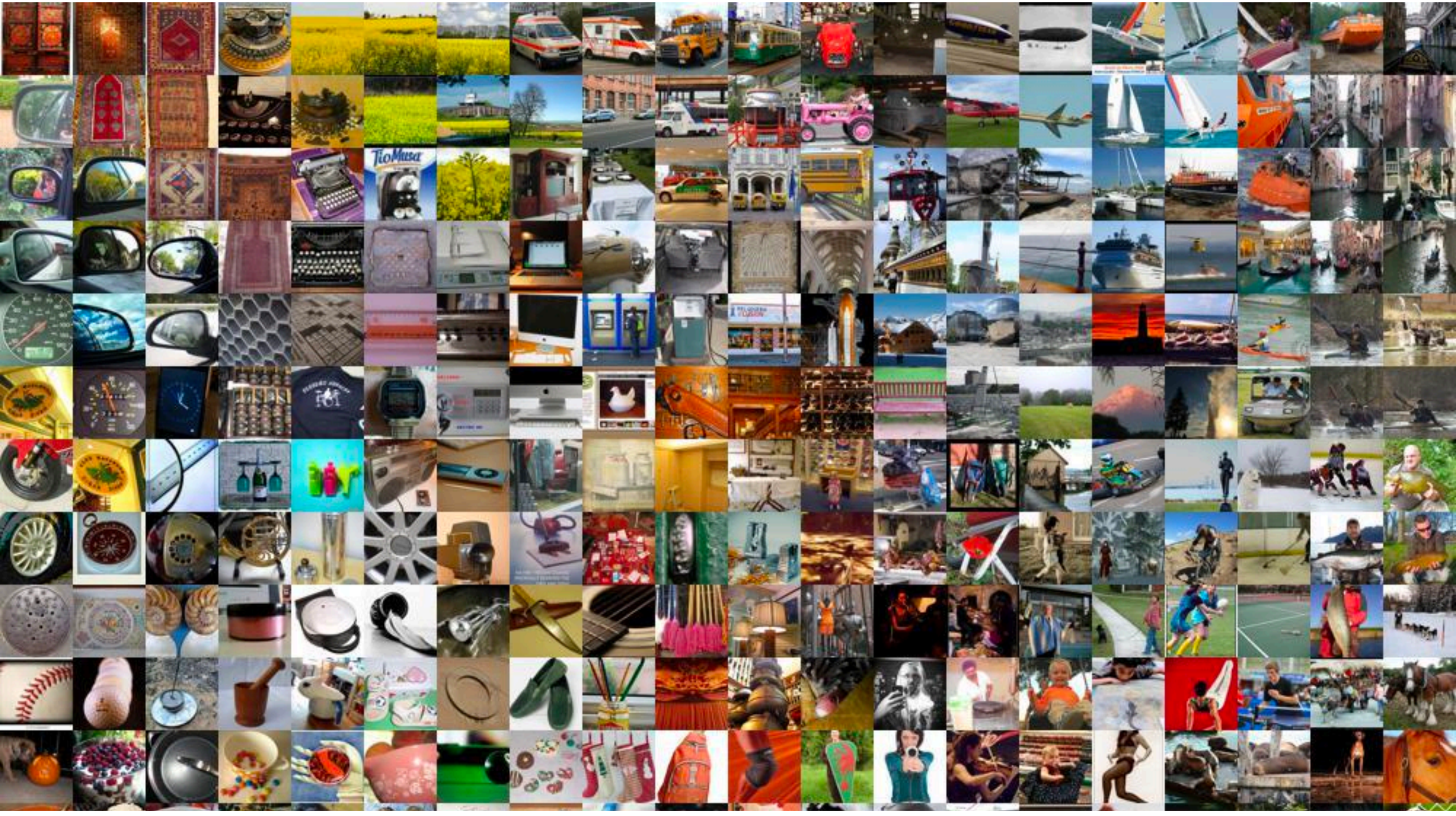
Practical advantages of SwAV

- Trains on 4-8 GPUs
- **Faster convergence** than prior work (SimCLR, MoCov2)
  - Smaller compute requirements.
  - **2x faster** than MoCo-v2 on 8 GPUs
    - 72% after 100h vs. 71% after 200h

- Better results

Code & Models - https://github.com/facebookresearch/swav

Pretrained on ImageNet without labels


TioMusa

Curated pretraining data

ImageNet data is "curated"

- All images belong to 1000 classes
- All images contain a prominent object
- Very limited clutter

Curated pretraining data

Pretraining on non-ImageNet data hurts performance

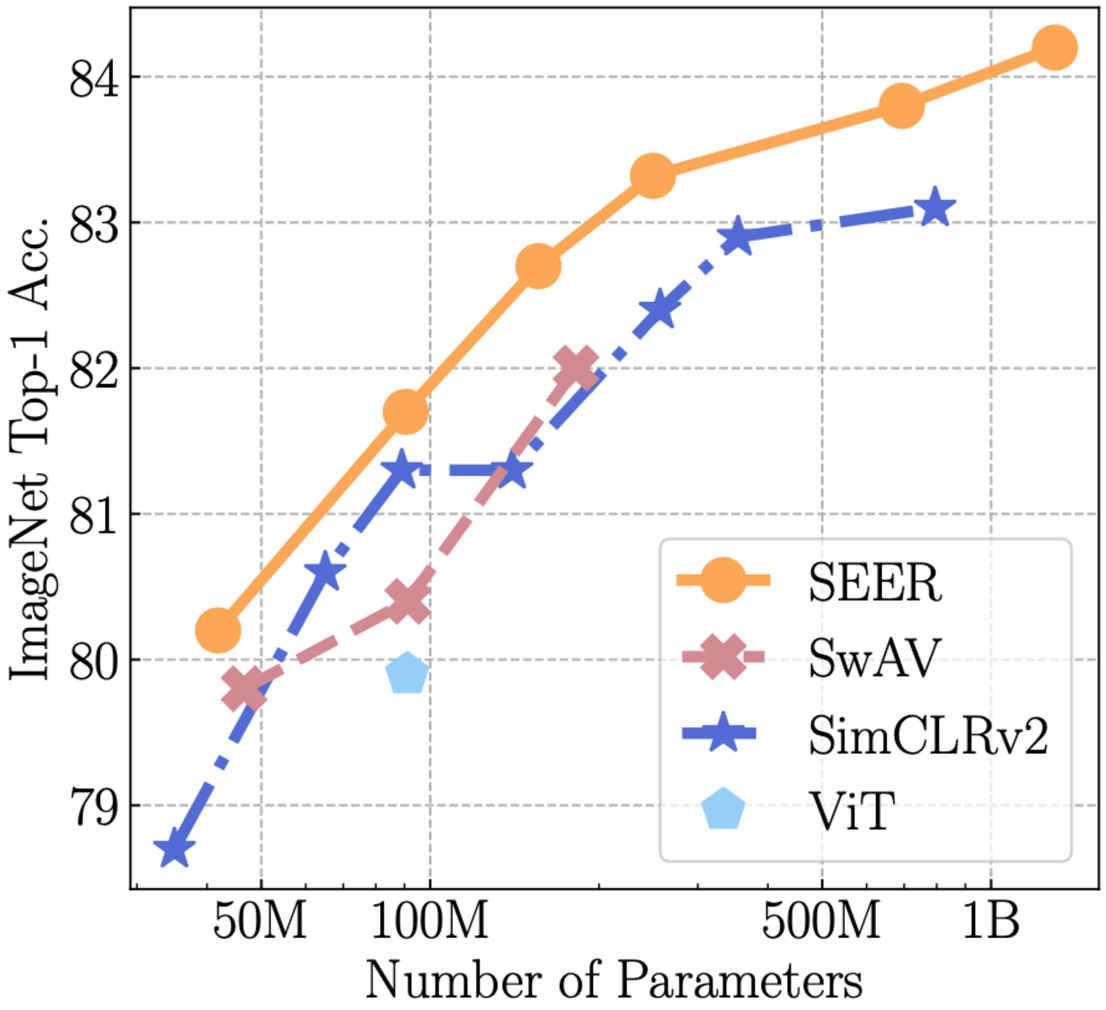
Demystifying Contrastive Self-Supervised Learning: Invariances, Augmentations and Dataset Biases - Purushwalkam et al., 2020

Real world data

Images have

- different distributions (cartoon images, memes)
- no single prominent object

SEER: Learning from uncurated images



Train on 1.3 billion random images

Images are NOT filtered in any way

RegNet trained on 1.3B random internet images

ResNet trained on ImageNet

ResNet (modified) trained on ImageNet

Vision Transformer trained on ImageNet

SEER - Goyal et al., 2021

SEER: Learning from uncurated images

Method	Pretrain images	Curated	Arch.	Params	ImageNet top-1
Hashtag prediction	1B	Yes	X101-32x8d	91M	82.6
SEER	1B	No	X101-32x8d	91M	81.6

SEER uses no labels and works on random images

Comparable performance to networks trained on curated data with weak supervision

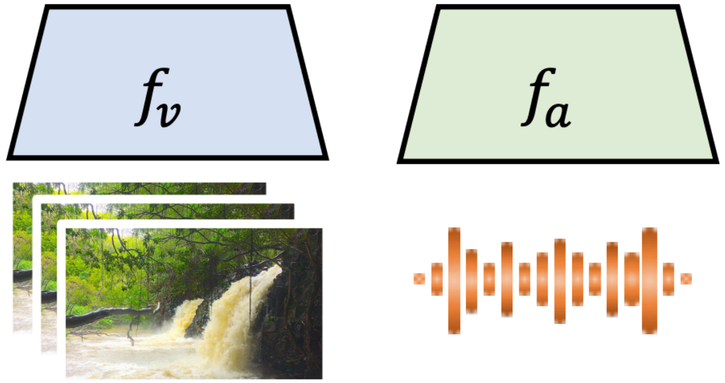
Architecture: ResNeXt-101-32x8d

Audio Visual Instance Discrimination with Cross Modal Agreement (AVID + CMA)

