# Self-Supervised Learning of Pretext-Invariant Representations

School of Industrial and Management Engineering, Korea University

Jong Kook, Heo

KOREA UNIVERSITY

DMQA

# Contents

# Research Purpose

❖ PIRL: Self-Supervised Learning of Pretext-Invariant Representations

- Facebook AI Resarch 에서 연구, 2022년 5월 20일 기준 약 619회 인용
- PIRL 이전 Pretext-task 기반의 SSL 방법들이 이미지 변환에 공변(Covariant)한다는 문제 지적
- Semantic Representation 은 Pretext task 의 이미지 변환에 Invariant 한 Representation 이어야함

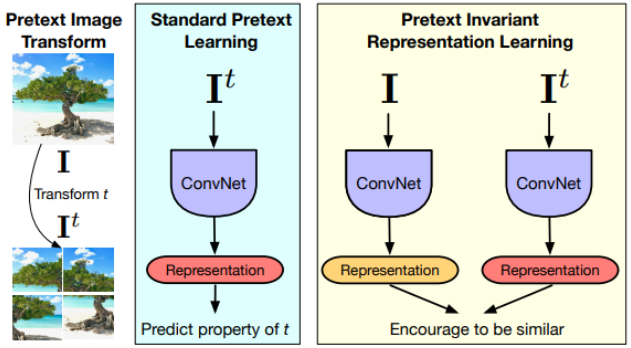



**Figure 1: Pretext-Invariant Representation Learning (PIRL).** Many pretext tasks for self-supervised learning [18, 46, 76] involve transforming an image $\mathbf{I}$, computing a representation of the transformed image, and predicting properties of transformation $t$ from that representation. As a result, the representation must *covary* with the transformation $t$ and may not contain much semantic information. By contrast, PIRL learns representations that are *invariant* to the transformation $t$ and retain semantic information.

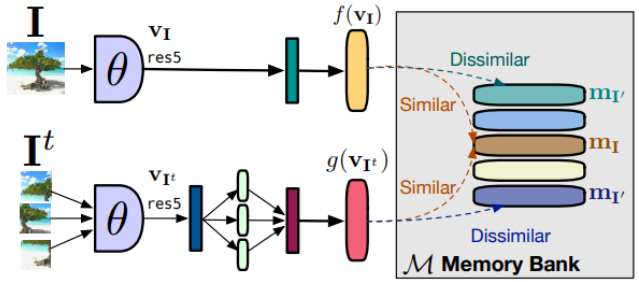



**Figure 3: Overview of PIRL.** Pretext-Invariant Representation Learning (PIRL) aims to construct image representations that are invariant to the image transformations $t \in \mathcal{T}$. PIRL encourages the representations of the image, $\mathbf{I}$, and its transformed counterpart, $\mathbf{I}^t$, to be similar. It achieves this by minimizing a contrastive loss (see Section 2.1). Following [72], PIRL uses a memory bank, $\mathcal{M}$, of negative samples to be used in the contrastive learning. The memory bank contains a moving average of representations, $\mathbf{m_I} \in \mathcal{M}$, for all images in the dataset (see Section 2.2).

