VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning

Bardes et al. | ICLR 2022

Cheng Jiang | October 24, 2023
EECS 598-007 Biomedical AI | University of Michigan

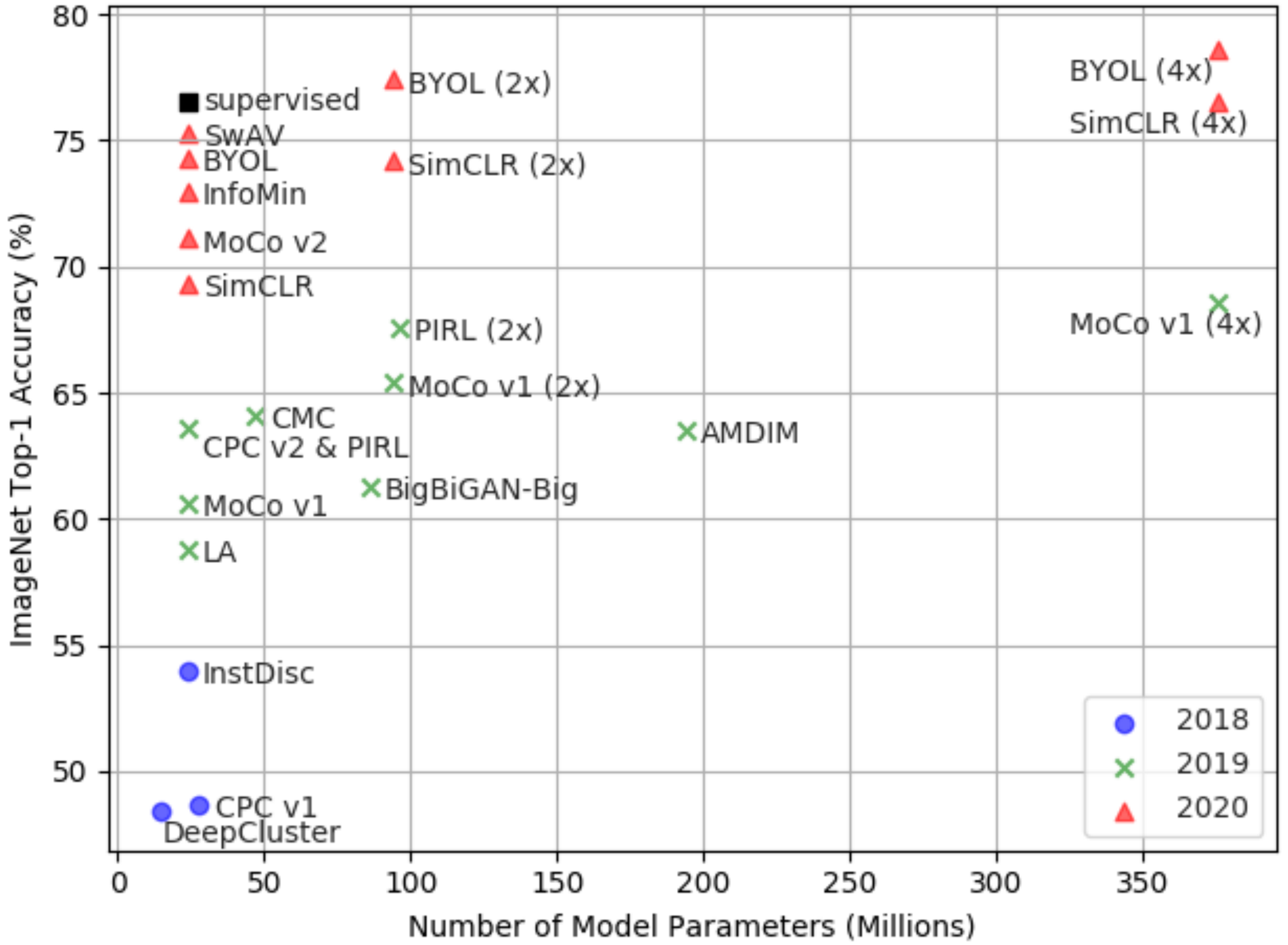
Introduction

Methods

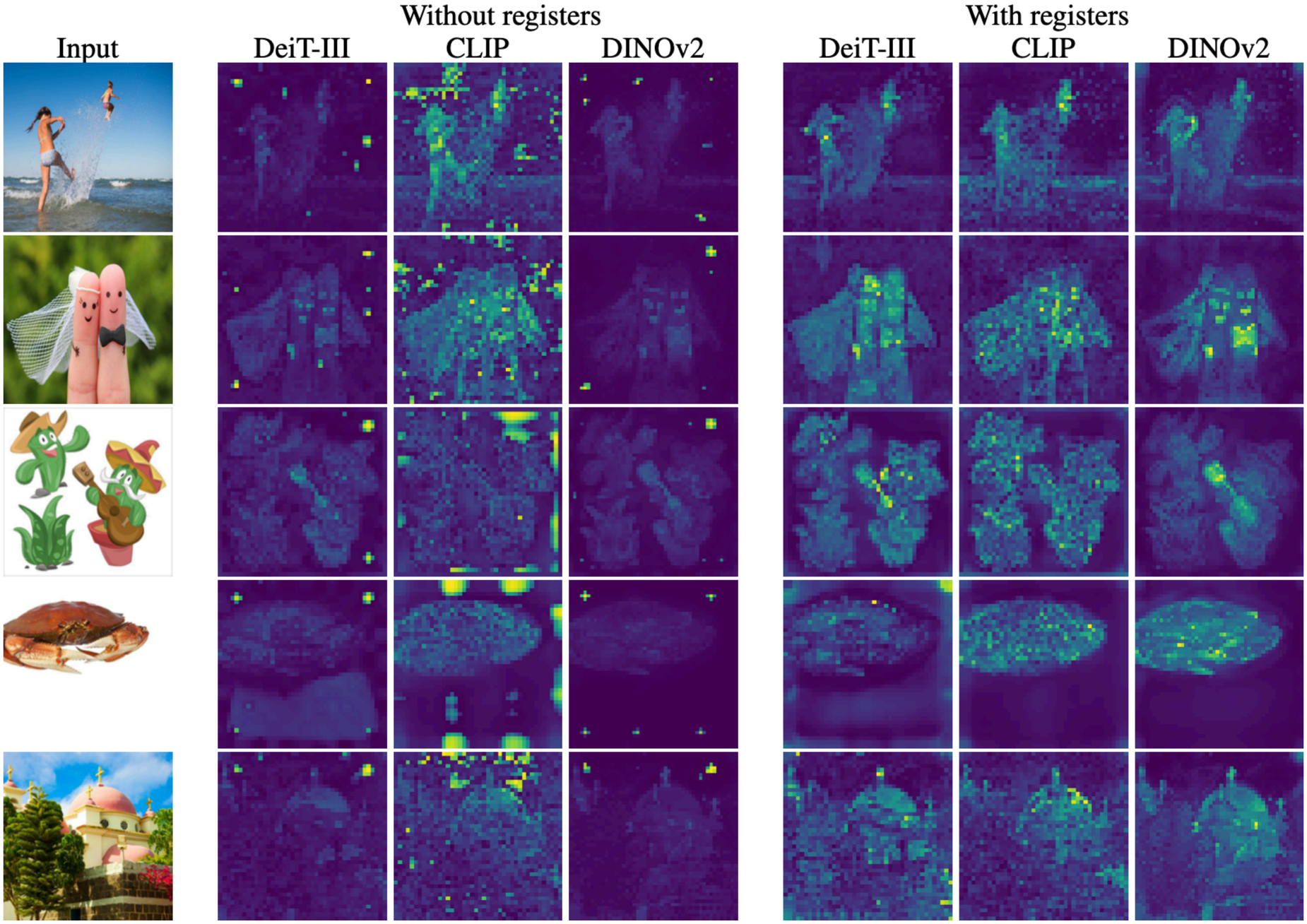
Experimentation

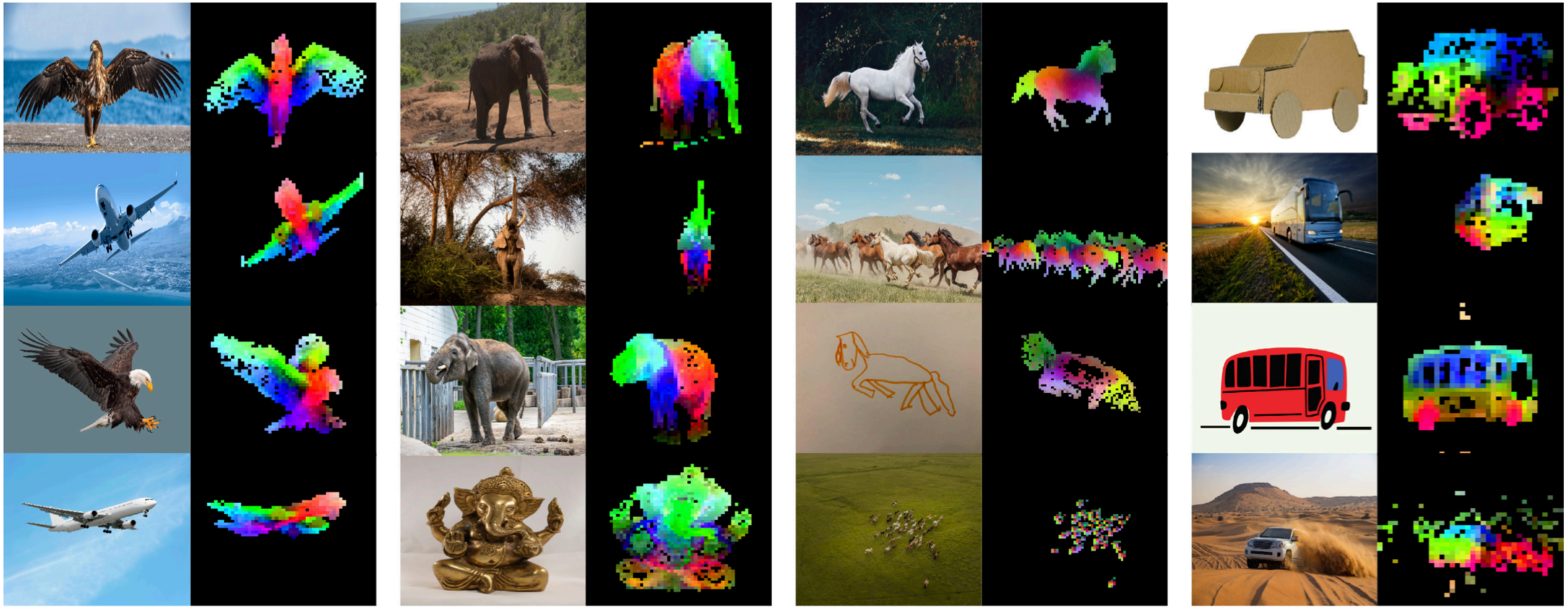
Results

Discussion + Conclusion



Self-supervised learning: Generative or contrastive, Liu et al. IEEE Trans Knowl Data Eng 2021


Without registers
With registers
Input
DeiT-III
CLIP
DINOv2
DeiT-III
CLIP
DINOv2

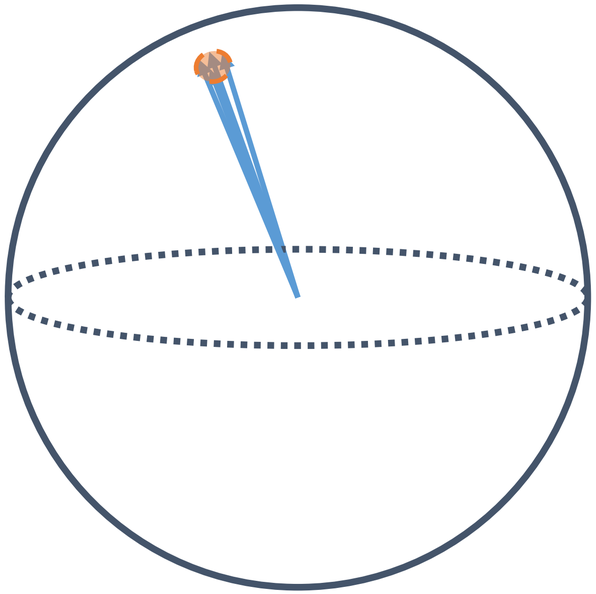


Self-supervised methods

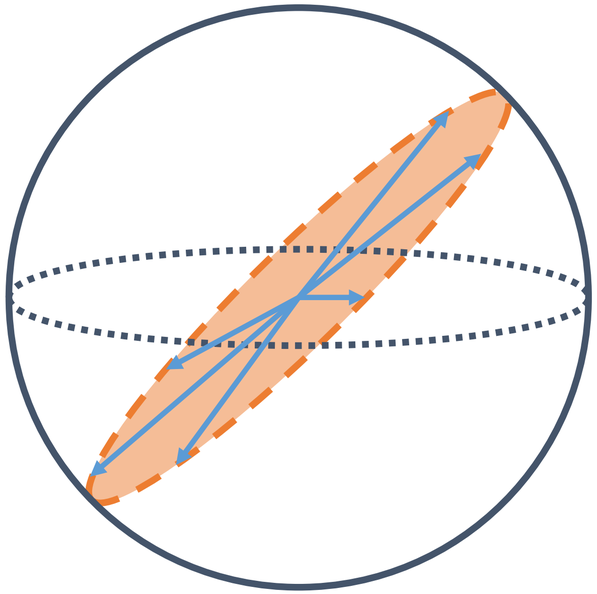
Deep metric learning	Self distillation	Masked image modeling	Canonical correlation analysis
SimCLR nnCLR DirectCLR	MoCo SimSiam BYOL DINO	BEiT MAE MSN I-JEPA	SwAV BarlowTwins VICReg

Key challenge: prevent informational collapse

Complete collapse



Dimensional collapse



Preventing collapse: strategies and tricks

Strong augmentations

Large batch sizes

Memory bank

Feature quantization

Batch- or feature-wise normalization

Moving average momentum encoder

Asymmetric model architecture

Stop-gradient operation

Pixel space image reconstruction

“[A] critical problem in BYOL [is that] the dynamics of learning and how they avoid collapse, is not fully understood.”