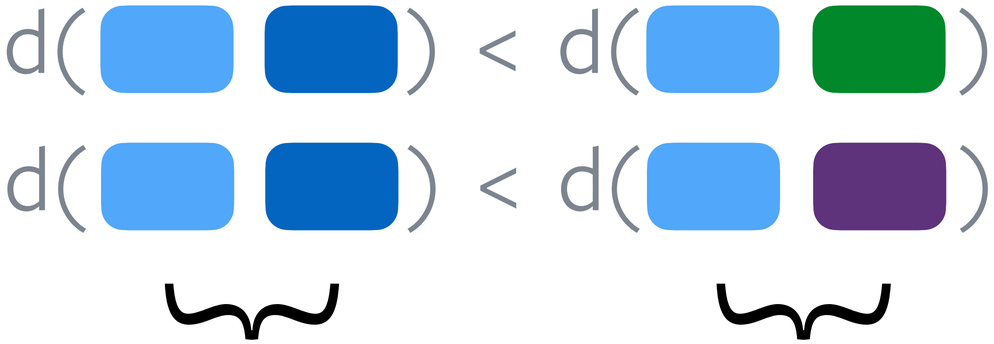
Pedro Morgado, Nuno Vasconcelos, Ishan Misra

https://github.com/facebookresearch/AVID-CMA

Contrastive (Audio Video Instance Discrimination)



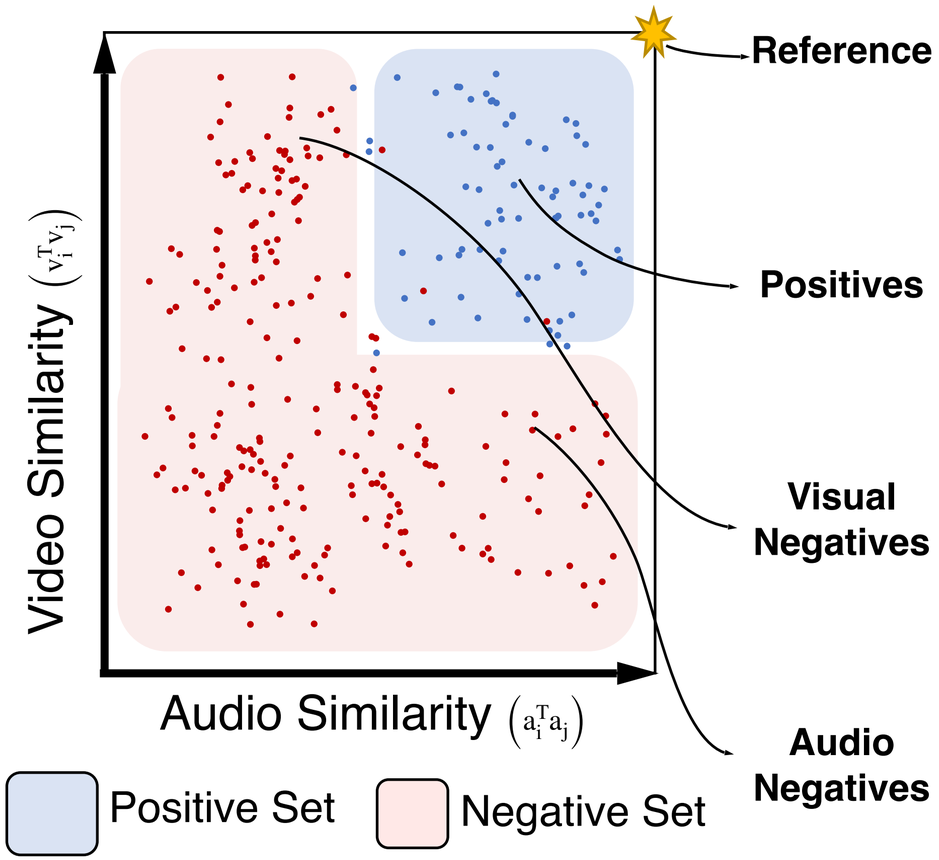
Positives **Negatives**

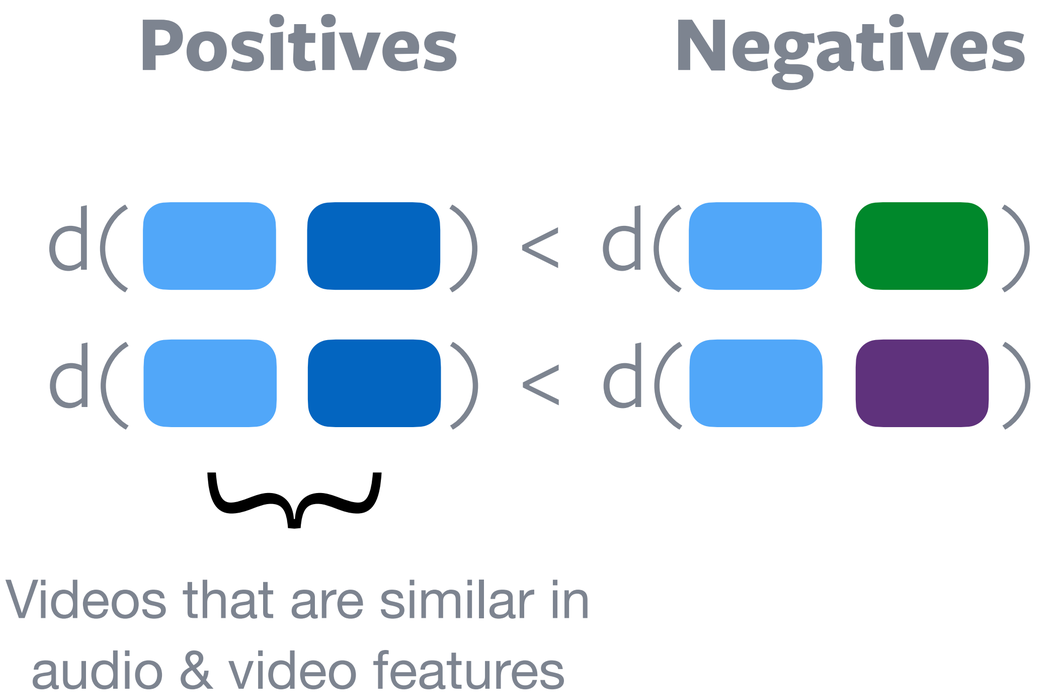


Audio & Video (same sample)

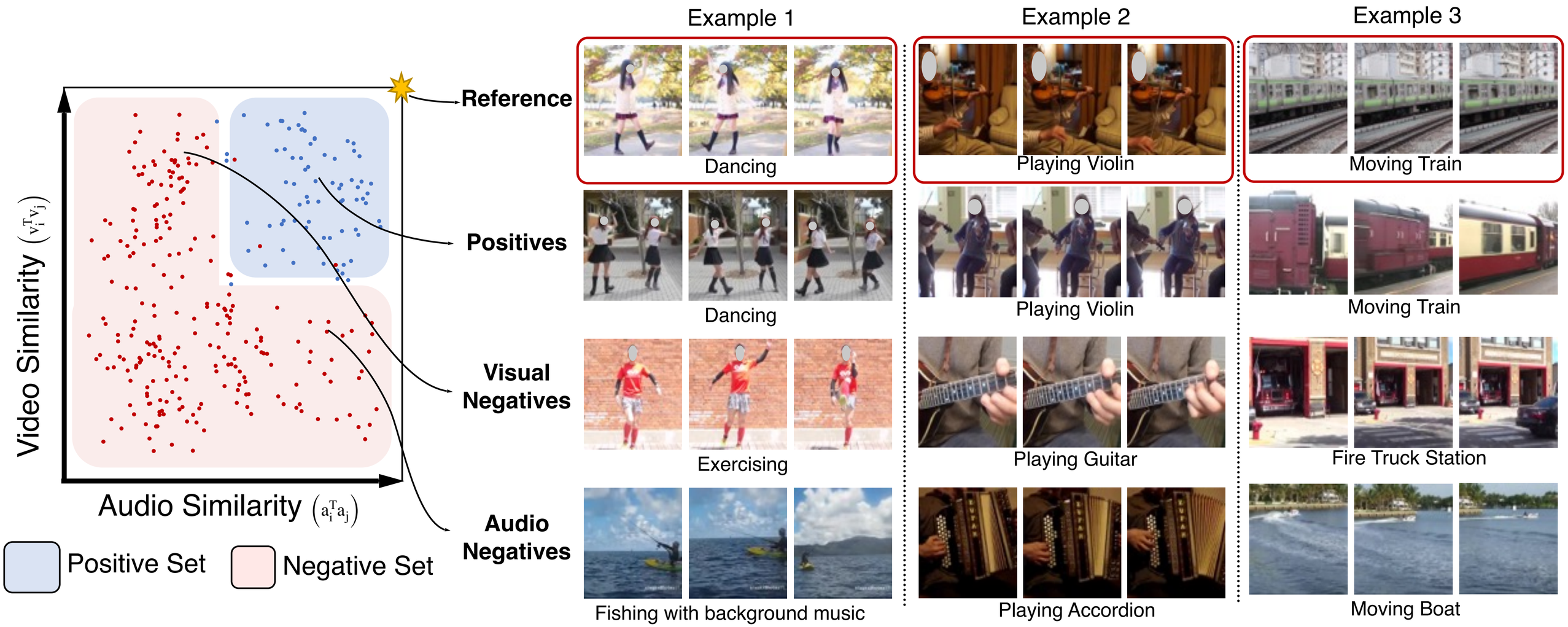
Relate to other video/audio using negatives

Grouping using Audio-visual Agreements (CMA)





Grouping using Audio-visual Agreements (CMA)



