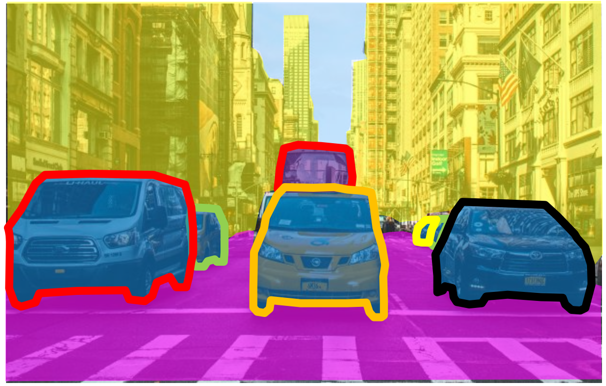
Photogrammetry & Robotics Lab

Machine Learning for Robotics and Computer Vision

Beyond Supervised Learning

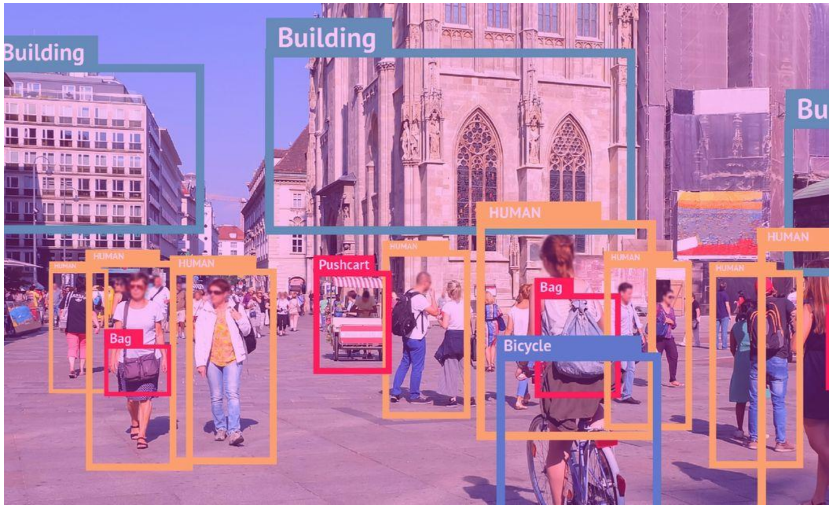
Jens Behley

Last Lecture



- Fine-grained scene understanding:
  - Semantic Segmentation
  - Instance Segmentation
  - Panoptic Segmentation
- Discussed common, popular approaches for segmentation in these domains.

Data, data, data



- Deep learning brought astonishing progress in visual perception
- Supervised learning on large annotated datasets made progress possible

Labeling data is expensive

- Labeling data is tedious and expensive
- Examples
  - Cityscapes: ~1.5 h per image ➔ 7500 h/312 days for 5k images
  - Mapillary Vistas: ~1.5 h per image → 4.2 years for 25k images
  - MS COCO: 22k h (category labeling) + 10k h (instance spotting) + 26k h (instance segmentation*) → 6.6 years
- Not included: Validation of annotations!

* 22h per 1000 segments, ~1.2M instances for 80 classes

Large datasets needed?

- Capacity of deep neural networks very large (millions of parameters)
- Commonly: More parameters = more training data

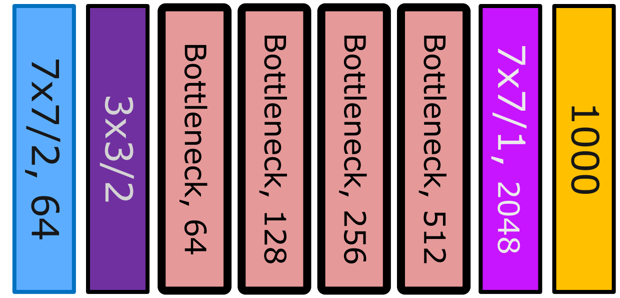
- **Question:** Do we always need first to invest lot of time and money to get labeled data?

- **Answer:** No!

