11. Representation Learning

GEV6135 Deep Learning for Visual Recognition and Applications

Kibok Lee

Assistant Professor of

Applied Statistics / Statistics and Data Science

Nov 24, 2022

연세대학교
YONSEI UNIVERSITY

Assignment 6

- Due **Friday 12/2, 11:59pm KST**
- Convolutional networks
  - Modularized implementation (loss is given!)
  - BatchNorm is given, but should be plugged into DeepConvNet

- Before submitting your work, we recommend you
  - Re-download clean files
  - Copy-paste your solution to clean py
  - Re-run clean ipynb only once
- If you feel difficult, consider to take **option 2**.

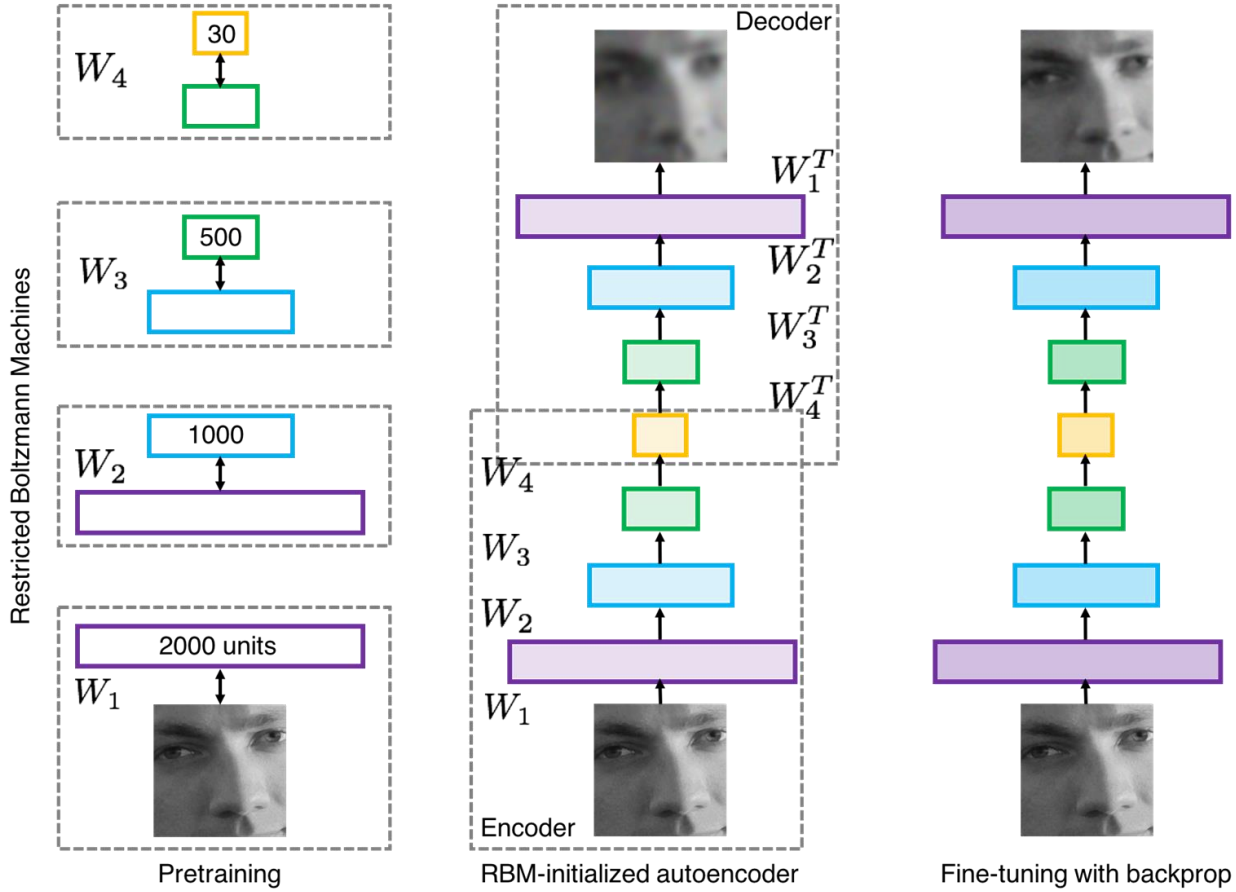
Outline

- Trends of Representation Learning by Decade
- Supervised Learning for Transfer Learning
- Self-Supervised Learning (Roughly, ~ 2019)
- Self-Supervised Learning (Roughly, 2020 ~)

Representation Learning in the 2000s

- To make DNNs working
  - Training DNNs was challenging
  - Not a mainstream research topic at the time

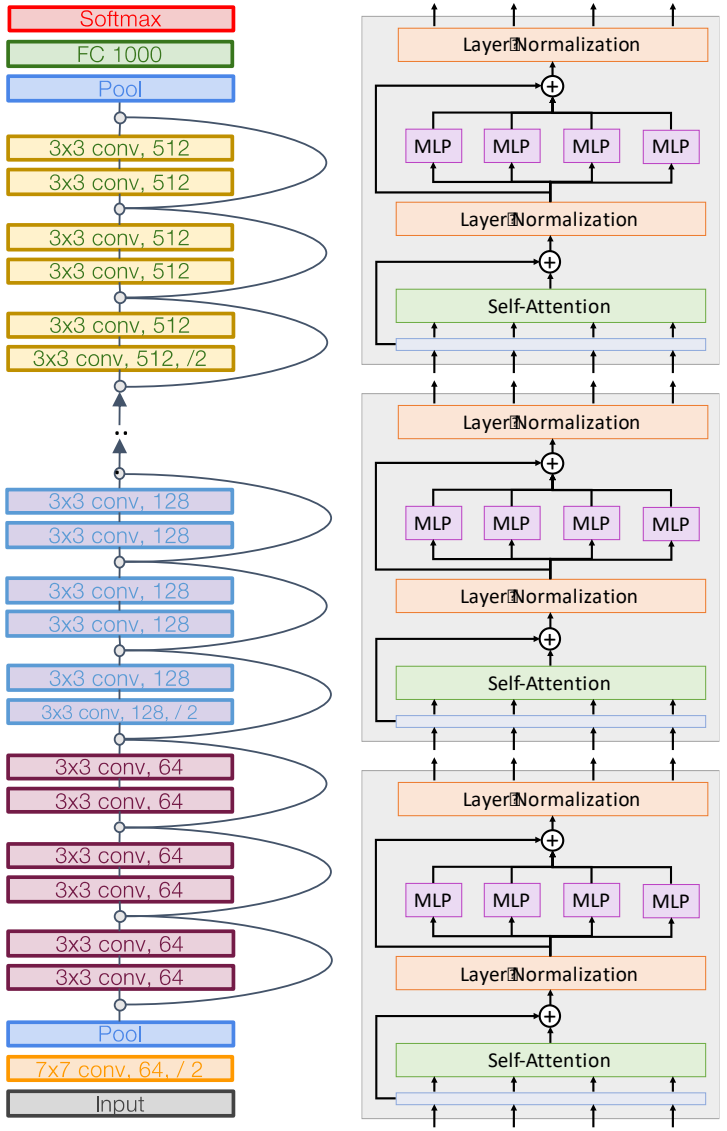
- Layer-wise RBM
- Sparse coding
- Autoencoding



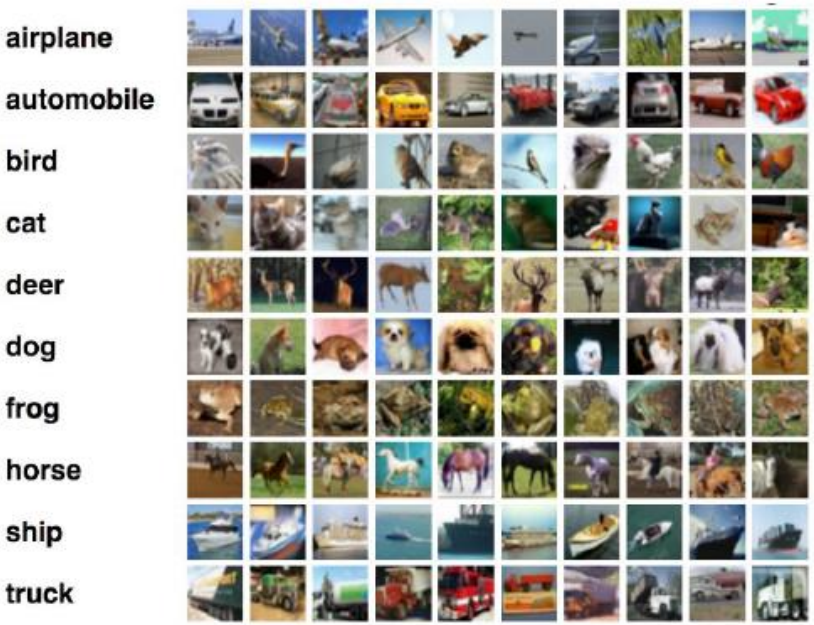
Hinton and Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks", Science 2006
Bengio et al, "Greedy Layer-Wise Training of Deep Networks", NeurIPS 2007
Lee et al, "Sparse deep belief net model for visual area V2", NeurIPS 2007
Lee et al, "Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations", ICML 2009
Vincent et al, "Stacked Denoising Autoencoders", JMLR 2010

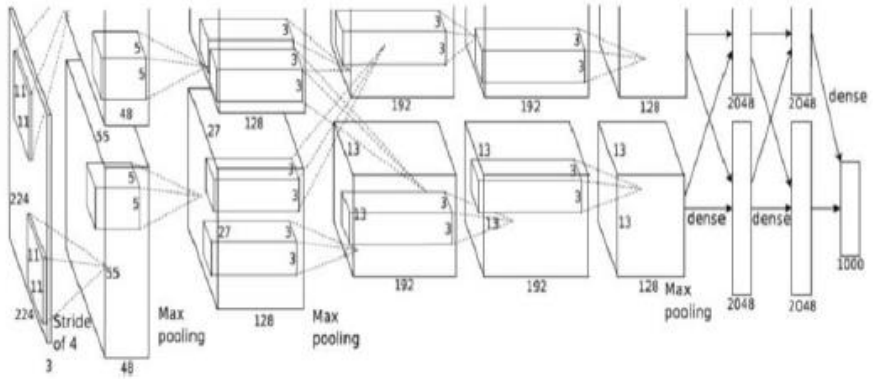
Representation Learning in the 2020s

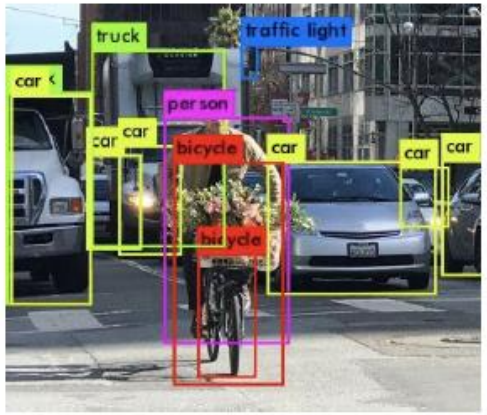
- To train DNNs efficiently
  - Also better few-shot performance
- Supervised Learning for Transfer Learning
  - State-of-the-art representation learning in the 2010s
  - Still competitive if no unlabeled data in the 2020s
- Self-Supervised Learning
  - Taking advantage of huge unlabeled datasets
  - Outperforms supervised learning in the 2020s



How Deep Learning Works







- Define the set of (visual) concepts to be learned
- Collect diverse and large number of examples for each of them
- Train a deep model for several GPU hours (or days)

Supervised Learning Is Expensive

- Supervised learning requires to curate large human-annotated datasets
  - Annotating + cleaning raw data
- Time consuming and expensive

  - Annotating such image: ~ 1.5h
- Error prone (human mistakes)

Supervised Learning Is Expensive

- Assume you want to label **1M** images. How much will it cost?

(1,000,000 images) (Small to medium sized dataset)

× (10 seconds/image) (Fast annotation)

× (1/3600 hours/second)

× (10k won/hour) (Low wage paid to annotator)

= 2777,7778 won

(Other assumptions: one annotator per image, no benefits / payroll tax / crowdsourcing fee for annotators; not accounting for time to set up tasks for annotators, etc. Real costs could easily be 3x this or more)

Supervised Learning Is Expensive

- Assume you want to label **1B** images. How much will it cost?

(1,000,000,000 images) (Large dataset)

× (10 seconds/image) (Fast annotation)

× (1/3600 hours/second)

× (10k won/hour) (Low wage paid to annotator)

= **277,7777,7778 won**

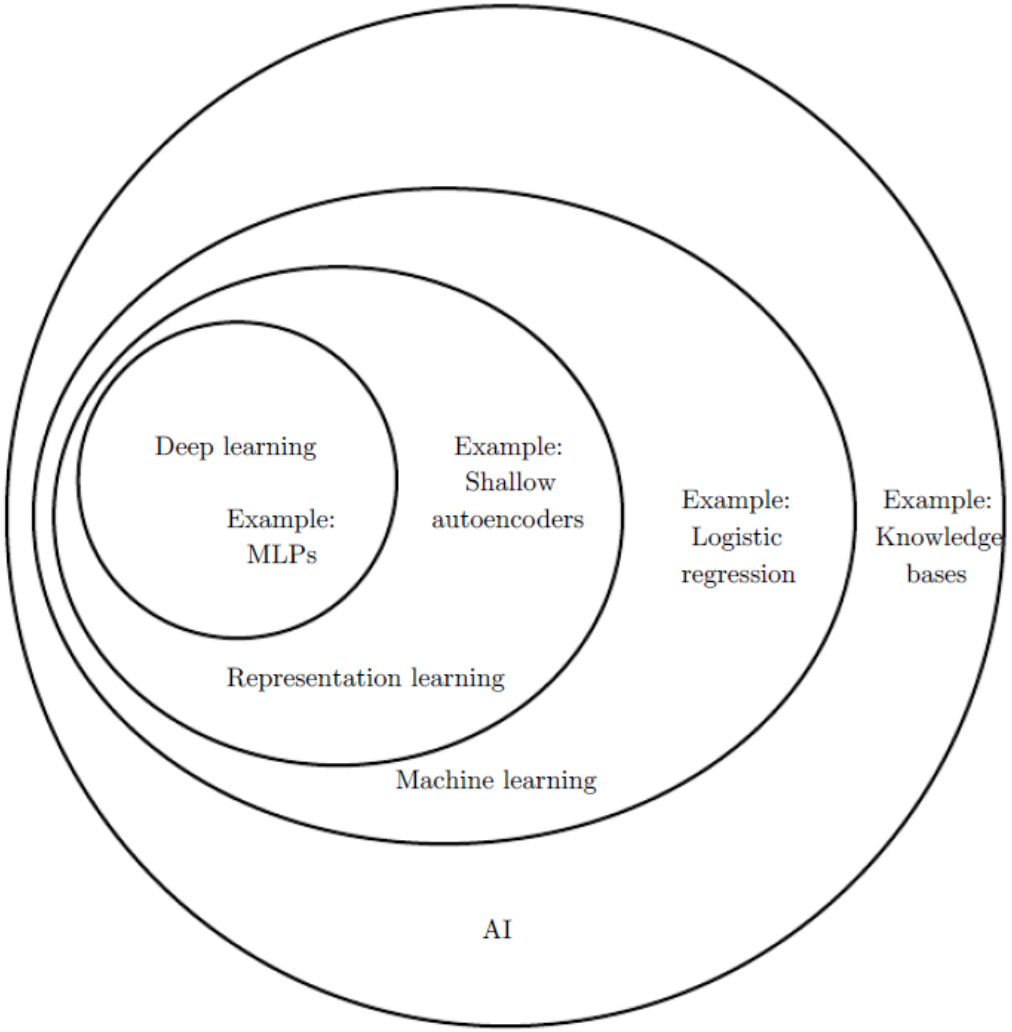
(Other assumptions: one annotator per image, no benefits / payroll tax / crowdsourcing fee for annotators; not accounting for time to set up tasks for annotators, etc. Real costs could easily be 3x this or more)

Supervised Learning Is Not How We Learn

Babies don’t get supervision for everything they see!

Representation Learning

- "Good representations" are
  - reusable, smooth, spatially coherent, disentangled, hierarchical, semantically meaningful
- (Most) Deep learning models consist of backbone and task-specific layers
  - Backbone = a good feature extractor
    - C.f. before DL, hand-designed features
  - Task-specific layers = classifier/regressor
- **Q. Can we pretrain a good backbone for efficient deep learning?**



https://dmitry.ai/t/topic/175

Supervised Learning for Transfer Learning

Supervised Learning for Transfer Learning

1. Pretrain on a supervised learning task
   - Usually ImageNet-1k classification task

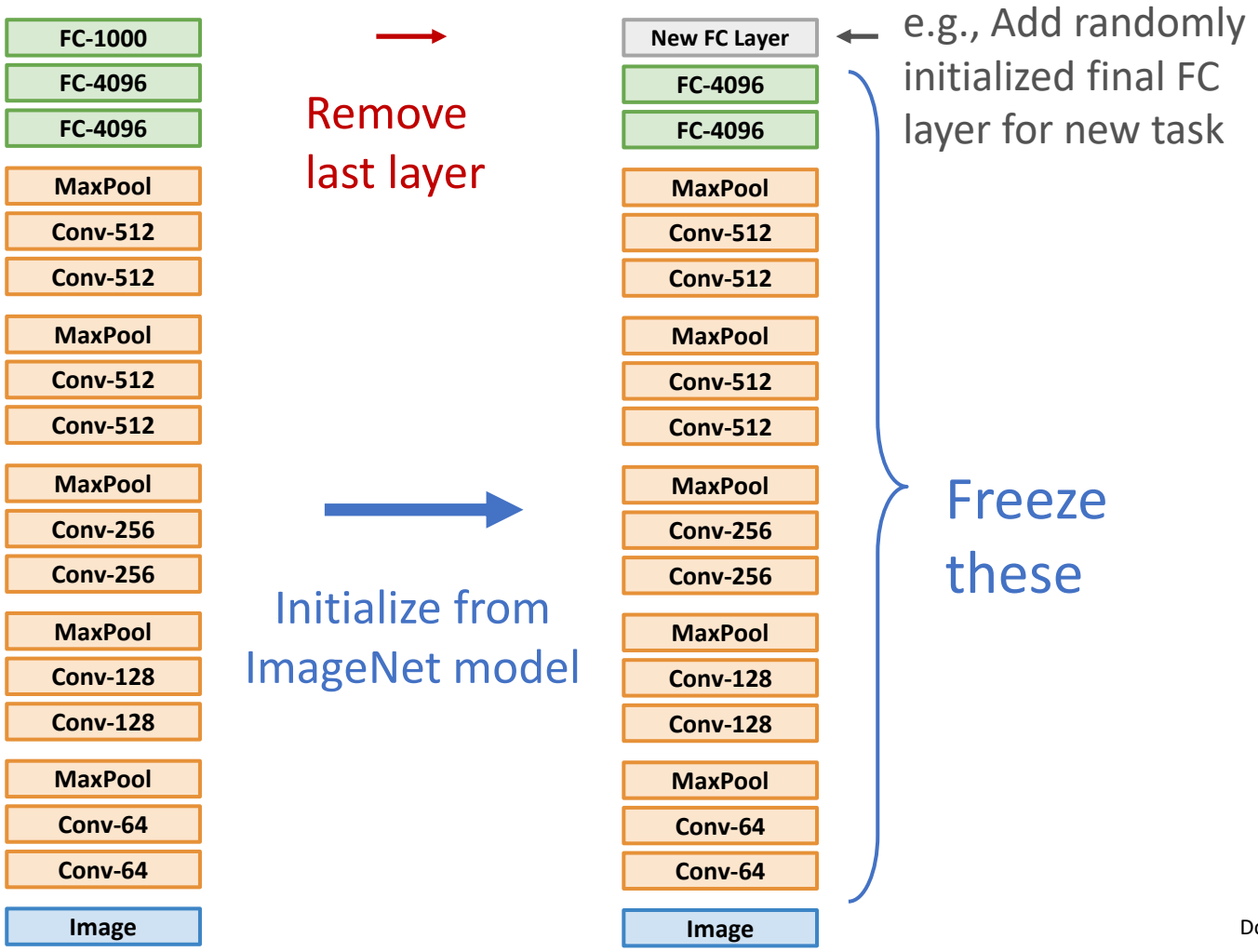
2. Remove the task-specific layer
   - Usually keep all conv layers and remove the linear layer/all MLPs

- This simple idea surprisingly works well!