NeurIPS Reviewer a9mx

Idea: maximize information content of embedding vectors

Introduction

Methods

Experimentation

Results

Discussion + Conclusion

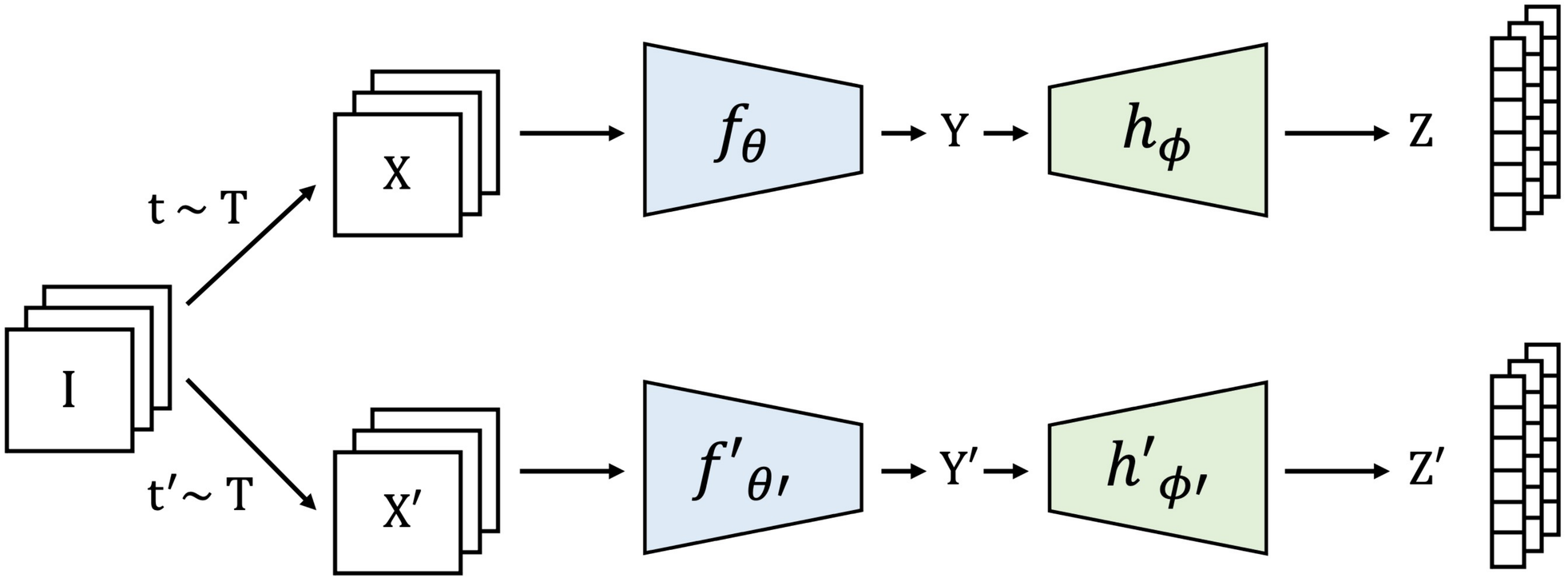
VICReg loss functions

Invariance *s*	Learn invariance between views
Variance *v*	Force batch embeddings to be different
Covariance *c*	Decorrelate the variables of each embedding

VICReg loss functions

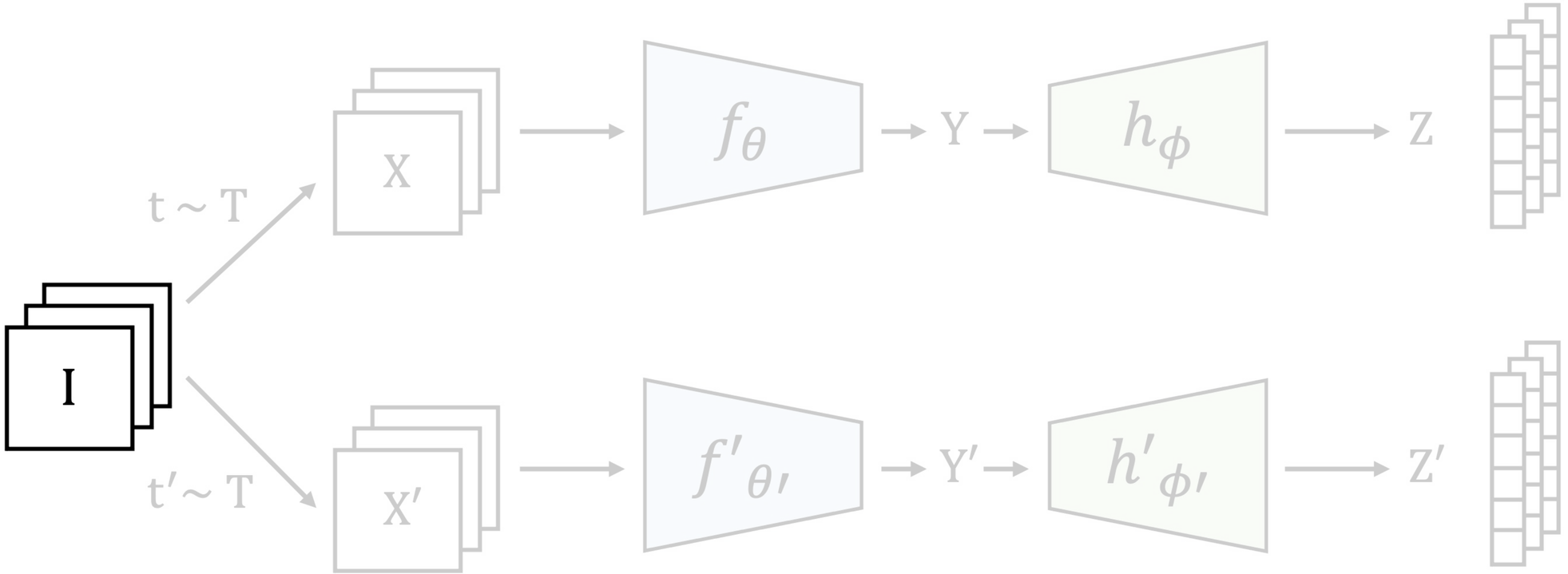
Invariance *s*	Learn invariance
Variance *v*	Prevent collapse
Covariance *c*	Prevent collapse