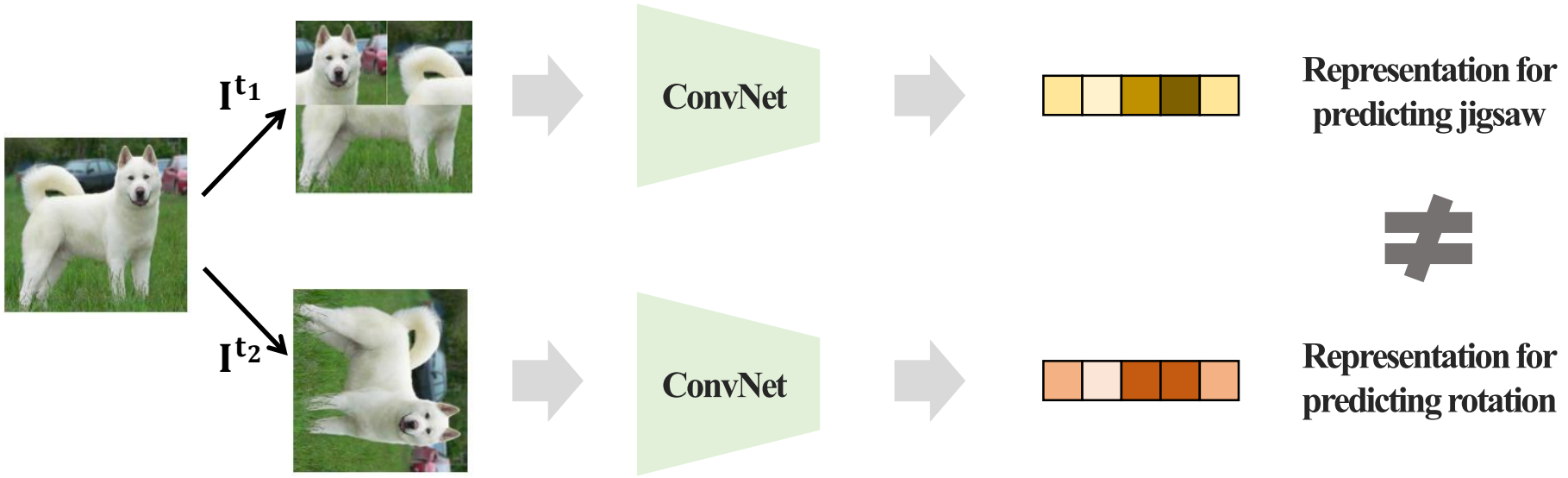


DMQA

# Research Purpose

❖ PIRL: Self-Supervised Learning of Pretext-Invariant Representations

- Facebook AI Resarch 에서 연구, 2022년 5월 20일 기준 약 619회 인용
- **PIRL 이전 Pretext-task 기반의 SSL 방법들이 이미지 변환에 공변(Covariant)한다는 문제 지적**
- Semantic Representation 은 Pretext task 의 이미지 변환에 Invariant 한 Representation 이어야함



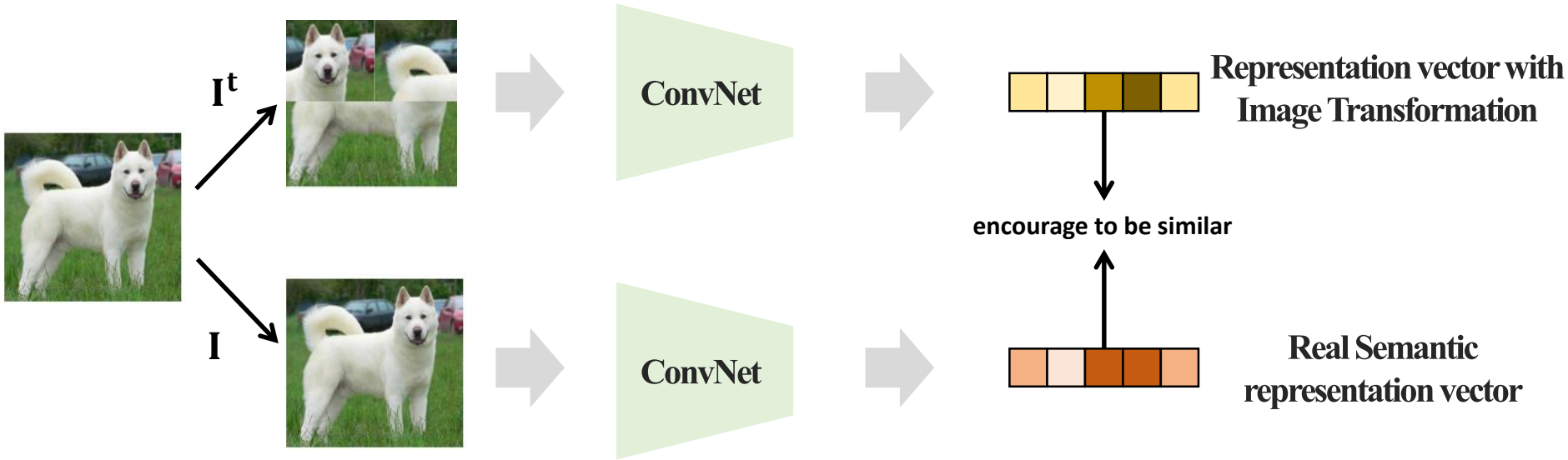



DMQA

# Research Purpose

❖ PIRL: Self-Supervised Learning of Pretext-Invariant Representations

- Facebook AI Resarch 에서 연구, 2022년 5월 20일 기준 약 619회 인용
- PIRL 이전 Pretext-task 기반의 SSL 방법들이 이미지 변환에 공변(Covariant)한다는 문제 지적
- **Semantic Representation 은 Pretext task 의 이미지 변환에 Invariant 한 Representation 이어야함**