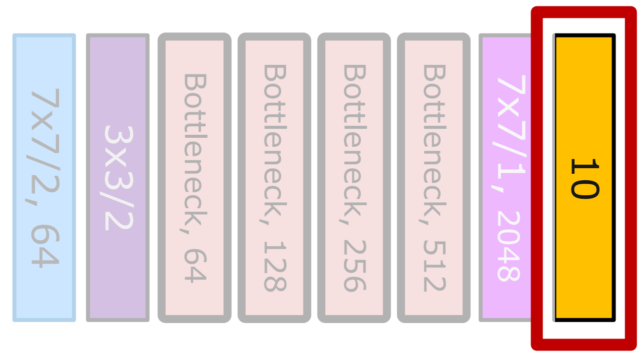
Pre-training & Fine-tuning

Stage 1:
Pre-training

(ImageNet)

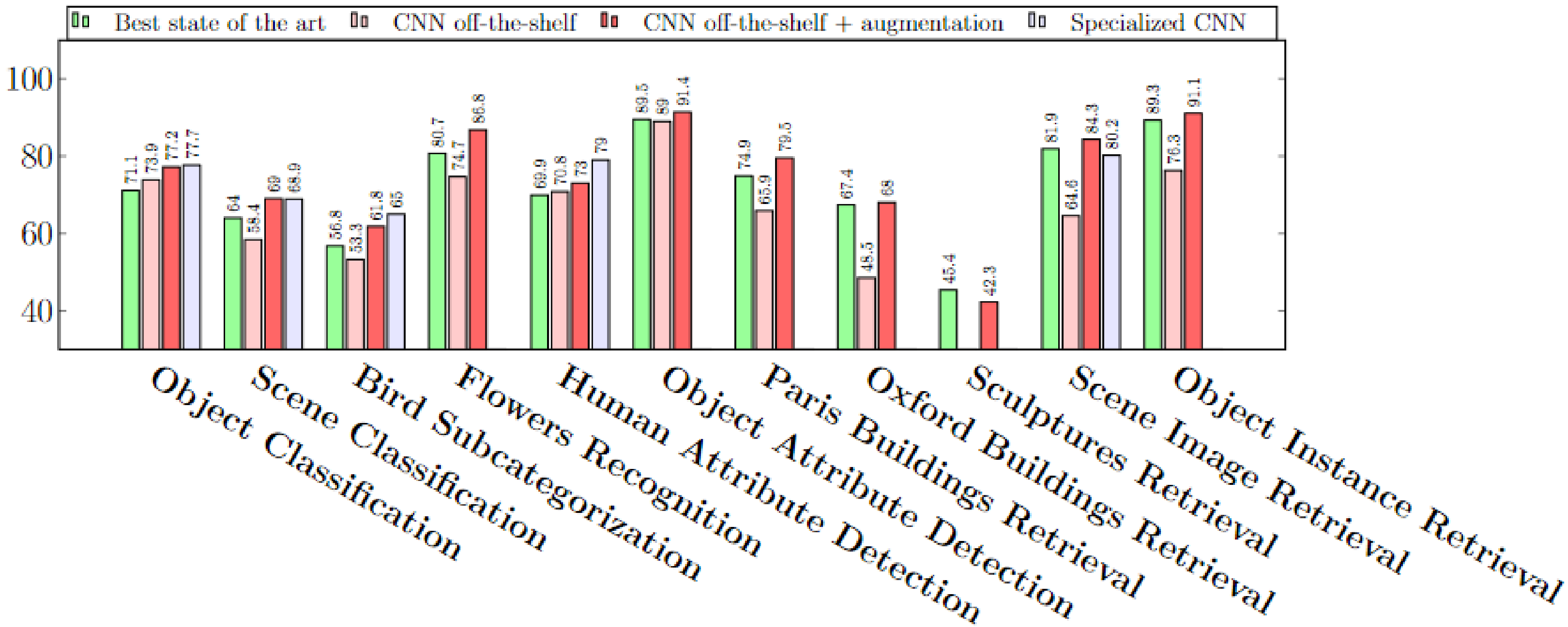


Stage 2:
Fine-tuning
(Targeted dataset)



- **Idea:** Take weights from ImageNet and train only part of the network for novel task/dataset
- Training with pre-trained weights is faster and less data intensive!

