Self-supervised Learning

- 2022 겨울 방학 세미나 -

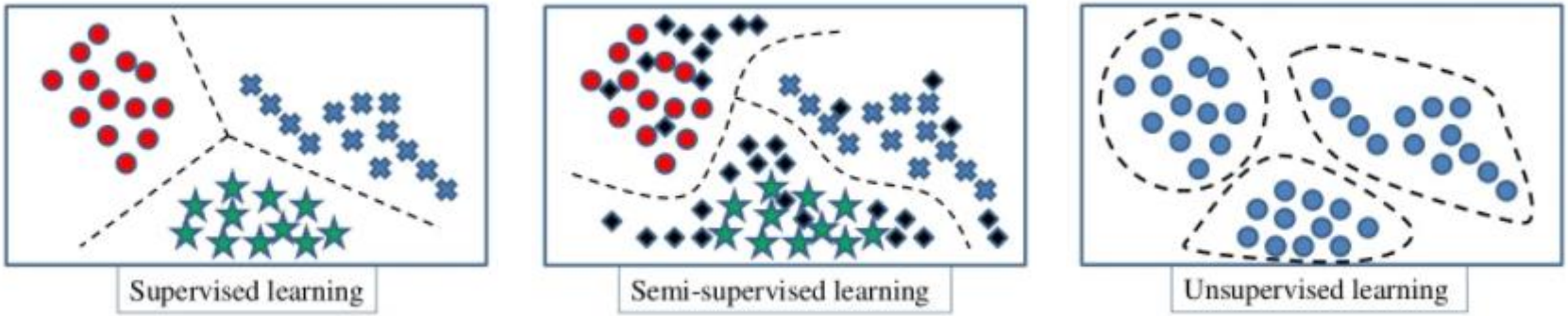
이 창 헌
Vision and Display System Lab.
Sogang University

Outline

- Background
  - Self-supervised learning
- Contrastive Learning methods
- Masked image modeling methods
  - MAE: Masked Autoencoders Are Scalable Vision Learners

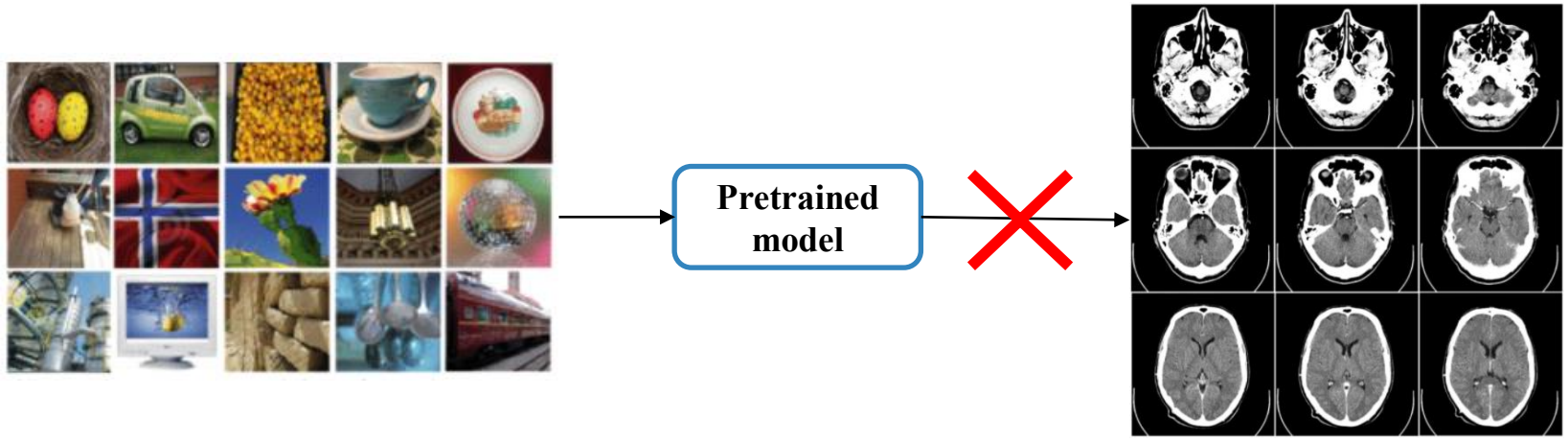
Background

- Supervised learning vs …
  - Transfer learning, domain adaptation
    - 유사한 task에서 학습된 network의 weight를 가져와서 target task에 적용
  - Semi-supervised learning
    - 데이터셋 중에 일부만 labeling을 하여 학습
  - Weakly-supervised learning
    - 기존의 label (segmentation mask)보다 적은 비용으로 얻은 label(bounding box)으로 학습
  - Unsupervised learning
    - Label되지 않은 데이터셋을 사용하여 학습



Background

- Self-supervised learning
  - Unsupervised learning의 한 분야
    - 딥러닝 모델 학습 시 초기 학습 속도와 성능을 높이기 위해 pretrained weight 를 사용
      - Pretraining에 사용하고자 하는 데이터셋이 존재하나 labeling 되어 있지 않은 경우 unsupervised learning을 수행할 필요가 있음
        - ✓의료 영상의 분야와 같이 특수한 영상을 사용하는 경우