Transfer Learning with CNNs: Feature Extraction

1. Train on ImageNet

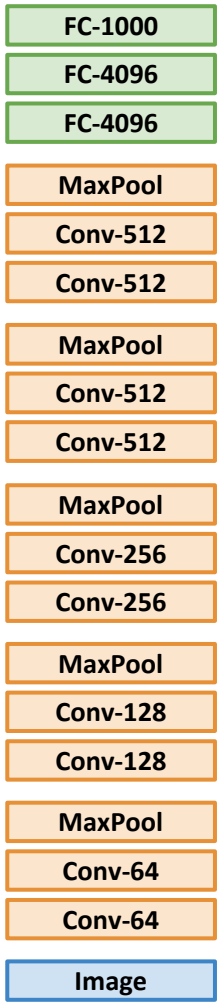
2. Use CNN as a feature extractor



Donahue et al, "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition", ICML 2014

Transfer Learning with CNNs: Fine-Tuning

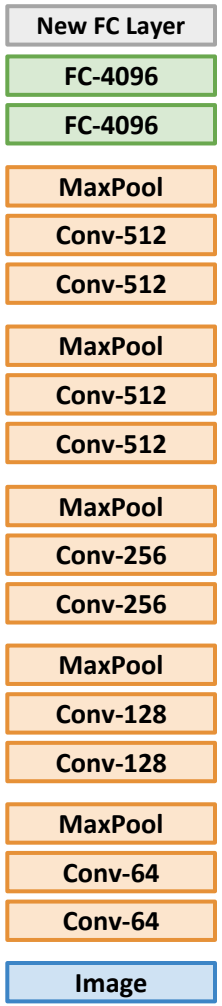
1. Train on ImageNet



Remove last layer

Initialize from ImageNet model

2. Fine-tune CNN



e.g., Add randomly initialized final FC layer for new task

Continue training entire model for new task

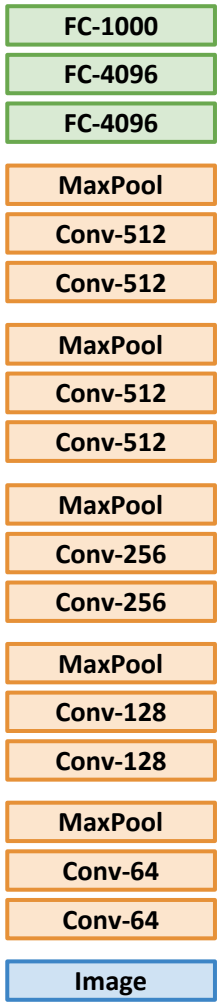
Some tricks:

- Train with frozen feature extraction first before fine-tuning
- Lower the learning rate: use ~1/10 of LR used in original training
- Sometimes freeze lower layers to save computation
- Train with BatchNorm in “test” mode

Donahue et al, “DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition”, ICML 2014

Transfer Learning with CNNs: Fine-Tuning

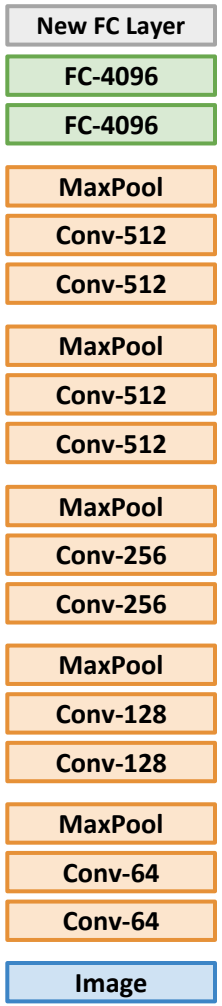
1. Train on ImageNet



Remove last layer

Initialize from ImageNet model

2. Fine-tune CNN



e.g., Add randomly initialized final FC layer for new task

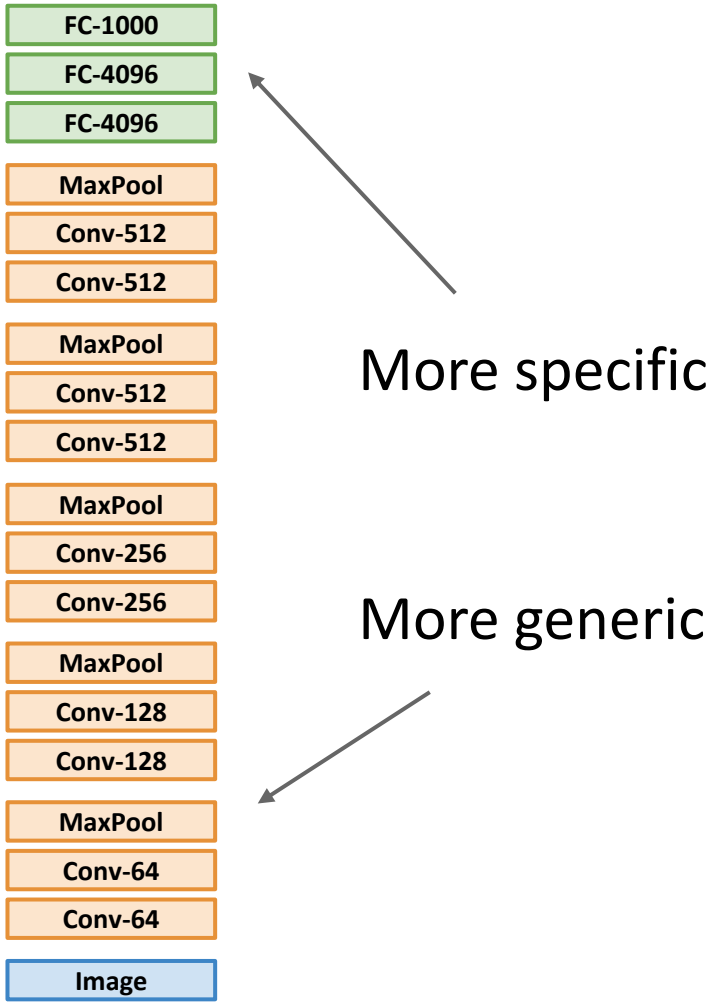
Continue training entire model for new task

Compared with Feature Extraction, Fine-Tuning:

- Requires more data
- Is more computationally expensive
- Can give higher accuracies

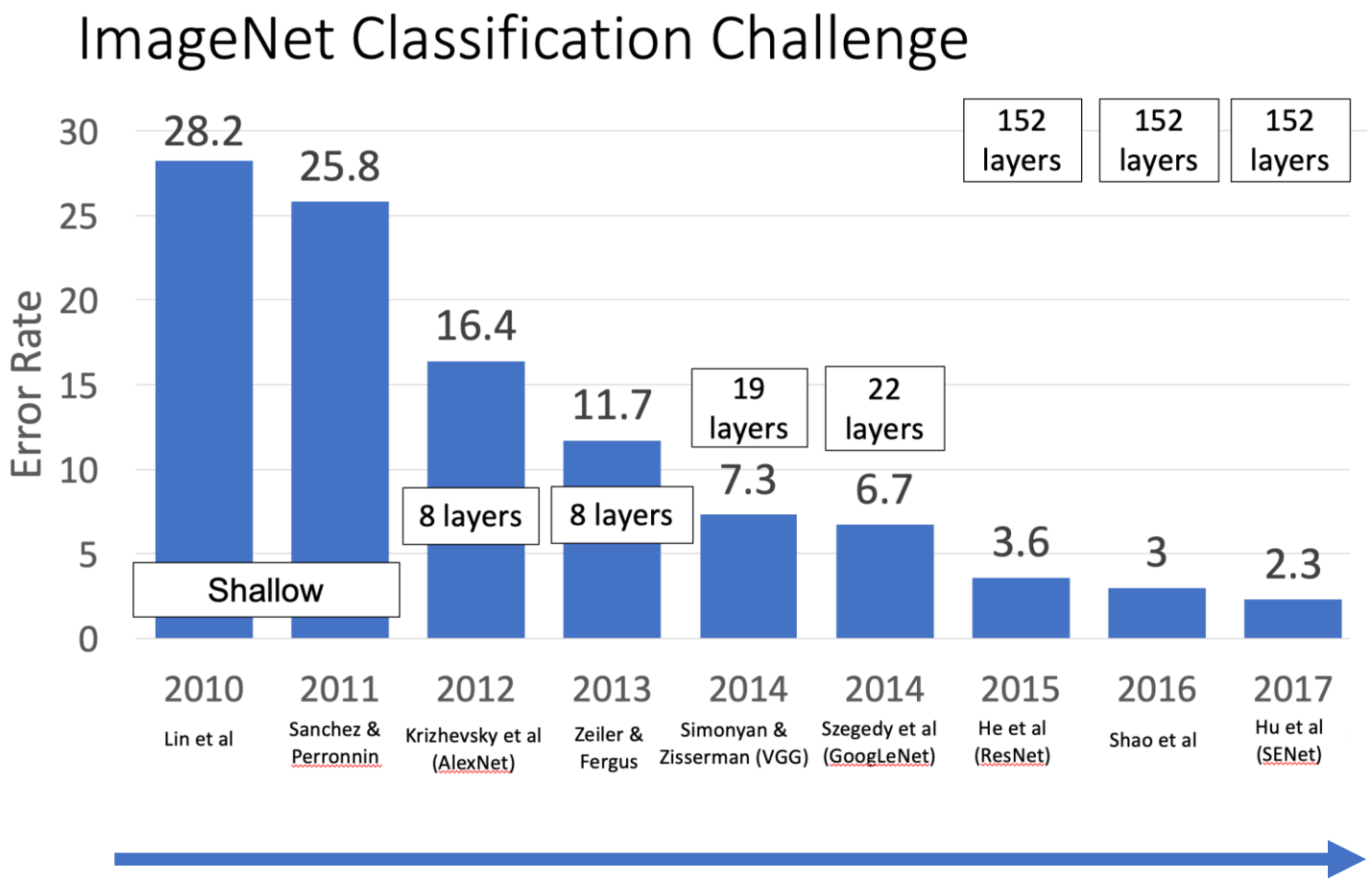
Donahue et al, "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition", ICML 2014

Transfer Learning with CNNs



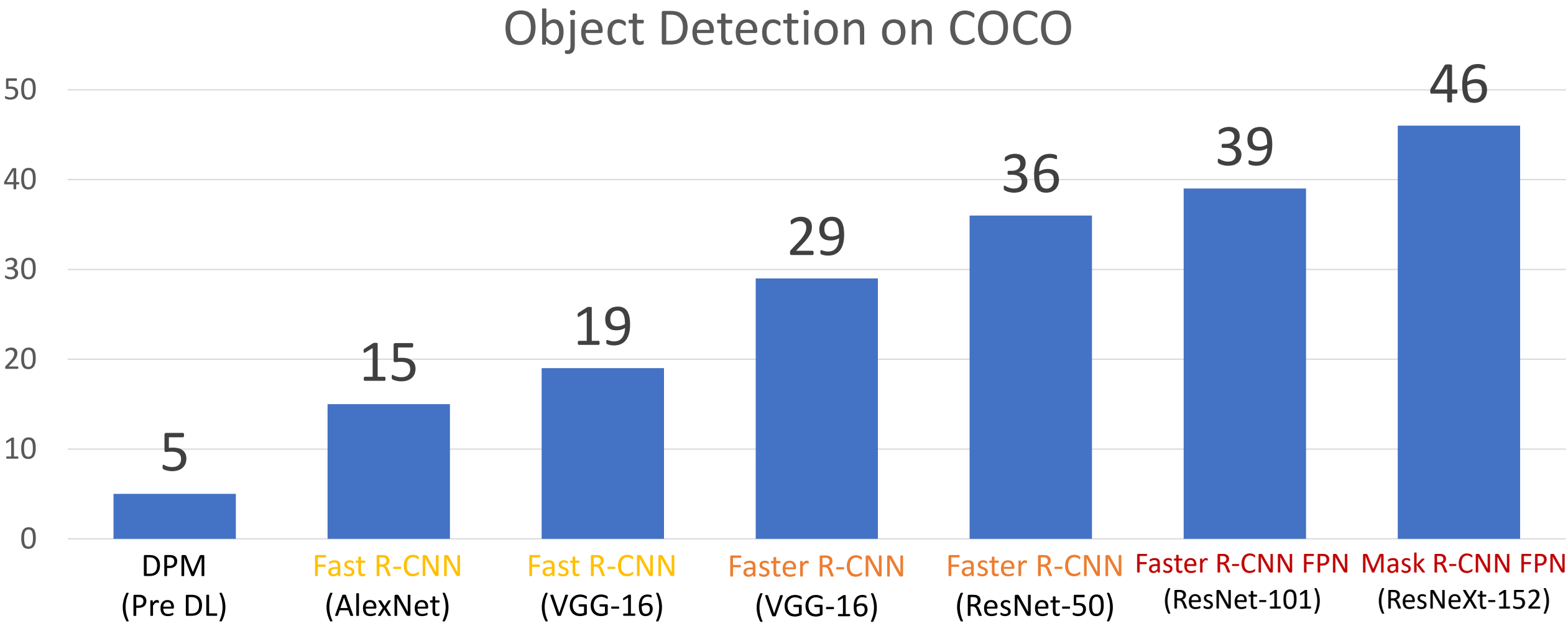
	Dataset similar to ImageNet	**Dataset very different from ImageNet**
very little data (10s to 100s)	Use Linear Classifier on top layer	You're in trouble… Try linear classifier from different stages
quite a lot of data (100s to 1000s)	Finetune a few layers	Finetune a larger number of layers

Transfer Learning with CNNs: Architecture Matters!



Improvements in CNN architectures lead to improvements in many downstream tasks thanks to transfer learning!

Transfer Learning with CNNs: Architecture Matters!



Ross Girshick, "The Generalized R-CNN Framework for Object Detection", ICCV 2017 Tutorial on Instance-Level Visual Recognition

Self-Supervised Learning (Roughly, ~ 2019)

Self-Supervised Representation Learning

- Building methods that learn representations from “raw” data – no annotations required

- **Unsupervised Learning**: Model isn’t told what to predict. This is somewhat old terminology, not used as much these days.

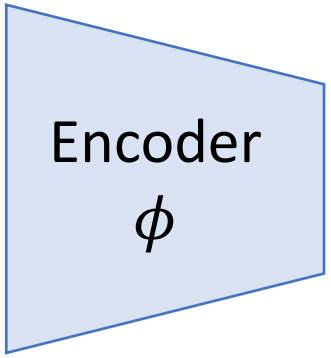
- **Self-Supervised Learning**: Model is trained to predict some naturally-occurring signal in the raw data rather than human annotations.

- **Semi-Supervised Learning**: Train jointly with some labeled data and (a lot of) unlabeled data.

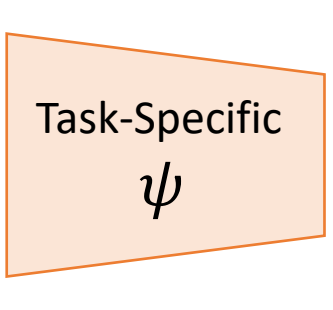
Self-Supervised Learning Pipeline

Stage 1: Pretraining (Representation learning)
Pretrain a network on a pretext task that doesn't require supervision;

Input Image: x



Features: $\phi(x)$

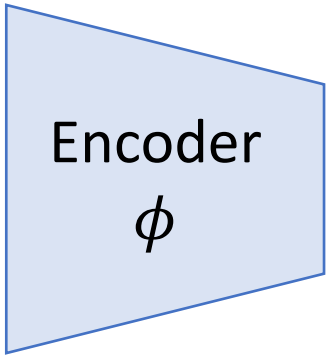


Prediction: $\hat{y}$

Loss: $L(\hat{y}, y)$

Stage 2: Transfer learning
Transfer encoder to downstream tasks via linear classifier, K-NN, fine-tuning

Input Image: x



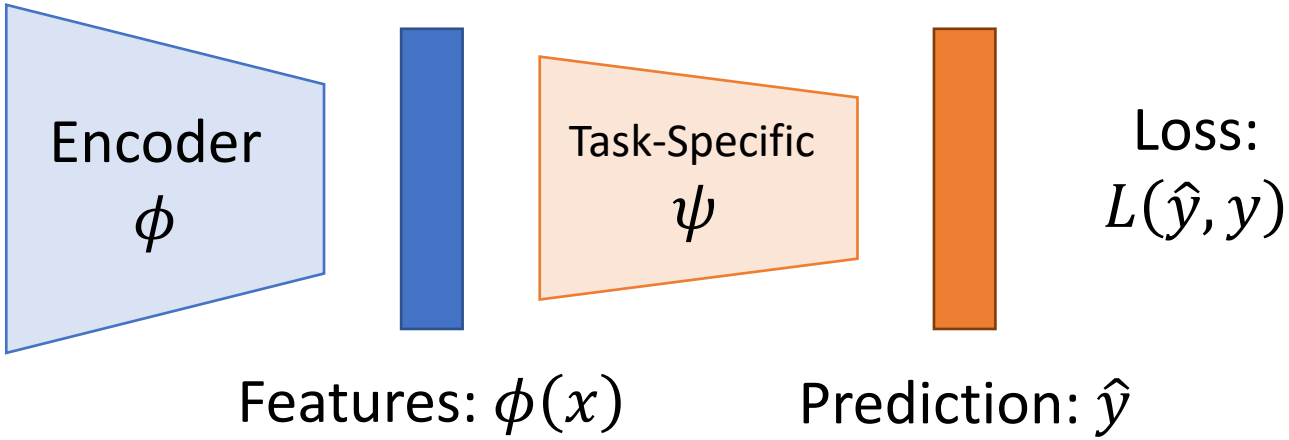
Features: $\phi(x)$

Downstream tasks:
Image classification,
object detection,
semantic segmentation

Self-Supervised Learning Pipeline

Stage 1: Pretraining (Representation learning)
Pretrain a network on a pretext task that doesn't require supervision;

Input Image: x

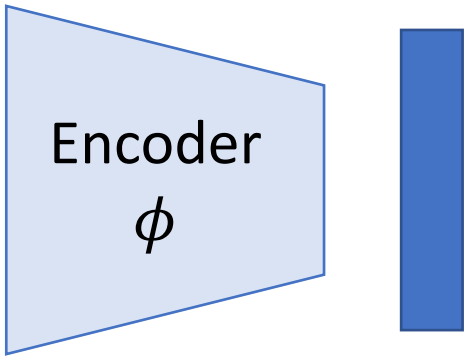


Features: $\phi(x)$

Prediction: $\hat{y}$

Stage 2: Transfer learning
Transfer encoder to downstream tasks via linear classifier, K-NN, fine-tuning

Input Image: x



Features: $\phi(x)$

Goal: Pretrain + Transfer does better than supervised pretraining, and better than directly training on downstream

Pretext Tasks

Generative: Predict a part of the input signal

- Autoencoding
- Autoregression
- GANs
- Inpainting
- Colorization

Discriminative: Predict something about the input signal

- Context prediction
- Clustering
- Rotation prediction
- Contrastive learning

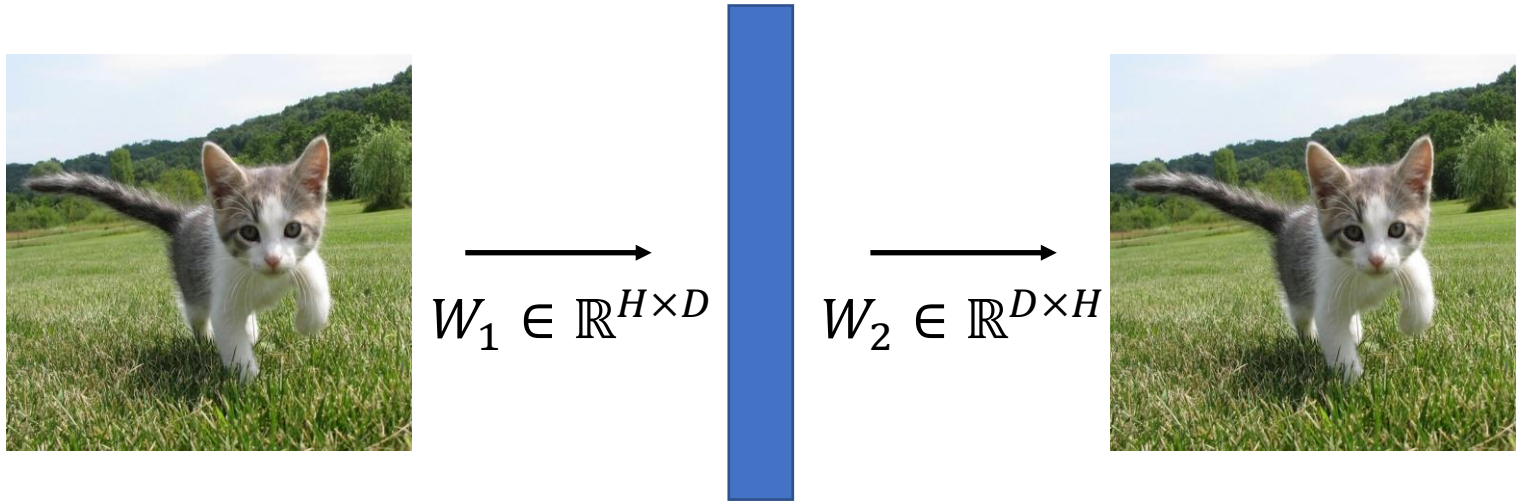
Multimodal: Use some additional signal in addition to RGB images

- Video
- 3D
- Sound
- Language

Pretext Task: Autoencoding

- Autoencoder tries to reconstruct inputs.
- Hidden layer (hopefully) learns good representations.
- H < D to get non-trivial hidden representations

$$L(x) = R(x, \hat{x}) = \|x - \hat{x}\|_2^2$$



Input Image: $x \in \mathbb{R}^D$

Hidden Layer: $\mathrm{h} \in \mathbb{R}^H$

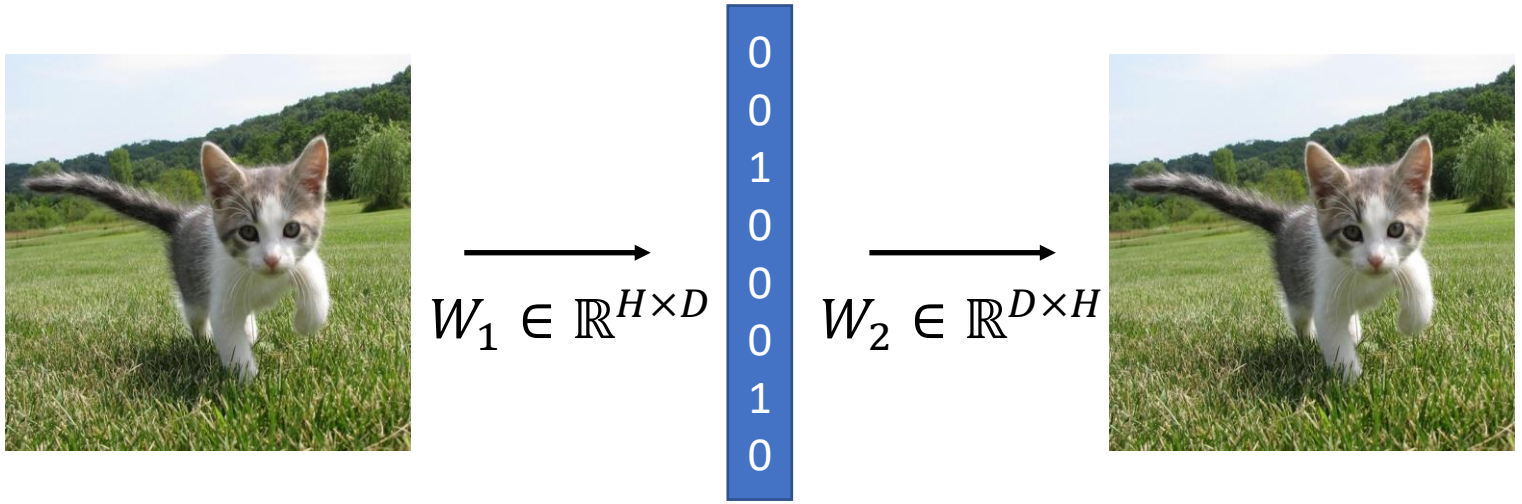
Reconstructed Image: $\hat{x} \in \mathbb{R}^D$