Pre-training on different tasks

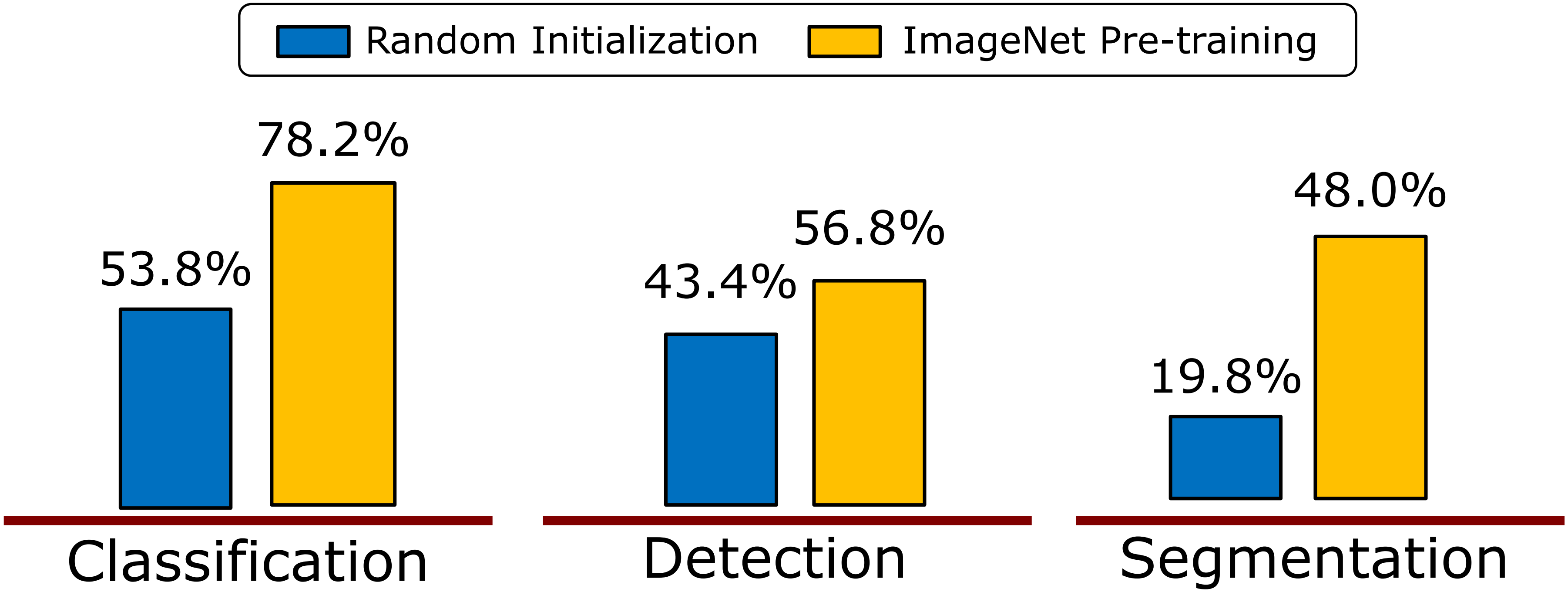


- ImageNet pre-trained features (with a bit data augmentation) performs well over a wide range of vision tasks
- Surprisingly beats consistently “traditional” state-of-the-art methods

[Razavian, 2014]

Figure from [Razavian, 2014]