# Overview

❖ Loss Function : Noise Contrastive Estimator

- Positive Pair : 원본 이미지($\mathbf{I}$)와 변환된 이미지($\mathbf{I^t}$)
- Negative Pairs : 변환된 이미지($\mathbf{I^t}$)와 N개의 다른 원본 이미지($\mathbf{I'}$)
- Similarity Function : Cosine Similarity
- NCE 는 Binary Event $(\mathbf{I}, \mathbf{I}^t)$ 가 발생할 확률을 아래와 같이 모사(Softmax Function)

$$h(\mathbf{v_I}, \mathbf{v_{I^t}}) = \frac{\exp\left(\frac{s(\mathbf{v_I}, \mathbf{v_{I^t}})}{\tau}\right)}{\exp\left(\frac{s(\mathbf{v_I}, \mathbf{v_{I^t}})}{\tau}\right) + \sum_{\mathbf{I'} \in \mathcal{D}_N} \exp\left(\frac{s(\mathbf{v_{I^t}}, \mathbf{v_{I'}})}{\tau}\right)}.$$

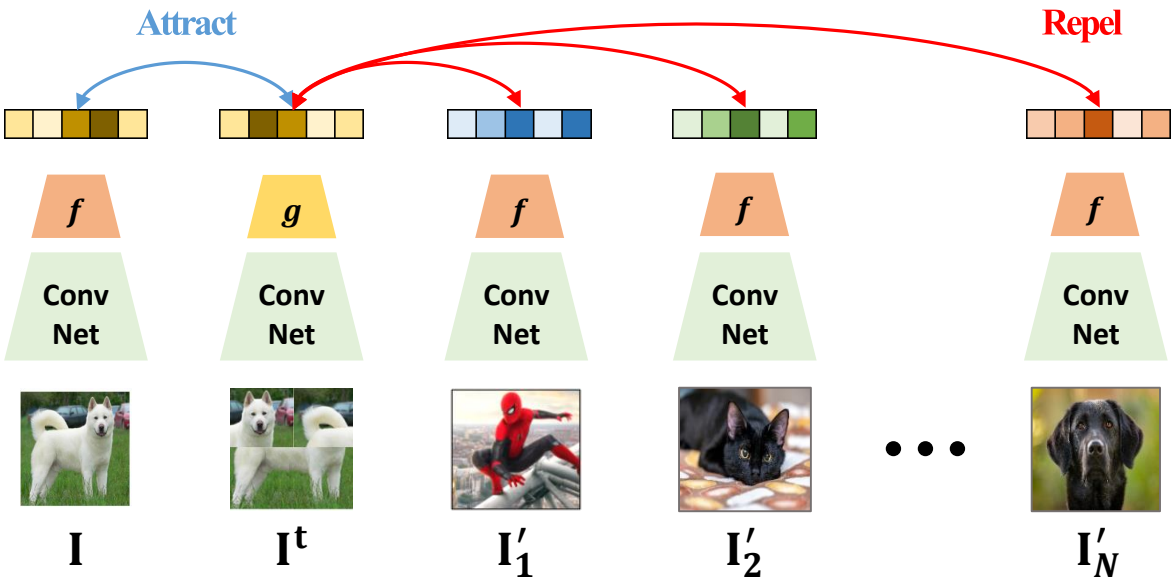
- Convolution Feature $\mathbf{v}$ 로부터 직접 유사도를 계산하지 않고 projection head 를 거친 후에 유사도 계산
- 원본 이미지에는 $f$ , 변환된 이미지에는 $g$ 라는 서로 다른 projection head 사용