Joint embedding model architecture



Joint embedding model architecture

Batch of images

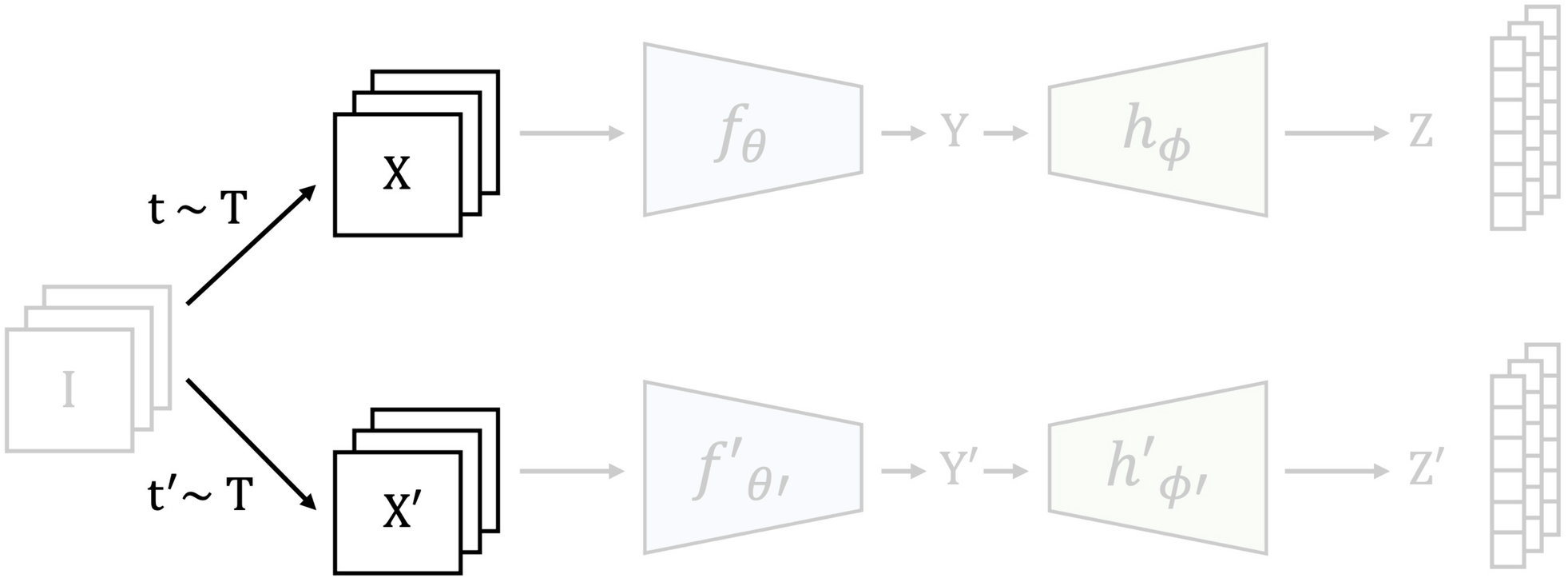


Joint embedding model architecture

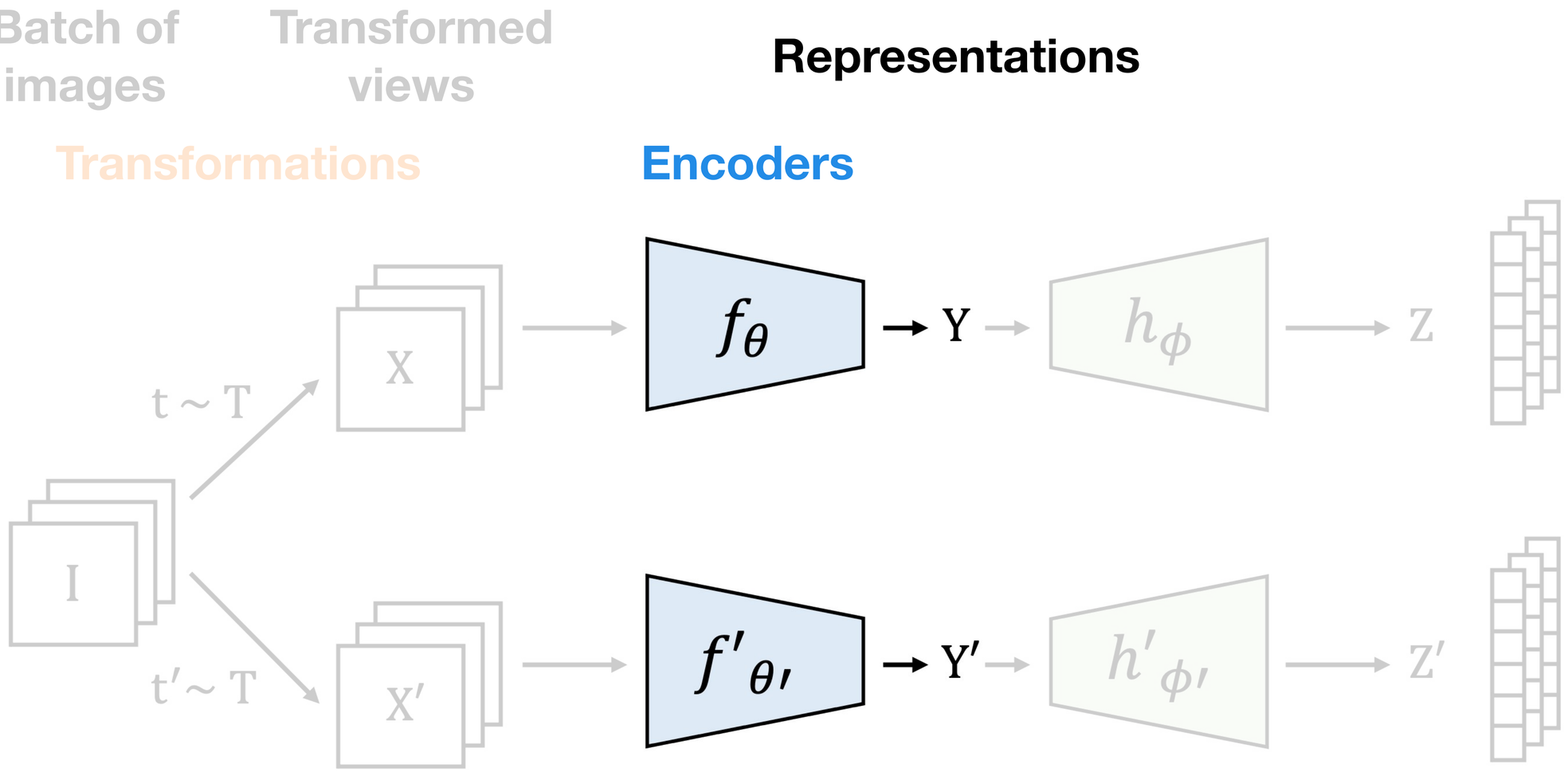
Batch of images

Transformed views

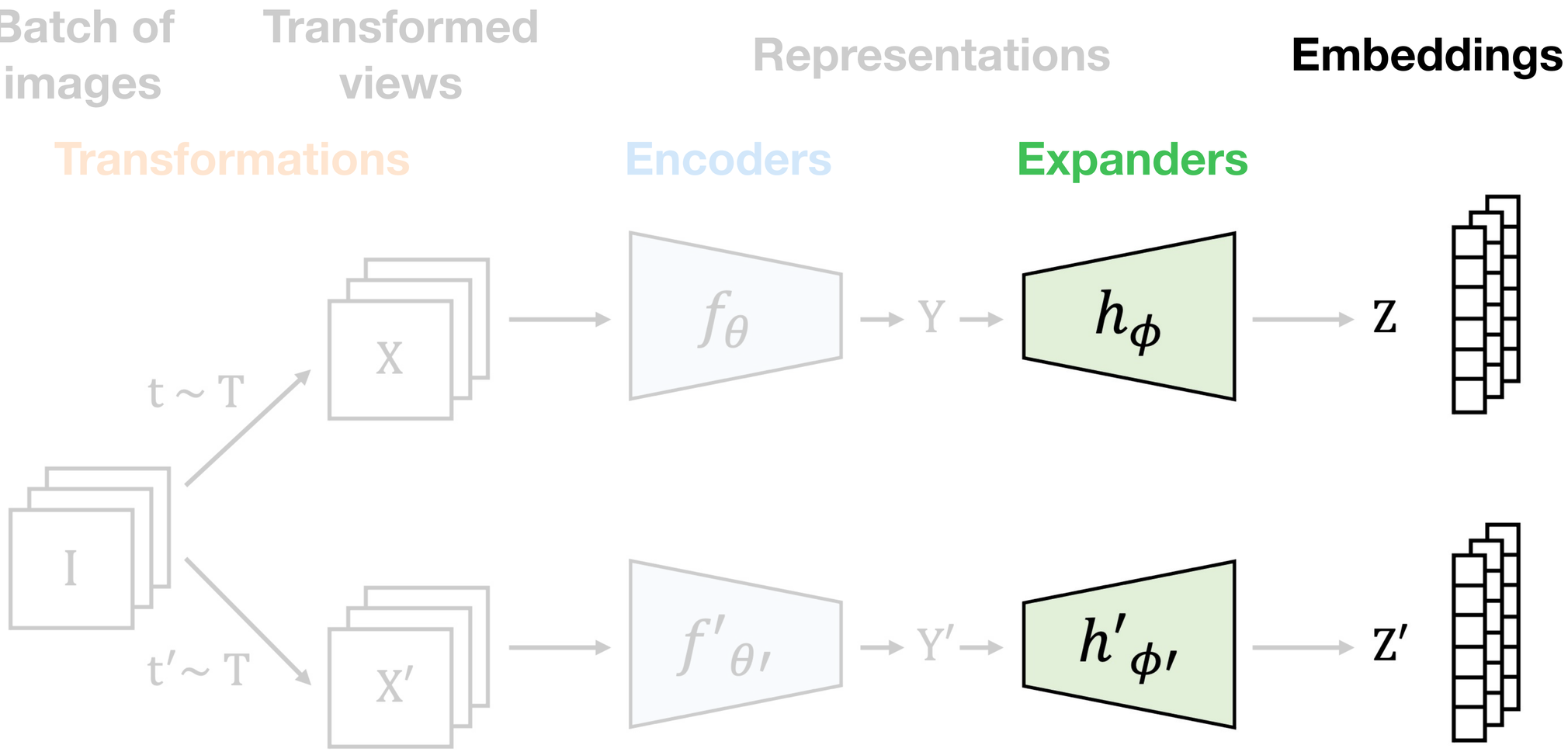
Transformations



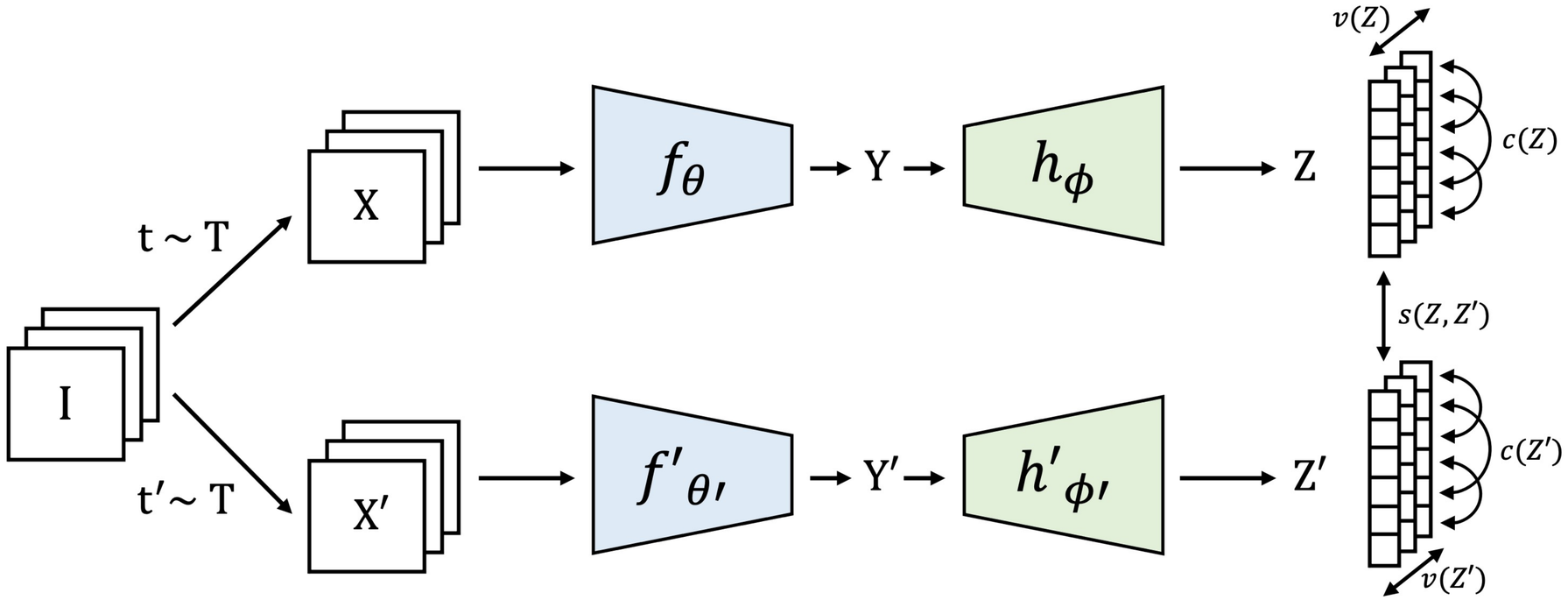
Joint embedding model architecture



Joint embedding model architecture

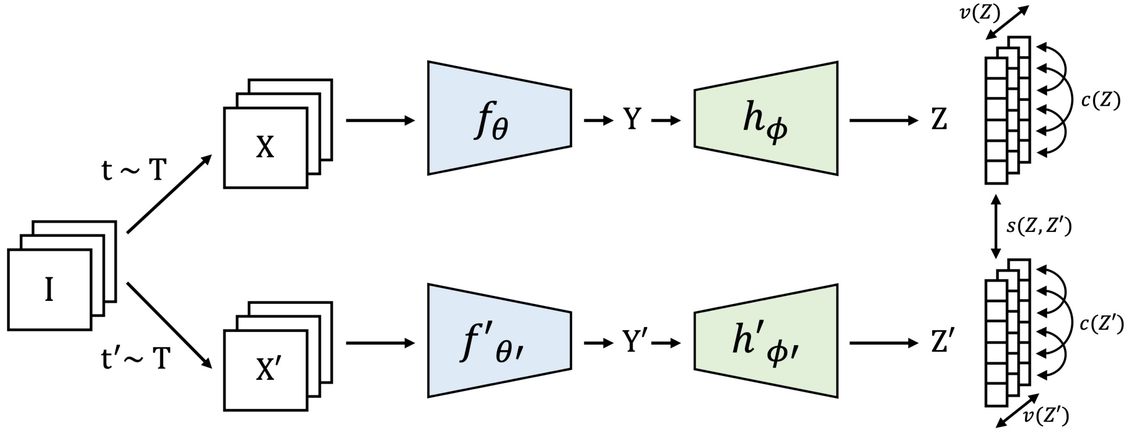