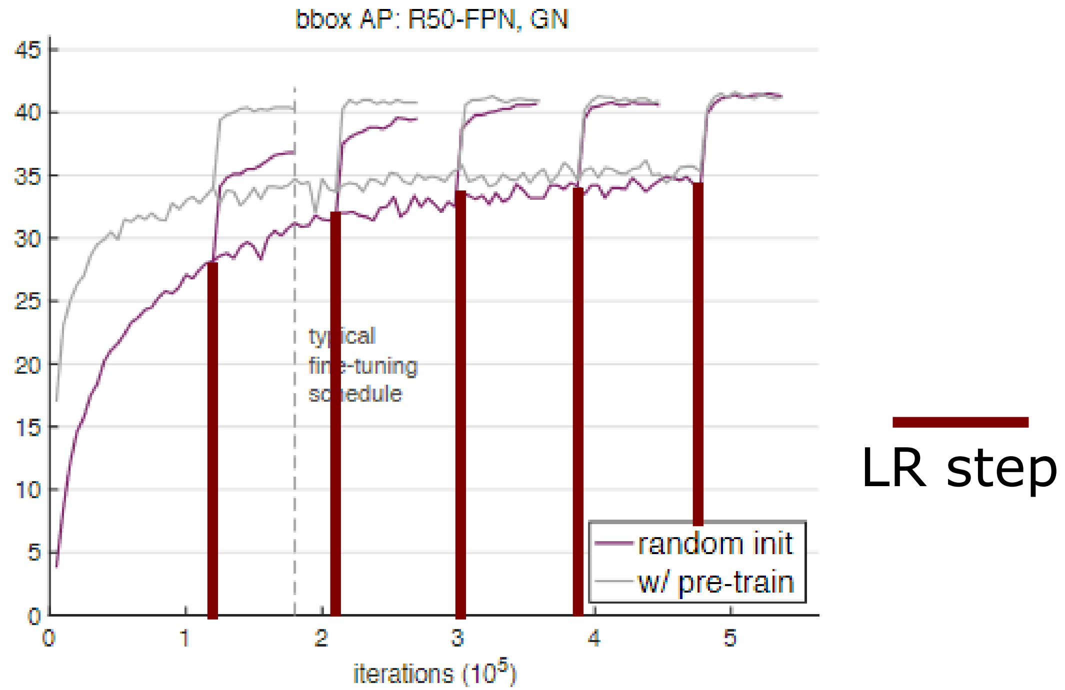
Pre-training vs. Random Init



- **Example:** Pascal VOC Classification, Detection, and Segmentation with same CNN backbone (AlexNet)
- Strong results of ImageNet pre-training vs. models trained "from scratch" on smaller dataset!

Figure from [Pathak, 2016]