Lee et al, "Efficient Sparse Coding Algorithms", NeurIPS 2006; Ranzato et al, "Efficient Learning of Sparse Representations with an Energy-Based Model", NeurIPS 2006; Lee et al, "Sparse deep belief net models for visual area V2", NeurIPS 2007; Ng, "Sparse Autoencoder", CS294A Lecture Notes

Pretext Task: Sparse Autoencoder

- Reconstruct inputs with **sparse activations** (mostly 0)
- Many ways to implement sparsity penalties $L(x) = R(x, \hat{x}) + \lambda S(h)$

$$= \|x - \hat{x}\|_2^2 + \lambda \|h\|_1$$



Input Image: $x \in \mathbb{R}^D$

Hidden Layer: $\mathrm{h} \in \mathbb{R}^H$

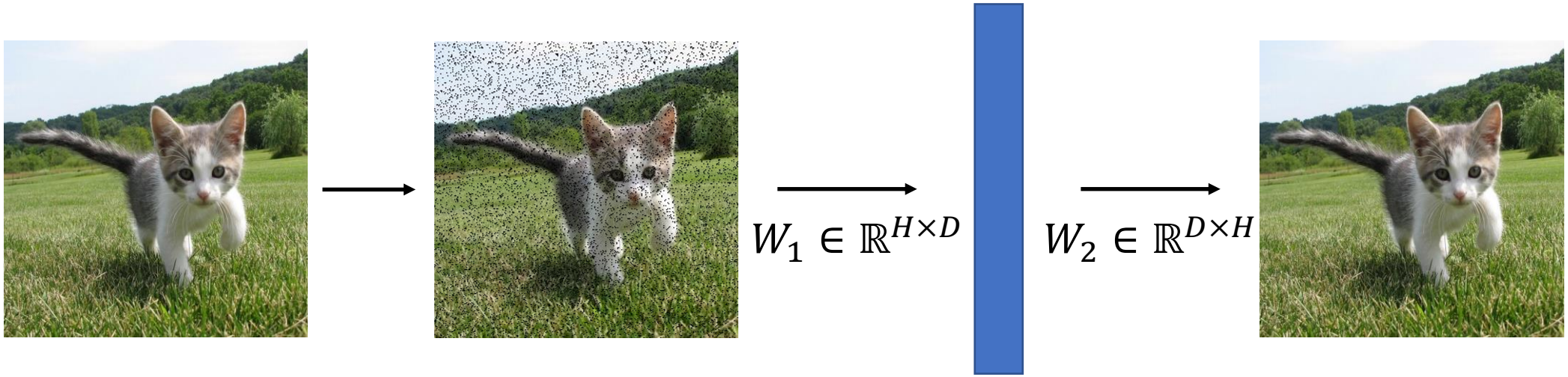
Reconstructed Image: $\hat{x} \in \mathbb{R}^D$

Lee et al, "Efficient Sparse Coding Algorithms", NeurIPS 2006; Ranzato et al, "Efficient Learning of Sparse Representations with an Energy-Based Model", NeurIPS 2006; Lee et al, "Sparse deep belief net models for visual area V2", NeurIPS 2007; Ng, "Sparse Autoencoder", CS294A Lecture Notes

Pretext Task: Denoising Autoencoder

- Reconstruct noisy inputs (pixels randomly set to zero)

$$L(x) = R(x, \hat{x}) = \|x - \hat{x}\|_2^2$$



Input Image: $x \in \mathbb{R}^D$

Corrupted Image: $x \in \mathbb{R}^D$

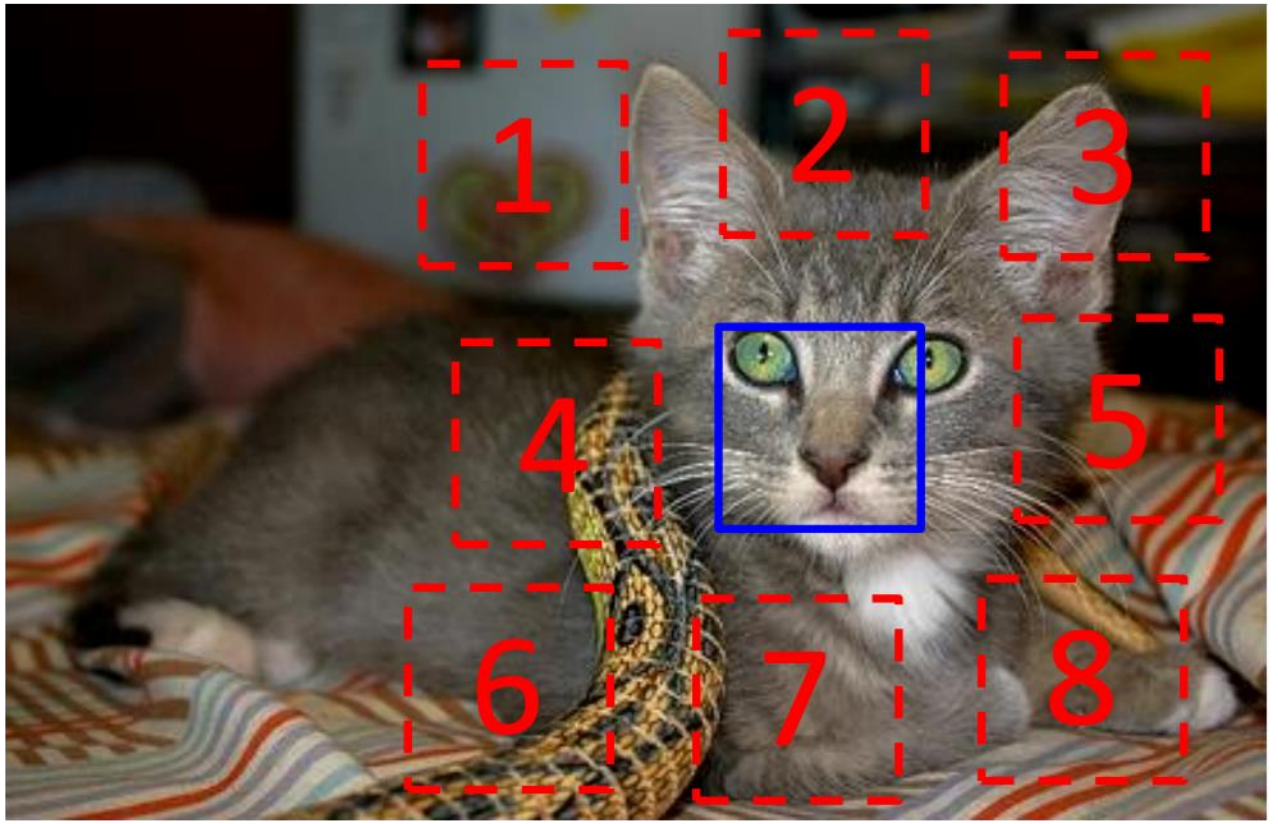
Hidden Layer: $\mathrm{h} \in \mathbb{R}^H$

Reconstructed Image: $\hat{x} \in \mathbb{R}^D$

Vincent et al, "Extracting and Composing Robust Features with Denoising Autoencoders", ICML 2008

Pretext Task: Context Prediction

- Model predicts relative location of two patches from the same image.
- Discriminative pretraining task
- Intuition: Requires understanding objects and their parts



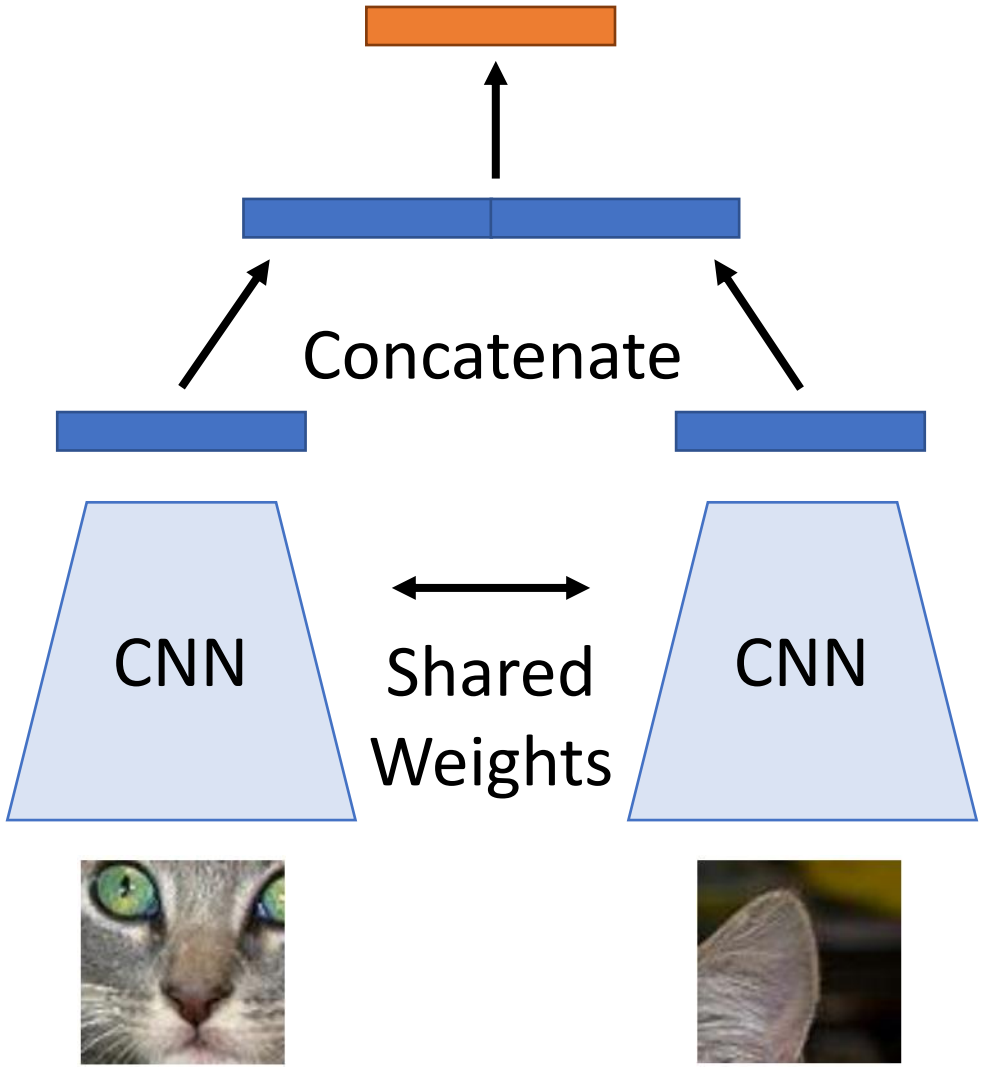
Doersch et al, “Unsupervised Visual Representation Learning by Context Prediction”, ICCV 2015

Pretext Task: Context Prediction

Model predicts relative location of two patches from the same image.
Discriminative pretraining task

Intuition: Requires understanding objects and their parts

Two networks with shared weights sometimes called a ”Siamese network”



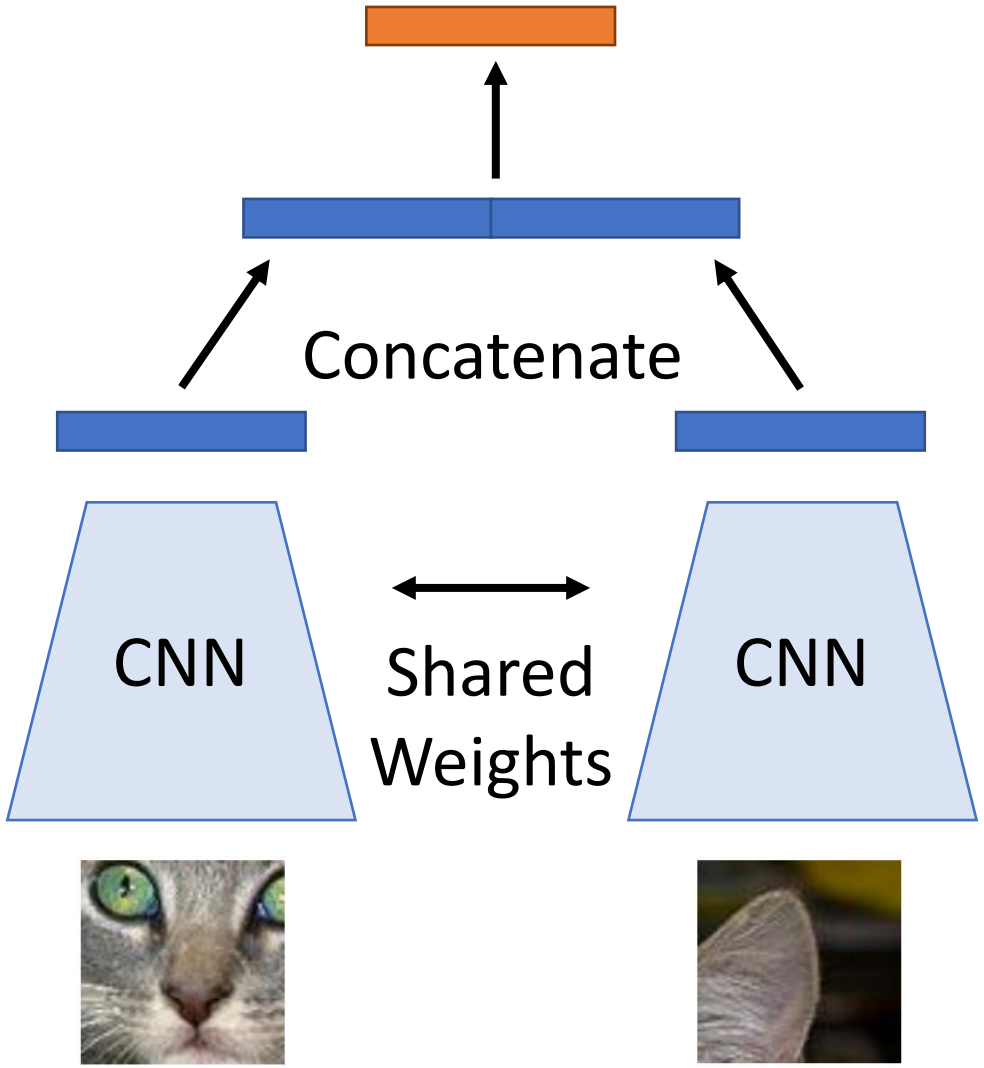
Doersch et al, “Unsupervised Visual Representation Learning by Context Prediction”, ICCV 2015

Pretext Task: Context Prediction

Model predicts relative location of two patches from the same image.
Discriminative pretraining task

Intuition: Requires understanding objects and their parts

"For experiments, we use a ConvNet trained on a K40 GPU for approximately four weeks."

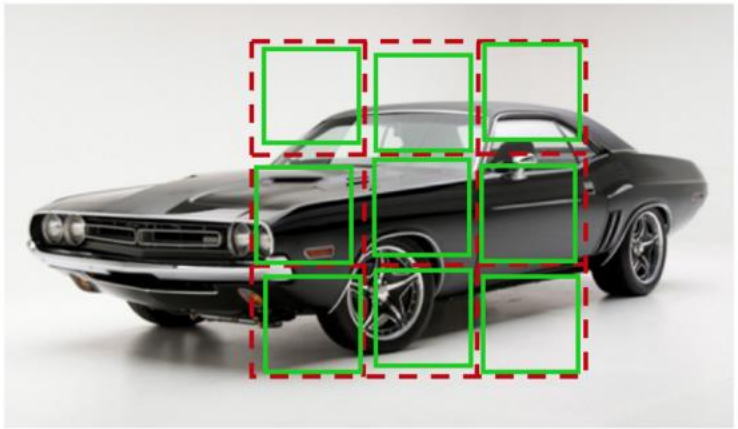


Doersch et al, "Unsupervised Visual Representation Learning by Context Prediction", ICCV 2015

Pretext Task: Solving Jigsaw Puzzles

Problem: These methods only work on patches, not whole images!

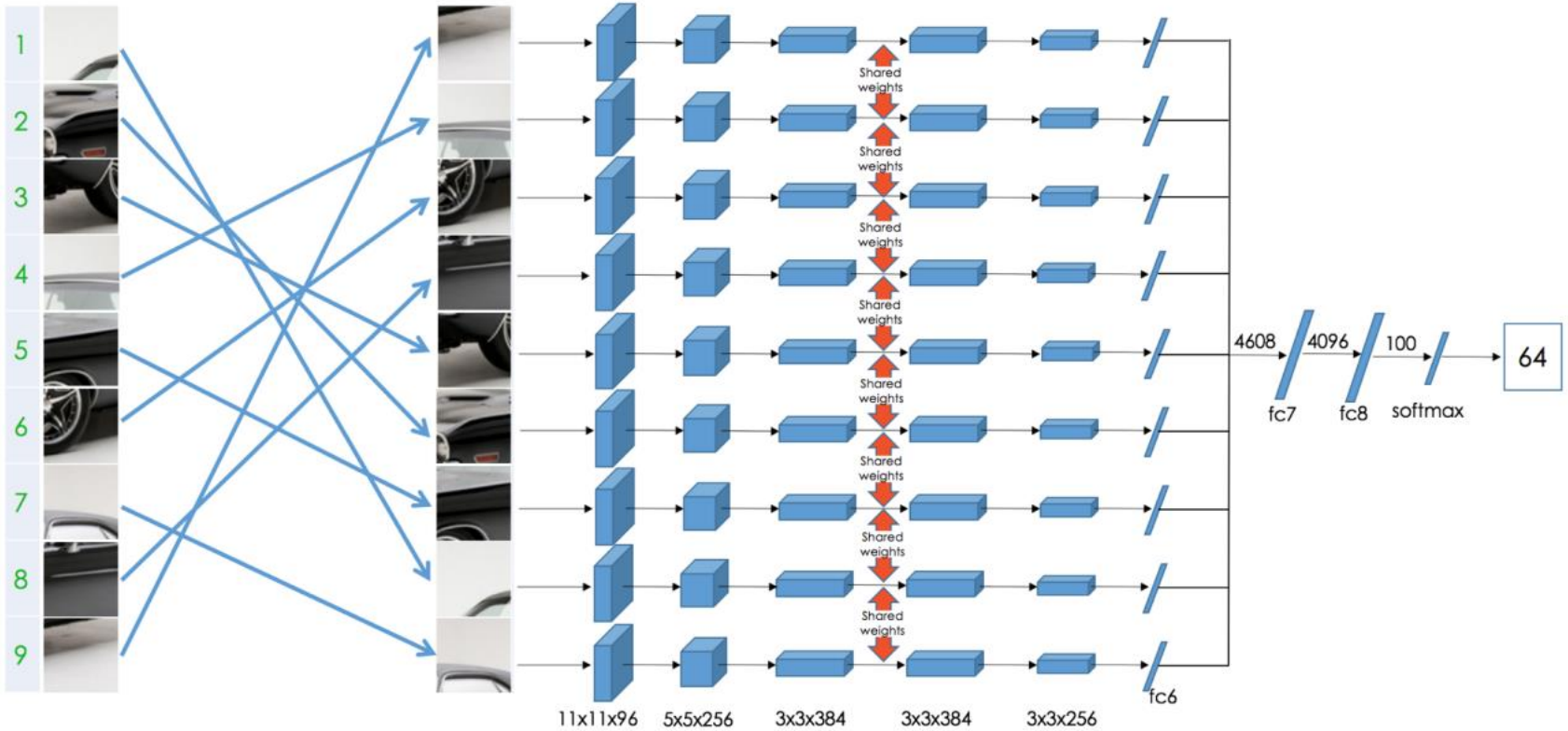
- Rather than predict relative position of two patches, instead predict permutation to “unscramble” 9 shuffled patches



Permutation Set

index	permutation
64	9,4,6,8,3,2,5,1,7

Reorder patches according to the selected permutation



Noroozi and Favoro, “Unsupervised learning of visual representations by solving jigsaw puzzles”, ECCV 2016

Pretext Task: Inpainting

Input Image

Encoder: ϕ

Decoder: ψ

Predict Missing Pixels

Human Artist

Pathak et al, “Context Encoders: Feature Learning by Inpainting”, CVPR 2016

Pretext Task: Inpainting

Input Image

Encoder: ϕ

Decoder: ψ

Predict Missing Pixels

L2 Loss
(Best for feature learning)

Pathak et al, “Context Encoders: Feature Learning by Inpainting”, CVPR 2016

Pretext Task: Inpainting

Input Image

Encoder: ϕ

Decoder: ψ

Predict Missing Pixels

L2 + Adversarial Loss
(Best for nice images)