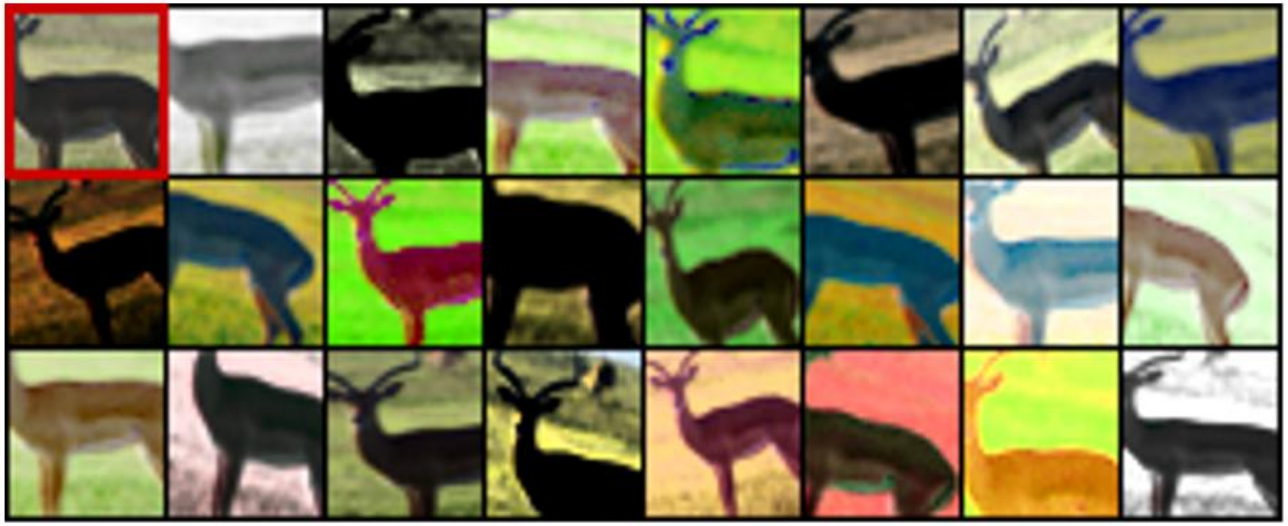


<Natural images from ImageNet>

<Medical images for target task>

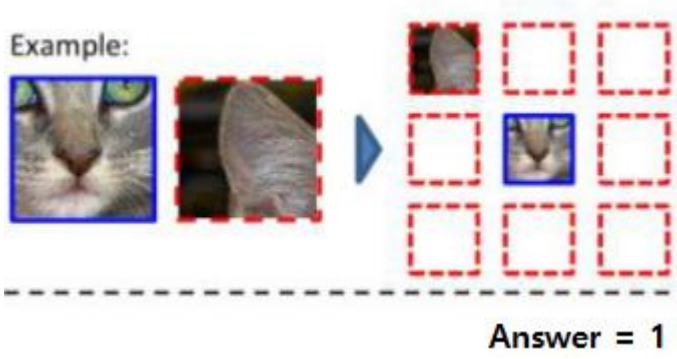
Background

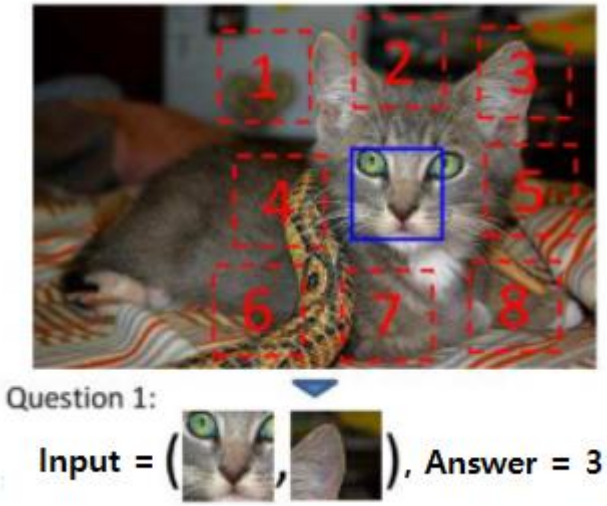
- Pretext tasks examples
  - Exempler[1] (2014 NIPS)
    - Object 가 존재하는 영역을 patch로 crop하고 augmentation을 적용하여 patch 개수를 늘림
      - 같은 patch로부터 생성된 patch는 모두 같은 class로 구분하도록 학습
    - Dataset 크기만큼 classifier가 구분해야 하는 class가 늘어남



Background

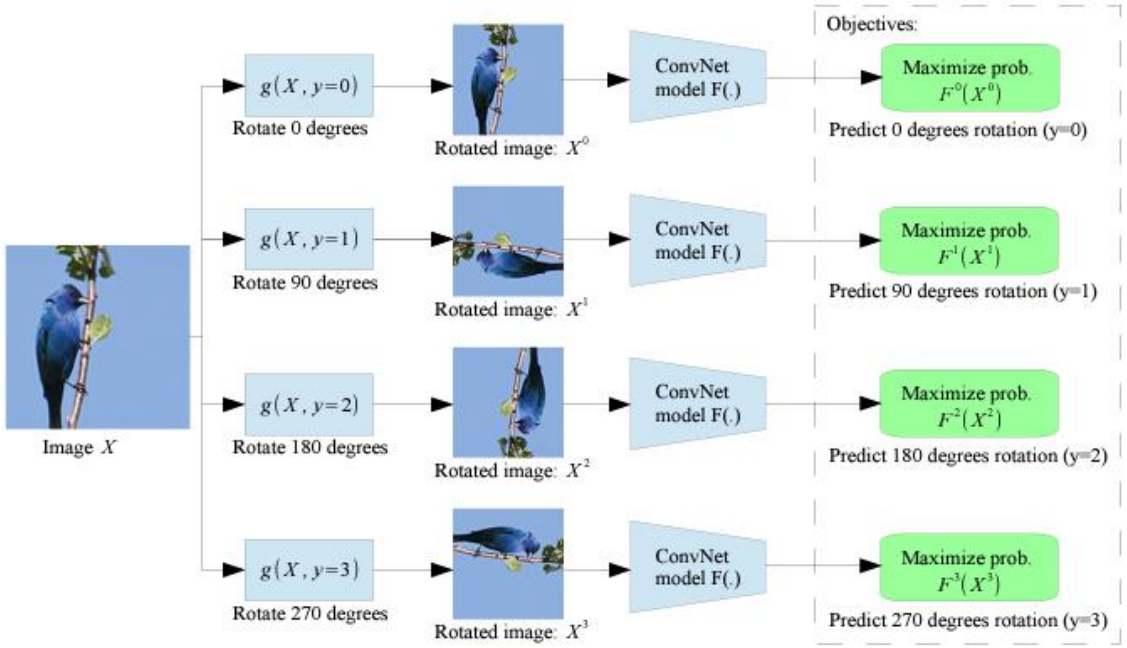
- Pretext tasks examples
  - Context Prediction[1] (2015 ICCV)
    - 8 class classification task
    - 중앙 위치의 patch + 8 방향 중 random patch가 주어졌을 때, random patch의 상대적인 위치를 classify하는 문제





Background

- Pretext tasks examples
  - Rotation[1] (2018 ICLR)
    - 4 class classification task
    - Input image에 0°, 90°, 180°, 270° 회전을 random하게 적용 후, 원본을 기준으로 적용된 회전 각도를 classify하는 문제