Pre-training vs. Random Init

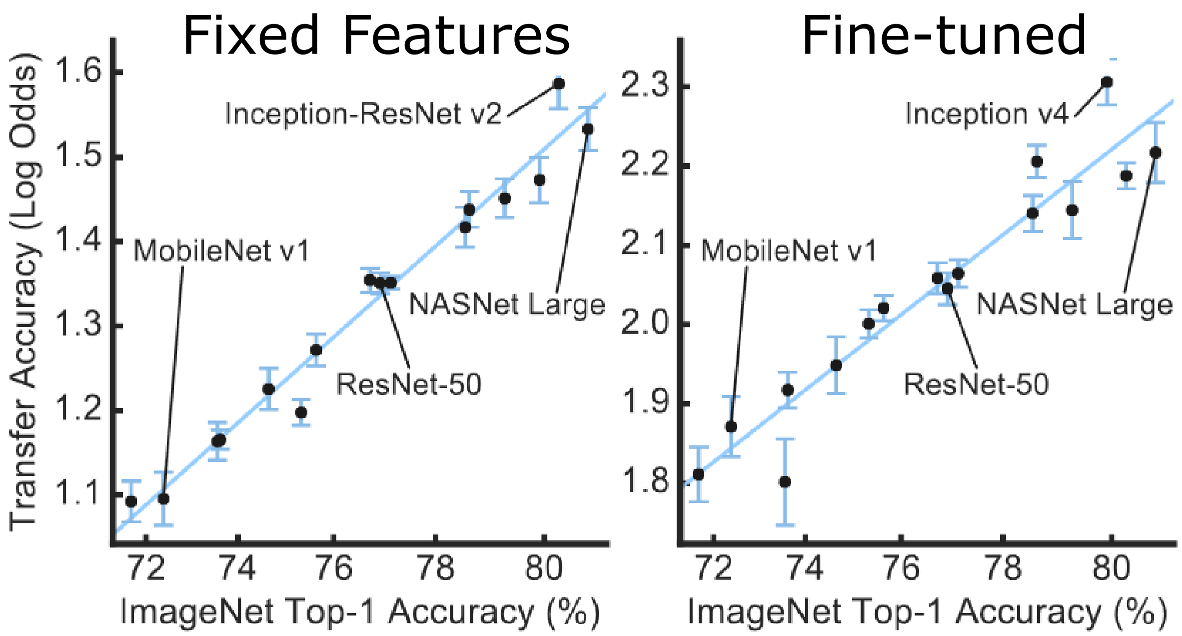


- ImageNet pre-training speeds up convergence
- But: ImageNet pre-training not necessarily leads to better performance in the end
- Requirement: **Enough** target data + **time** available

[He, 2019]

Figure from [He, 2019]

Influence of CNN Architecture



- Study on 16 architectures and performance on 12 target datasets
- Takeaway: Better performance on ImageNet leads to better transfer to other datasets!

[Kornblith, 2019]

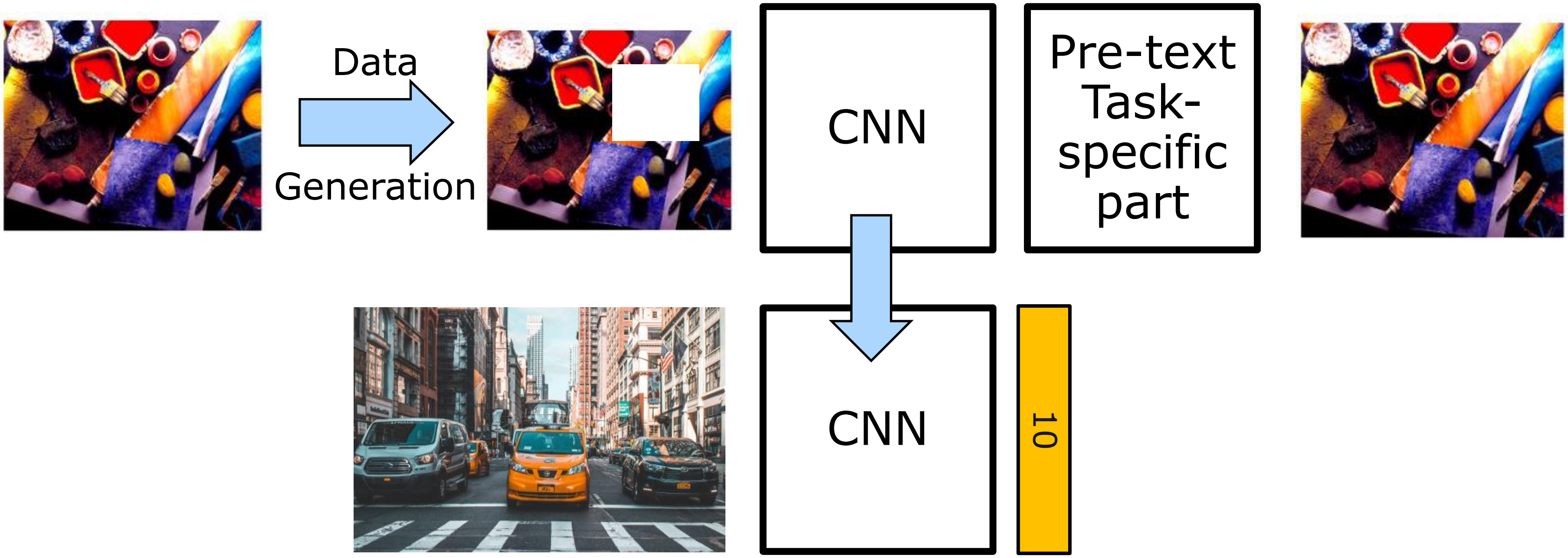
Figure from [Kornblith, 2019]

Domain/Modality Gap

- But performance usually degrades when features not specifically learned for the task or data
- Domain gap: ImageNet → Satellite, Medical images
- Modality gap: RGB vs. RGB-D vs. Hyperspectral Cameras