Pathak et al, “Context Encoders: Feature Learning by Inpainting”, CVPR 2016

Pretext Task: Colorization

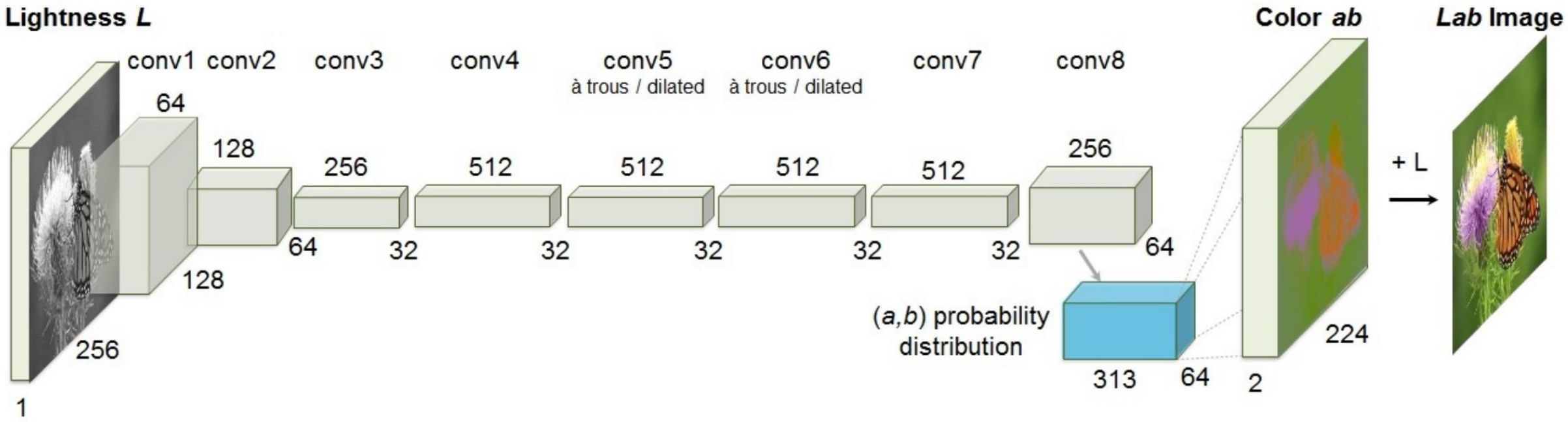
- Colorize grayscale images

Input: Grayscale Image

Output: Color Image

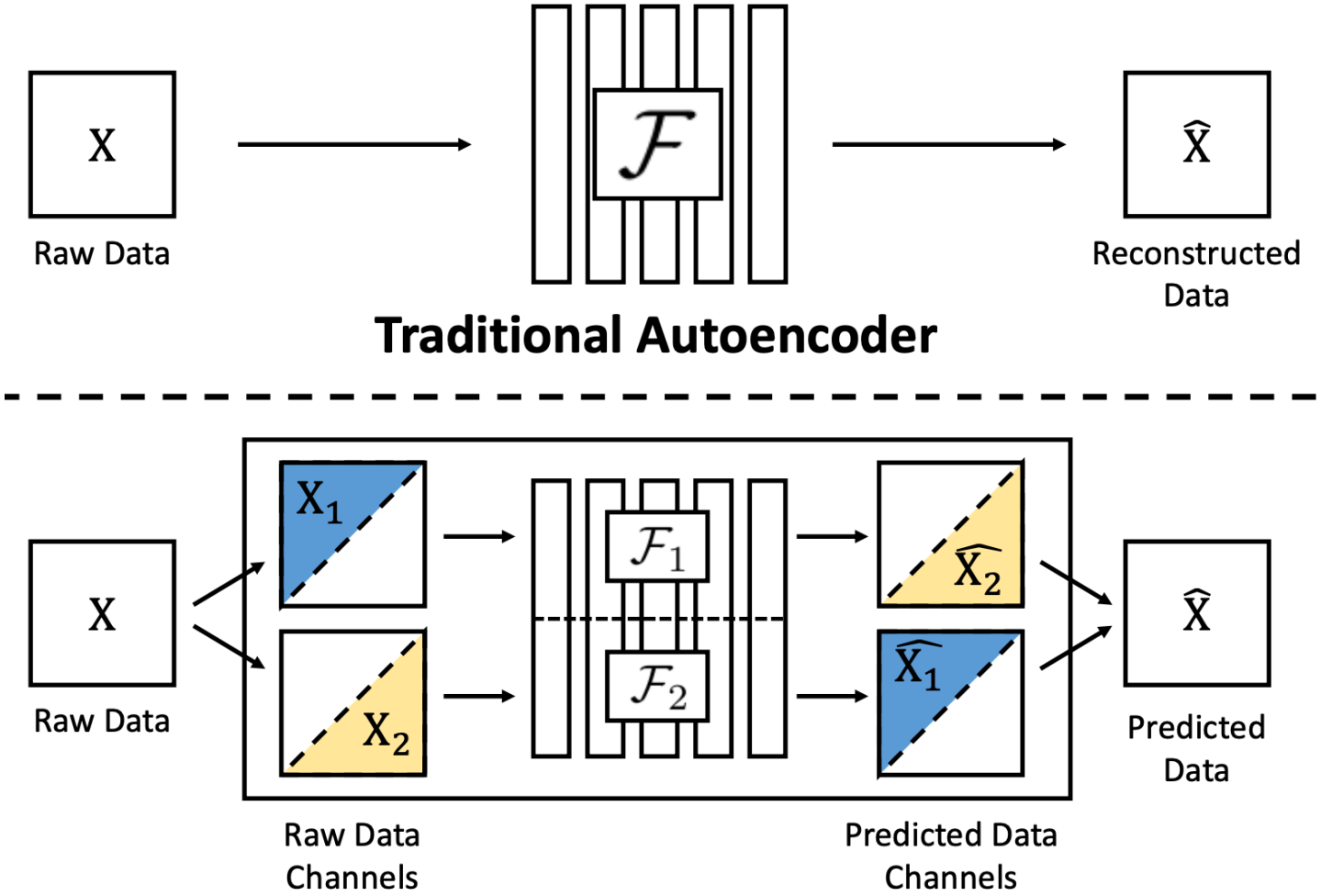
Zhang et al, “Colorful Image Colorization”, ECCV 2016

Pretext Task: Colorization



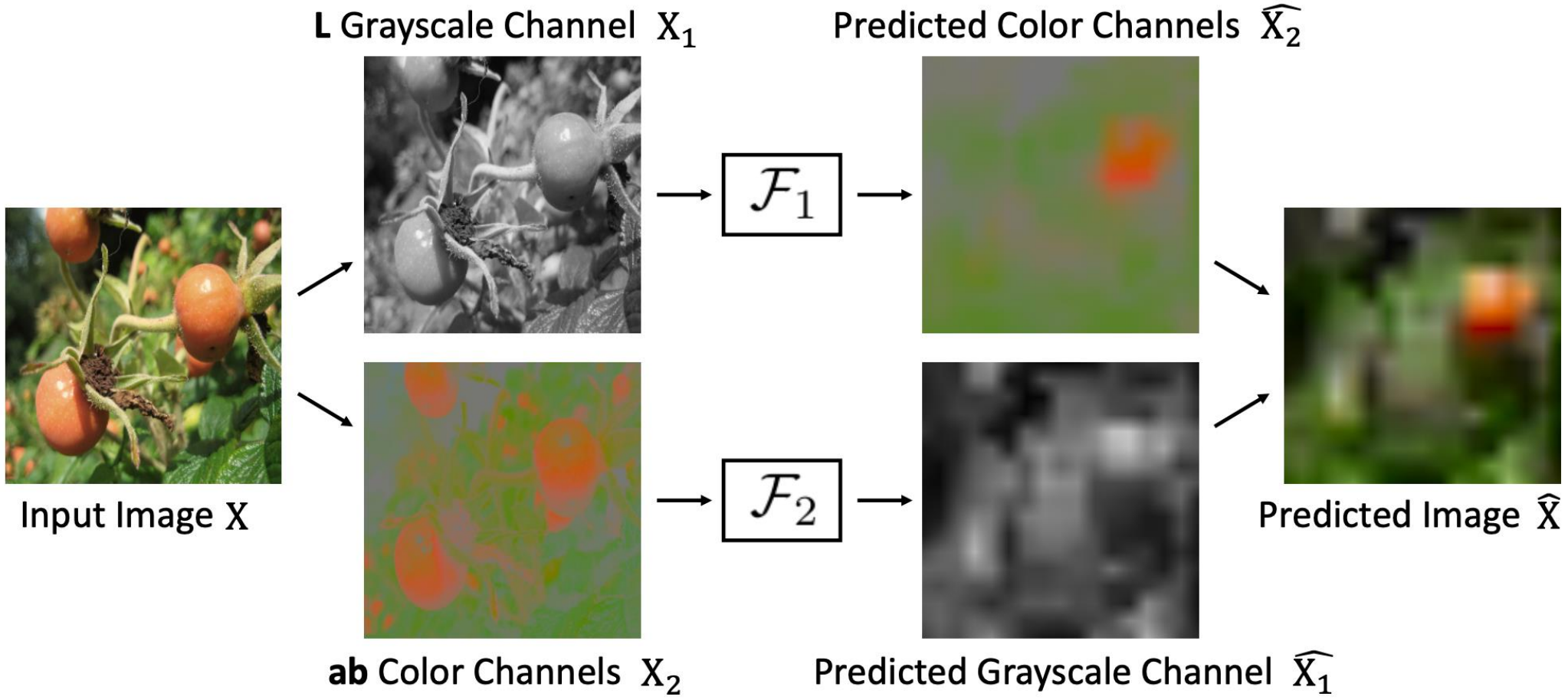
Zhang et al, “Colorful Image Colorization”, ECCV 2016

Colorization Extension: Split-Brain Autoencoder



Zhang et al, "Split-Brain Autoencoders: Unsupervised Learning by Cross-Channel Prediction", CVPR 2017

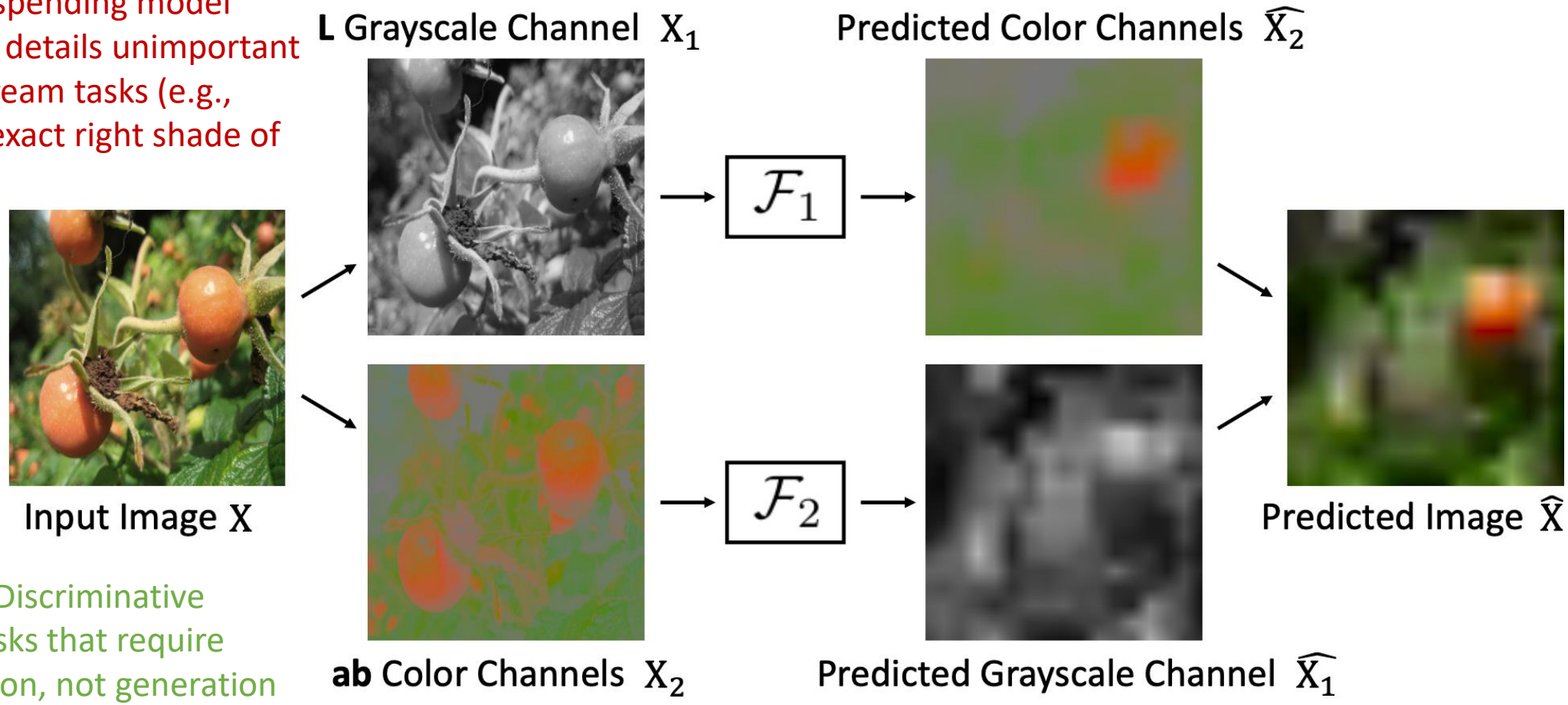
Colorization Extension: Split-Brain Autoencoder



Zhang et al, "Split-Brain Autoencoders: Unsupervised Learning by Cross-Channel Prediction", CVPR 2017

Colorization Extension: Split-Brain Autoencoder

Concern: Generative pretexts encourage spending model capacity on details unimportant for downstream tasks (e.g., regressing exact right shade of orange)



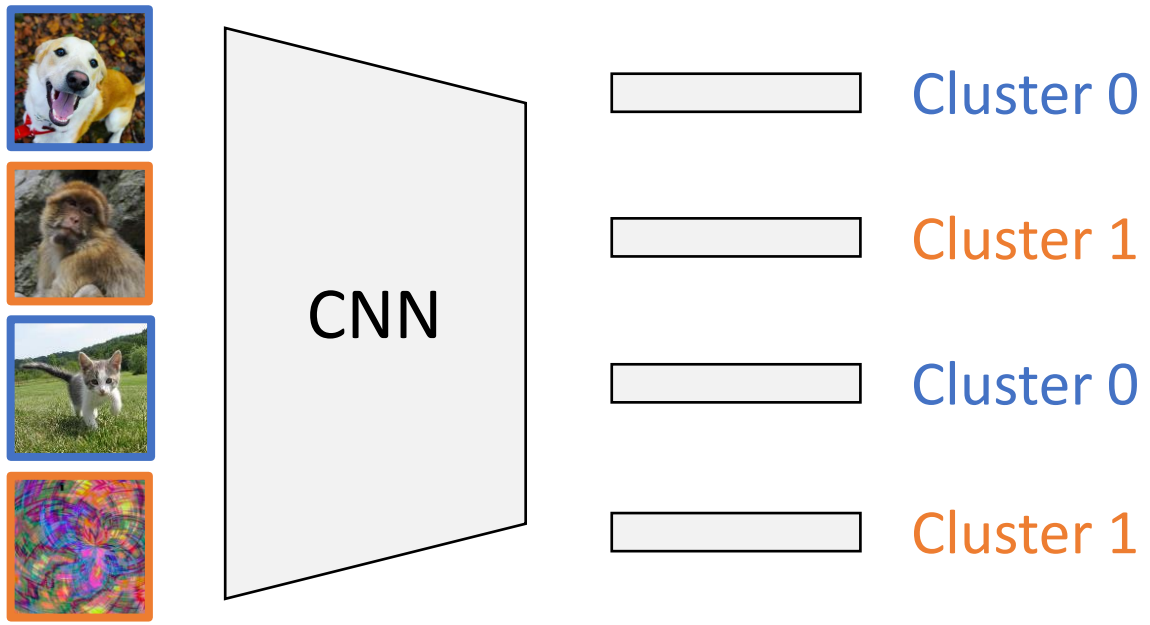
Solution: Discriminative pretext tasks that require classification, not generation

Zhang et al, "Split-Brain Autoencoders: Unsupervised Learning by Cross-Channel Prediction", CVPR 2017

Pretext Task: Deep Clustering

- A naïve example of deep clustering:

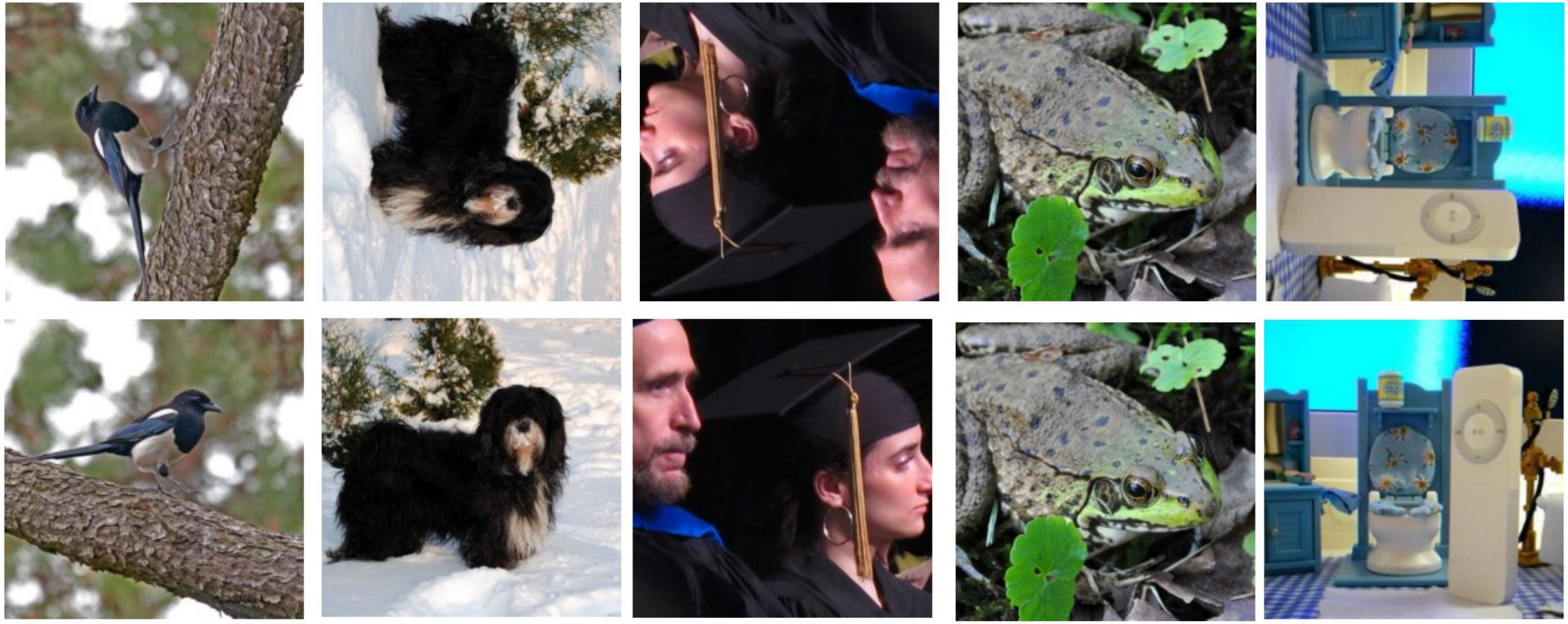
1. Randomly initialize a CNN
2. Run many images through CNN, get their final-layer features
3. Cluster the features with K-Means; record cluster assignments
4. Use cluster assignments as pseudo-labels for each image; train the CNN to predict cluster assignments
5. GOTO 2.



Caron et al, "Deep Clustering for Unsupervised Learning of Visual Features", ECCV 2018
Caron et al, "Unsupervised Pre-Training of Image Features on Non-Curated Data", ICCV 2019
Yan et al, "ClusterFit: Improving Generalization of Visual Representations", CVPR 2020
Caron et al, "Unsupervised Learning of Visual Features by Contrasting Cluster Assignments", NeurIPS 2020

Pretext Task: Rotation Prediction

- 4-way classification task: How much was each image rotated? (0, 90, 180, or 270 degrees)



90 270 180 0 270

Gidaris et al, “Unsupervised representation learning by predicting image rotations”, ICLR 2018

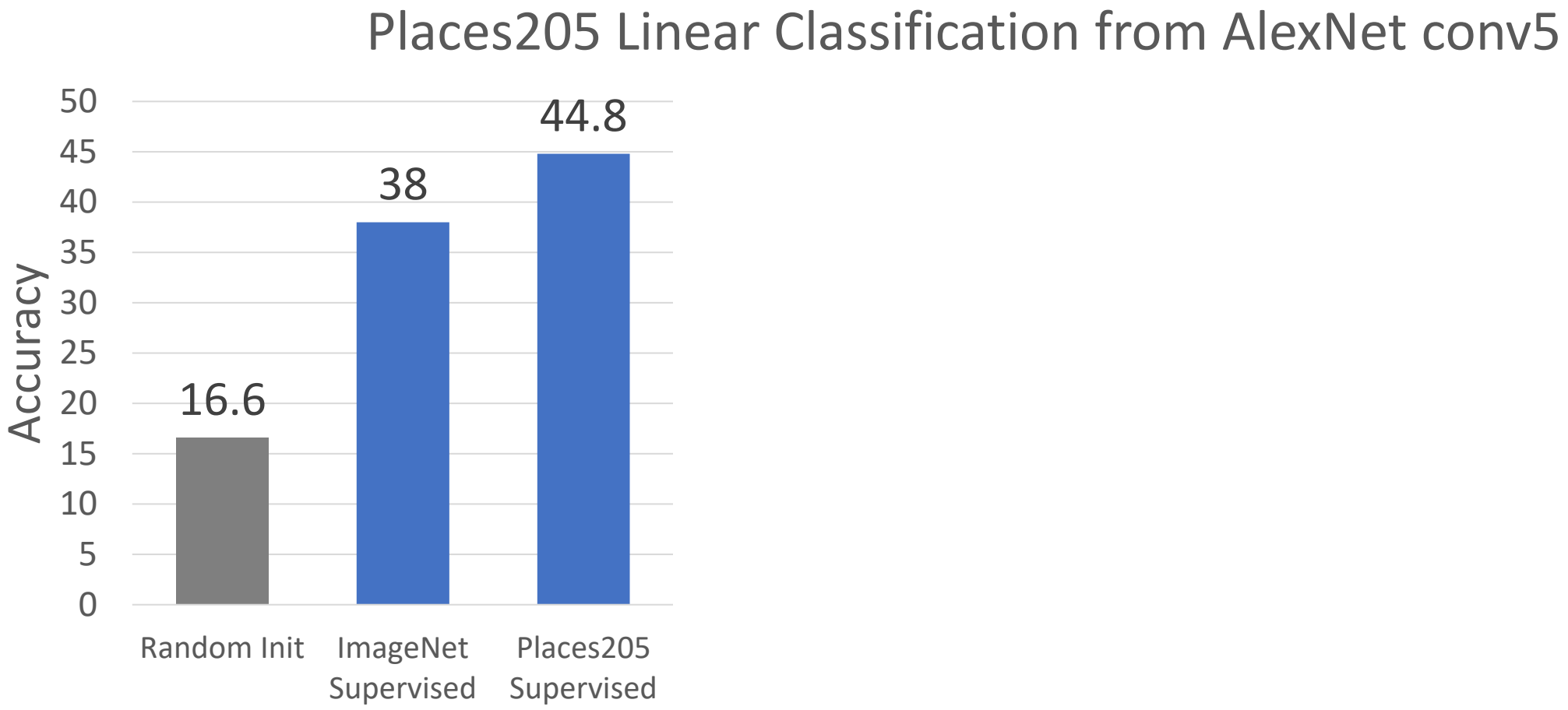
Which SSL Method is best?