Many ways to avoid trivial solutions

Similarity Maximization Objective

- Contrastive learning
  - MoCo, PIRL, SimCLR
- Clustering
  - DeepCluster, SeLA, SwAV
- Distillation
  - BYOL, SimSiam

Redundancy Reduction Objective

- Redundancy Reduction
  - Barlow Twins

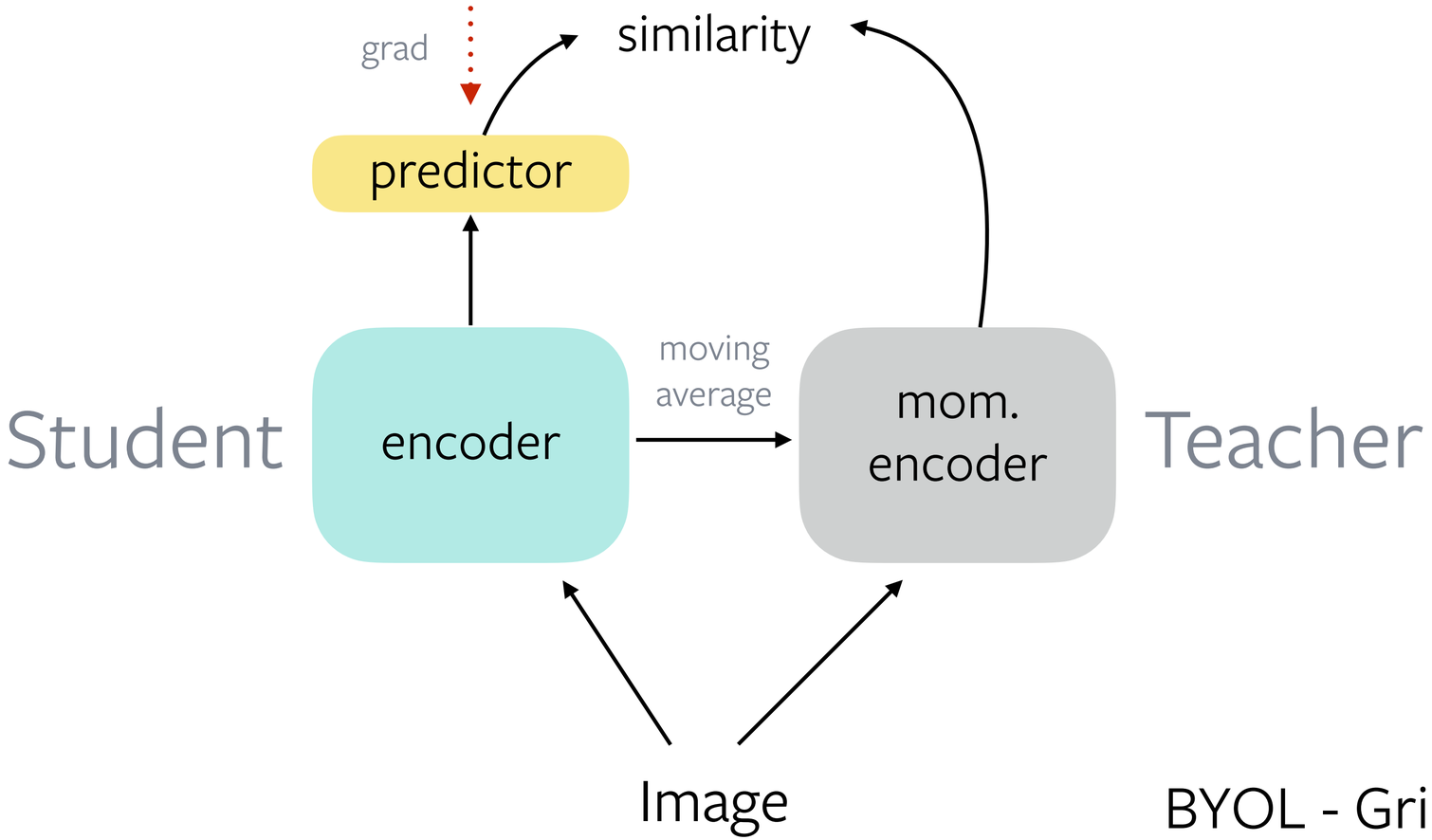
Distillation

- What we want $$f_\theta(I) = f_\theta(\text{augment}(I))$$
- How we do it $$f_\theta^{\text{student}}(I) = f_\theta^{\text{teacher}}(\text{augment}(I))$$
- Prevent trivial solutions by asymmetry
  - Asymmetric **learning rule** between student teacher
  - Asymmetric **architecture** between student teacher

BYOL

- What we want $$f_\theta(I) = f_\theta(\text{augment}(I))$$

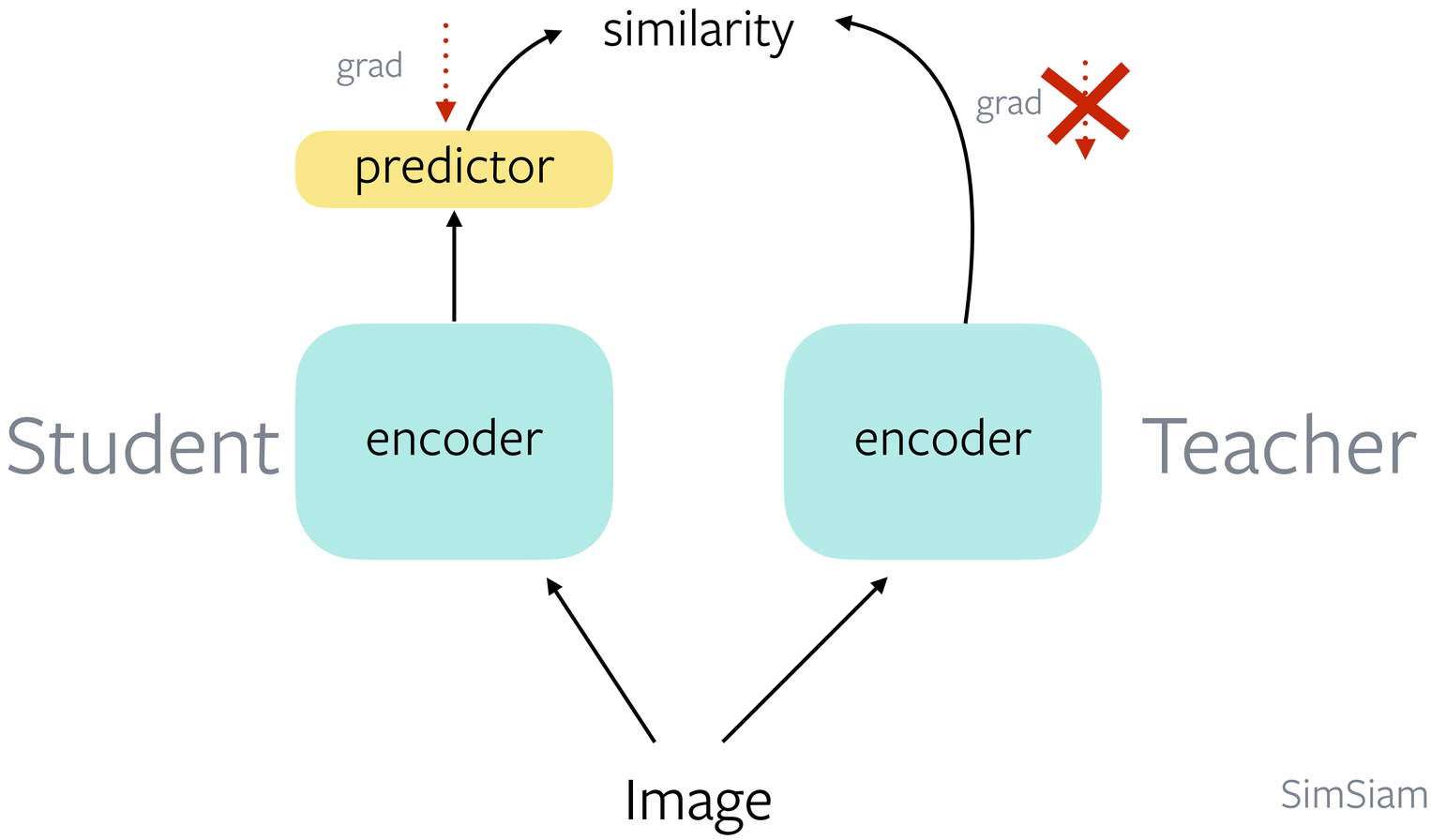
- How we do it $$f_\theta^{\text{student}}(I) = f_\theta^{\text{teacher}}(\text{augment}(I))$$



BYOL - Grill et al., 2020

SimSiam

- What we want $$f_\theta(I) = f_\theta(\text{augment}(I))$$



SimSiam - Chen & He 2020

Many ways to avoid trivial solutions

Similarity Maximization Objective

- Contrastive learning
  - MoCo, PIRL, SimCLR
- Clustering
  - DeepCluster, SeLA, SwAV
- Distillation
  - BYOL, SimSiam

Redundancy Reduction Objective

- Redundancy Reduction
  - Barlow Twins

Barlow Twins: Self-supervised Learning via Redundancy Reduction

Jure Zbontar*, Li Jing*, Ishan Misra, Yann LeCun, Stéphane Deny

https://github.com/facebookresearch/barlowtwins

Horace Barlow's Efficient Coding Hypothesis

- Inspired by Information Theory
- Neurons communicate via "spiking codes"
- Spiking codes aim to reduce redundancy between neurons