VICReg loss functions



$$\ell(Z, Z') = \lambda s(Z, Z') + \mu[v(Z) + v(Z')] + \nu[c(Z) + c(Z')]$$

VICReg loss functions



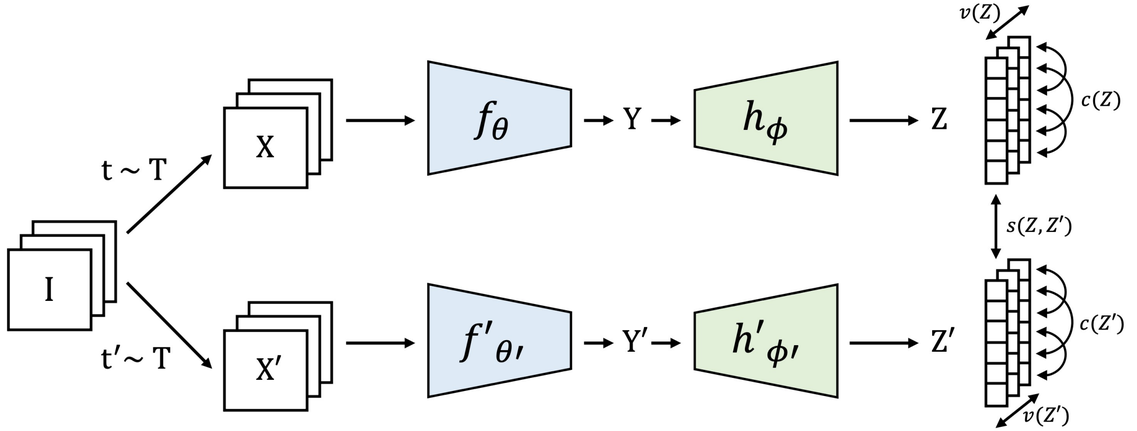
Invariance ***s*** Learn invariance between views

Variance ***v*** Force batch embeddings to be different

Covariance ***c*** Decorrelate the variables of each embedding

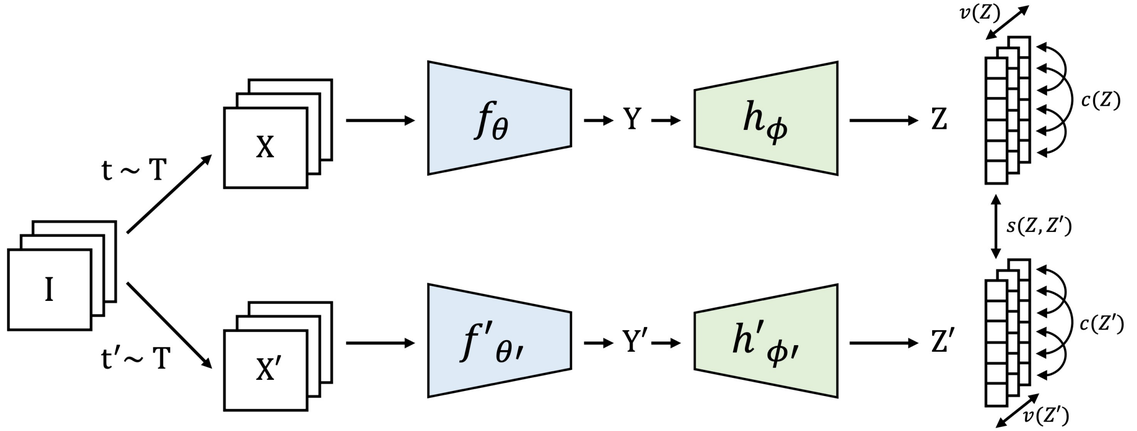
$$\ell(Z, Z') = \lambda s(Z, Z') + \mu[v(Z) + v(Z')] + \nu[c(Z) + c(Z')]$$