Fair evaluation of SSL methods is very hard! No theory, so we need to rely on experiment

Many choices in experimental setup, huge variations from paper to paper:

- CNN architecture? AlexNet, ResNet50, something else?
- Pretraining dataset? ImageNet, or something else?
- Downstream task? ImageNet classification, detection, something else?
- Pretraining hyperparameters? Learning rates, training iterations, data augmentation?
- Transfer learning protocol?
  - Linear probe? From which layer? How to train linear models? SGD, something else? Transfer learning hyperparameters? Data augmentation or BatchNorm during transfer learning?
  - Fine-tune? From which layer? Architecture of “head” you attach? Linear or nonlinear? Fine-tuning hyperparameters?
  - K-NN? What value of K? Normalization on features?

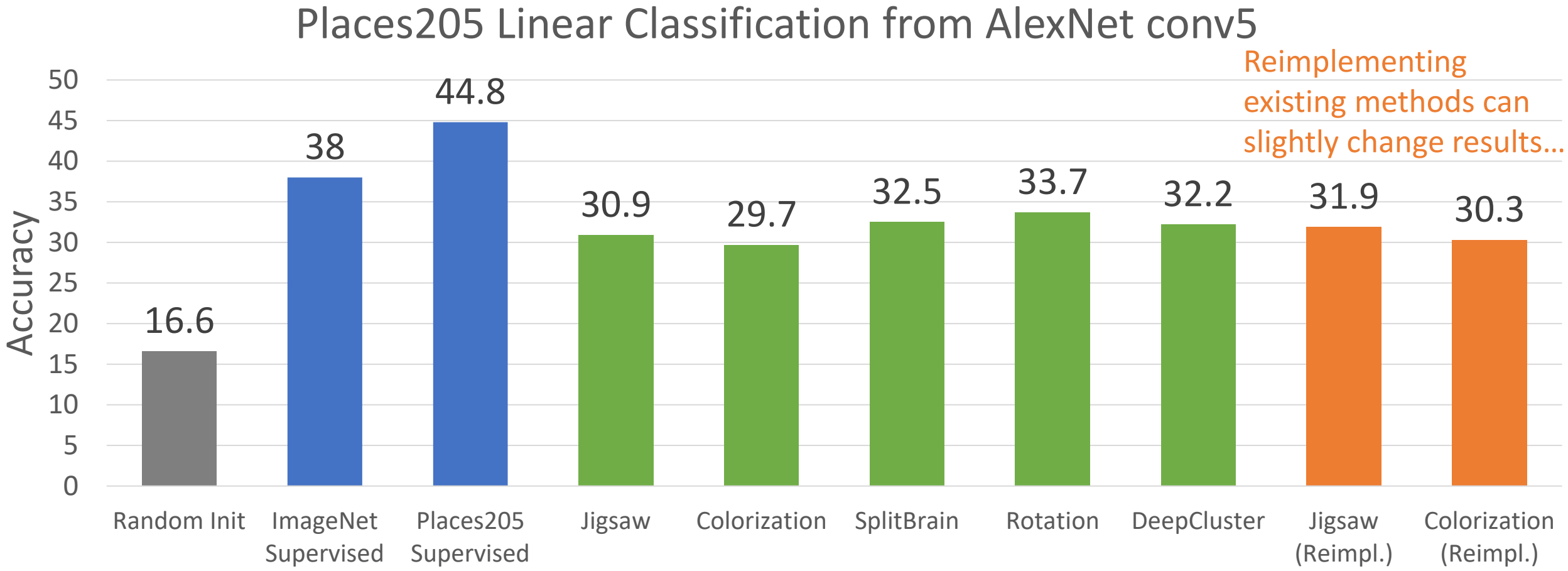
Supervised vs. Self-Supervised Pretraining



Goyal et al, "Scaling and Benchmarking Self-Supervised Visual Representation Learning", ICCV 2019

Supervised vs. Self-Supervised Pretraining

- As of 2019, SSL gave worse features than supervised pretraining

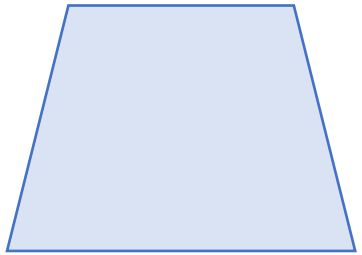


Goyal et al, "Scaling and Benchmarking Self-Supervised Visual Representation Learning", ICCV 2019

Self-Supervised Learning for Natural Language

Computer Vision

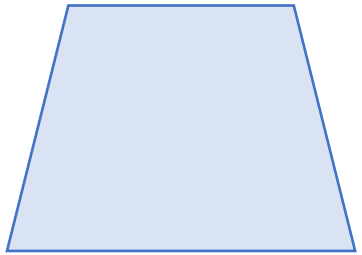
Image Features:
H x W x C



Input Image

Natural Language Processing

Word Features
L x C

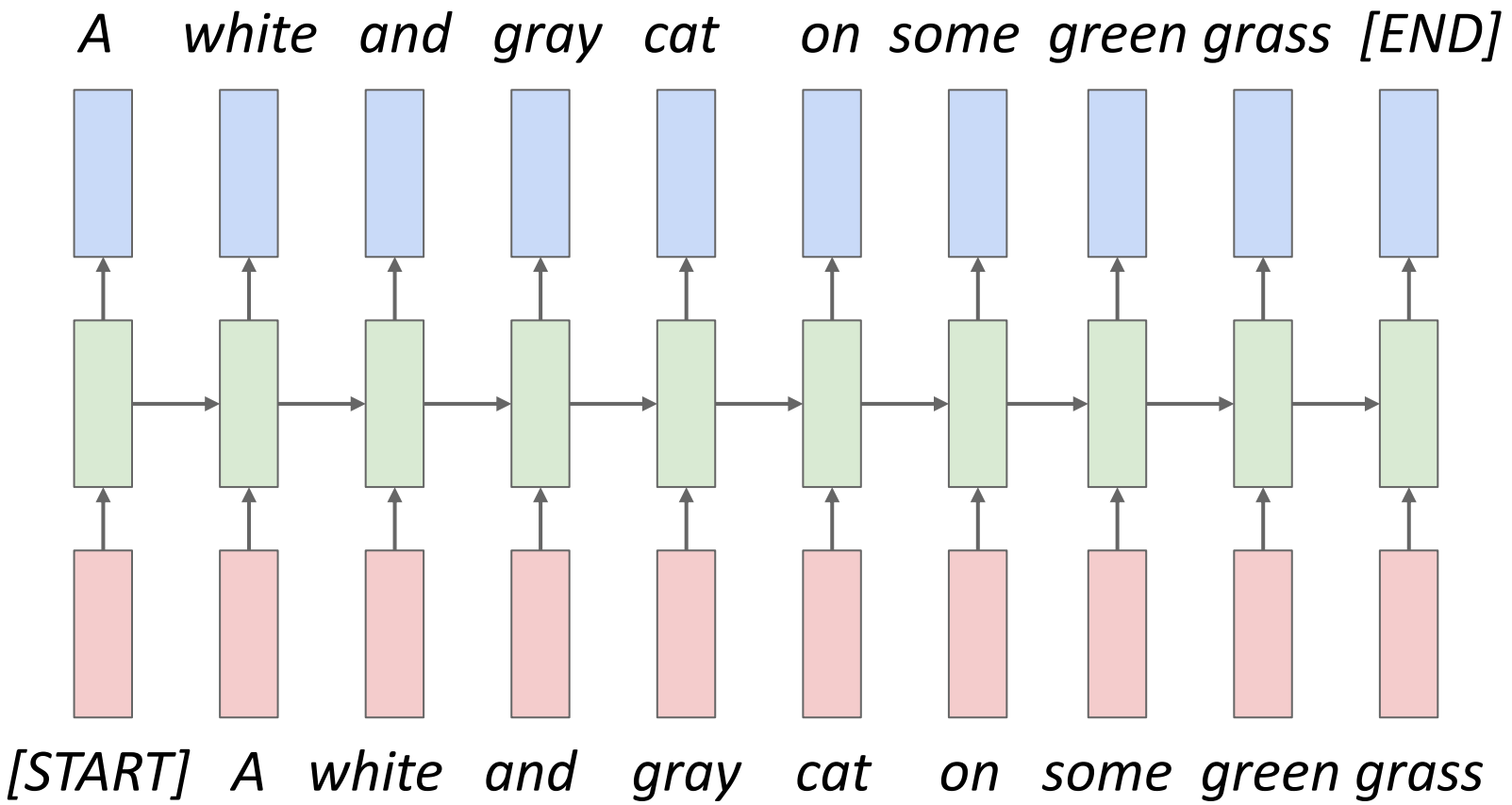


A white and gray cat standing outside on the grass

Input Sentence (L words)

Self-Supervised Learning for Natural Language

RNN language models train on raw text – no human labels required!
Their hidden states give features that transfer to many downstream tasks!



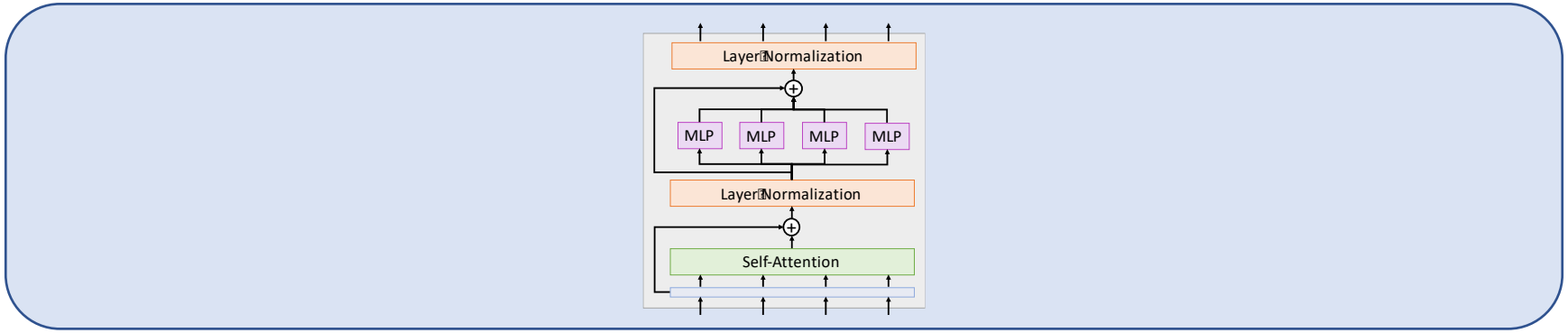
Peters et al, “Deep contextualized word representations”, NAACL 2018

Self-Supervised Learning for Natural Language

Transformer-based language models work even better! Can scale up to very large datasets, and give extremely powerful features that transfer to downstream tasks

Wildly successful: larger models, larger datasets give better features that improve performance on many downstream NLP tasks. The dream of SSL made real!

A white and gray cat on some green grass [END]



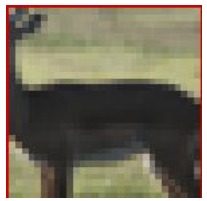
[START] A white and gray cat on some green grass

Radford et al, "Language models are unsupervised multitask learners", 2019
Brown et al, "Language Models are Few-Shot Learners", arXiv 2020
Rae et al, “Scaling Language Models: Methods, Analysis, & Insights from Training Gopher”, arXiv 2021

Self-Supervised Learning (Roughly, 2020 ~)

Pretext Task: Exemplar CNN

Quiz: What is this?



Different data augmentations (scale, shift, color jitter) of the same initial image patch

Answer: Deer!

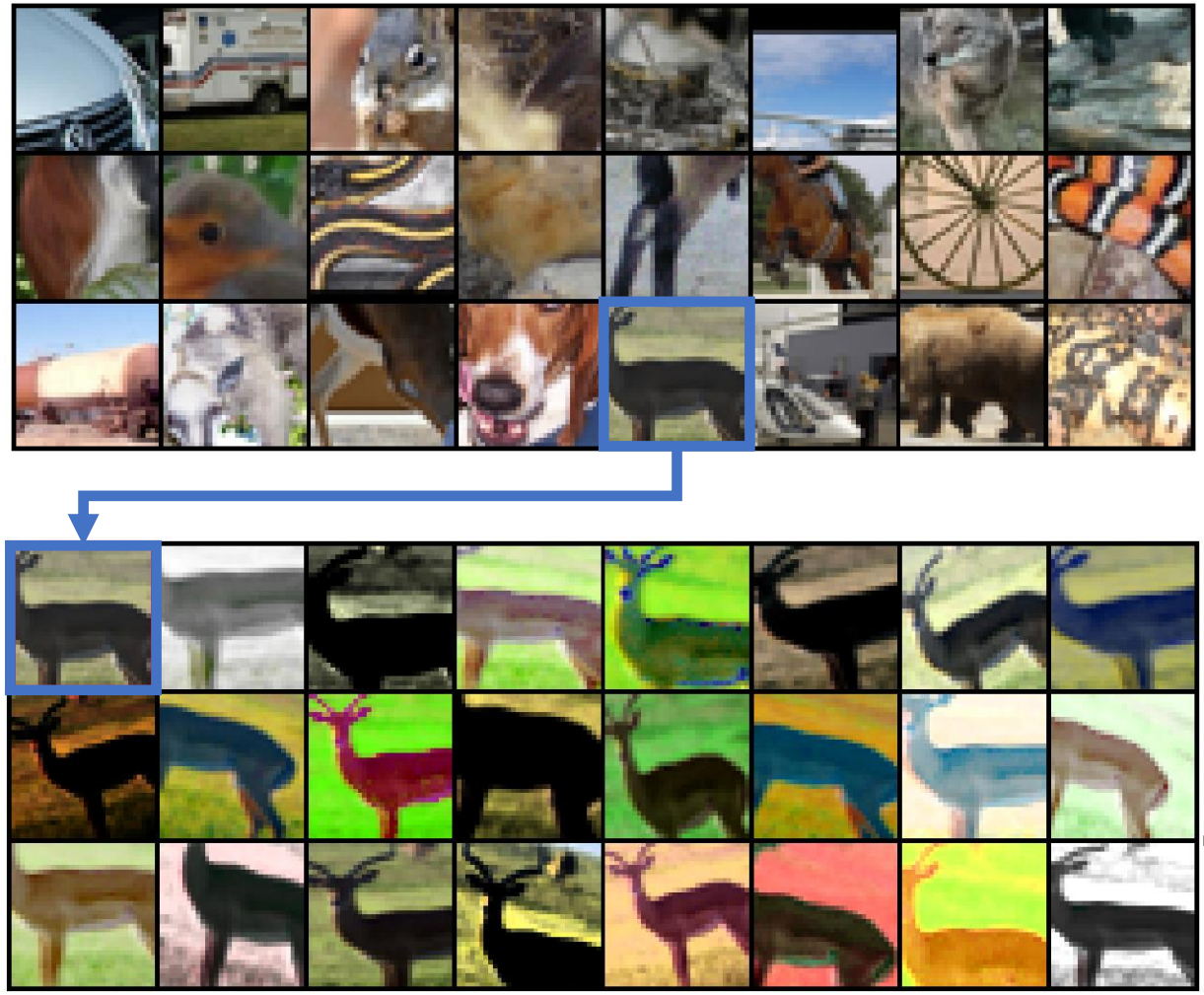
Dosovitskiy et al, “Discriminative Unsupervised Feature Learning with Exemplar Convolutional Neural Networks”, NeurIPS 2014; TPAMI 2016

Pretext Task: Exemplar CNN

Given an initial dataset of N images

Problem: number of parameters in the final layer depends on N; hard to scale

Sample K different augmentations for each; now have K*N total patches



Predicts which of the N original images it came from (N-way classification)

CNN

CNN inputs an augmented patch

Dosovitskiy et al, "Discriminative Unsupervised Feature Learning with Exemplar Convolutional Neural Networks", NeurIPS 2014; TPAMI 2016

Pretext Task: Contrastive Learning

- Assume we don't have labels for images, but we know whether some pairs of images are similar or dissimilar
- Let $d = \|\phi(x_1) - \phi(x_2)\|_2$ be the Euclidean distance btw features
- Problem: where to get positive and negative pairs?

Similar images should have similar features

CNN

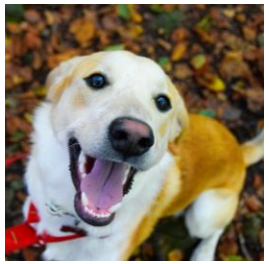
CNN

$L_S(x_1, x_2) = d^2$

Pull features together

Dissimilar images should have dissimilar features

CNN



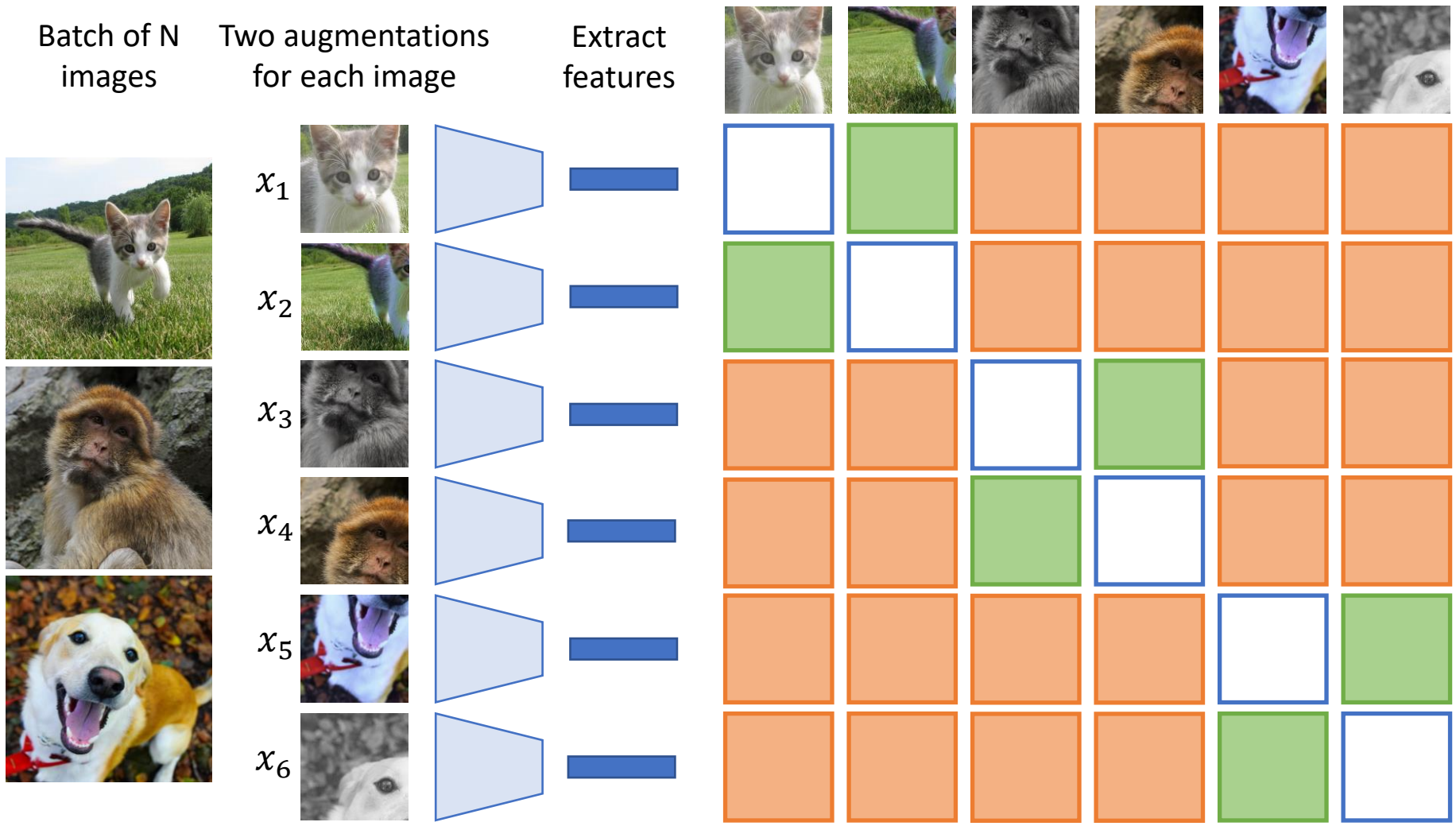
CNN

$L_D(x_1, x_2)$
$= \max(0, m - d)^2$

Push features apart
(up to margin m)

Hadsell et al, "Dimensionality Reduction by Learning and Invariant Mapping", CVPR 2006

Pretext Task: Contrastive Learning



- Each image tries to predict which of the other 2N-1 images came from the same original image.
- Similarity between x_i and x_j:

$$s_{i,j} = \frac{\phi(x_i)^T \phi(x_j)}{\|\phi(x_i)\| \cdot \|\phi(x_j)\|}$$

- If (x_i, x_j) is a positive pair, then loss for x_i is:

$$L_i = -\log \frac{\exp(s_{i,j}/\tau)}{\sum_{\substack{k=1 \\ k \neq i}}^{2N} \exp(s_{i,k}/\tau)}$$

(τ is a temperature)

- Interpretation: Cross-entropy loss over the other 2N-1 elements in the batch.