Redundancy Reduction

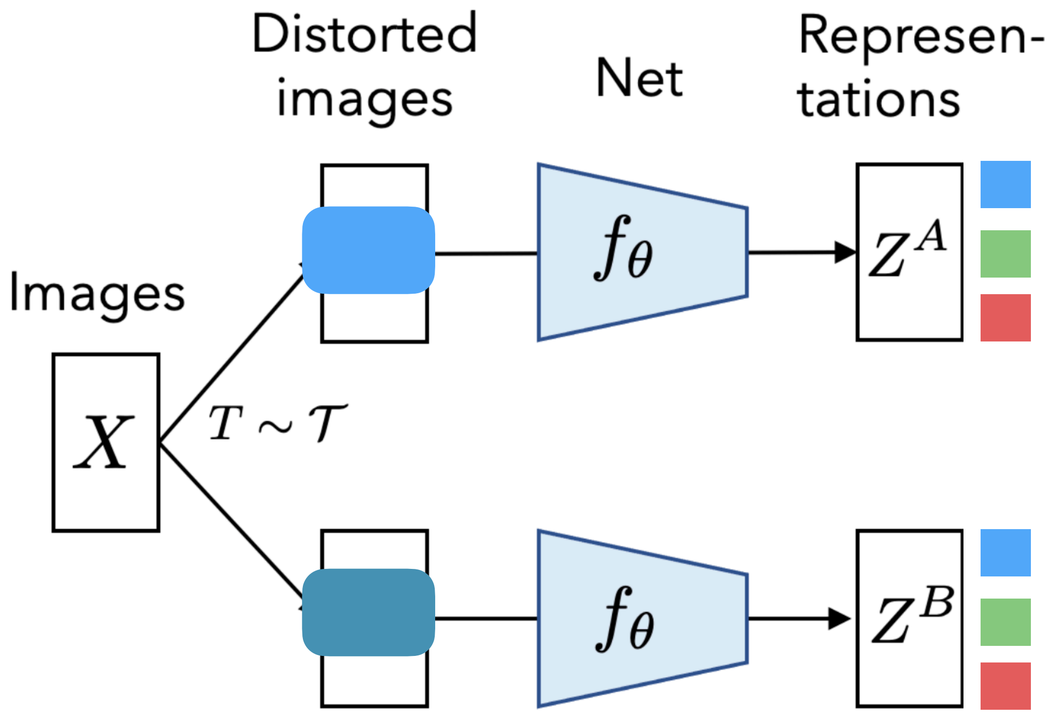
- N neurons produce a representation: N dimensional feature
- Each neuron should satisfy
  - Invariance -- be invariant under different data augmentation
  - Independent of other neurons -- reduce redundancy

- VERY roughly speaking

$$f_\theta(I)[i] = f_\theta(\text{augment}(I))[i]$$

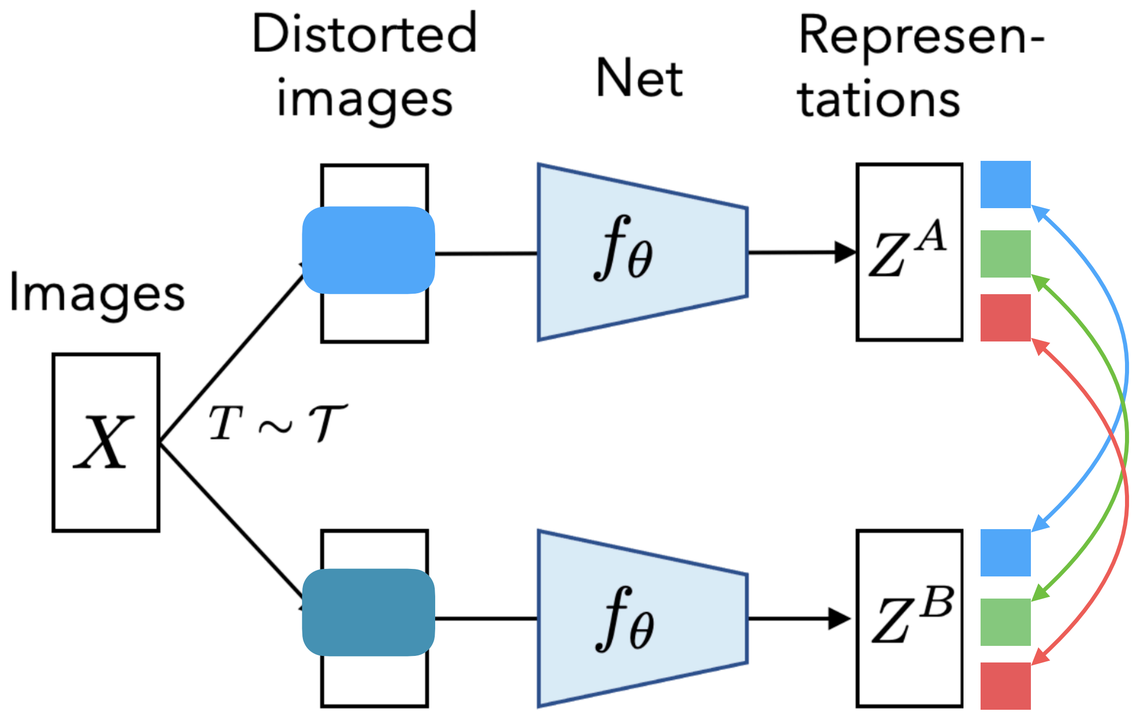
$$f_\theta(I)[i] \neq f_\theta(\text{augment}(I))[j]$$