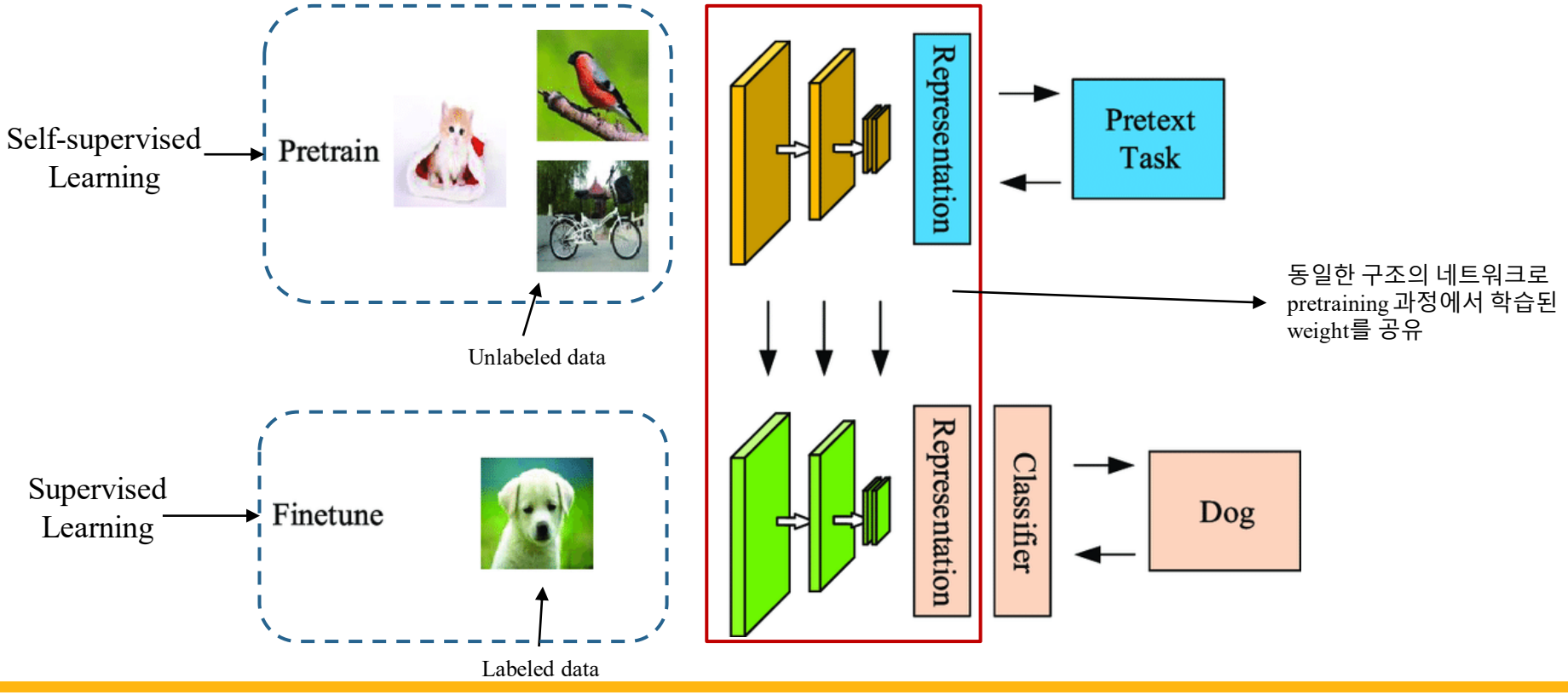


Background

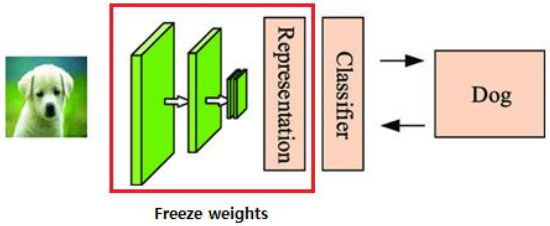
- Self-supervised learning 평가 방법

  1) Pretext task에서 unlabeled data를 사용하여 network를 학습(pretrain)

  2) Pretrain 된 network를 downstream task로 transfer learning 후 성능을 평가



Background



Finetuning 과정에서는 추가된 last layer(linear classifier)만 학습됨

- ImageNet linear probing 성능 비교
  - Pretrain model을 transfer하는 방법을 비교했을 때 supervised learning이 더 높은 classification accuracy 달성
  - Transfer 효과를 비교하기 위해서 30 epoch만 학습하였기 때문에 random initial weight를 가지고 scratch부터 train하는 방법으로 높은 성능을 달성하지 못함

Supervised learning(pretraining)

Training from scratch

Self-supervised learning(pretraining)

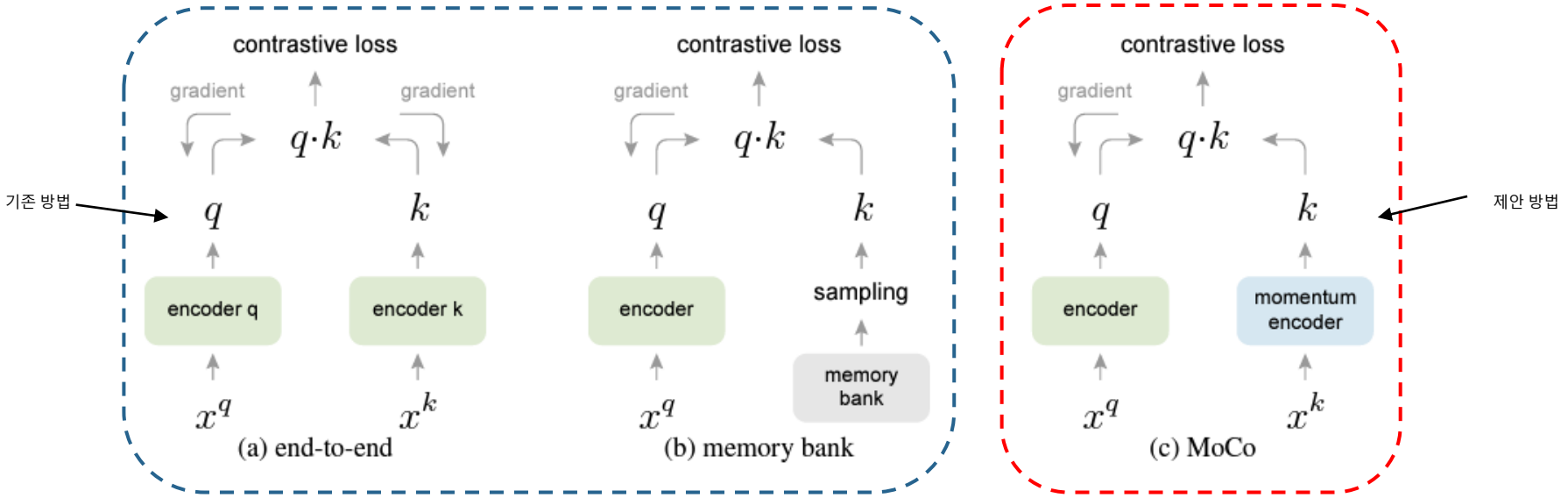
Method	Conv1	Conv2	Conv3	Conv4	Conv5
ImageNet labels	19.3	36.3	44.2	48.3	50.5
Random	11.6	17.1	16.9	16.3	14.1
Random rescaled Krähenbühl et al. (2015)	17.5	23.0	24.5	23.2	20.6
Context (Doersch et al., 2015)	16.2	23.3	30.2	31.7	29.6
Context Encoders (Pathak et al., 2016b)	14.1	20.7	21.0	19.8	15.5
Colorization (Zhang et al., 2016a)	12.5	24.5	30.4	31.5	30.3
Jigsaw Puzzles (Noroozi & Favaro, 2016)	18.2	28.8	34.0	33.9	27.1
BIGAN (Donahue et al., 2016)	17.7	24.5	31.0	29.9	28.0
Split-Brain (Zhang et al., 2016b)	17.7	29.3	35.4	35.2	32.8
Counting (Noroozi et al., 2017)	18.0	30.6	34.3	32.5	25.7
(Ours) RotNet	**18.8**	**31.7**	**38.7**	**38.2**	**36.5**

Contrastive Learning methods

- Contrastive learning
  - ImageNet linear evaluation시 supervised learning으로 학습한 pretrained model에 근접한 성능 달성
  - Method
    - Batch의 모든 image에 서로 다른 augmentation을 가한 pair를 생성함
    - 같은 image이지만 다른 augmentation이 가해진 sample을 positive pair, 다른 image에 augmentation이 가해진 sample들을 negative pair라고 정의함
    - Image를 encoder에 통과시켰을 때 positive pair들의 feature representation은 거리가 가까워지도록, negative pair와의 feature representation은 거리가 멀어지도록 학습
  - Contrastive learning의 한계
    - Negative pair의 개수에 따라서 성능이 좌우됨
      - ⁘ 학습 시 large batch size (256 ~ 8192) 필요
    - Negative pair를 정의하는 augmentation 중 어떤 것을 사용하는지에 따라 성능 차이가 심함
      - ⁘ 적절한 augmentation 선정 필요

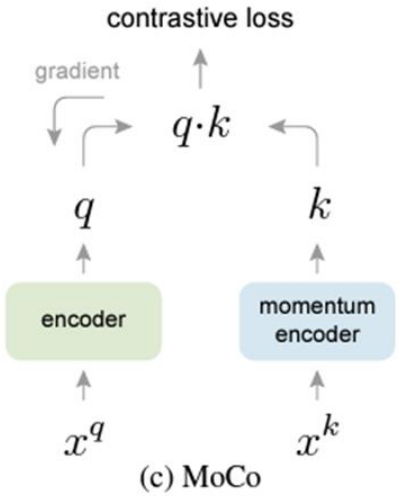
Contrastive Learning methods

- MoCo v1[1] (CVPR 2020)
  - 기존의 contrastive learning methods의 문제점 해결
    - Batch size: end-to-end 방식
    - Inconsistency: memory bank 방식
    - Momentum encoder를 사용하는 dynamic dictionary를 사용하여 문제 해결