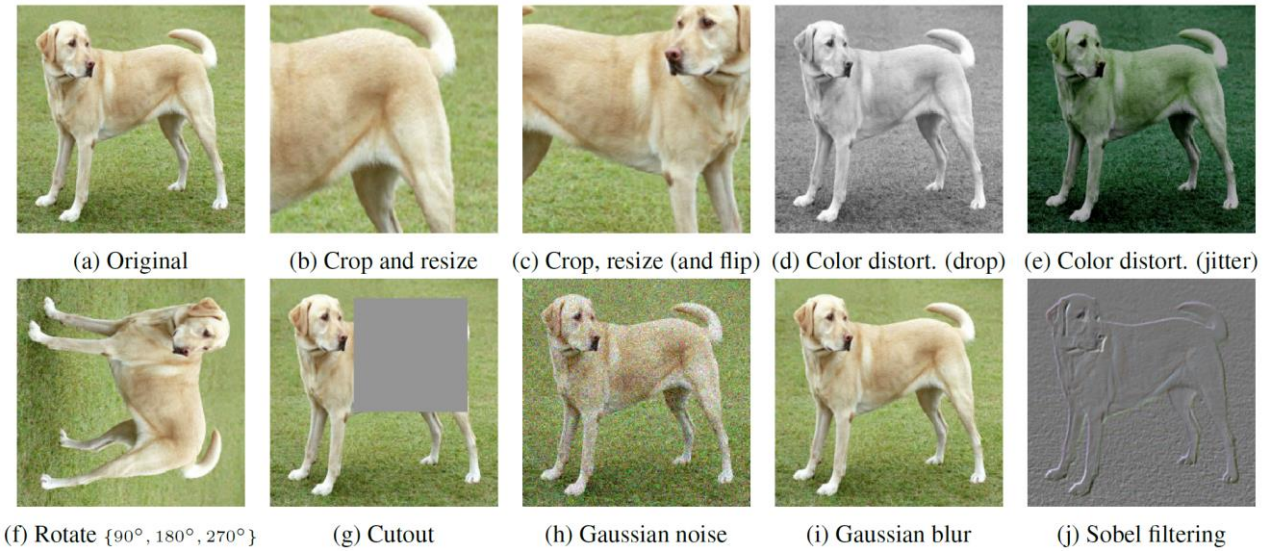
Sohn et al, "Improved Deep Metric Learning with Multi-class N-pair Loss Objective", NeurIPS 2016
Wu et al, "Unsupervised Feature Learning by Non-Parametric Instance-Level Discrimination", CVPR 2018
Van den Oord et al, "Representation Learning with Contrastive Predictive Coding", NeurIPS 2018

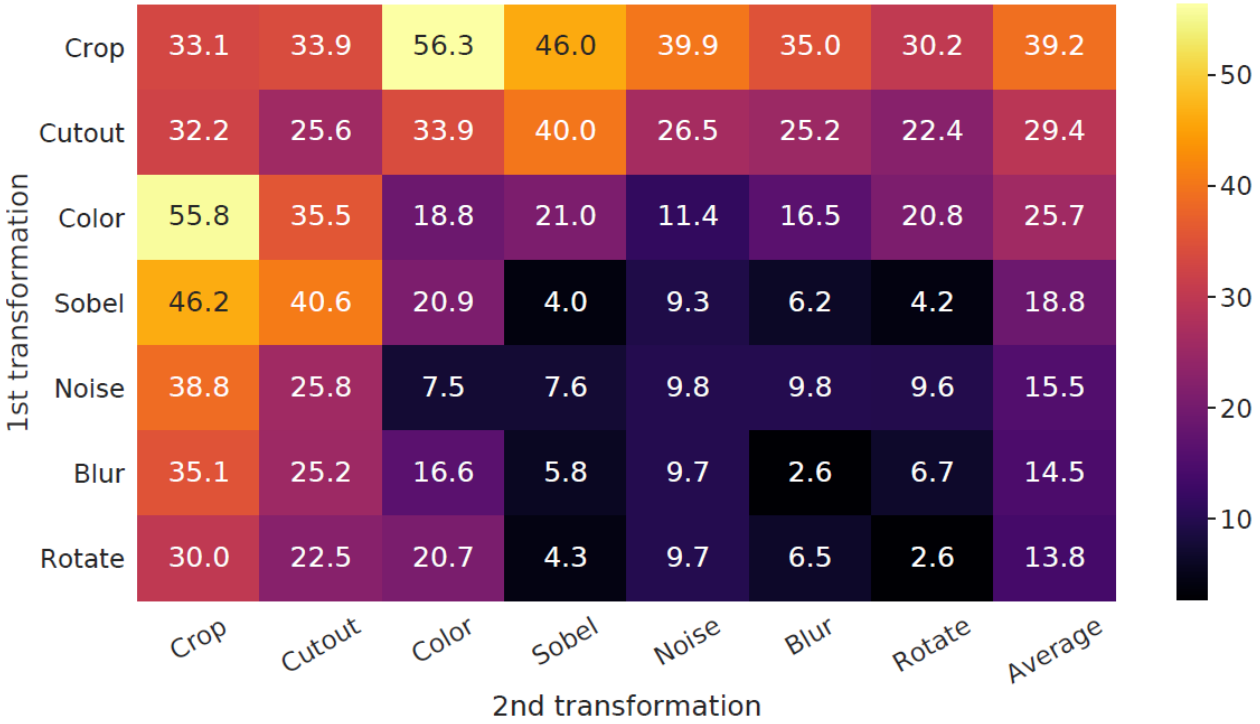
Hjelm et al, "Learning deep representations by mutual information estimation and maximization", ICLR 2019
Bachman et al, "Learning Representations by Maximizing Mutual Information Across Views", NeurIPS 2019
Henaff et al, "Data-Efficient Image Recognition with Contrastive Predictive Coding", ICML 2020

Tian et al, "Contrastive Multiview Coding", ECCV 2020
He et al, "Momentum Contrast for Unsupervised Visual Representation Learning", CVPR 2020
Chen et al, "A Simple Framework for Contrastive Learning of Visual Representations", ICML 2020

Impact of Data Augmentation

- SimCLR: Composition of data augmentation operations is crucial for learning good representations.
  - Random resized crop, color jitter, Gaussian blurring, …

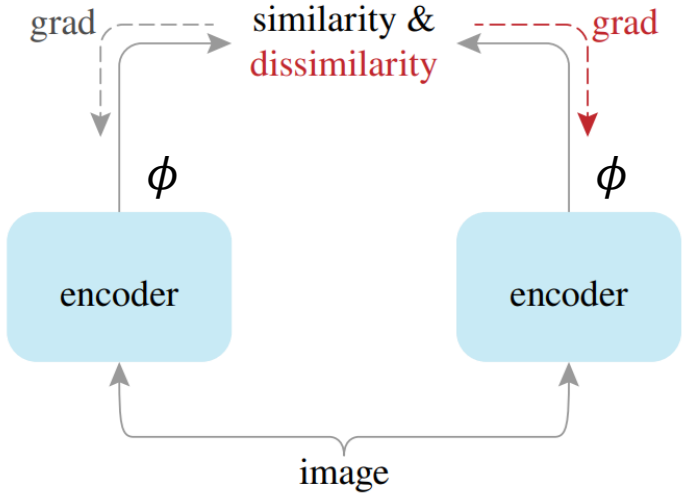




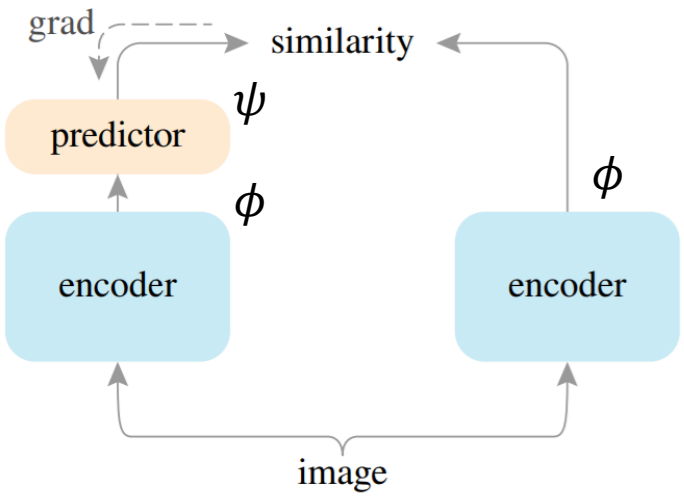
Chen et al, "A Simple Framework for Contrastive Learning of Visual Representations", ICML 2020

Contrastive vs. Feature Reconstruction

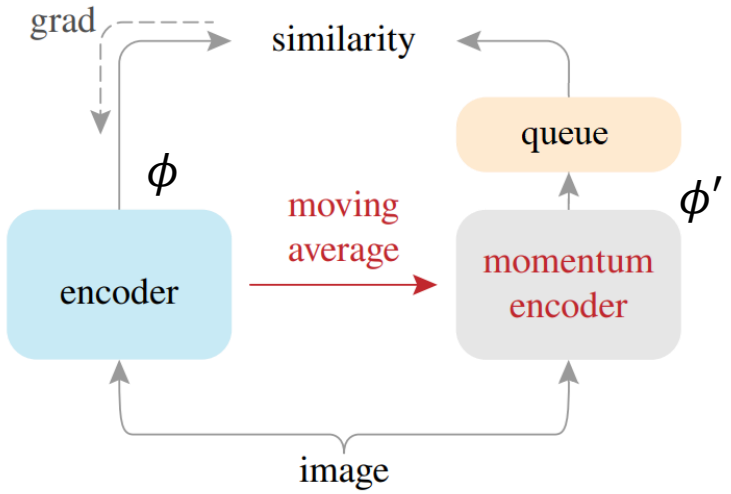
- SimCLR $L_i = -\log \frac{\exp(s_{i,j}/\tau)}{\sum_{\substack{k=1 \\ k \neq i}}^{2N} \exp(s_{i,k}/\tau)}$
  - Contrastive learning
  - 2N-2 negatives
- MoCo $L_i = -\log \frac{\exp(s_{i,j}/\tau)}{\exp(s_{i,j}/\tau) + \sum_{k=1}^{K} \exp(s_{i,k}/\tau)}$
  - Contrastive learning
  - Momentum encoder
  - Negatives from queue (size K)
- BYOL $L_i = -\frac{\psi(\phi(x_i))^T \phi'(x_j)}{\|\psi(\phi(x_i))\| \cdot \|\phi'(x_j)\|}$
  - Feature reconstruction
  - Momentum encoder
- SimSiam $L_i = -\frac{\psi(\phi(x_i))^T \phi(x_j)}{\|\psi(\phi(x_i))\| \cdot \|\phi(x_j)\|}$
  - Feature reconstruction



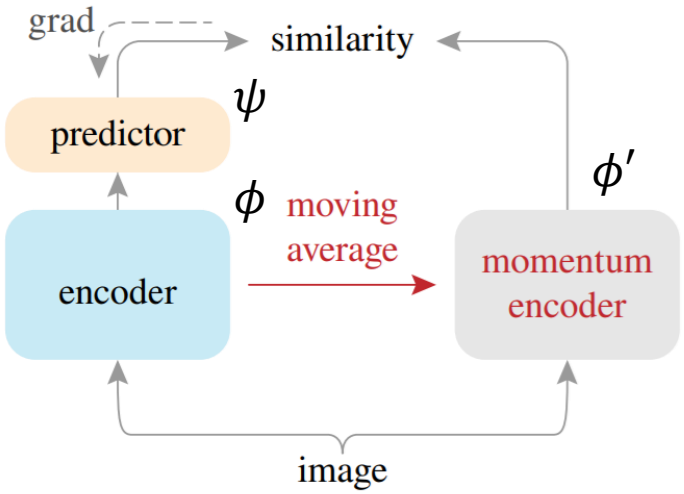
SimCLR



SimSiam



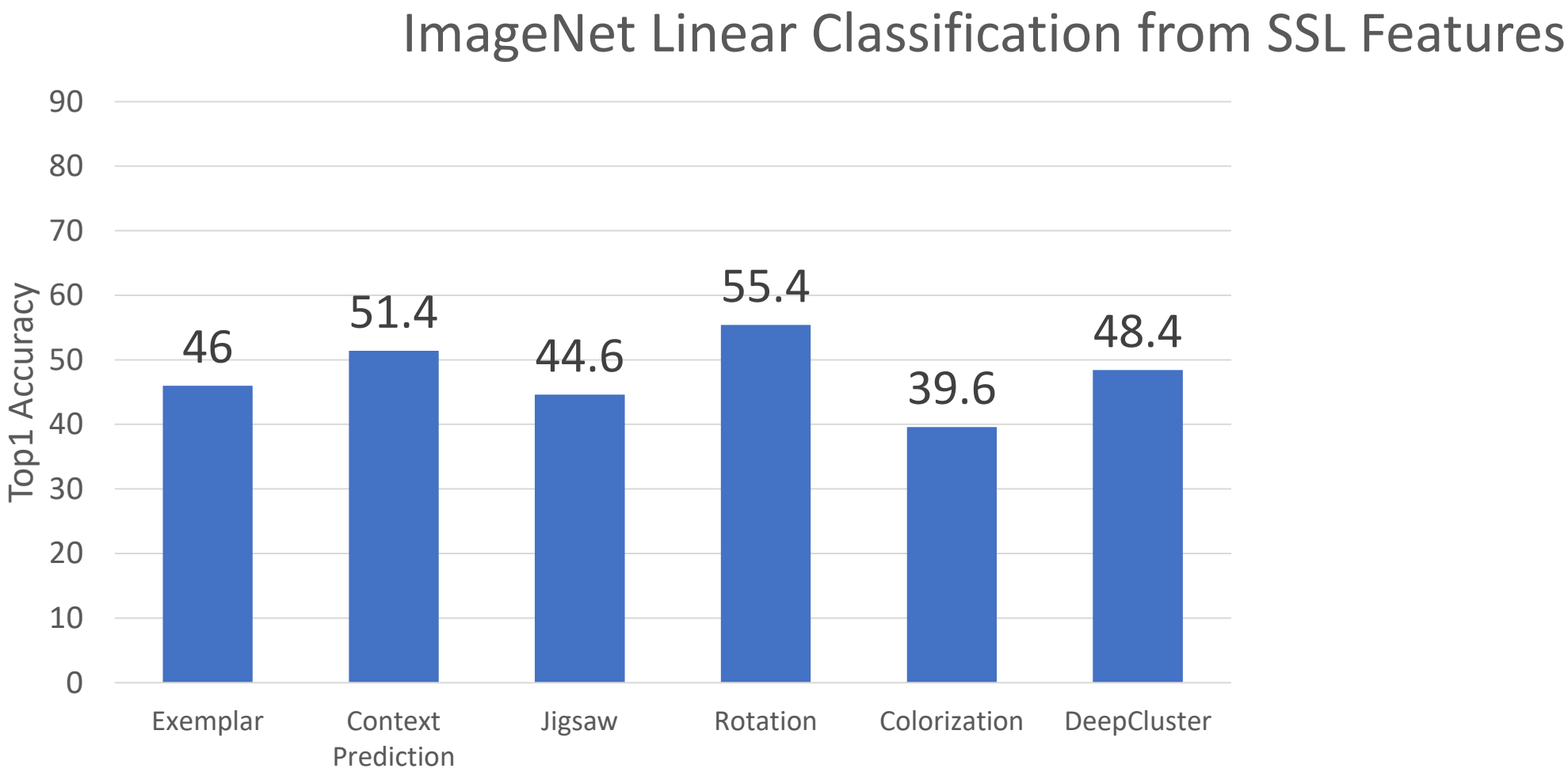
MoCo



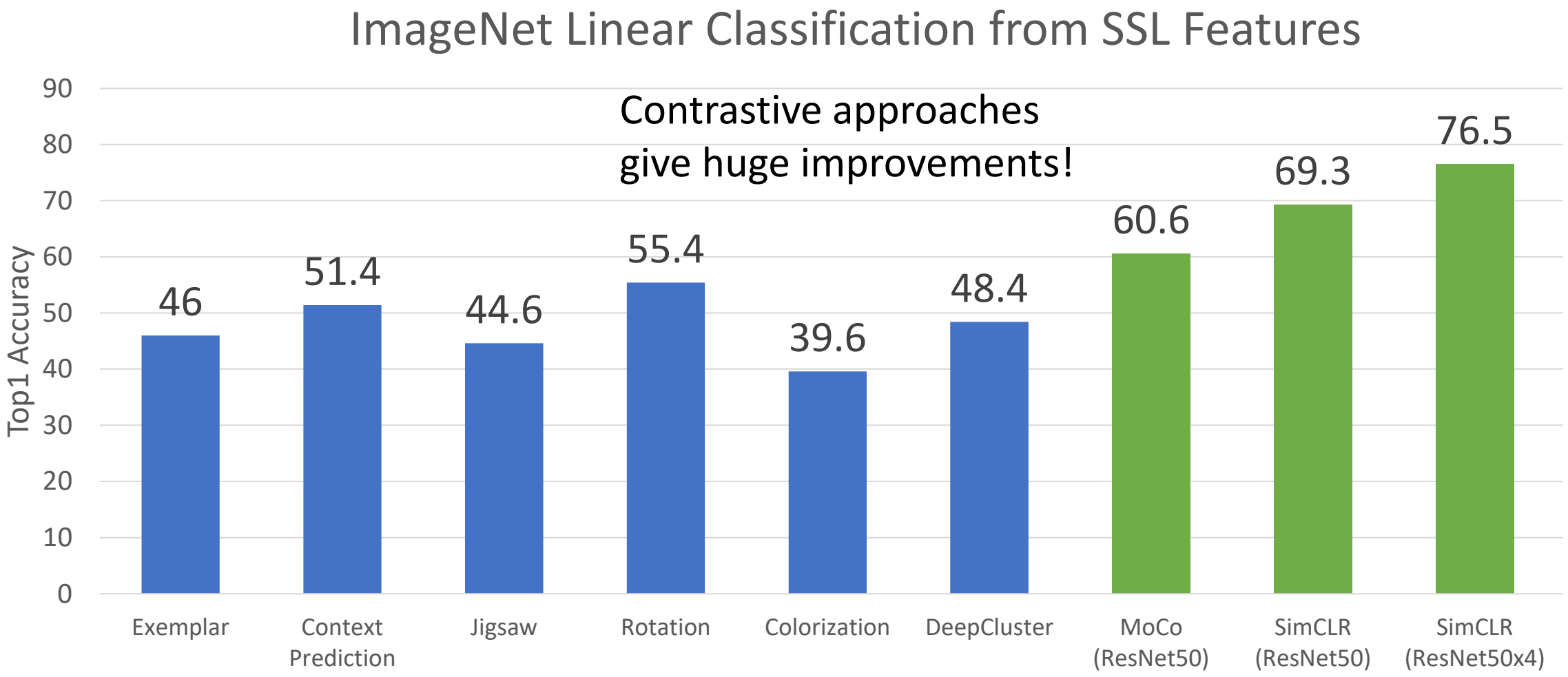
BYOL

He et al, "Momentum Contrast for Unsupervised Visual Representation Learning", CVPR 2020
Chen et al, "A Simple Framework for Contrastive Learning of Visual Representations", ICML 2020
Grill et al, "Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning", NeurIPS 2020
Chen and He, "Exploring simple Siamese representation learning", CVPR 2021

Comparison of SSL Methods

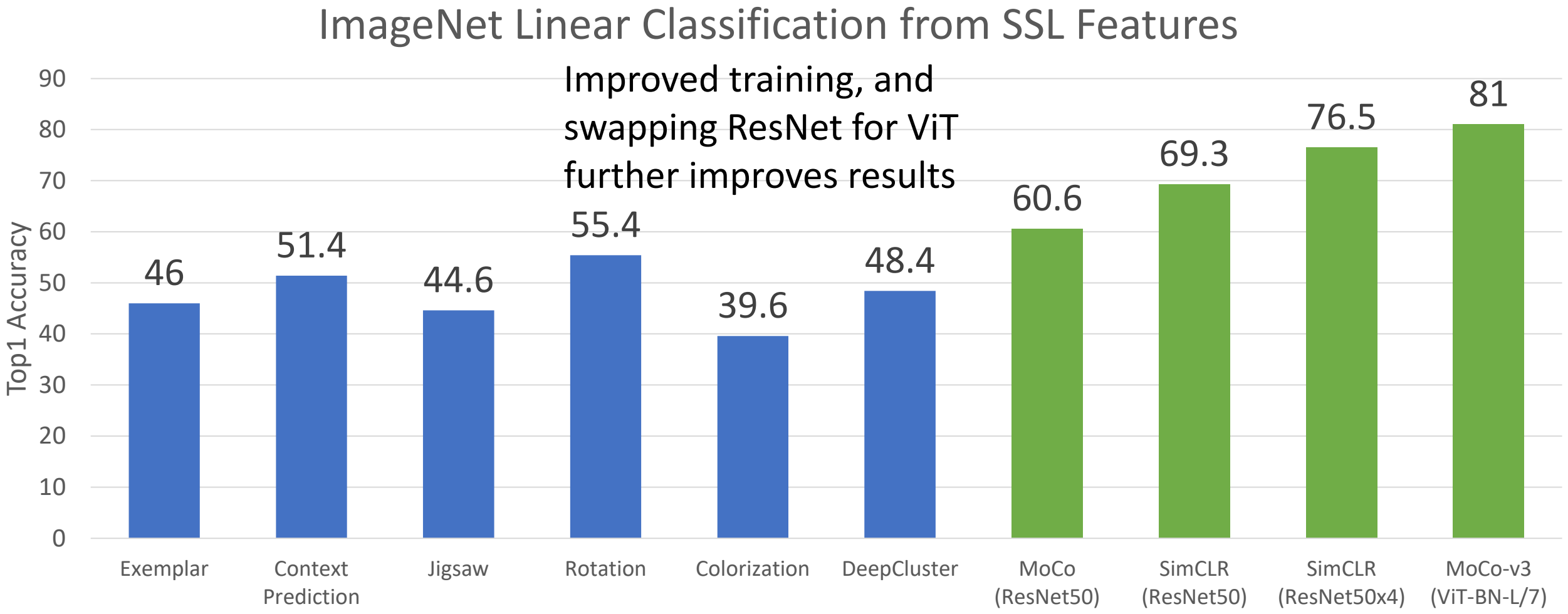


Comparison of SSL Methods



He et al, "Momentum Contrast for Unsupervised Visual Representation Learning", CVPR 2020
Chen et al, "A Simple Framework for Contrastive Learning of Visual Representations", ICML 2020
Chen et al, "An Empirical Study of Training Self-Supervised Vision Transformers", ICCV 2021