Barlow Twins



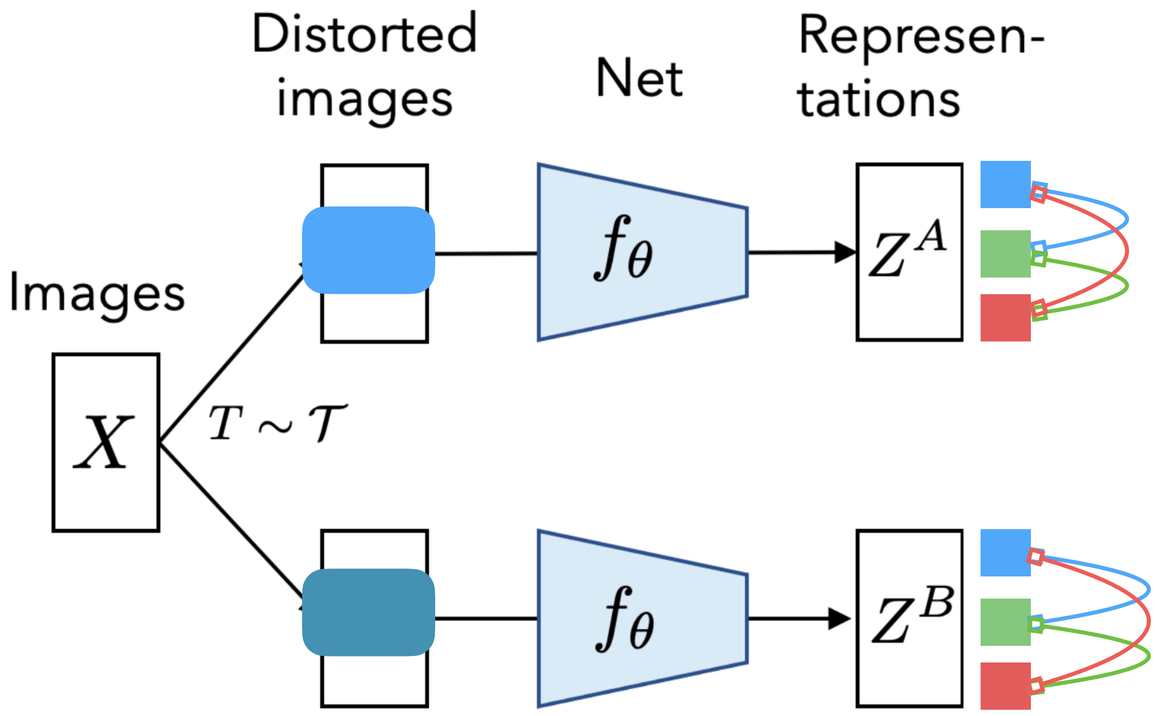
Barlow Twins - Zbontar et al., 2020

Barlow Twins - Invariance



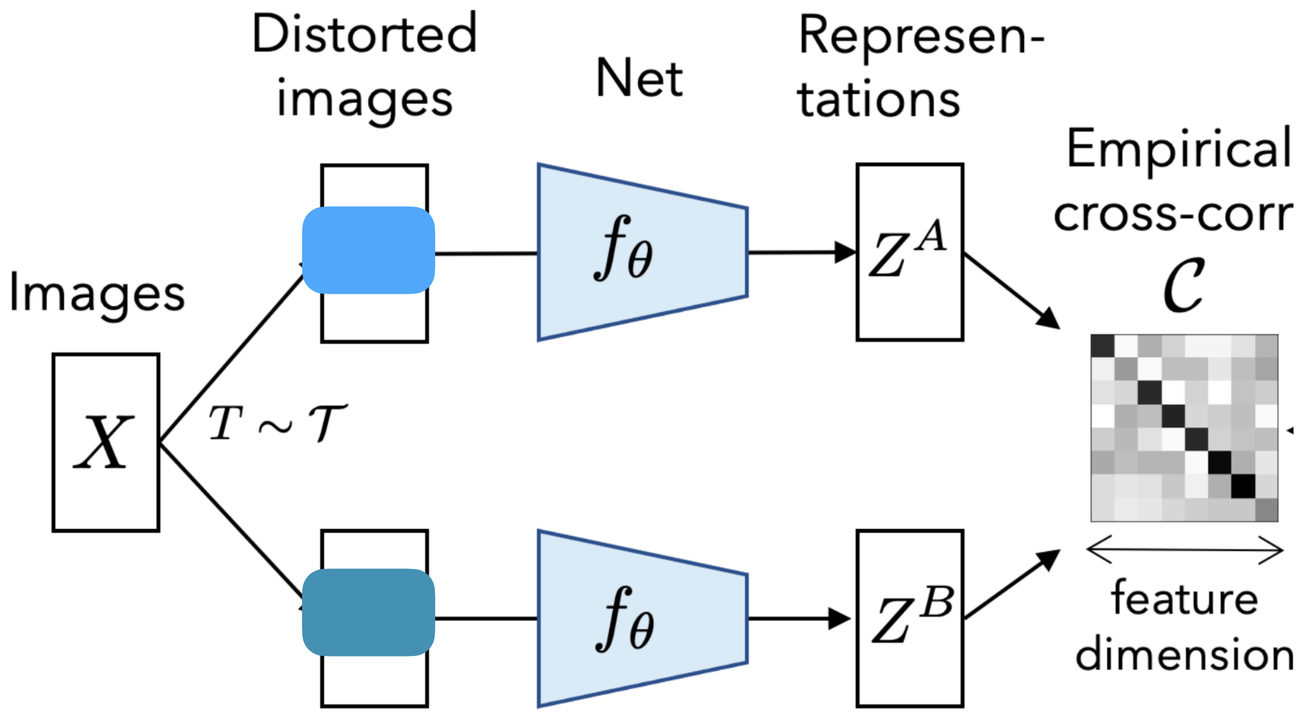
Barlow Twins - Zbontar et al., 2020

Barlow Twins - Redundancy Reduction



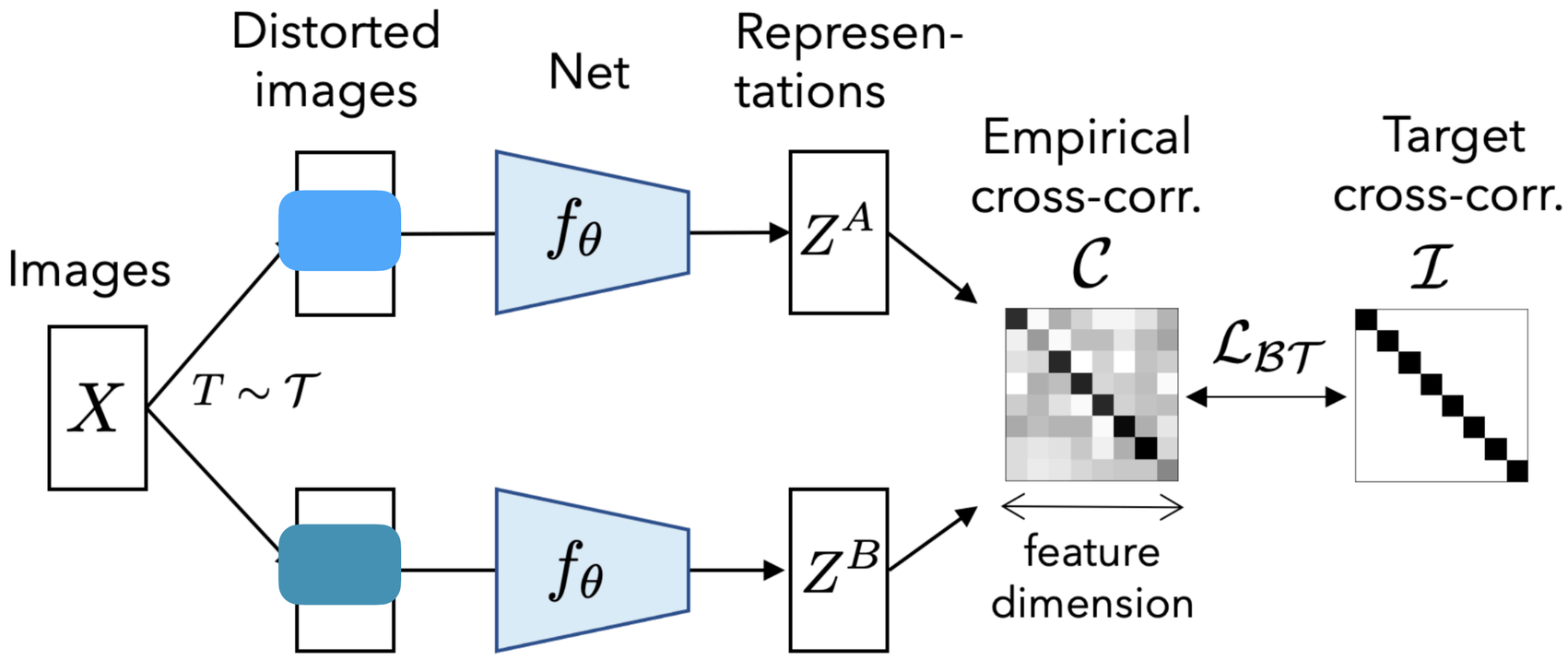
Barlow Twins - Zbontar et al., 2020

Barlow Twins



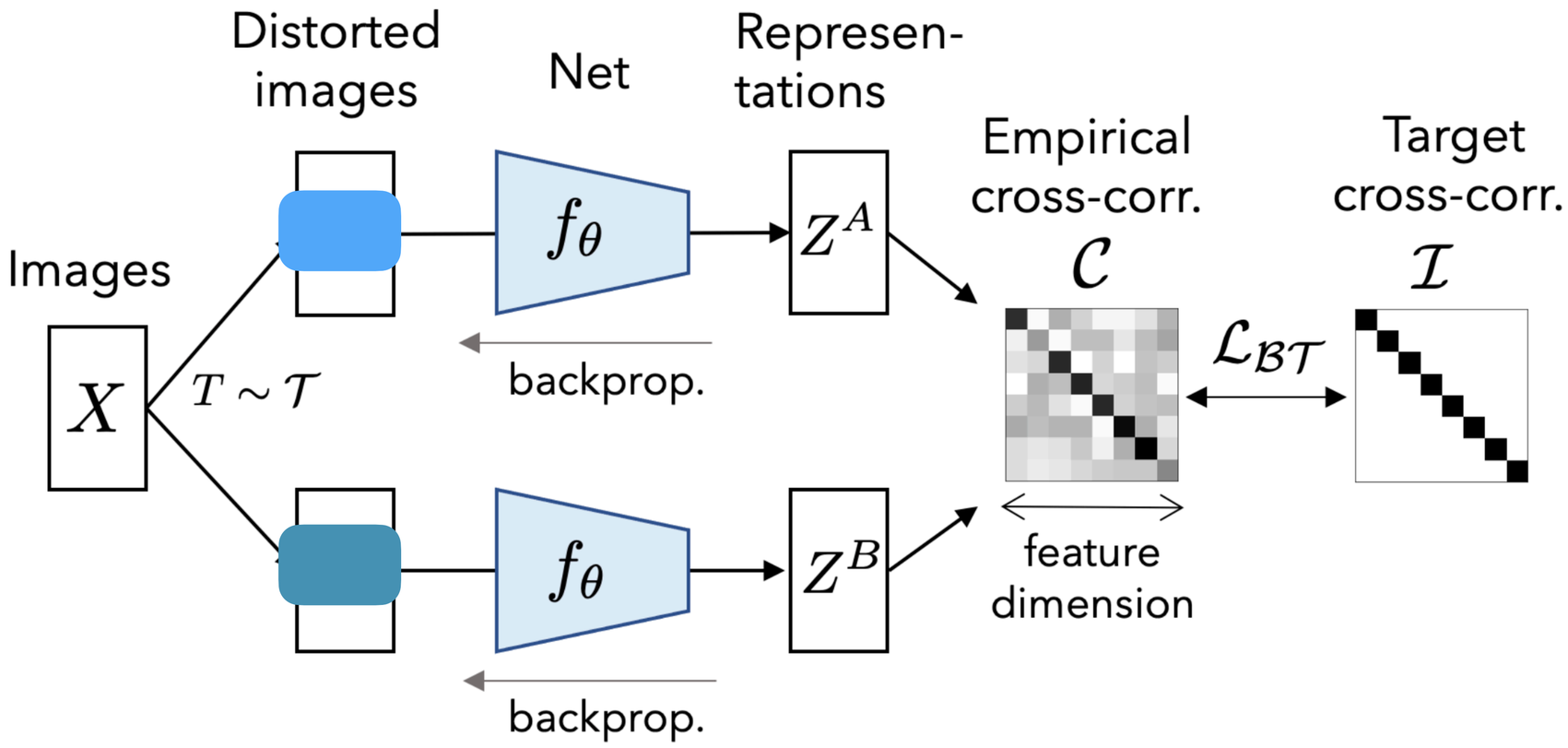
Barlow Twins - Zbontar et al., 2020

Barlow Twins - Loss



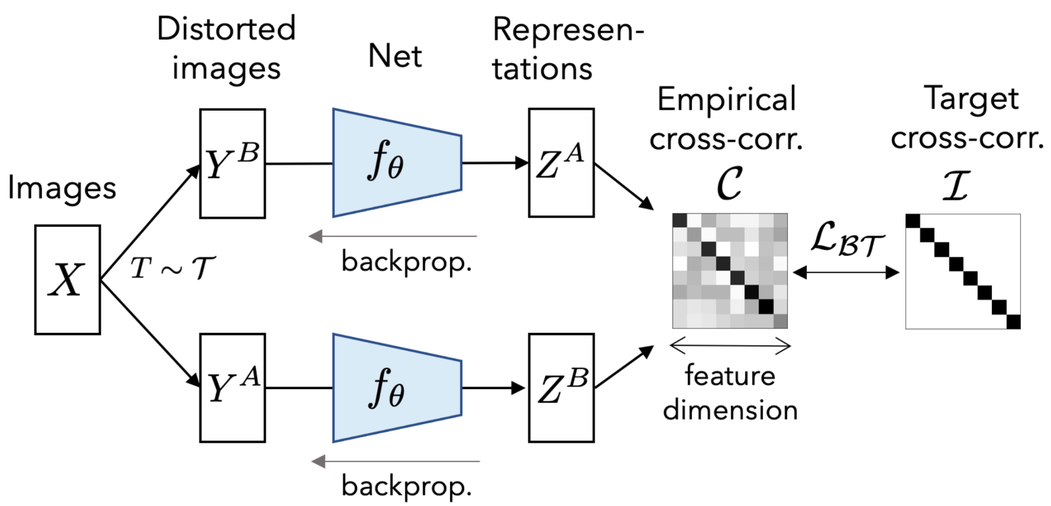
Barlow Twins - Zbontar et al., 2020

Barlow Twins - Loss



Barlow Twins - Zbontar et al., 2020

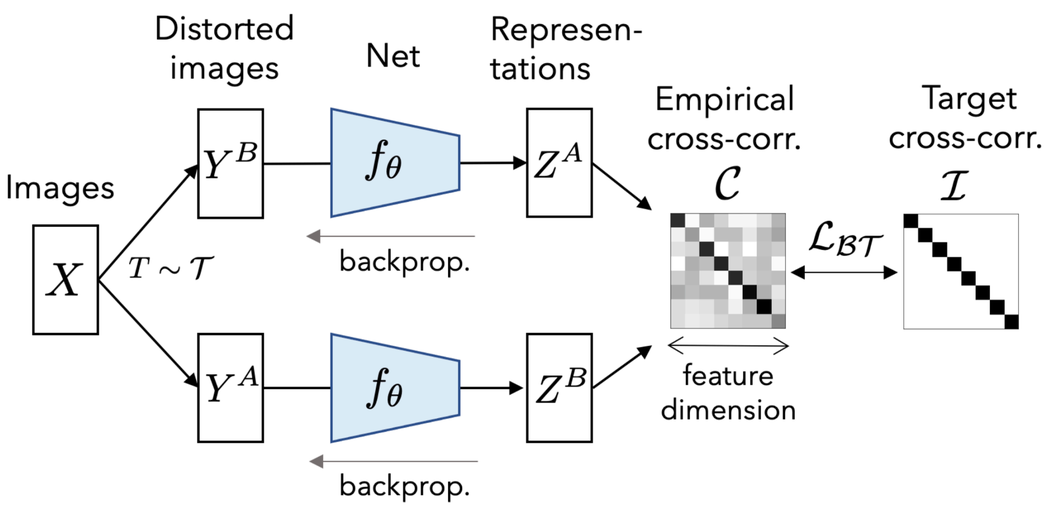
Barlow Twins Objective Function



$$\mathcal{C}_{ij} \triangleq \frac{\sum_b z^A_{b,i} z^B_{b,j}}{\sqrt{\sum_b \left(z^A_{b,i}\right)^2} \sqrt{\sum_b \left(z^B_{b,j}\right)^2}}$$

$$\mathcal{L}_{\mathcal{BT}} \triangleq \underbrace{\sum_i (1 - \mathcal{C}_{ii})^2}_{\text{invariance term}} + \lambda \underbrace{\sum_i \sum_{j \neq i} \mathcal{C}_{ij}{}^2}_{\text{redundancy reduction term}}$$

Trivial Solutions?



$$\mathcal{C}_{ij} \triangleq \frac{\sum_b z^A_{b,i} z^B_{b,j}}{\sqrt{\sum_b \left(z^A_{b,i}\right)^2}\sqrt{\sum_b \left(z^B_{b,j}\right)^2}}$$

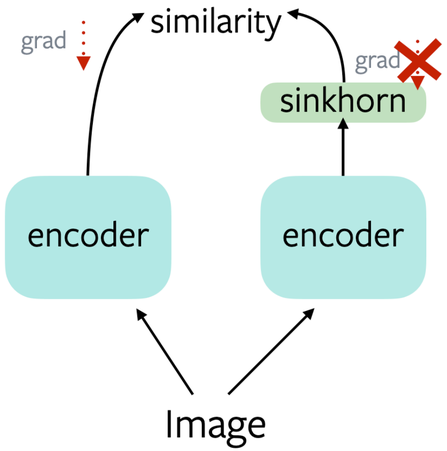
$$\mathcal{L}_{\mathcal{BT}} \triangleq \underbrace{\sum_i (1-\mathcal{C}_{ii})^2}_{\text{invariance term}} + \lambda \underbrace{\sum_i \sum_{j \neq i} \mathcal{C}_{ij}{}^2}_{\text{redundancy reduction term}}$$