Contrastive Learning methods

- MoCo v1[1] (CVPR 2020)
  - Momentum encoder
    - Momentum encoder parameters : θ_k
    - Visual representation encoder parameters : θ_q
    - Encoder 에서 학습된 weight 에 momentum 을 주어 update
      - ∴ Update 수식 $\theta_k \leftarrow m\theta_k + (1 - m)\,\theta_q$
    - Momentum coefficient m : $m \in [0,1]$, default = 0.999

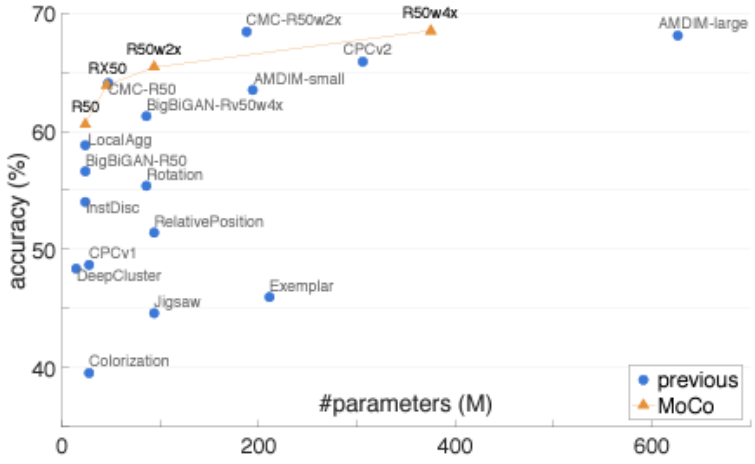


(c) MoCo

  - Dynamic dictionary
    - Memory bank 방식은 모든 sample을 encoder에 통과시켜 추출한 feature representation을 memory bank 에 저장 후 sampling 하여 사용할 때마다 update
    - MoCo는 FIFO 방식으로 가장 최근의 mini batch의 sample들을 momentum encoder에 통과시켜 추출한 feature representation을 dictionary (default size = 65536) 에 저장하고 negative pair로 사용
      - ∴ Momentum encoder는 서서히 update되기 때문에 inconsistency가 해결됨

Contrastive Learning methods

- MoCo v1[1] (CVPR 2020)
  - Downstream task에서 성능 비교



<기존 방법들과 ImageNet에서 linear probing 성능 비교>

pre-train	COCO keypoint detection AP^{kp}	AP^{kp}_{50}	AP^{kp}_{75}
random init.	65.9	86.5	71.7
super. IN-1M	65.8	86.9	71.9
MoCo IN-1M	66.8 (+1.0)	87.4 (+0.5)	72.5 (+0.6)
MoCo IG-1B	66.9 (+1.1)	87.8 (+0.9)	73.0 (+1.1)

pre-train	COCO dense pose estimation AP^{dp}	AP^{dp}_{50}	AP^{dp}_{75}
random init.	39.4	78.5	35.1
super. IN-1M	48.3	85.6	50.6
MoCo IN-1M	50.1 (+1.8)	86.8 (+1.2)	53.9 (+3.3)
MoCo IG-1B	50.6 (+2.3)	87.0 (+1.4)	54.3 (+3.7)

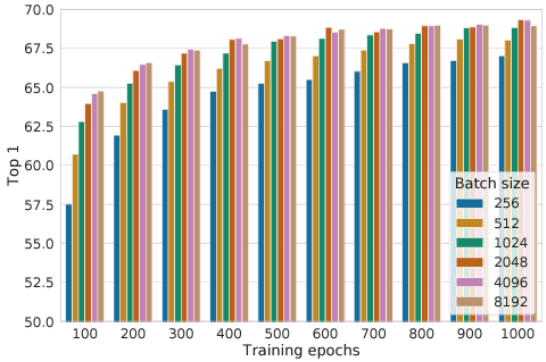
pre-train	LVIS v0.5 instance segmentation AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
random init.	22.5	34.8	23.8
super. IN-1M†	24.4	37.8	25.8
MoCo IN-1M	24.1 (−0.3)	37.4 (−0.4)	25.5 (−0.3)
MoCo IG-1B	24.9 (+0.5)	38.2 (+0.4)	26.4 (+0.6)

pre-train	Cityscapes instance seg. AP^{mk}	AP^{mk}_{50}	Semantic seg. (mIoU) Cityscapes	VOC
random init.	25.4	51.1	65.3	39.5
super. IN-1M	32.9	59.6	74.6	74.4
MoCo IN-1M	32.3 (−0.6)	59.3 (−0.3)	75.3 (+0.7)	72.5 (−1.9)
MoCo IG-1B	32.9 (0.0)	60.3 (+0.7)	75.5 (+0.9)	73.6 (−0.8)

< Downstream task에서 supervised pretrained model과 성능 비교>

Contrastive Learning methods

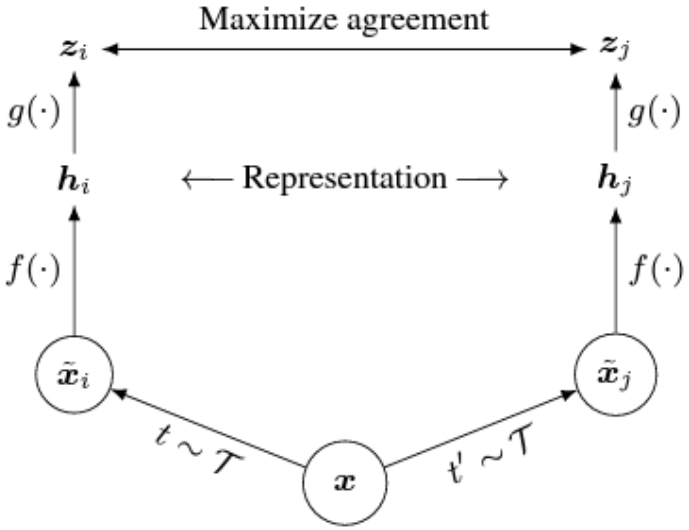
- SimCLR v1[1] (ICML 2020)
  - End-to-end 방식 사용
    - Default batch size = 4096
  - Base encoder network $f(\cdot)$ 뒤에 projection head (MLP) g(·)추가
  - Data augmentation 최적화를 위한 search 진행



<Batchsize에 따른 ImageNet linear probing 성능 비교>

Method	Architecture	Param (M)	Top 1
Methods using ResNet-50:			
Local Agg.	ResNet-50	24	60.2
MoCo	ResNet-50	24	60.6
PIRL	ResNet-50	24	63.6
CPC v2	ResNet-50	24	63.8
SimCLR (ours)	ResNet-50	24	**69.3**
Methods using other architectures:			
Rotation	RevNet-50 (4×)	86	55.4
BigBiGAN	RevNet-50 (4×)	86	61.3
AMDIM	Custom-ResNet	626	68.1
CMC	ResNet-50 (2×)	188	68.4
MoCo	ResNet-50 (4×)	375	68.6
CPC v2	ResNet-161 (*)	305	71.5
SimCLR (ours)	ResNet-50 (2×)	94	74.2
SimCLR (ours)	ResNet-50 (4×)	375	**76.5**