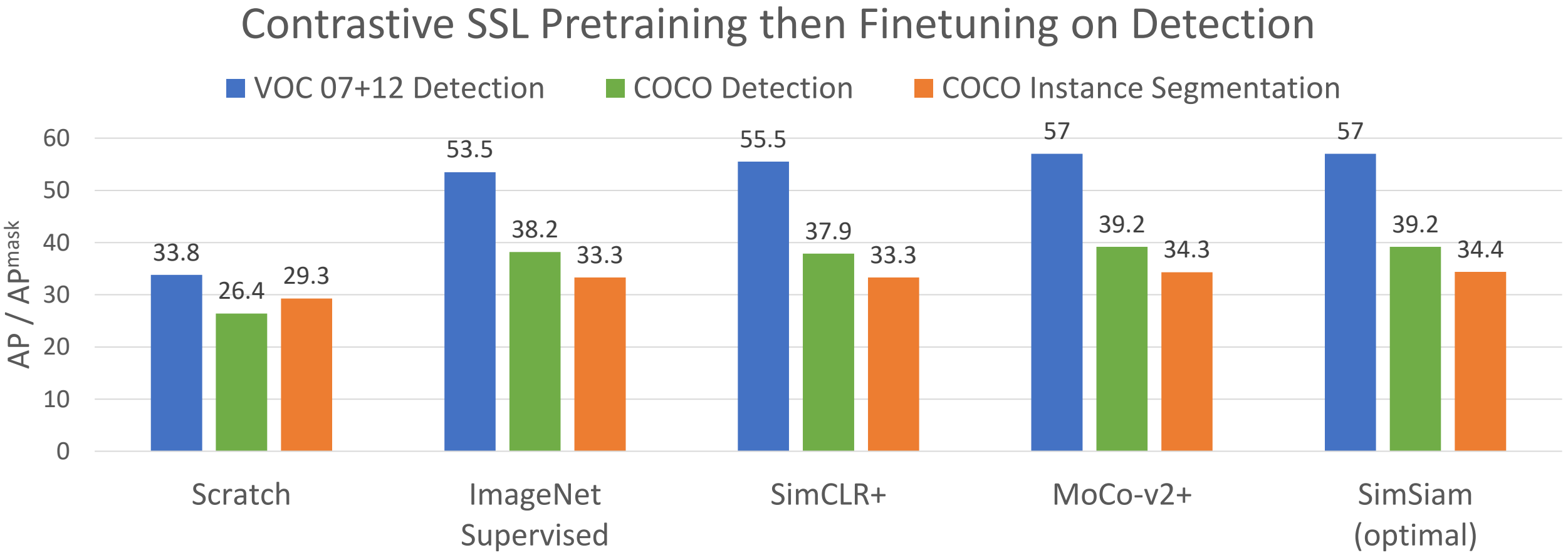
Comparison of SSL Methods



He et al, "Momentum Contrast for Unsupervised Visual Representation Learning", CVPR 2020
Chen et al, "A Simple Framework for Contrastive Learning of Visual Representations", ICML 2020
Chen et al, "An Empirical Study of Training Self-Supervised Vision Transformers", ICCV 2021

Supervised vs. Self-Supervised Learning

- SSL pretraining on ImageNet matches its supervised counterpart

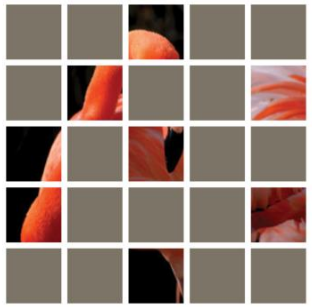


He et al, "Momentum Contrast for Unsupervised Visual Representation Learning", CVPR 2020
Chen et al, "A Simple Framework for Contrastive Learning of Visual Representations", ICML 2020
Chen et al, "Improved Baselines with Momentum Contrastive Learning", arXiv 2020
Chen and He, "Exploring simple Siamese representation learning", CVPR 2021

Masked Autoencoders (MAE)

A ~~new~~ old method dethrones contrastive learning? Denoising Autoencoder with Vision Transformer

Divide image into nonoverlapping patches, discard most of them



input

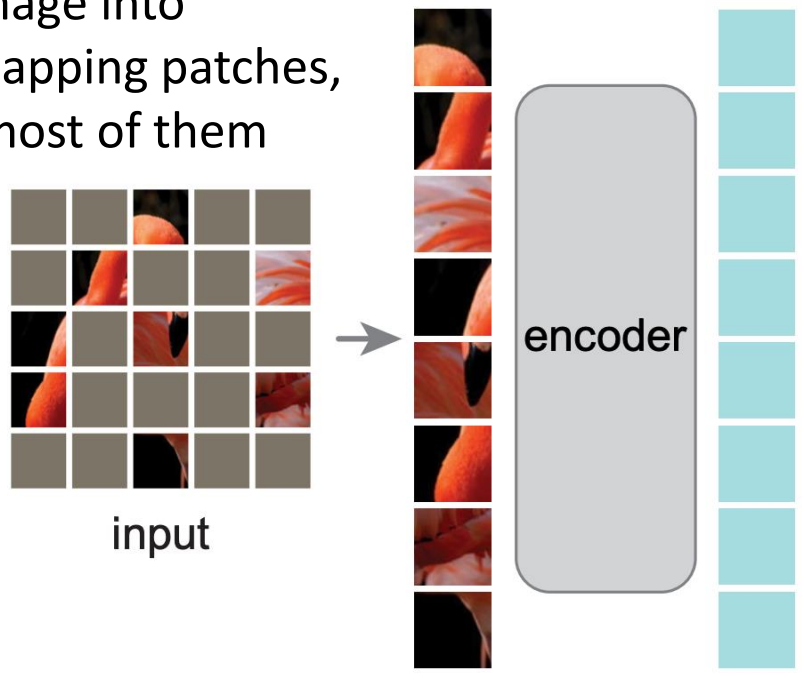
He et al, “Masked Autoencoders are Scalable Vision Learners”, CVPR 2022

Masked Autoencoders (MAE)

A ~~new~~ old method dethrones contrastive learning? Denoising Autoencoder with Vision Transformer

Divide image into nonoverlapping patches, discard most of them

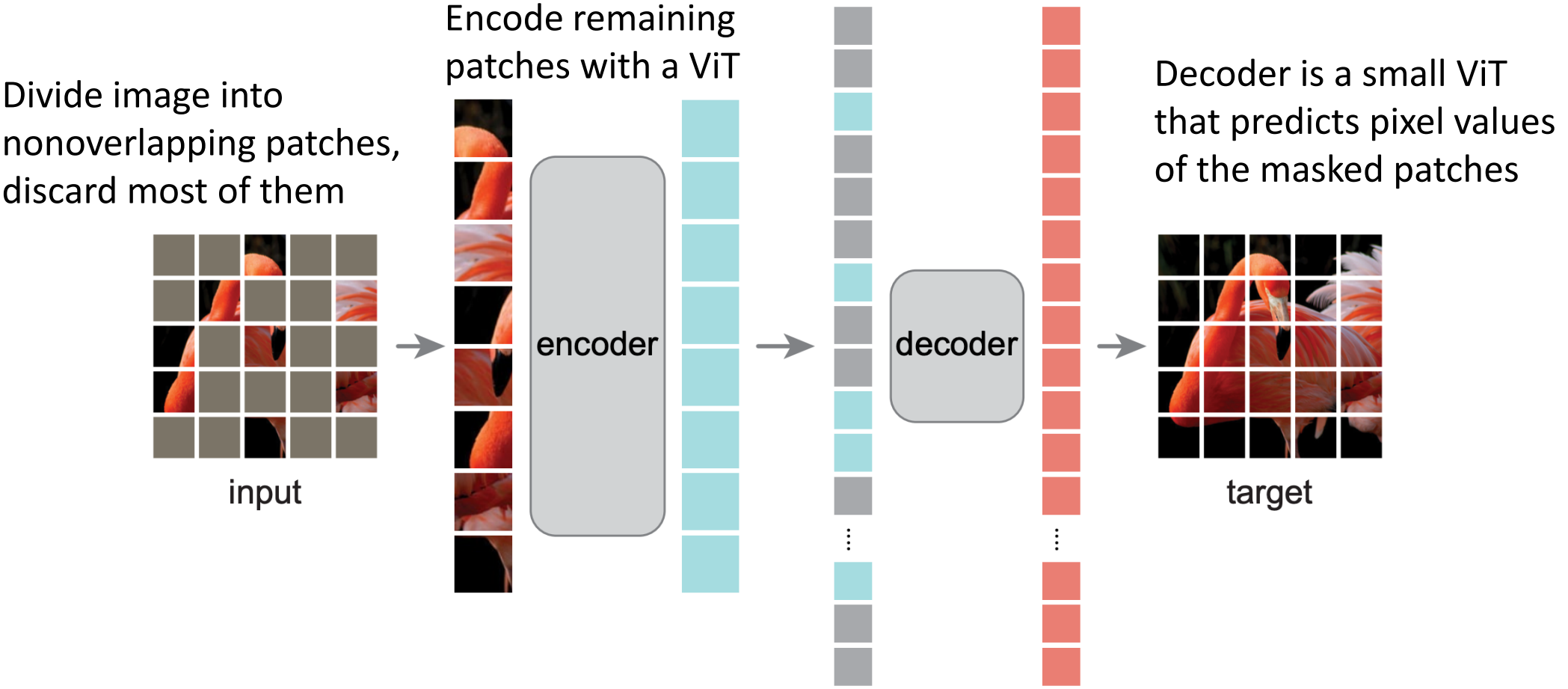
Encode remaining patches with a ViT



He et al, “Masked Autoencoders are Scalable Vision Learners”, CVPR 2022

Masked Autoencoders (MAE)

A ~~new~~ old method dethrones contrastive learning? Denoising Autoencoder with Vision Transformer



He et al, "Masked Autoencoders are Scalable Vision Learners", CVPR 2022

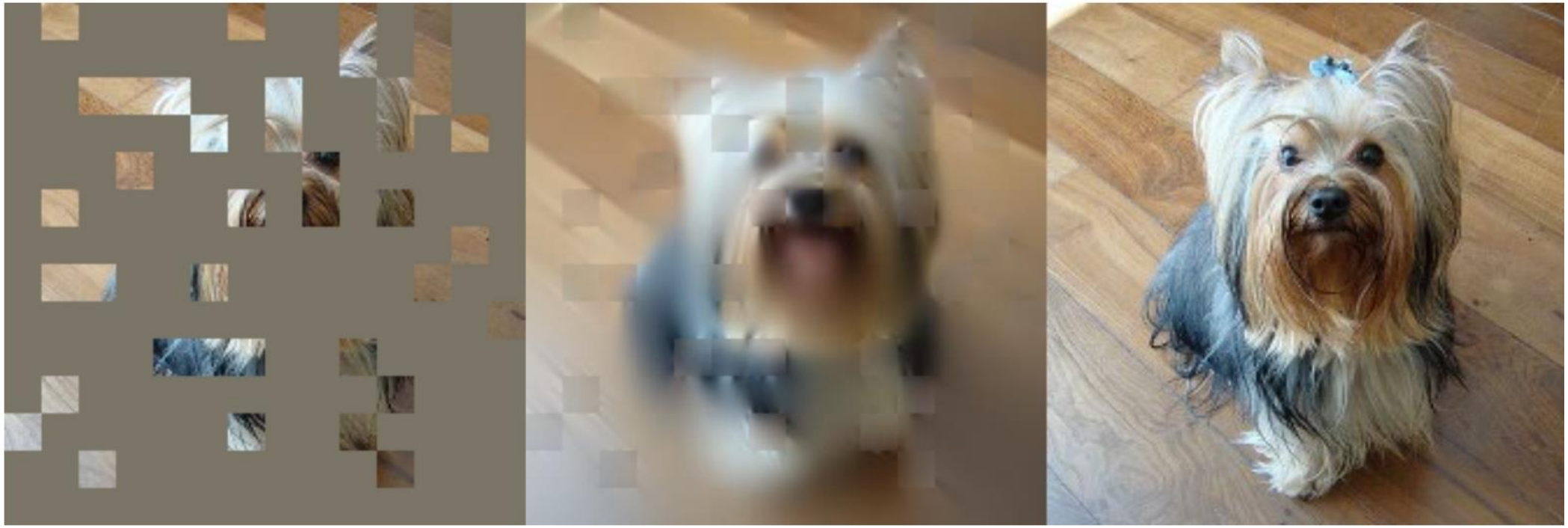
Masked Autoencoders (MAE): Reconstructions

Input Patches | Prediction | Actual Image

He et al, “Masked Autoencoders are Scalable Vision Learners”, CVPR 2022

Masked Autoencoders (MAE): Reconstructions

Input Patches | Prediction | Actual Image

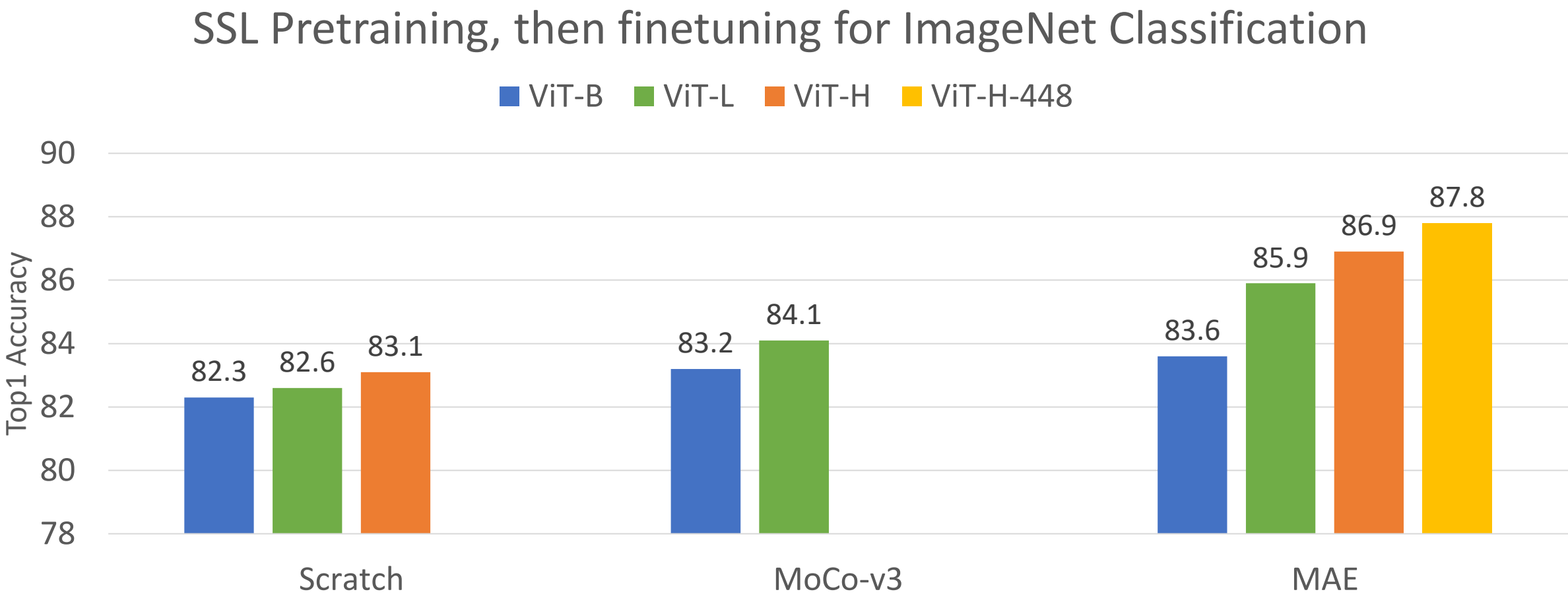


He et al, “Masked Autoencoders are Scalable Vision Learners”, CVPR 2022

Masked Autoencoders (MAE): Reconstructions

Input Patches | Prediction | Actual Image

He et al, “Masked Autoencoders are Scalable Vision Learners”, CVPR 2022



MAE Pretraining outperforms training from scratch, and allows scaling to larger ViT models

He et al, “Masked Autoencoders are Scalable Vision Learners”, CVPR 2022

What Is Going on These Days

- Active ongoing researches on self-supervised learning
  - SimCLR is cited > 7k times in < 3 years (Feb 13, 2020 ~ Nov 24, 2022)

A Simple Framework for Contrastive Learning of Visual ... - arXiv

T Chen 저술 · 2020 · 7103회 인용 — Abstract: This paper presents SimCLR: **a simple framework for contrastive learning of visual representations**. We simplify recently…

- Further improvements
  - [SwAV] Caron et al, “Unsupervised Learning of Visual Features by Contrasting Cluster Assignments”, NeurIPS 2020
  - [OBoW] Gidaris et al, “OBoW: Online Bag-of-Visual-Words Generation for Self-Supervised Learning”, CVPR 2021
  - [DeiT] Brigato et al, “Training data-efficient image transformers & distillation through attention”, ICML 2021
  - [Barlow Twins] Zbontar et al, “Barlow Twins: Self-Supervised Learning via Redundancy Reduction”, ICML 2021
  - [MoCo v3] Chen et al., “An empirical study of training self-supervised Vision Transformers”, ICCV 2021
  - [DINO] Caron et al, “Emerging Properties in Self-Supervised Vision Transformers”, ICCV 2021
  - [VICReg] Bardes et al, “VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning”, ICLR 2022
  - [BEiT] Bao et al, “BEiT: BERT Pre-Training of Image Transformers”, ICLR 2022
  - [SimMIM] Xie et al, “SimMIM: A Simple Framework for Masked Image Modeling”, CVPR 2022
  - [MSN] Assran et al, “Masked Siamese Networks for Label-Efficient Learning”, arXiv 2022
  - [MCMAE] Gao et al, “MCMAE: Masked Convolution Meets Masked Autoencoders”, NeurIPS 2022
  - [CAE] X Chen et al, “Context Autoencoder for Self-Supervised Representation Learning”, arXiv 2022

An Issue on Self-Supervised Learning in Vision

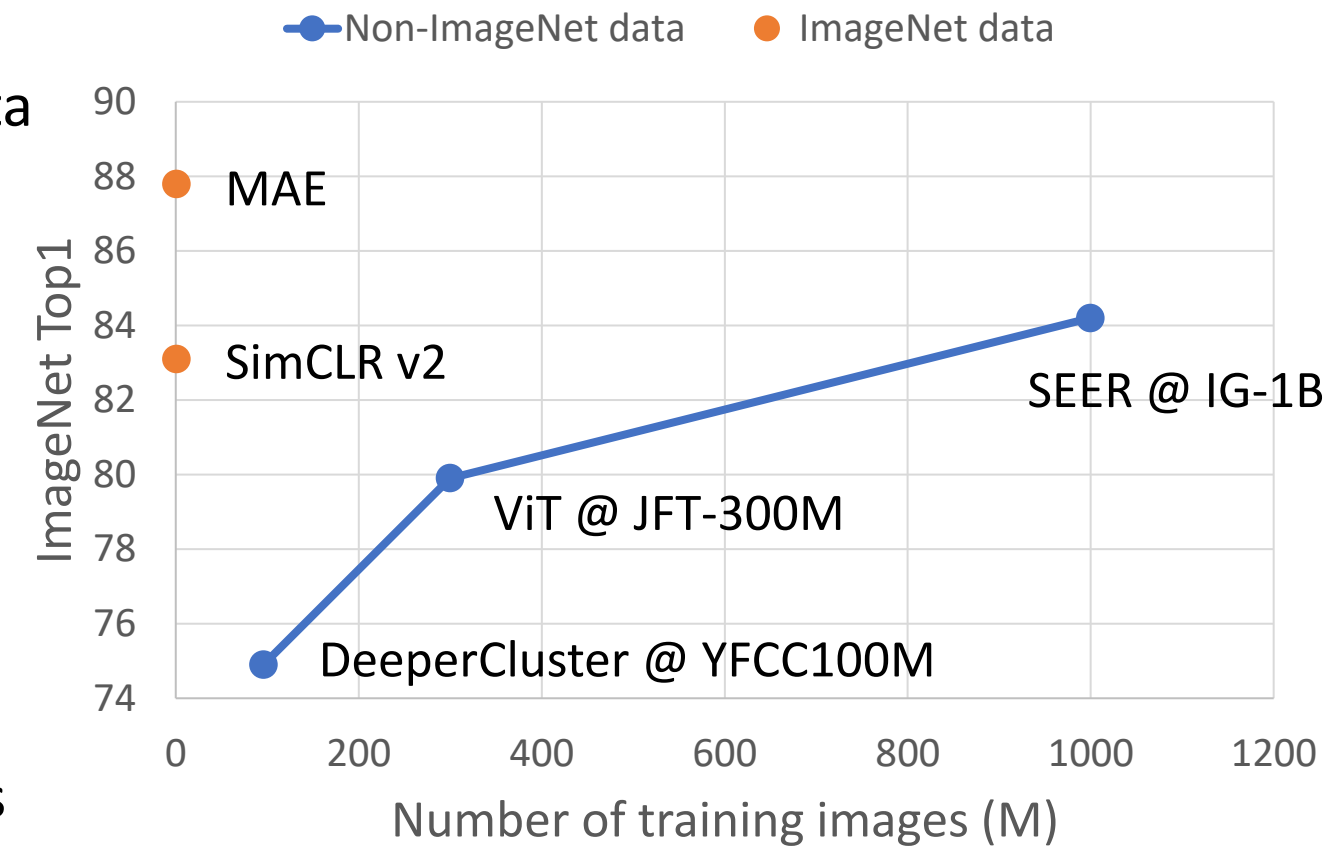
The motivation of SSL is scaling to large data that can't be labeled.

Most papers pretrain on (unlabeled) ImageNet, then evaluate on ImageNet!

Unlabeled ImageNet is still curated: single object per image, balanced classes.

Self-Supervised Learning on larger datasets hasn't been as successful as NLP

Idea: What if we go beyond isolated images?