Center Z^A and Z^B before computing cross-correlation

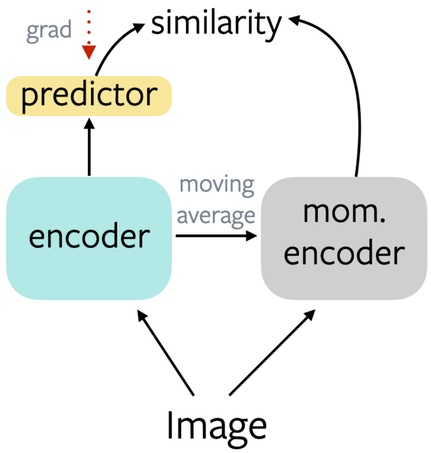
Prevents trivial solutions without

- "Negatives" like in contrastive learning

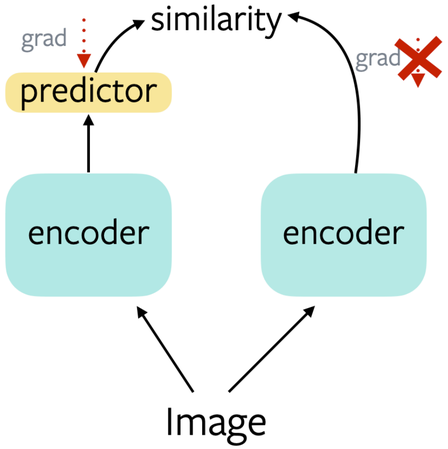
Prevents trivial solutions without Asymmetric Learning



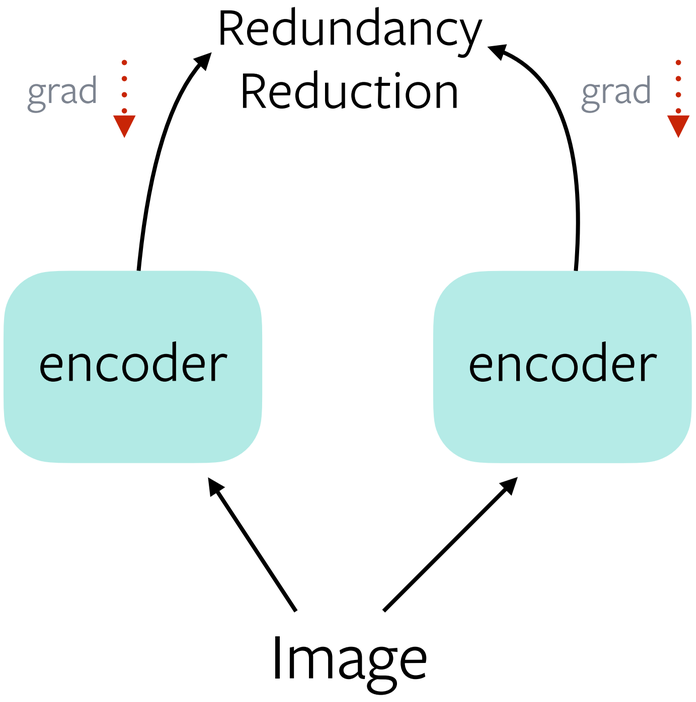
SwAV - Caron et al., 2020



BYOL - Grill et al., 2020



SimSiam - Chen & He, 2020



Barlow Twins

Barlow Twins - easy to implement

```
# f: encoder network
# lambda: weight on the off-diagonal terms
# N: batch size
# D: dimensionality of the representation
#
# mm: matrix-matrix multiplication
# off_diagonal: off-diagonal elements of a matrix
# eye: identity matrix

for x in loader: # load a batch with N samples
    # two randomly augmented versions of x
    y_a, y_b = augment(x)

    # compute representations
    z_a = f(y_a) # NxD
    z_b = f(y_b) # NxD

    # normalize repr. along the batch dimension
    z_a_norm = (z_a - z_a.mean(0)) / z_a.std(0) # NxD
    z_b_norm = (z_b - z_b.mean(0)) / z_b.std(0) # NxD

    # cross-correlation matrix
    c = mm(z_a_norm.T, z_b_norm) / N # DxD

    # loss
    c_diff = (c - eye(D)).pow(2) # DxD
    # multiply off-diagonal elems of c_diff by lambda
    off_diagonal(c_diff).mul_(lambda)
    loss = c_diff.sum()

    # optimization step
    loss.backward()
    optimizer.step()
```