# Overview

## ❖ Loss Function : Modification1

- Convolution Feature $\mathbf{v}$ 로부터 직접 유사도를 계산하지 않고 projection head 를 거친 후에 유사도 계산
- 원본 이미지에는 $f$ , 변환된 이미지에는 $g$ 라는 서로 다른 projection head 사용
- NCE Loss 는 아래와 같이 변형됨

$$L_{\mathrm{NCE}}\left(\mathbf{I}, \mathbf{I}^{t}\right) = -\log\left[h\left(f(\mathbf{v}_{\mathbf{I}}), g(\mathbf{v}_{\mathbf{I}^{t}})\right)\right] - \sum_{\mathbf{I}' \in \mathcal{D}_N} \log\left[1 - h\left(g(\mathbf{v}_{\mathbf{I}}^{t}), f(\mathbf{v}_{\mathbf{I}'})\right)\right]. \quad (4)$$






DMQA

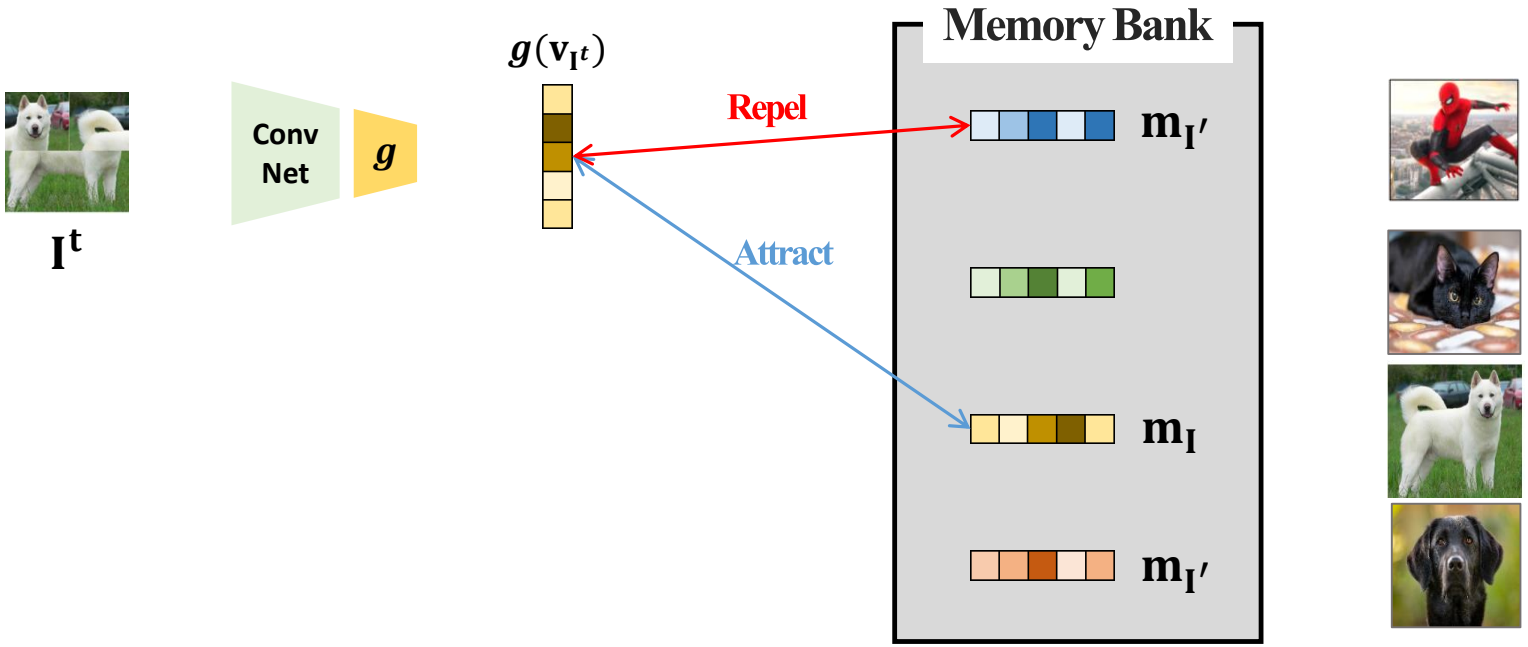
# Overview

## ❖ Loss Function : Modification2

- Memory Bank : 각기 다른 **원본 이미지에 대한 representation vector** 를 Memory Bank 에 저장하여 Negative Sample 을 랜덤 샘플링(NPID와 동일)

  *변환된 이미지에 대한 Representation Vector 는 저장하지 않음!