Caron et al, "Unsupervised pre-training of images features on non-curated data", ICCV 2019
Chen et al, "Big self-supervised models are strong semi-supervised learners", NeurIPS 2020
Dosovitskiy et al, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale", ICLR 2021
Goyal et al, "Self-supervised Pretraining of Visual Features in the Wild", arXiv 2021
He et al, "Masked Autoencoders are Scalable Vision Learners", CVPR 2022

Multimodal Self-Supervised Learning

Don’t learn from isolated images -- take images together with some **context**

Video: Image together with adjacent video frames

Agrawal et al, “Learning to See by Moving”, ICCV 2015
Wang et al, “Unsupervised Learning of Visual Representations using Videos”, ICCV 2015
Pathak et al, “Learning Features by Watching Objects Move”, CVPR 2017

Sound: Image with audio track from video

Owens et al, “Ambient Sound Provides Supervision for Visual Learning”, ECCV 2016
Arandjelovic and Zisserman, “Look, Listen and Learn”, ICCV 2017

3D: Image with depth map or point cloud

Xie et al, “PointContrast: Unsupervised Pre-training for 3D Point Cloud Understanding”, ECCV 2020
Zhang et al, “Self-supervised pretraining of 3D features on any point-cloud”, CVPR 2021

Language: Image with natural-language text

Sariyildiz et al, “Learning Visual Representations with Caption Annotations”, ECCV 2020
Desai and Johnson, “VirTex: Learning Visual Representations from Textual Annotations”, CVPR 2021
Radford et al, “Learning Transferable Visual Models form Natural Language Supervision”, ICML 2021
Jia et al, “Scaling up Visual and Vision-Language Representation Learning with Noisy Text Supervision”, ICLR 2021
Desai et al, “RedCaps: Web-curated Image-Text data created by the people, for the people”, NeurIPS 2021

Why Language?

Large dataset of
(image, caption)

a dog with his head out the window of the car

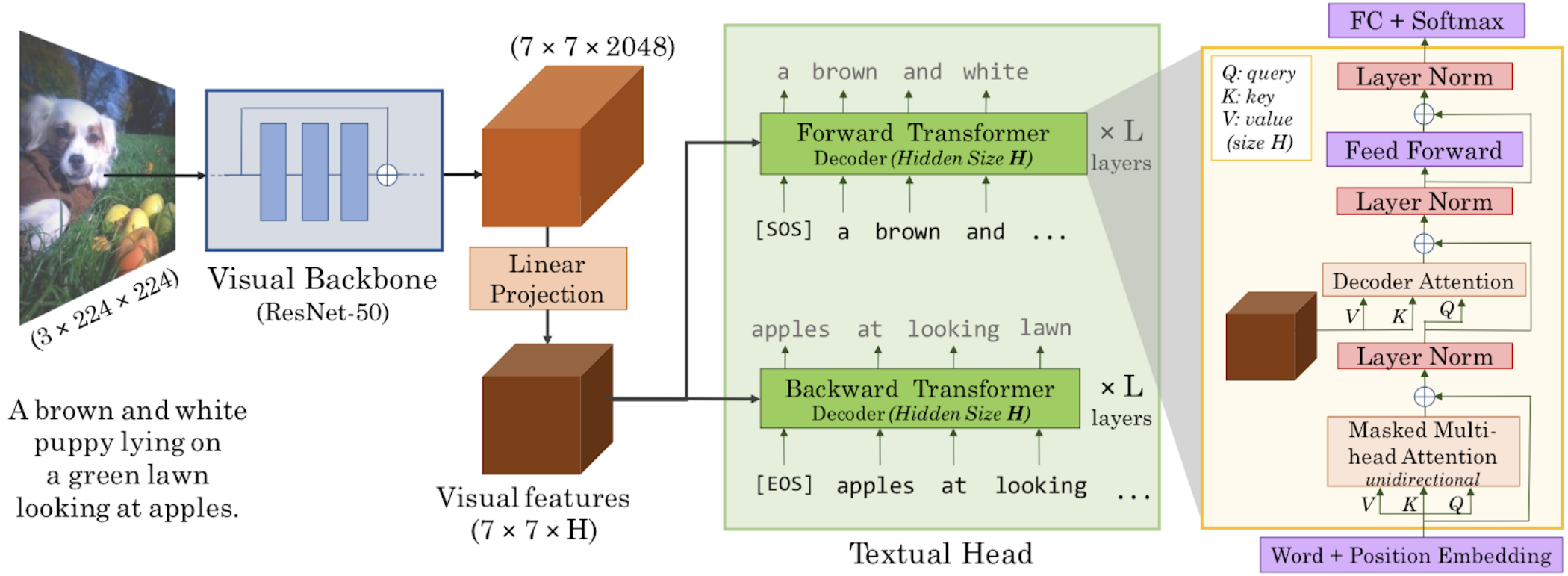
a black and orange cat is resting on a keyboard and yellow back scratcher

1. **Semantic density**: Just a few words give rich information

2. **Universality**: Language can describe any concept

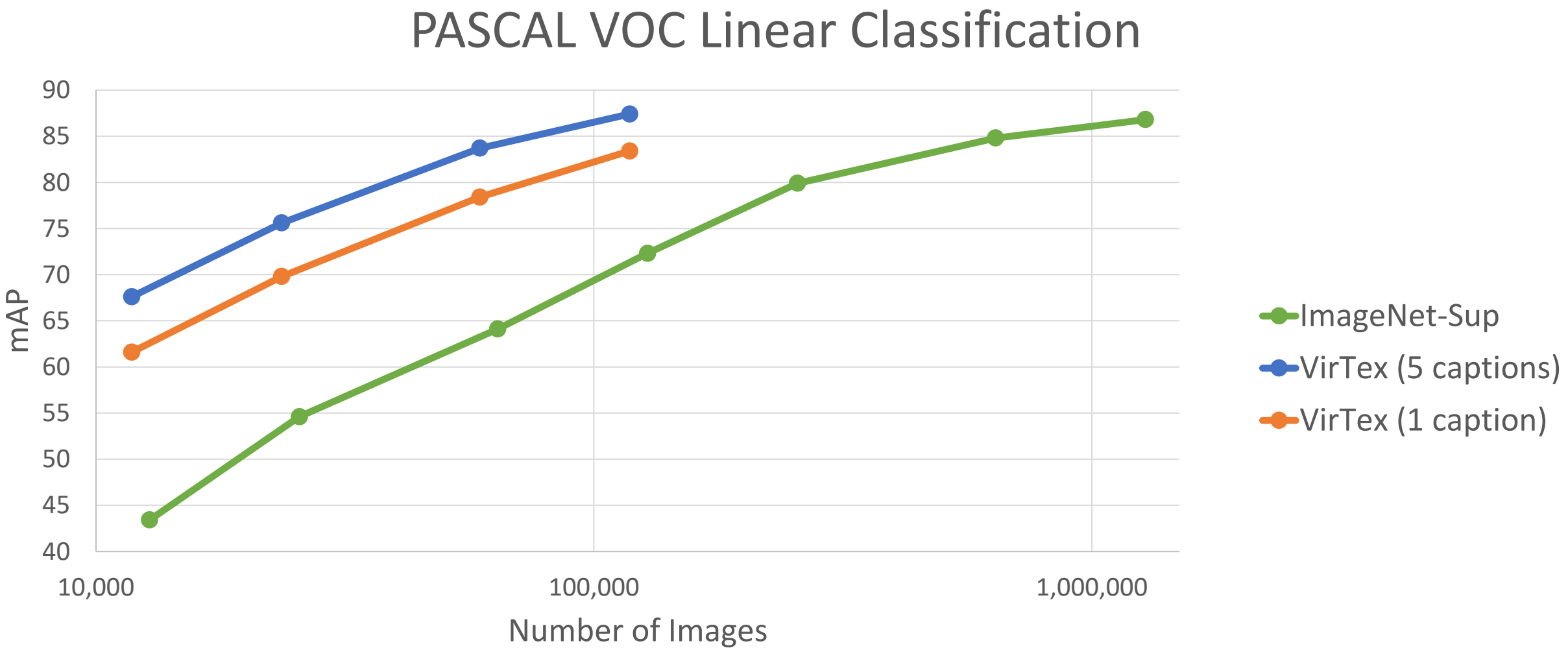
3. **Scalability**: Non-experts can easily caption images; data can also be collected from the web at scale

Generating Captions



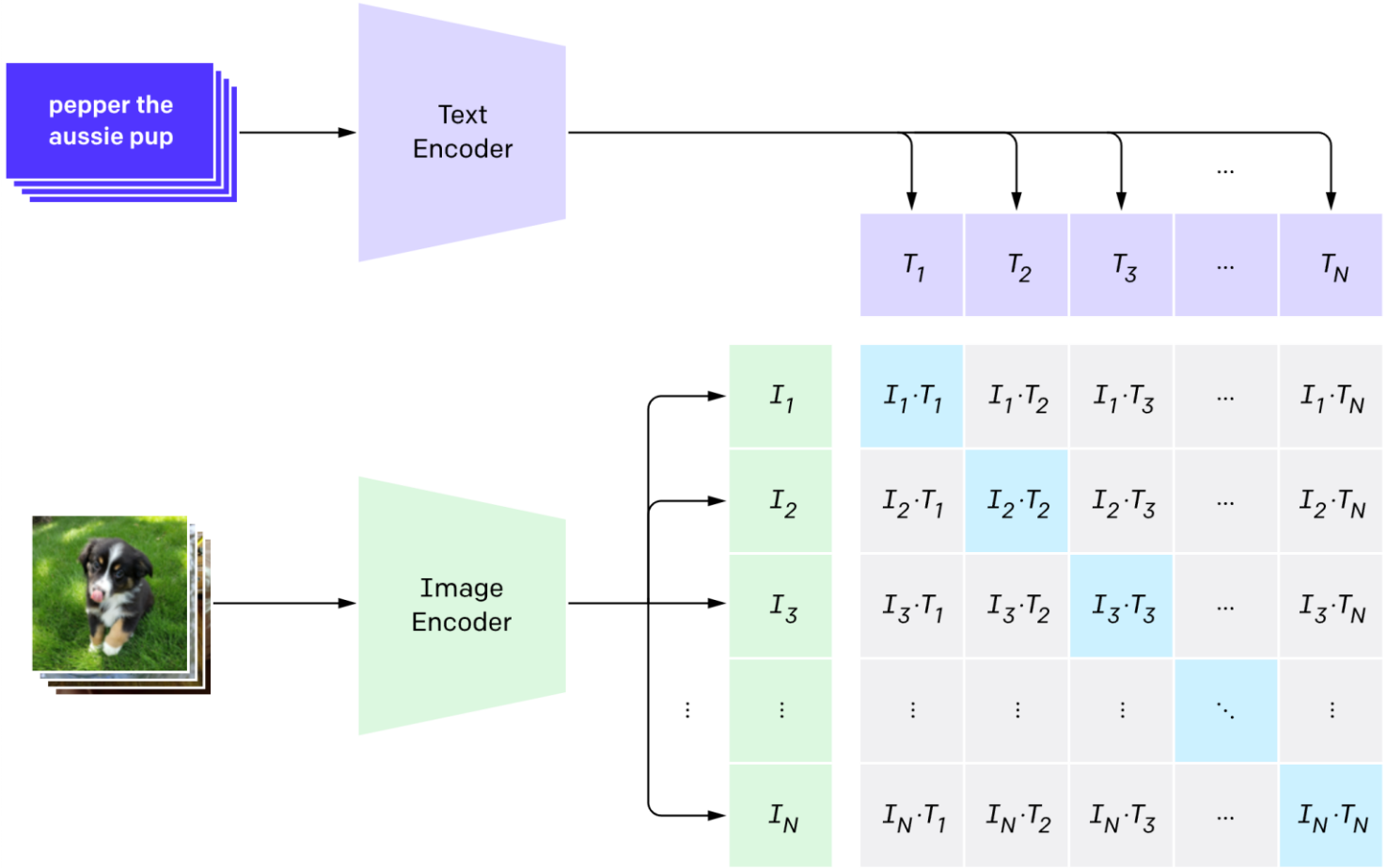
Desai and Johnson, "Desai and Johnson, "VirTex: Learning Visual Representations from Textual Annotations", CVPR 2021

Generating Captions



Desai and Johnson, “VirTex: Learning Visual Representations from Textual Annotations”, CVPR 2021

Matching Images and Text: CLIP



Contrastive loss: Each image predicts which caption matches

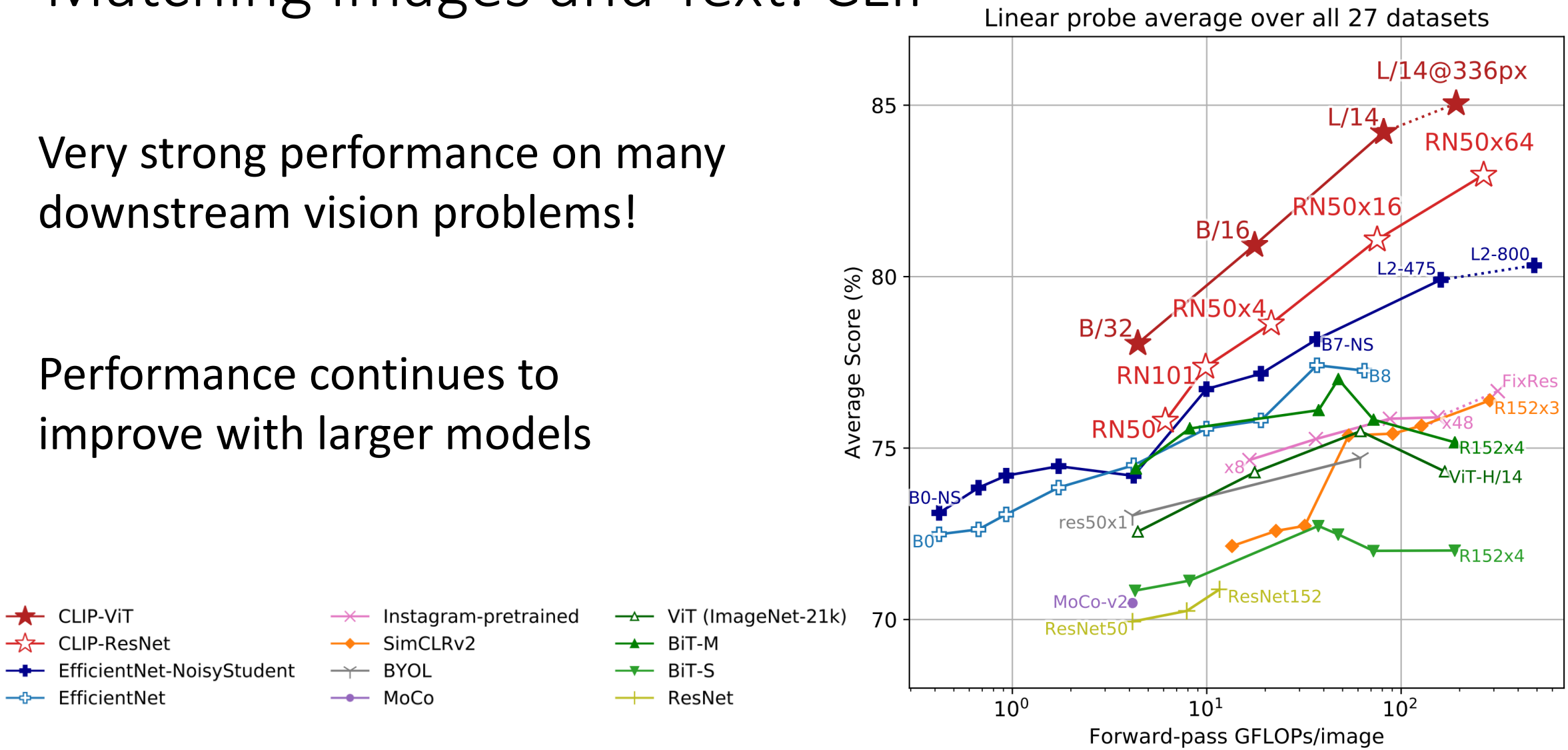
Large-scale training on 400M (image, text) pairs from the internet

Radford et al, "Learning Transferable Visual Models form Natural Language Supervision", ICML 2021
Jia et al, "Scaling up Visual and Vision-Language Representation Learning with Noisy Text Supervision", ICLR 2021

Matching Images and Text: CLIP

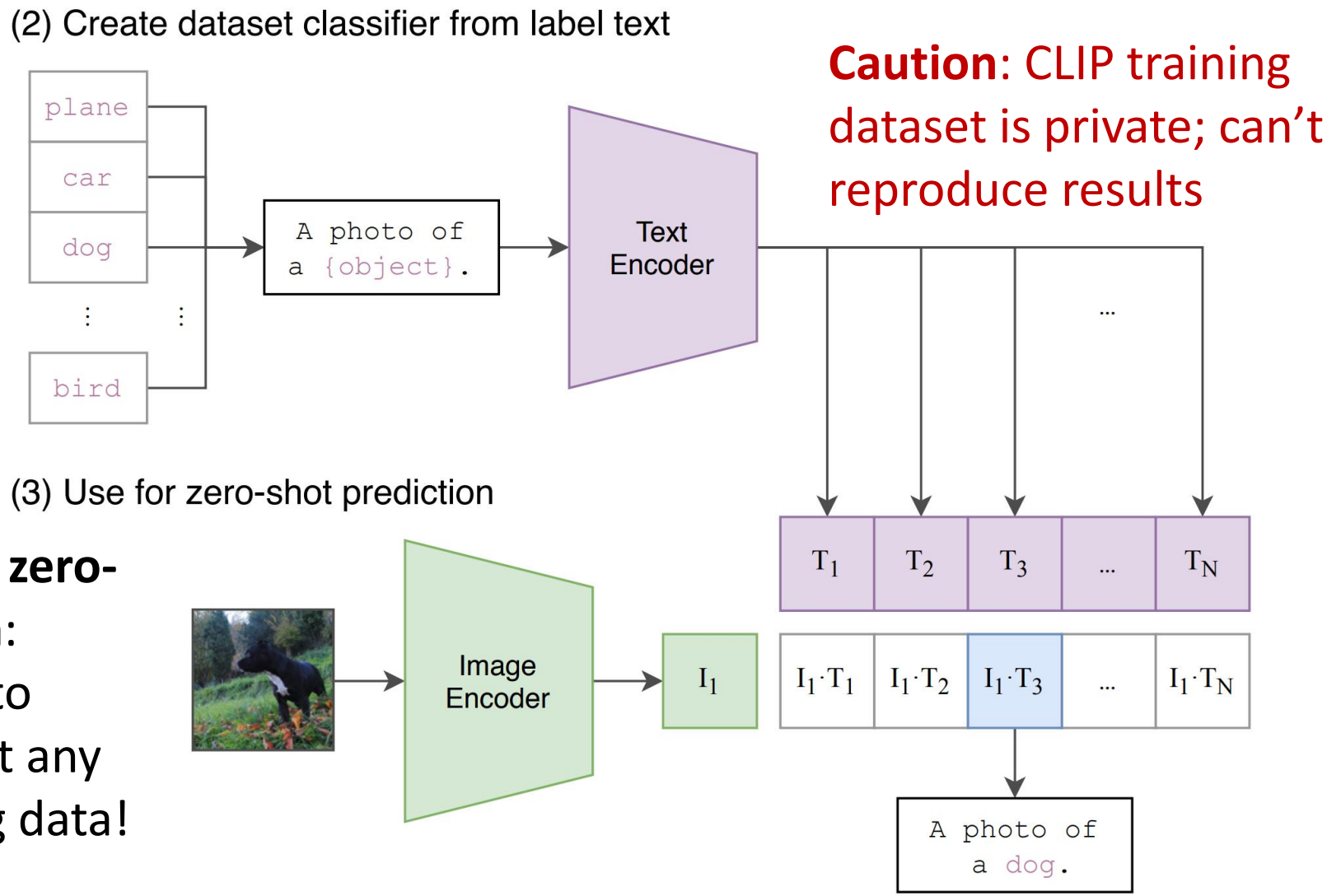
Very strong performance on many downstream vision problems!

Performance continues to improve with larger models



Radford et al, “Learning Transferable Visual Models form Natural Language Supervision”, ICML 2021

CLIP: Zero-Shot Classification



Caution: CLIP training dataset is private; can't reproduce results

Language enables **zero-shot classification**: Classify images into categories without any additional training data!

Radford et al, "Learning Transferable Visual Models form Natural Language Supervision", ICML 2021

DALL-E 2: Text-to-Image Generation

A rabbit detective sitting on a park bench and reading a newspaper in a victorian setting

A shark and a dolphin cruise hand-in-hand with an undersea city in the background

Robot dinosaurs versus monster trucks in the colosseum

Ramesh et al, "DALL·E 2", 2022. https://openai.com/dall-e-2/
Source: https://twitter.com/sama/status/1511724264629678084

Phenaki: Text-to-Video Generation

A photorealistic teddy bear is swimming in the ocean at San Francisco
The teddy bear goes under water
The teddy bear keeps swimming under the water with colorful fishes
A panda bear is swimming under water

A teddy bear diving in the ocean
A teddy bear emerges from the water
A teddy bear walks on the beach
Camera zooms out to the teddy bear in the campfire by the beach

Side view of an astronaut is walking through a puddle on mars
The astronaut is dancing on mars
The astronaut walks his dog on mars
The astronaut and his dog watch fireworks

Villegas et al, “Phenaki: Variable Length Video Generation From Open Domain Textual Description”, arXiv 2022

Source: https://phenaki.research.google/

Representation Learning Summary

- Transferring Supervised Learning model was pervasive in the 2010s
- Self-Supervised Learning (SSL) aims to scale up to larger datasets without human annotation; promising strategy in the 2020s
- First train for a **pretext** task, then **transfer** to **downstream** tasks
- Many pretext tasks: autoencoding, context prediction, jigsaw, inpainting, colorization, clustering, rotation prediction, …
- SSL has been successful in natural language processing
- Intense research on SSL in vision; current best are
  - Contrastive learning (and variants like feature reconstruction)
  - Masked autoencoding
- Multimodal SSL has been very successful; it seems promising!