Transfer Performance

Simpler method that performs as well as SOTA

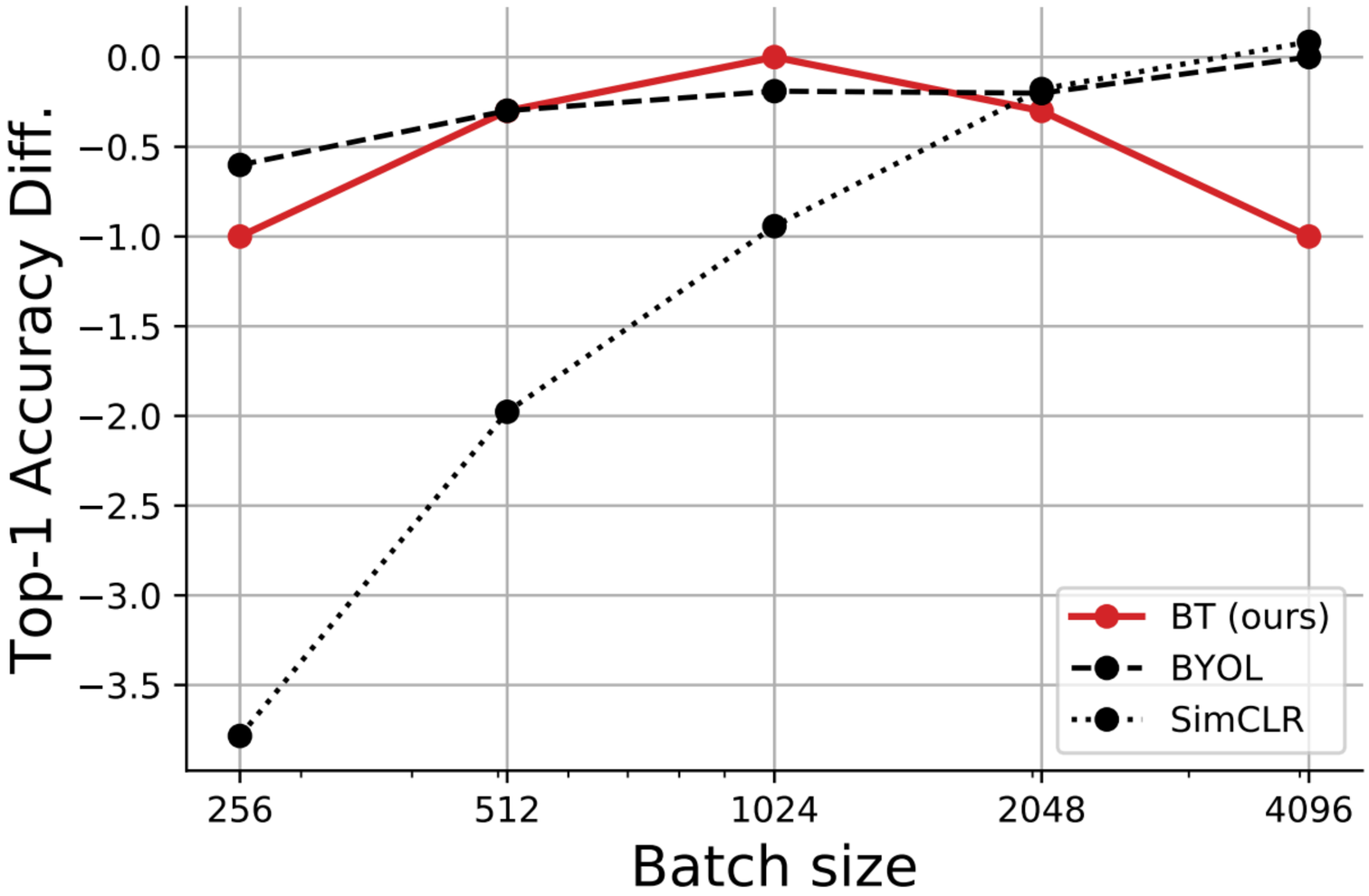
Method	Top-1		Top-5	
	1%	10%	1%	10%
Supervised	25.4	56.4	48.4	80.4
PIRL	-	-	57.2	83.8
SIMCLR	48.3	65.6	75.5	87.8
BYOL	53.2	68.8	78.4	89.0
SWAV (w/ multi-crop)	53.9	**70.2**	78.5	**89.9**
BARLOW TWINS (ours)	**55.0**	69.7	**79.2**	89.3

Finetuning on ImageNet

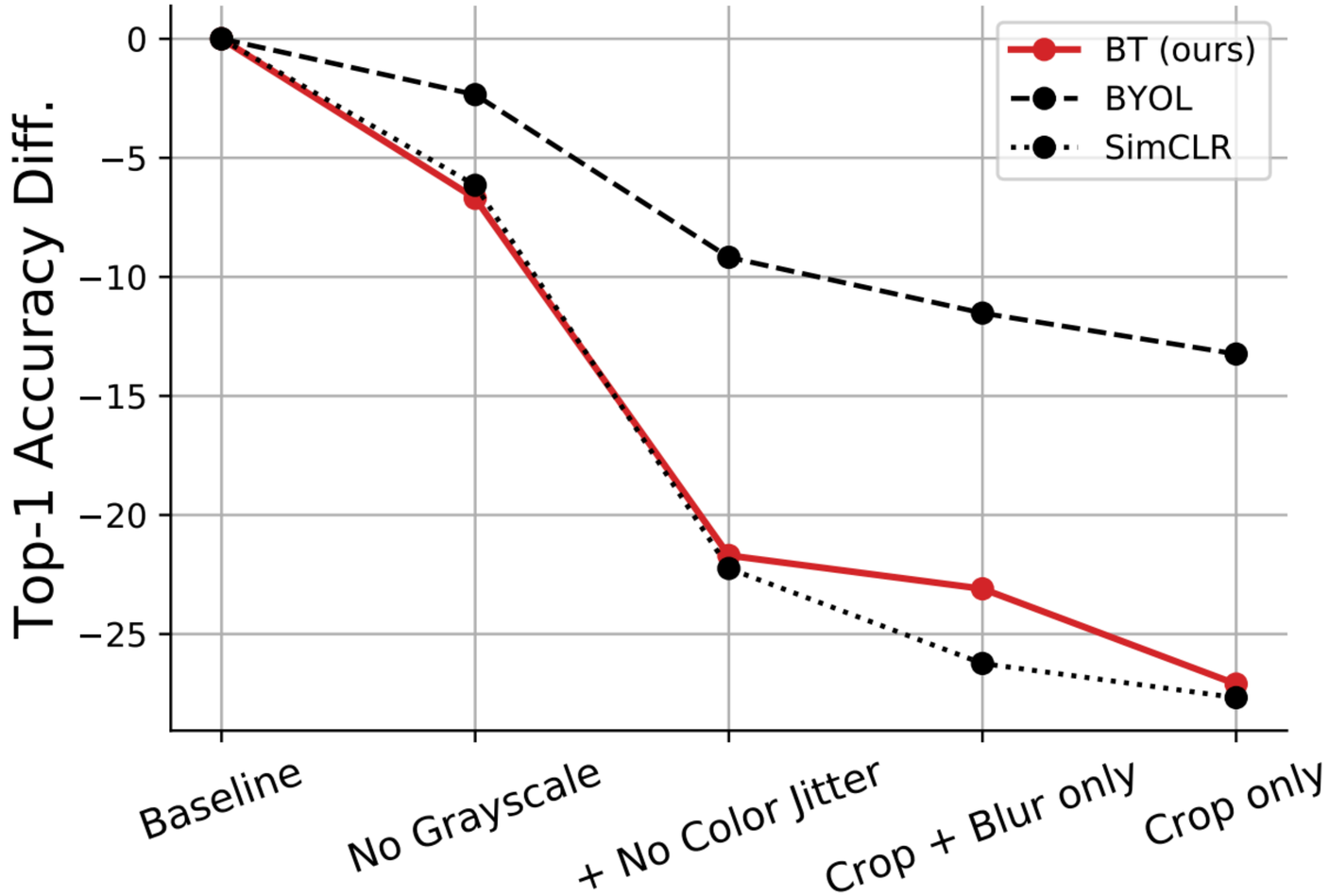
Method	Places-205	VOC07	iNat18
Supervised	53.2	87.5	46.7
SimCLR	52.5	85.5	37.2
MoCo-v2	51.8	86.4	38.6
SwAV	52.8	86.4	39.5
SwAV (w/ multi-crop)	56.7	88.9	48.6
BYOL	54.0	86.6	47.6
BARLOW TWINS (ours)	54.1	86.2	46.5