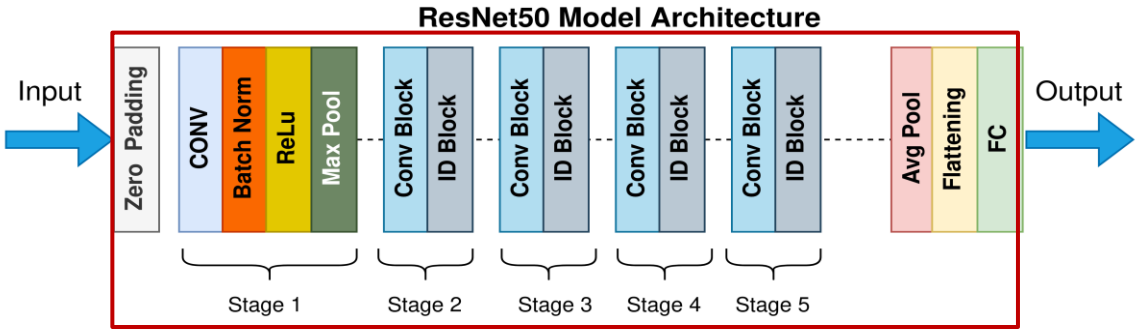
<기존 방법들과 ImageNet에서 linear probing 성능 비교>



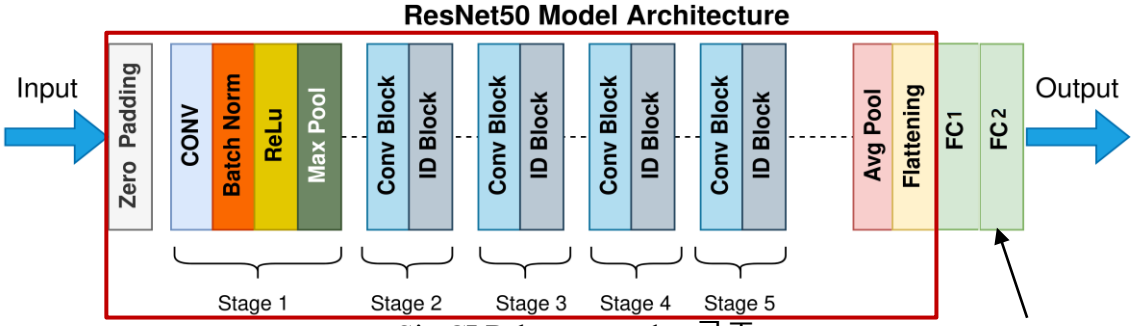
<SimCLR framework>

Contrastive Learning methods

- SimCLR v1[1] (ICML 2020)
  - Projection head g(·)는 pre-training 과정에서만 사용됨
    - 학습이 완료된 후에는 projection head를 제외한 구조에서 visual representation을 추출

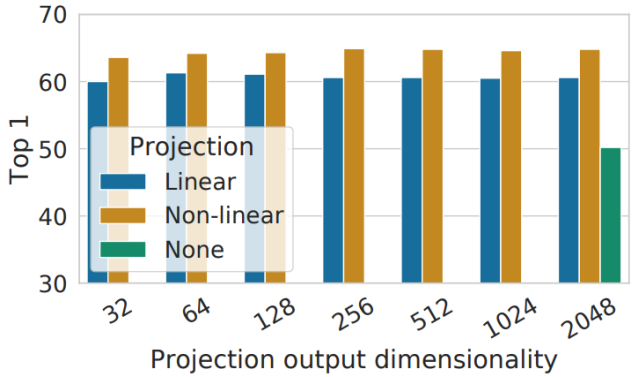


<MoCo base encoder 구조>



<SimCLR base encoder 구조>

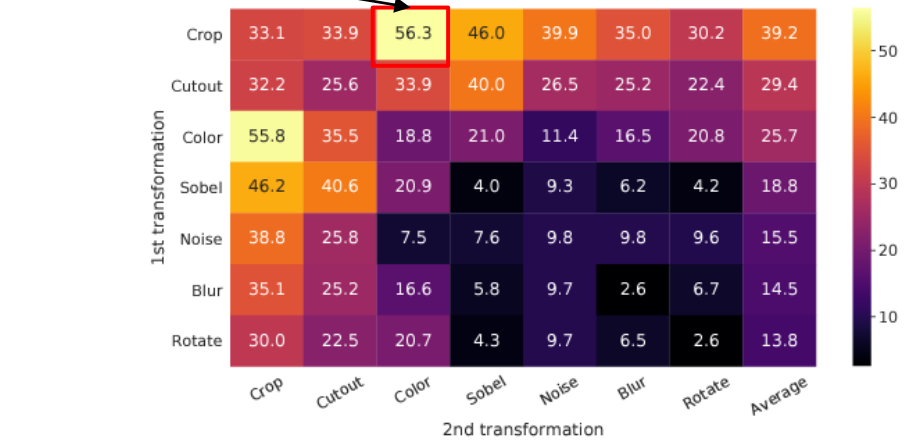
Pre-training에만 사용됨



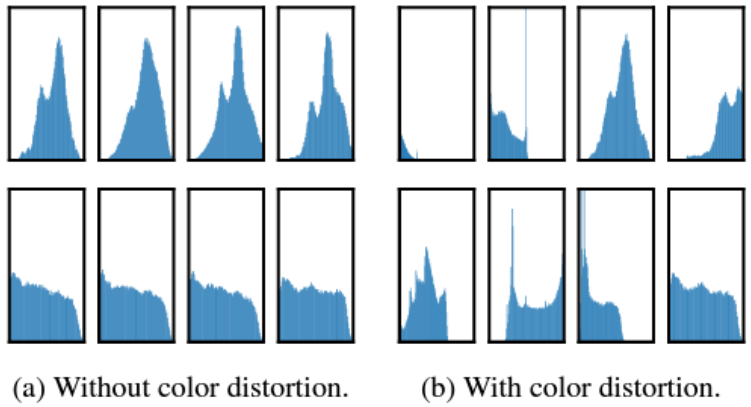
<Projection head 구조에 따른 성능 비교>

Contrastive Learning methods

- SimCLR v1[1] (ICML 2020)
  - Data augmentation 최적화를 위한 search 진행
    - Contrastive learning method는 하나의 sample에 두 개의 서로 다른 augmentation을 적용
      - 선택된 data augmentation pair에 따라서 성능이 크게 좌우됨
    - Crop과 함께 사용 시 가장 좋은 결과를 취득할 수 있는 augmentation search
      - ImageNet은 data sample의 크기가 일정하지 않기 때문에 crop은 반드시 필요
      - Crop + color distortion 조합이 가장 높은 성능 달성



<Augmentation 조합 성능 비교>

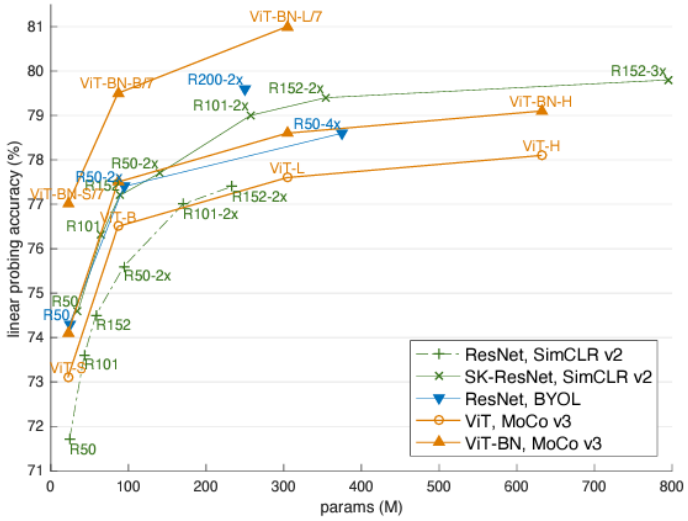


(a) Without color distortion. (b) With color distortion.