Invariance loss s



$$s(Z, Z') = \frac{1}{n} \sum_i \|z_i - z'_i\|_2^2$$

Variance loss *v*



$$v(Z) = \frac{1}{d}\sum_{j=1}^{d} \max(0, \gamma - S(z^j, \epsilon))$$

Target stdev — γ

Feature stdev — $S(z^j, \epsilon)$

Average over features — $\frac{1}{d}\sum_{j=1}^{d}$

Hinge loss — $\max(0, \gamma - S(z^j, \epsilon))$

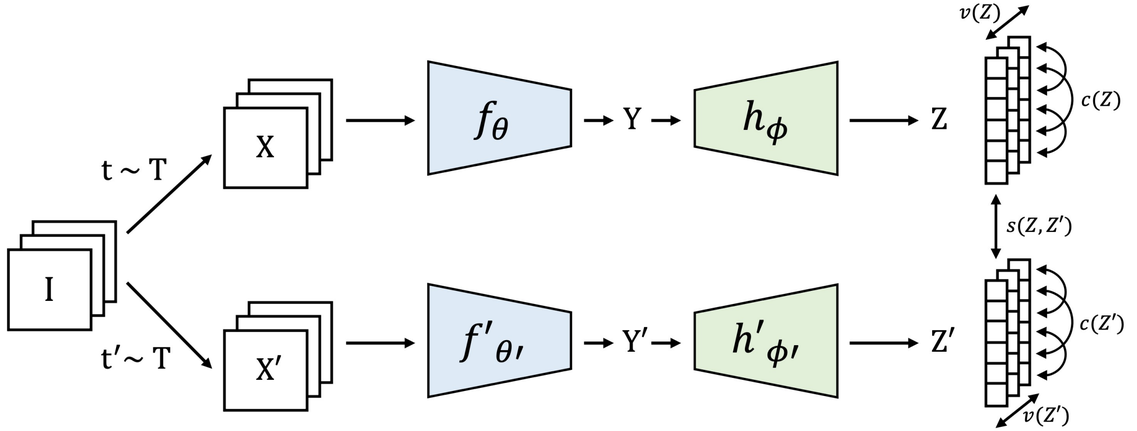
$$S(x, \epsilon) = \sqrt{\mathrm{Var}(x) + \epsilon}$$

Variance along batch dimension — $\mathrm{Var}(x)$

Numerical stability — ϵ

Covariance loss *c*

Inspired by BarlowTwins



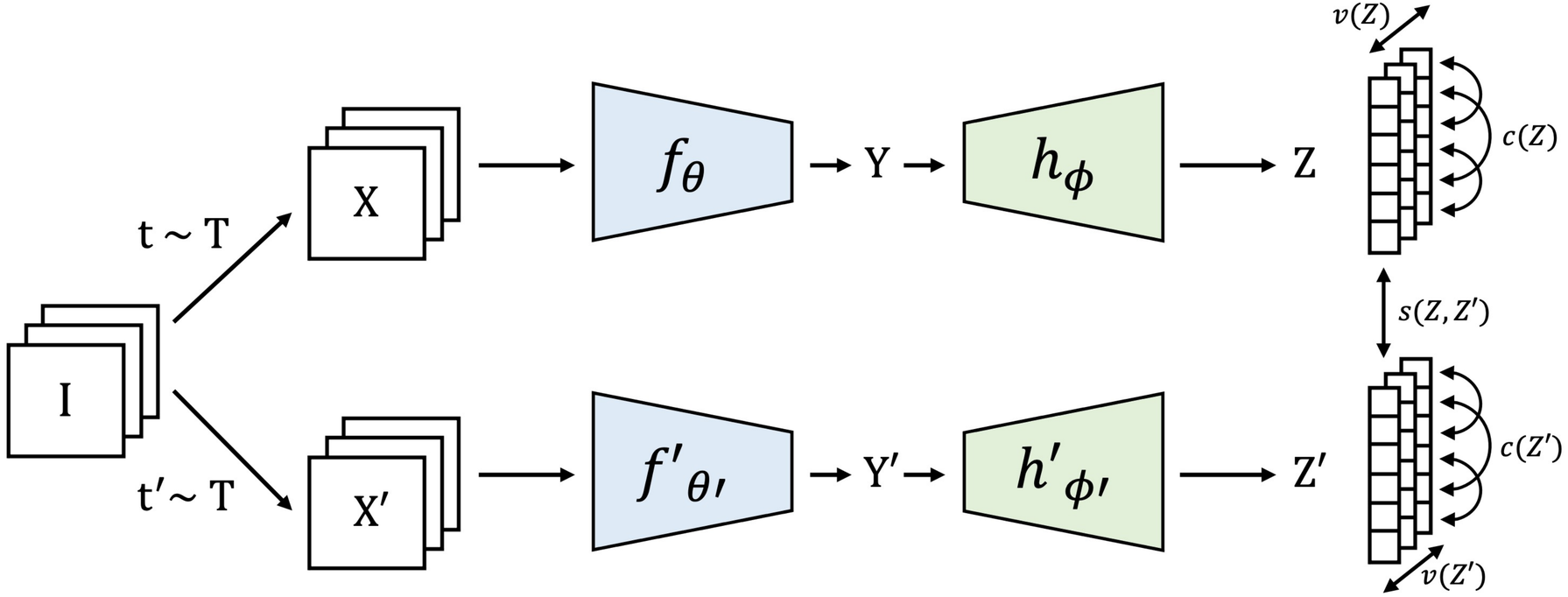
$$c(Z) = \frac{1}{d}\sum_{i \neq j}[C(Z)]^2_{i,j}$$

Embedding covariance matrix

$$C(Z) = \frac{1}{n-1}\sum_{i=1}^{n}(z_i - \bar{z})(z_i - \bar{z})^T, \quad \text{where} \quad \bar{z} = \frac{1}{n}\sum_{i=1}^{n} z_i$$

VICReg loss functions

$$\ell(Z, Z') = \lambda s(Z, Z') + \mu[v(Z) + v(Z')] + \nu[c(Z) + c(Z')]$$



Yann LeCun @ylecun · May 12, 2021

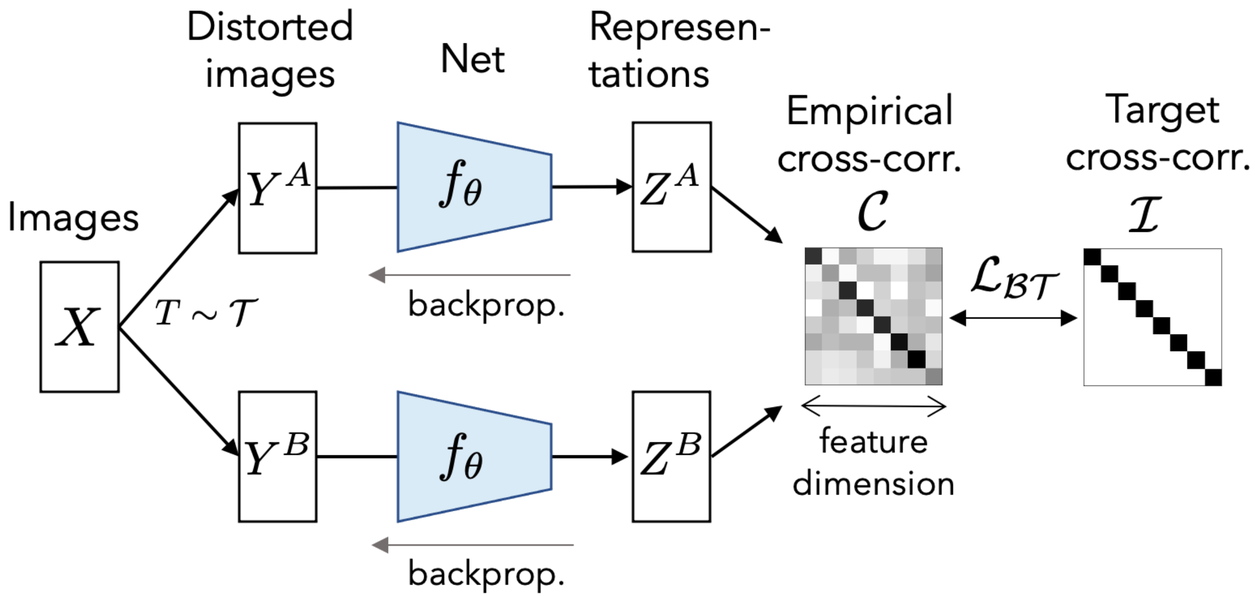
VICReg is a loss for JAE with 3 terms:
1. Variance: Hinge loss to maintain the std-dev of each component of Gx(x) & Gy(y) above a margin
2. Invariance: ||Gx(x)-Gy(y)||^2
3. Covariance: sum of the squares of the off-diag terms of the covariance matrices of Gx(x) and Gy(y).
3/N