Linear Classifiers

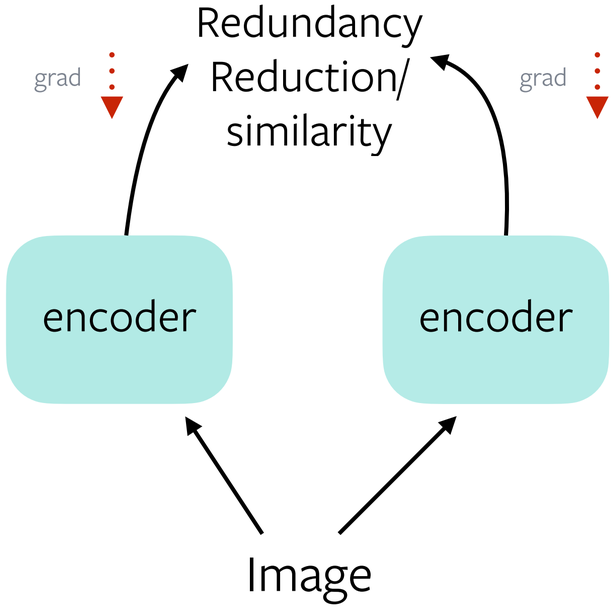
Stability Across Batchsizes



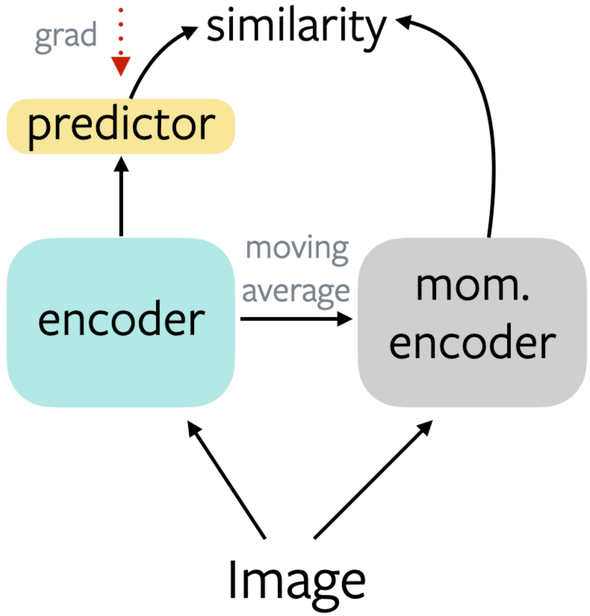
Effect of Data Augmentation



Barlow Twins
SimCLR

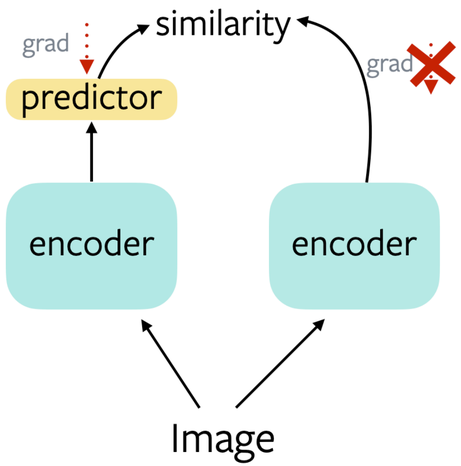


BYOL

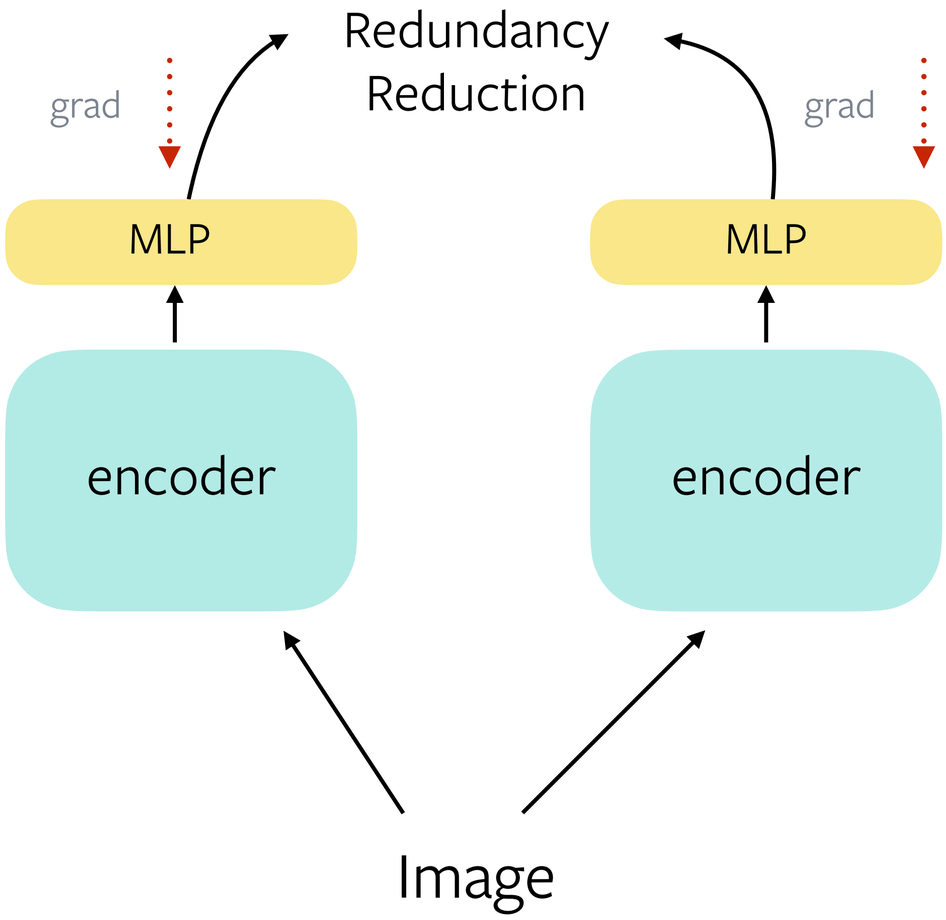


Is Asymmetric Learning/Architecture needed?

case	stop-gradient	predictor	Top-1	Top-5
Baseline	-	-	71.4	90.2
(a)	✓	-	70.5	89.0
(b)	-	✓	70.2	89.0
(c)	✓	✓	61.3	83.5

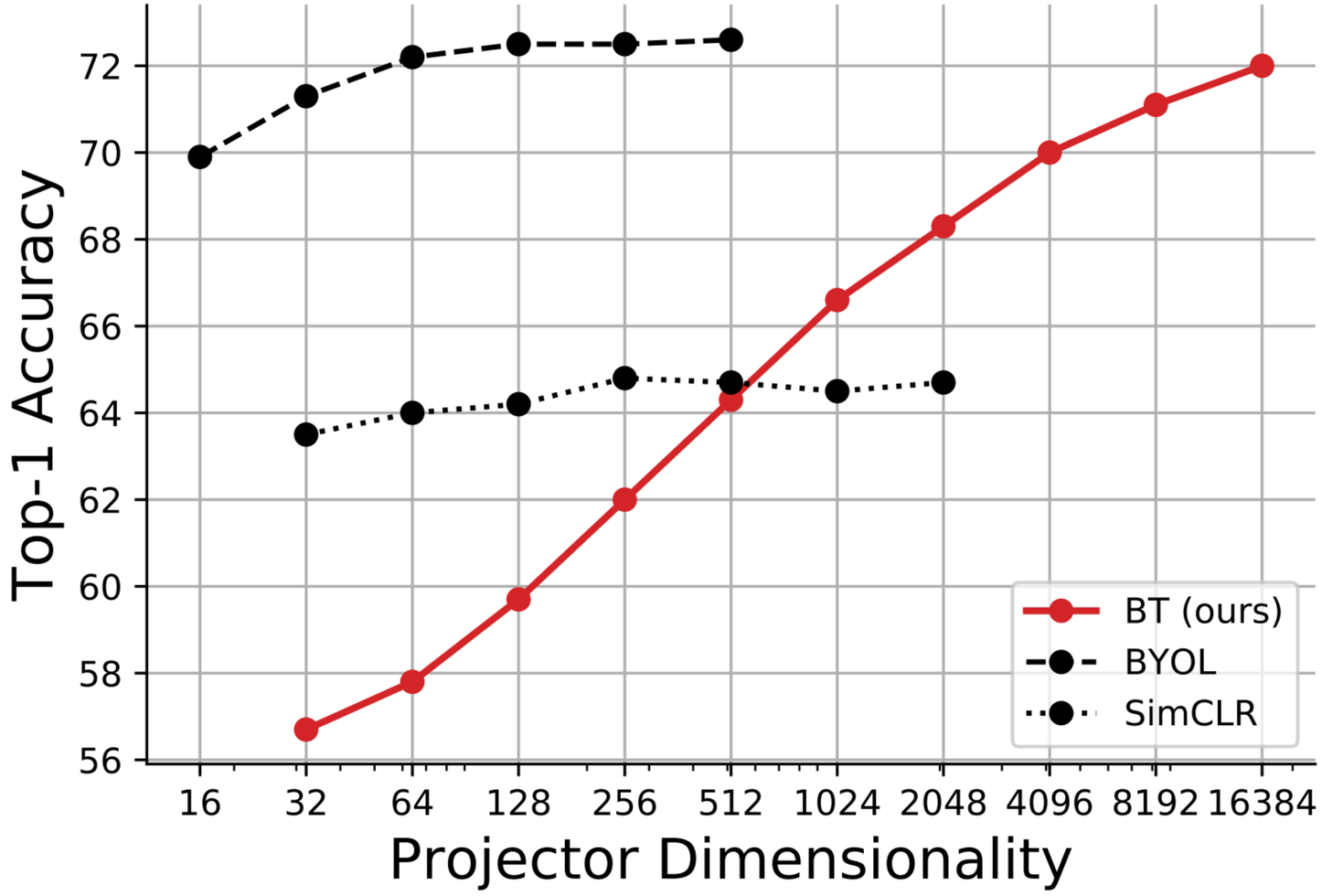


Number of non-redundant neurons



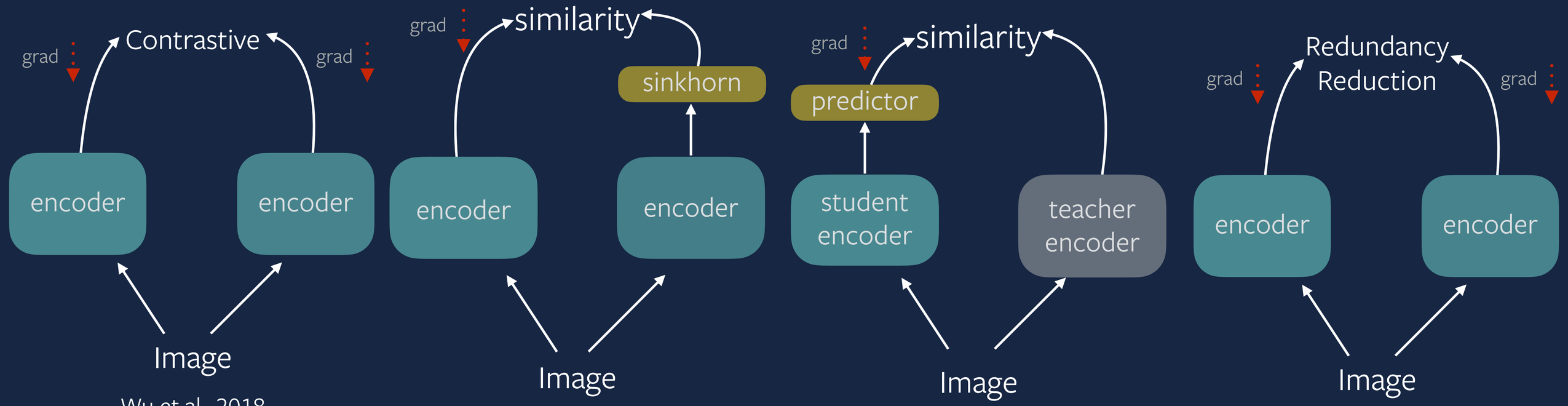
Change repr. dim used in pretraining.
All methods evaluated on 2048 dim features

Number of non-redundant neurons



Change repr. dim used in pretraining.
All methods evaluated on 2048 dim features

Thanks!



Wu et al., 2018
He et al., 2019
Misra and van der Maaten, 2019
Chen et al., 2020

Contrastive

Asano et al., 2019
Caron et al., 2020

Clustering

Grill et al., 2020
Chen & He, 2020

Distillation

Zbontar et al., 2021

Redundancy Reduction