<이미지의 pixel intensity에 대한 histogram>

Contrastive Learning methods

- MoCo vs SimCLR
  - MoCo v3[1] (ICCV 2021)
    - SimCLR[2] 의 projection head 개념 채용
      - Pretraining 과정에서 base encoder 뒤에 projection head + prediction head를 추가, momentum encoder 뒤에는 prediction head만 추가
    - 기존의 dictionary queue 구조대신 SimCLR과 batch size 4096의 end to end 방식 사용
    - Encoder의 backbone으로 ResNet50 대신 ViT-BN 사용



Masked image modeling methods

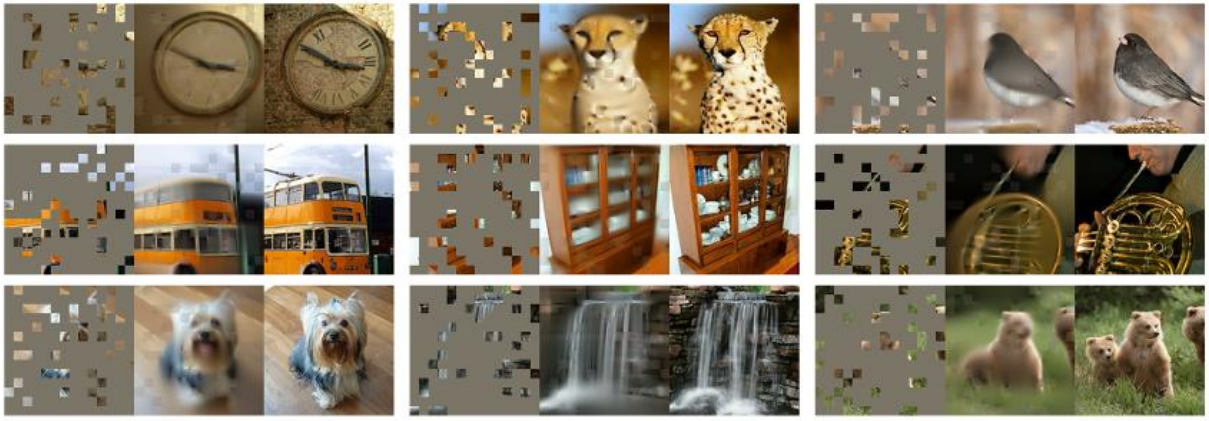
- MAE[1)] : Masked Autoencoders Are Scalable Vision Learners
  - Motivation
    - NLP 분야에서 BERT는 MLM(masked language model)의 self-supervised learning을 수행하여 downstream task에서 성능을 향상
      - MLM은 임의로 몇 개의 토큰을 mask하고, mask 된 토큰을 예측하는 것을 목표로 함
        - ✓기존의 left-to-right 구조와는 다르게 왼쪽 문맥과 오른쪽 문맥을 모두 학습할 수 있음
        - ✓Next-sentence-prediction task를 함께 수행 가능
      - Pretraining dataset
        - ✓BooksCorpus (800M words)
        - ✓English Wikipedia (2,500M words)
      - Pretrain된 BERT 모델을 다양한 NLP task에서 fine-tuning 시 높은 성능 달성

System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	Average -
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$BERT_{BASE}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$BERT_{LARGE}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

< GLUE dataset에서 BERT 성능 비교>

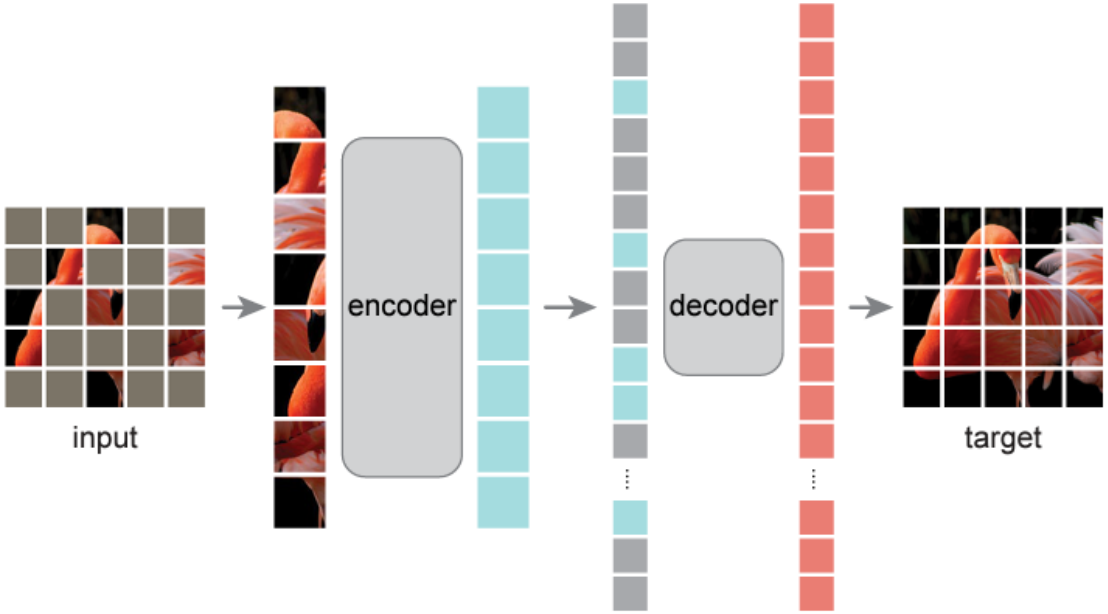
Masked image modeling methods

- MAE[1)] : Masked Autoencoders Are Scalable Vision Learners
  - Implementing masking to vision
    - MLM에서 motivation을 얻은 기존의 MIM 방법은 성능 향상에 성공하지 못함
      - Vision에서 다루는 image는 NLP가 다루는 word의 차이에 주목
        - ✓이미지는 spatial redundancy가 높기 때문에 주변 픽셀로부터 masking 된 픽셀을 예측 가능
        - ✓BERT는 데이터셋의 15%을 masking
        - ✓MAE는 image를 patch로 나누고 전체에서 75%를 masking하여 성능 향상