2 3 25

Relation to BarlowTwins

$$\ell_{\mathrm{BT}} \triangleq \underbrace{\sum_i (1 - \mathcal{C}_{ii})^2}_{\text{Invariance term}} + \lambda \underbrace{\sum_i \sum_{j \neq i} \mathcal{C}_{ij}{}^2}_{\text{Covariance term}}$$

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



Introduction

Methods

Experimentation

Results

Discussion + Conclusion

Implementation details

Key Parameter	Value
Pretraining dataset	ImageNet 1k
Backbone	ResNet-50
Projector	Three 8192-dim dense layers with batchnorm in first two
Batch size	2048
Optimizer	LARS
Base learn rate	0.2
Scheduler	Cosine decay with warm up

Hyperparameters

Hyperparameter	Value
Invariance coefficient λ	25
Variance coefficient μ	25
Covariance coefficient ν	1
Numerical stability constant ε	0.0001
Target embedding standard deviation γ	1

Introduction

Methods

Experimentation

Results

Discussion + Conclusion

ImageNet evaluations

Method	Linear		Semi-supervised			
	Top-1	Top-5	Top-1		Top-5	
			1%	10%	1%	10%
Supervised	76.5	-	25.4	56.4	48.4	80.4
MoCo He et al. (2020)	60.6	-	-	-	-	-
PIRL Misra & Maaten (2020)	63.6	-	-	-	57.2	83.8
CPC v2 Hénaff et al. (2019)	63.8	-	-	-	-	-
CMC Tian et al. (2019)	66.2	-	-	-	-	-
SimCLR Chen et al. (2020a)	69.3	89.0	48.3	65.6	75.5	87.8
MoCo v2 Chen et al. (2020c)	71.1	-	-	-	-	-
SimSiam Chen & He (2020)	71.3	-	-	-	-	-
SwAV Caron et al. (2020)	71.8	-	-	-	-	-
InfoMin Aug Tian et al. (2020)	73.0	91.1	-	-	-	-
OBoW Gidaris et al. (2021)	73.8	-	-	-	82.9	90.7
BYOL Grill et al. (2020)	74.3	91.6	53.2	68.8	78.4	89.0
SwAV (w/ multi-crop) Caron et al. (2020)	75.3	-	53.9	70.2	78.5	89.9
Barlow Twins Zbontar et al. (2021)	73.2	91.0	55.0	69.7	79.2	89.3
VICReg (ours)	73.2	91.1	54.8	69.5	79.4	89.5

Downstream tasks

Method	Linear Classification			Object Detection		
	Places205	VOC07	iNat18	VOC07+12	COCO det	COCO seg
Supervised	53.2	87.5	46.7	81.3	39.0	35.4
MoCo He et al. (2020)	46.9	79.8	31.5	-	-	-
PIRL Misra & Maaten (2020)	49.8	81.1	34.1	-	-	-
SimCLR Chen et al. (2020a)	52.5	85.5	37.2	-	-	-
MoCo v2 Chen et al. (2020c)	51.8	86.4	38.6	82.5	39.8	36.1
SimSiam Chen & He (2020)	-	-	-	82.4	-	-
BYOL Grill et al. (2020)	54.0	86.6	47.6	-	40.4†	37.0†
SwAV (m-c) Caron et al. (2020)	56.7	88.9	48.6	82.6	41.6	37.8
OBoW Gidaris et al. (2021)	56.8	89.3	-	82.9	-	-
Barlow Twins Grill et al. (2020)	54.1	86.2	46.5	82.6	40.0†	36.7†
VICReg (ours)	54.3	86.6	47.0	82.4	39.4	36.4

Multimodal pretraining

Method	Image-to-text			Text-to-Image		
	R@1	R@5	R@10	R@1	R@5	R@10
Contrastive (VSE++)	30.3	59.4	72.4	41.3	71.1	81.2
Barlow Twins	31.4	60,4	75.1	42.9	74.0	83.5
VICReg	33.6	62.7	77.9	45.2	76.1	84.2

Ablation: variance and covariance regularization

Method	ME	SG	PR	BN	No Reg	Var Reg	Var/Cov Reg
BYOL	✓	✓	✓	✓	69.3†	70.2	69.5
SimSiam		✓	✓	✓	67.9†	68.1	67.6
SimSiam		✓	✓		35.1	67.3	67.1
SimSiam		✓			collapse	56.8	66.1
VICReg			✓		collapse	56.2	67.3
VICReg			✓	✓	collapse	57.1	68.7
VICReg				✓	collapse	57.5	68.6†
VICReg					collapse	56.5	67.4

ME, Momentum Encoder, SG, stop-gradient, PR, predictor, BN, Batch normalization, No Reg, No additional regularization, Var Reg, Variance regularization, Var/Cov Reg, Variance and Covariance regularization.

Ablation: weight sharing

	SW R50	DW R50	DA R50/R101	DA R50/ViT-S
BYOL	69.3	✗	✗	✗
SimCLR	64.4	63.1	63.9	63.5
Barlow Twins	68.7	64.2	65.3	63.9
VICReg	68.6	66.5	68.1	66.2

SW, share weights, DW, share architecture with different weights, DA, different architectures

Ablation studies

Variance-covariance regularization

Method	λ	μ	ν	Top-1
Inv	1	0	0	collapse
Inv + Cov	25	0	1	collapse
Inv + Cov	0	25	1	collapse
Inv + Var	1	1	0	57.5
Inv + Var + Cov (VICReg)	1	1	1	collapse
	1	10	1	collapse
	10	1	1	collapse
	5	5	1	68.1
	10	10	1	68.2
	25	25	1	68.6
	50	50	1	68.3

Normalization

Representation	Embedding	Top-1
Std	None	68.6
Std	Std	68.4
None	Std	67.4
Std	None	67.2
None	l_2	65.1