- Memory bank 에 저장되는 원본 이미지($\mathbf{I}$)에 대한 Representation $\mathbf{m}_{\mathbf{I}}$는 이전 epoch 들의 $f(\mathbf{v}_{\mathbf{I}})$의 지수이동 평균






DMQA

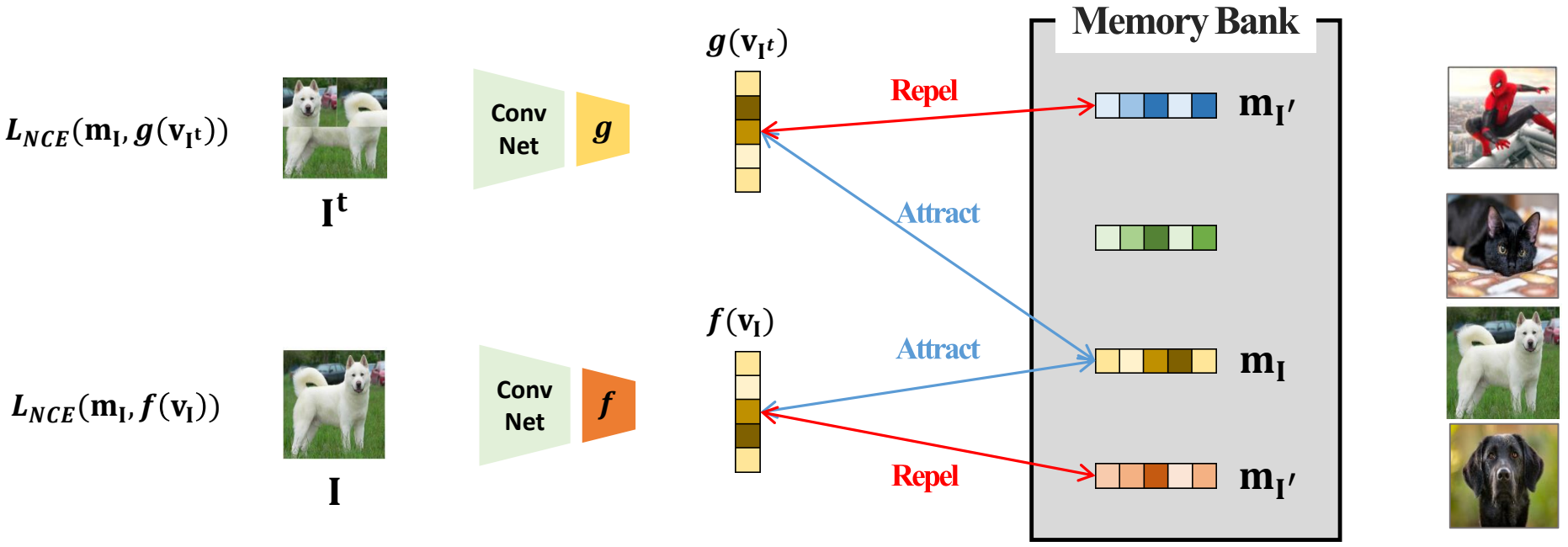
# Overview

❖ Loss Function : Modification3

- 이전 장의 Loss Function 은 서로 다른 원본 이미지 간의 비교는 진행하지 않았음
- 따라서 아래 두 Loss Function 의 Convex Optimization 을 통해 최종적인 Loss Function 을 제안

$$L\left(\mathbf{I}, \mathbf{I}^t\right) = \lambda L_{\mathrm{NCE}}(\mathbf{m}_{\mathbf{I}}, g(\mathbf{v}_{\mathbf{I}^t})) + (1-\lambda) L_{\mathrm{NCE}}(\mathbf{m}_{\mathbf{I}}, f(\mathbf{v}_{\mathbf{I}})). \quad (5)$$

- 이때 $\boldsymbol{\lambda} = \mathbf{0}$ 이면 NPID 와 동일






DMQA

# Experiment

## ❖ Experiment Overview

- Parameter Freezing : Feature Extractor 로써 인코더를 활용
- **Transfer Learning : Parameter Initialization 으로써 인코더를 활용**