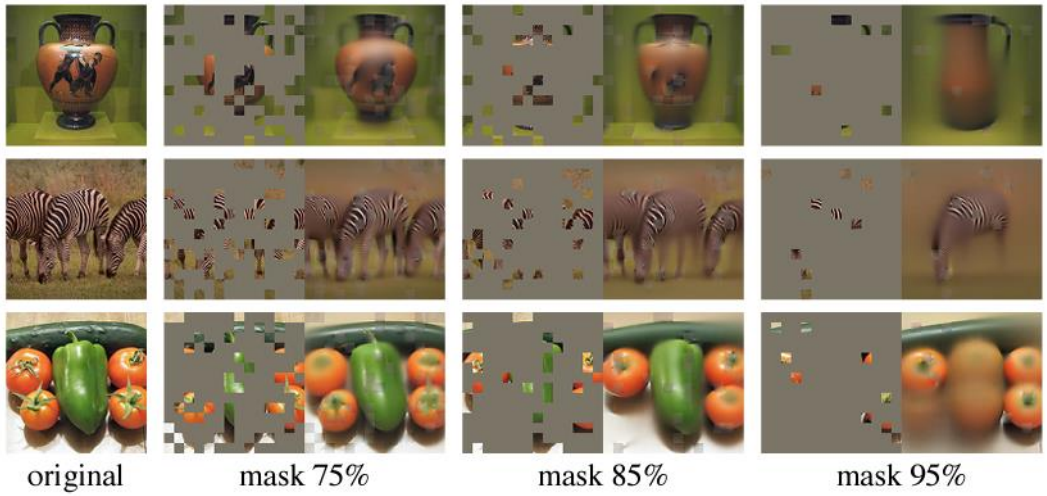
Masked image modeling methods

- MAE[1] : Masked Autoencoders Are Scalable Vision Learners
  - Method
    - Image에서 25%의 token만을 encoder의 입력으로 사용, latent vector에 mask token을 붙여서 decoder를 통해서 reconstruct



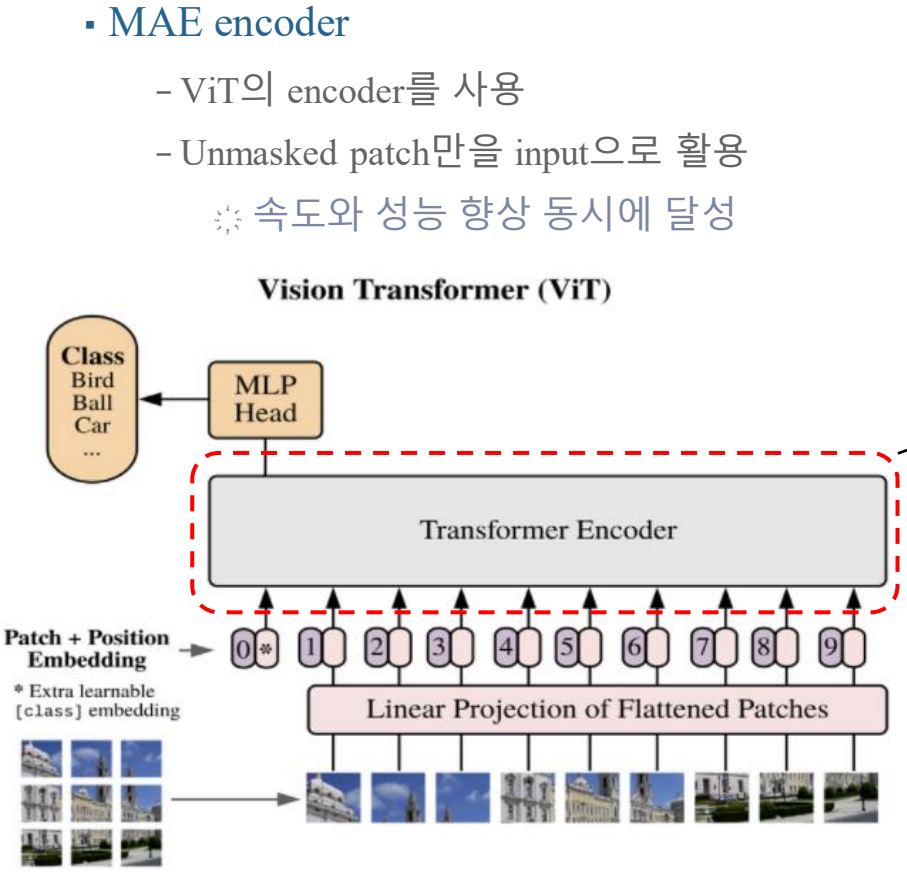
Masked image modeling methods

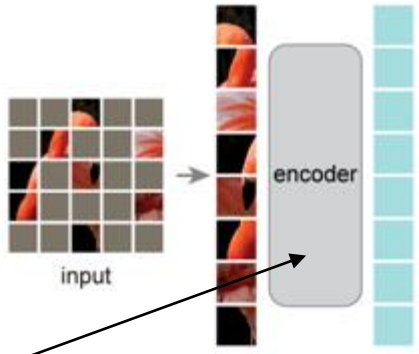
- MAE[1] : Masked Autoencoders Are Scalable Vision Learners
  - Masking
    - 중복되지 않는 patch로 이미지를 나누고 uniform distribution으로 patch를 sampling함
      - 특정 영역에 patch가 집중되는 bias 방지
    - 높은 masking 비율을 사용하여 주변 patch 정보로부터 쉽게 reconstruction을 수행할 수 없도록 유도함
      - 매우 높은 masking 비율을 사용하는 실험에서도 GT에 가까운 reconstruction 생성



Masked image modeling methods

- MAE[1)] : Masked Autoencoders Are Scalable Vision Learners
  - MAE encoder
    - ViT의 encoder를 사용
    - Unmasked patch만을 input으로 활용
      - 속도와 성능 향상 동시에 달성



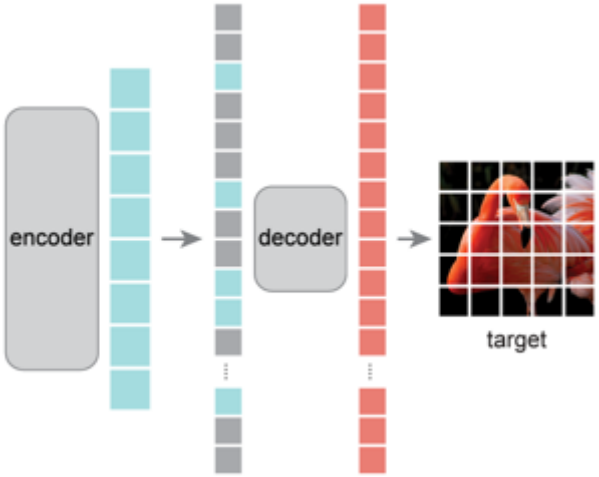


case	ft	lin	FLOPs
encoder w/ [M]	84.2	59.6	3.3×
encoder w/o [M]	**84.9**	**73.5**	**1**×

(c) **Mask token**. An encoder without mask tokens is more accurate and faster

Masked image modeling methods

- MAE[1)] : Masked Autoencoders Are Scalable Vision Learners
  - MAE decoder
    - Encoder와 비 대칭적인 light weight decoder 사용
      - 8 transformer blocks width 512-d
    - Reconstruction을 위해서 encoder에서 나온 visible patch와 mask token을 모두 사용
      - Positional embedding에 따라서 mask token을 원래 자리로 배치
    - Pre-training 과정에서만 사용되며 downstream task로 transfer 시에는 encoder만 사용함



blocks	ft	lin
1	84.8	65.5
2	**84.9**	70.0
4	**84.9**	71.9
8	**84.9**	**73.5**
12	84.4	73.3