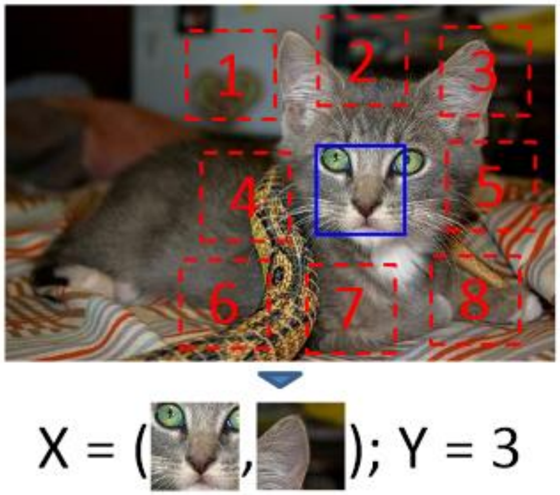
- So we are back at labeling lots of data?

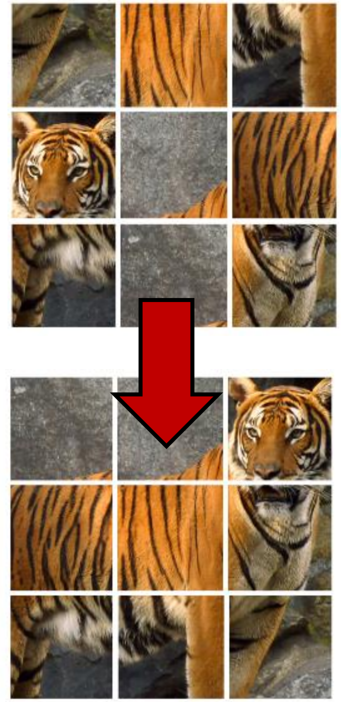
Pre-text Tasks

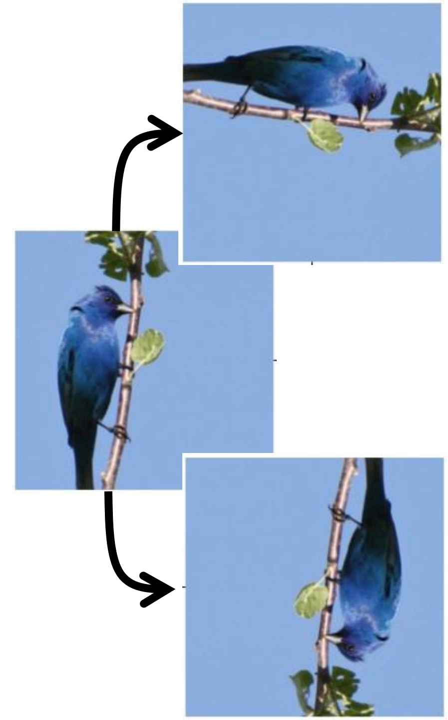


- Pre-train networks with other task, where it's easy to generate data
  → **Self-supervised learning**
- **Idea:** Learn good representation of data (say features) that can be exploited