## ❖ Object Detection

- Pre-Training : Faster RCNN 의 backbone 인 ResNet 50을 ImageNet Train 데이터 중 1.28M개를 통해 사전 학습
- Fine-Tuning : VOC07+12 Train 을 통해 전이 학습(지도 학습 수행)
- 평가지표 : AP-all, AP-50, AP-75
- ImageNet Supervised Pre-Training 보다 근소한 차이를 보이거나 우수한 성능을 보여줌

| Method | Network | $AP^{all}$ | $AP^{50}$ | $AP^{75}$ | $\Delta AP^{75}$ |
|---|---|---|---|---|---|
| Supervised | R-50 | 52.6 | **81.1** | 57.4 | =0.0 |
| Jigsaw [19] | R-50 | 48.9 | 75.1 | 52.9 | -4.5 |
| Rotation [19] | R-50 | 46.3 | 72.5 | 49.3 | -8.1 |
| NPID++ [72] | R-50 | 52.3 | 79.1 | 56.9 | -0.5 |
| PIRL **(ours)** | R-50 | **54.0** | 80.7 | **59.7** | **+2.3** |
| CPC-Big [26] | R-101 | – | 70.6* | – | |
| CPC-Huge [26] | R-170 | – | 72.1* | – | |
| MoCo [24] | R-50 | 55.2*† | 81.4*† | 61.2*† | |

**Table 1: Object detection on VOC07+12 using Faster R-CNN.** De-

# Experiment

## ❖ Experiment Overview

- **Parameter Freezing : Feature Extractor 로써 인코더를 활용**
- Transfer Learning : Parameter Initialization 으로써 인코더를 활용

## ❖ Image Classification with Linear Models

- Pre-Training : Faster RCNN 의 backbone 인 ResNet 50을 ImageNet Train 데이터 중 1.28M개를 통해 사전 학습, 인코더의 파라미터 고정
- Fine-Tuning : ImageNet,VOC07, Places205, iNaturalist 2018 4개에 데이터에 대해 Linear Classifier 학습

| Method | Parameters | Transfer Dataset | | | |
|---|---|---|---|---|---|
| | | ImageNet | VOC07 | Places205 | iNat. |
| ResNet-50 using evaluation setup of [19] | | | | | |
| Supervised | 25.6M | 75.9 | 87.5 | 51.5 | 45.4 |
| Colorization [19] | 25.6M | 39.6 | 55.6 | 37.5 | – |
| Rotation [18] | 25.6M | 48.9 | 63.9 | 41.4 | 23.0 |
| NPID++ [72] | 25.6M | 59.0 | 76.6 | 46.4 | 32.4 |
| MoCo [24] | 25.6M | 60.6 | – | – | – |
| Jigsaw [19] | 25.6M | 45.7 | 64.5 | 41.2 | 21.3 |
| PIRL (**ours**) | 25.6M | **63.6** | **81.1** | **49.8** | **34.1** |
| Different architecture or evaluation setup | | | | | |
| NPID [72] | 25.6M | 54.0 | – | 45.5 | – |
| BigBiGAN [12] | 25.6M | 56.6 | – | – | – |
| AET [76] | 61M | 40.6 | – | 37.1 | – |
| DeepCluster [6] | 61M | 39.8 | – | 37.5 | – |
| Rot. [33] | 61M | 54.0 | – | 45.5 | – |
| LA [80] | 25.6M | 60.2† | – | 50.2† | – |
| CMC [64] | 51M | 64.1 | – | – | – |
| CPC [51] | 44.5M | 48.7 | – | – | – |
| CPC-Huge [26] | 305M | 61.0 | – | – | – |
| BigBiGAN-Big [12] | 86M | 61.3 | – | – | – |
| AMDIM [4] | 670M | 68.1 | – | 55.1 | – |

**Table 2: Image classification with linear models.** Image-classification

# Conclusion

- Pretext Task 에 사용되는 이미지 변환 기법에 상관없이 적용할 수 있는 대조 학습을 제안

- 당시 제안되었던 Pretext-task 나 MoCo, CPC 등 다른 대조학습 기법보다도 우수한 성능을 보임

- Jigsaw Puzzle 로 학습한 것보다 Jigsaw Augmentation 을 통해 PIRL 을 학습하였을 경우 성능이 8%~18% 증가(PIRL 이 Transform Covariant 하지 않는 Semantic Representation 을 잘 추출하는 것을 입증)