Ablation studies

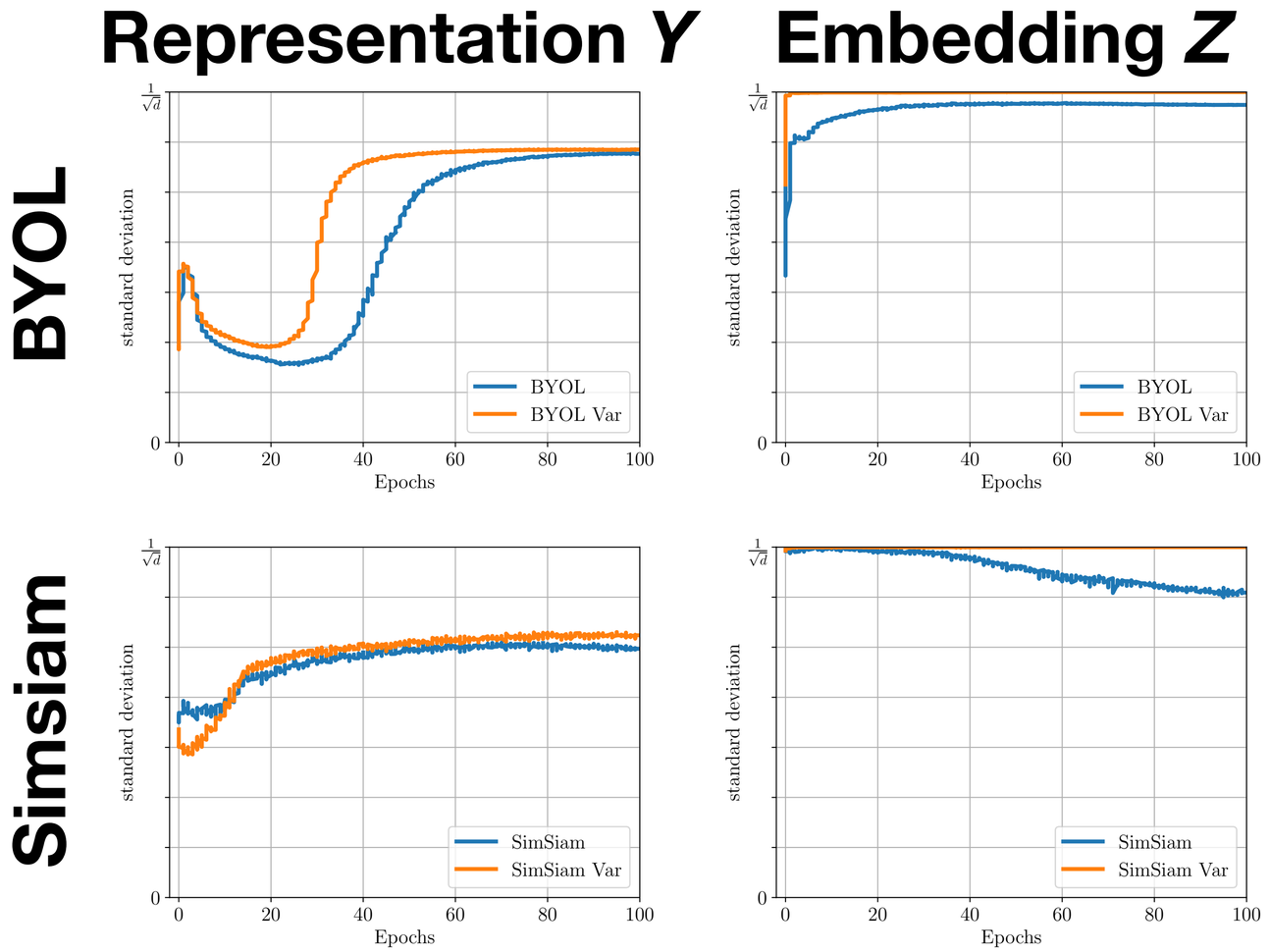
Expander size

Dimensionality	256	512	1024	2048	4096	8192	16834
Top-1	55.9	59.2	62.4	65.1	67.3	68.6	68.8

Batch size

Batch size	128	256	512	1024	2048	4096
Top-1	67.3	67.9	68.2	68.3	68.6	67.8

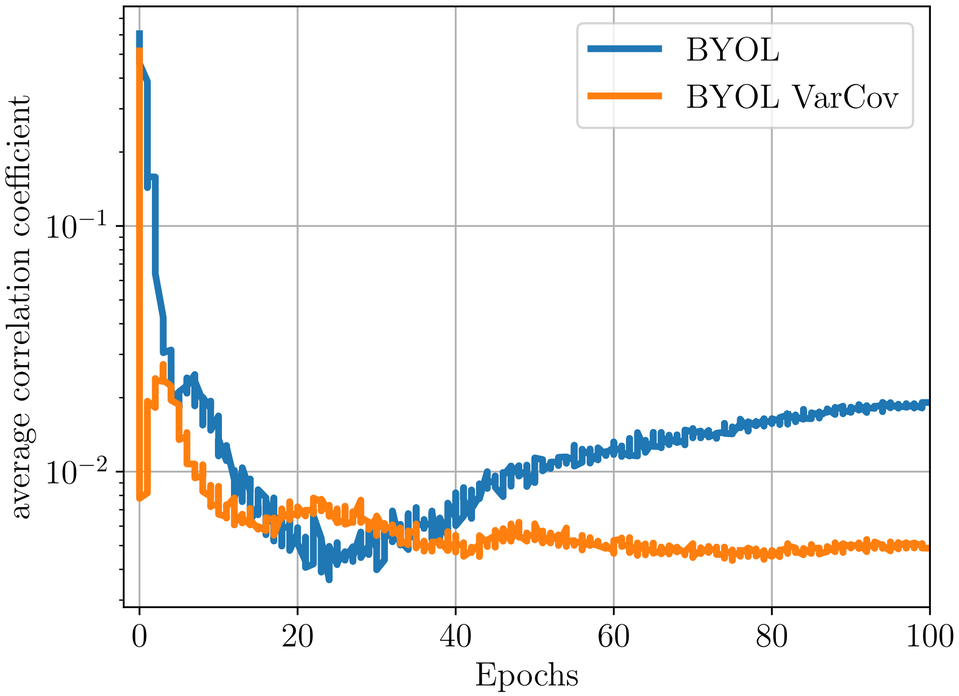
Feature standard deviation for BYOL and SimSiam



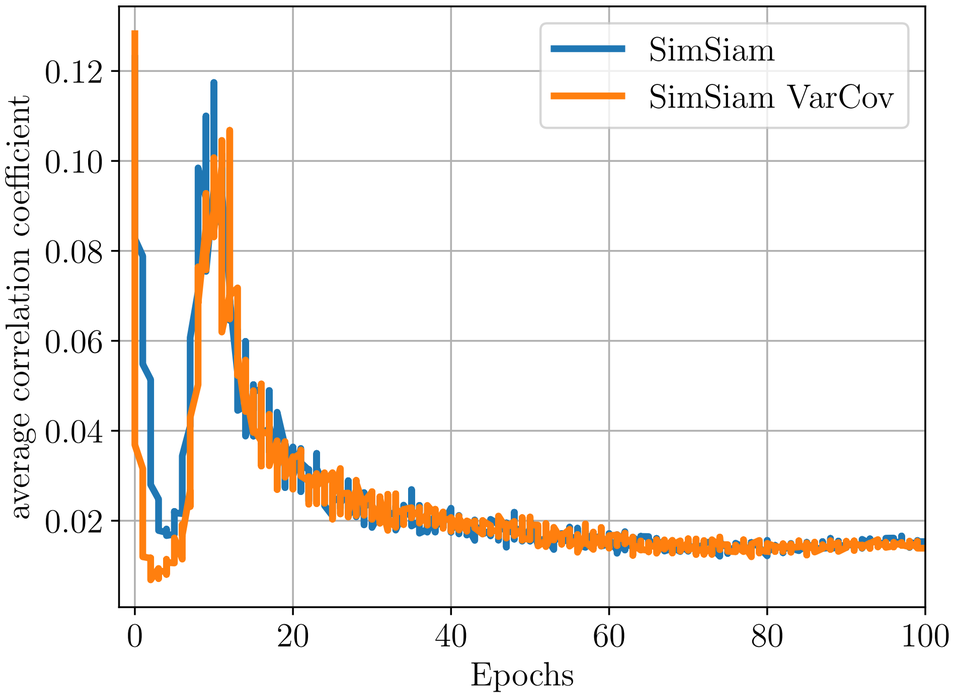
Feature covariance for BYOL and SimSiam

Representation *Y*

BYOL



Simsiam



“The computation time of the covariance matrix is quadratic in terms of the feature dimension, which slow the pre-training significantly.”

ICLR Reviewer QN96

Benchmarking

Method	time / 100 epochs	peak memory / GPU	Top-1 accuracy (%)
SwAV	9h	9.5G	71.8
SwAV (w/ multi-crop)	13h	12.9G	75.3
BYOL	10h	14.6G	74.3
Barlow Twins	12h	11.3G	73.2
VICReg	11h	11.3G	73.2

Introduction

Methods

Experimentation

Results

Discussion + Conclusion

“Authors give an explicit loss function to deal with the collapsed solution problem, which is understandable and explainable compared with BYOL and SimSiam. ”

ICLR Reviewer 8cmN

“The covariance term is borrowed from the Barlow Twins method, which decreases the originality and significance.”

NeurIPS Reviewer cVf4

“[The reported metrics are] not pushing the boundary of self-supervised learning.”

ICLR Reviewer dMLo

Invariance *s*

Learn invariance between views

Variance *v*

Force batch embedding vectors to be different

Covariance *c*

Decorrelate the variables of each embedding

On par with SOTA performance

Less architectural requirements

Self-supervised methods

Deep metric learning	Self distillation	Canonical correlation analysis	Masked image modeling
SimCLR nnCLR DirectCLR	MoCo SimSiam BYOL DINO	SwAV BarlowTwins VICReg	BEiT MAE MSN I-JEPA


UNIVERSITY OF
MICHIGAN

What is a trick that is still required to prevent collapse when training with VICReg loss?

Why is it important to use the standard deviation and not the variance in the *variance loss*?