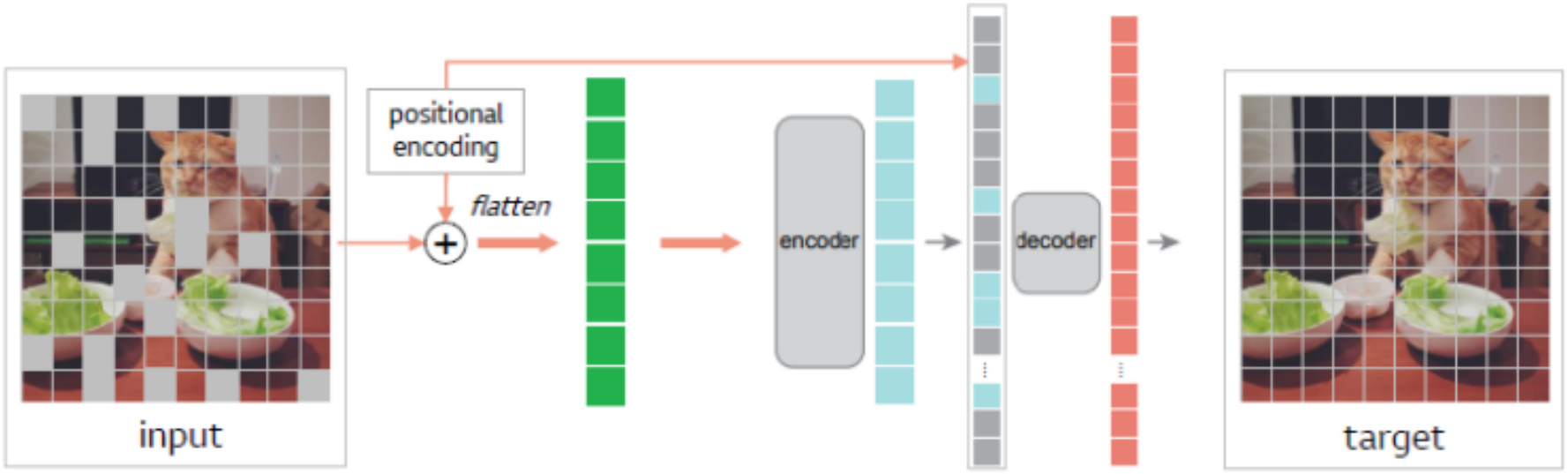
(a) **Decoder depth**. A deep decoder can improve linear probing accuracy.

dim	ft	lin
128	**84.9**	69.1
256	84.8	71.3
512	**84.9**	**73.5**
768	84.4	73.1
1024	84.3	73.1

(b) **Decoder width**. The decoder can be narrower than the encoder (1024-d).

Masked image modeling methods

- MAE[1] : Masked Autoencoders Are Scalable Vision Learners
  - Reconstruction target
    - Loss function으로 MSE(mean square error)를 사용
      - Target인 원본 이미지의 patch와 reconstructed patch의 차이를 줄여 나가도록 학습
  - Implementation



Masked image modeling methods

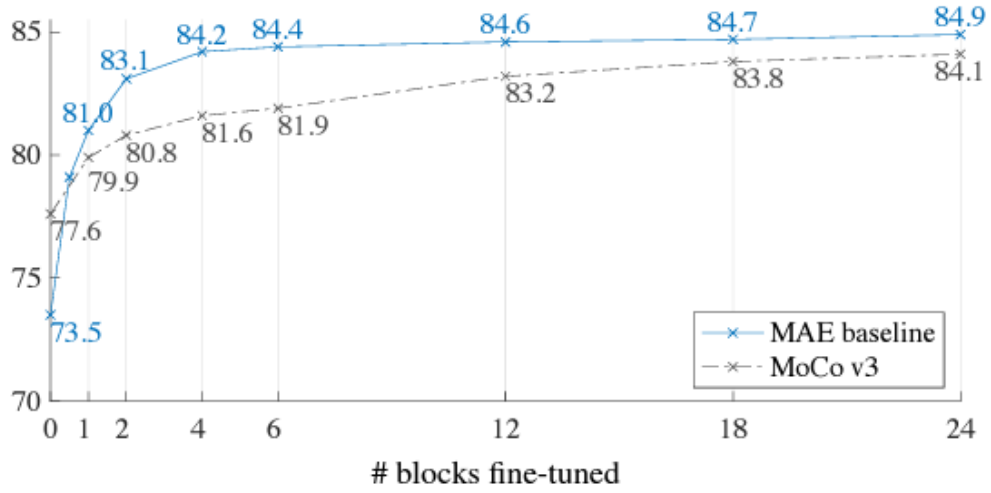
- MAE[1] : Masked Autoencoders Are Scalable Vision Learners
  - 논문 실험 결과
    - ImageNet linear probing에서 기존의 contrastive learning 방법에 비해 낮은 성능 달성
      - Pre-training 과정에서 사용된 decoder가 원인으로 생각됨
        - ✓ ViT encoder의 24개의 transformer block 중 단 하나의 block이라도 fine-tuning되면 contrastive learning 방법보다 높은 성능 달성

method	model	params	acc
iGPT [6]	iGPT-L	1362 M	69.0
iGPT [6]	iGPT-XL	6801 M	72.0
BEiT [2]	ViT-L	304 M	52.1†
MAE	ViT-B	86 M	68.0
MAE	ViT-L	304 M	75.8
MAE	ViT-H	632 M	76.6

Table 12. **Linear probing results of masked encoding methods.**

blocks	ft	lin
1	84.8	65.5
2	**84.9**	70.0
4	**84.9**	71.9
8	**84.9**	**73.5**
12	84.4	73.3

(a) **Decoder depth.** A deep decoder can improve linear probing accuracy.



<ViT-L encoder의 transformer block을 partial fine-tuning한 결과 비교>

Masked image modeling methods

- MAE[1)] : Masked Autoencoders Are Scalable Vision Learners
  - 논문 실험 결과
    - ImageNet finetuning 및 downstream task로 transfer learning 시 기존 방법들에 비해 높은 성능 달성

method	pre-train data	ViT-B	ViT-L	ViT-H	ViT-H$_{448}$
scratch, our impl.	-	82.3	82.6	83.1	-
DINO [5]	IN1K	82.8	-	-	-
MoCo v3 [9]	IN1K	83.2	84.1	-	-
BEiT [2]	IN1K+DALLE	83.2	85.2	-	-
MAE	IN1K	83.6	85.9	86.9	**87.8**

Table 3. **Comparisons with previous results on ImageNet-**

< 기존 방법들과 ImageNet finetuning 성능 비교 >

		AP^{box}		AP^{mask}	
method	pre-train data	ViT-B	ViT-L	ViT-B	ViT-L
supervised	IN1K w/ labels	47.9	49.3	42.9	43.9
MoCo v3	IN1K	47.9	49.3	42.7	44.0
BEiT	IN1K+DALLE	49.8	**53.3**	44.4	47.1
MAE	IN1K	**50.3**	**53.3**	**44.9**	**47.2**

Table 4. **COCO object detection and segmentation** using a ViT

method	pre-train data	ViT-B	ViT-L
supervised	IN1K w/ labels	47.4	49.9
MoCo v3	IN1K	47.3	49.1
BEiT	IN1K+DALLE	47.1	53.3
MAE	IN1K	**48.1**	**53.6**

Table 5. **ADE20K semantic segmentation** (mIoU) using Uper-

dataset	ViT-B	ViT-L	ViT-H	ViT-H$_{448}$	prev best
iNat 2017	70.5	75.7	79.3	**83.4**	75.4 [50]
iNat 2018	75.4	80.1	83.0	**86.8**	81.2 [49]
iNat 2019	80.5	83.4	85.7	**88.3**	84.1 [49]
Places205	63.9	65.8	65.9	**66.8**	66.0 [19]†
Places365	57.9	59.4	59.8	**60.3**	58.0 [36]‡

Table 6. **Transfer learning accuracy on classification datasets,**

< Downstream task에서 pretrained model 성능 비교>