Common Pre-text Tasks





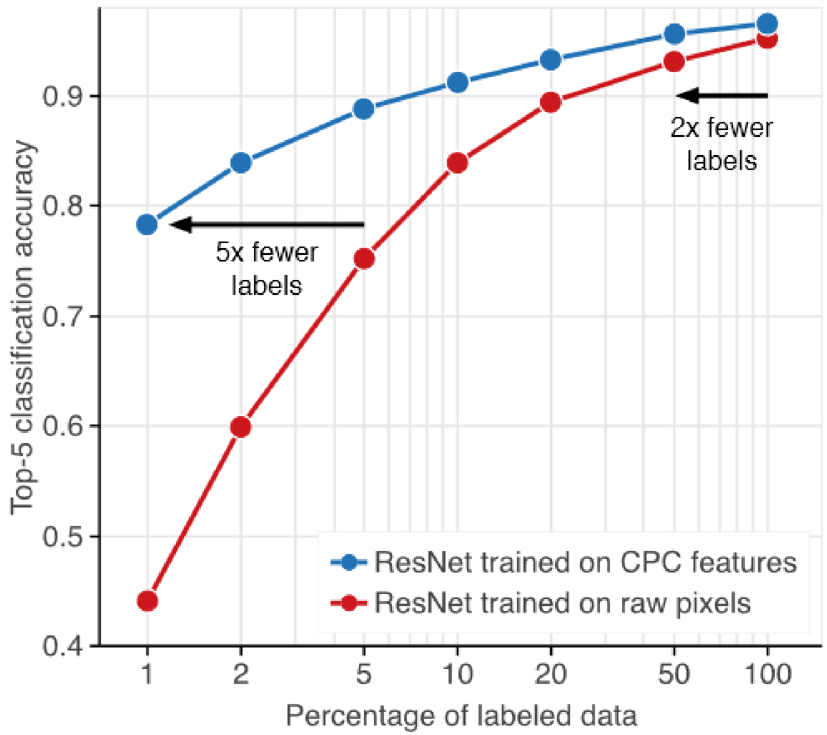


- Predict relative position of image patches
- Predict ordering of Jigsaw
- Predict rotation of images
- Common: Features must capture visual information

[Doersch, 2015][Noroozi, 2015][Gidaris, 2018]

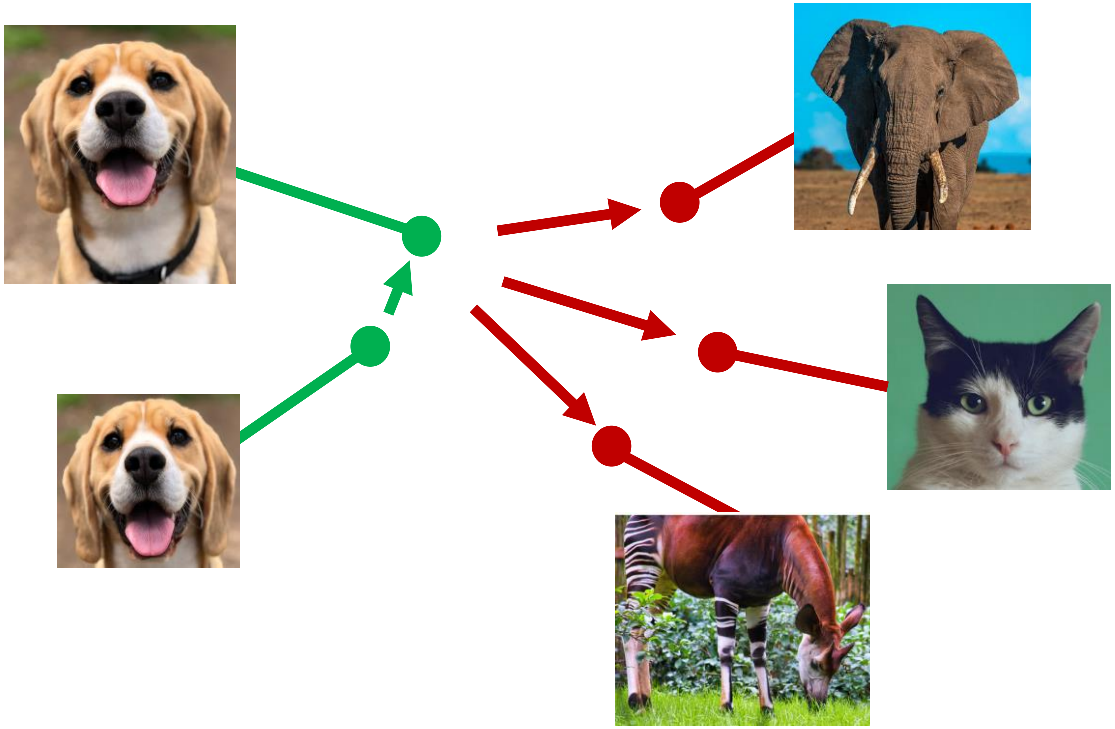
Figure from papers.

Prospect of Self-Supervision



- Pre-text tasks or self-supervised pre-training leads to more data-efficient learning
- Learn more generalizable models with fewer labels

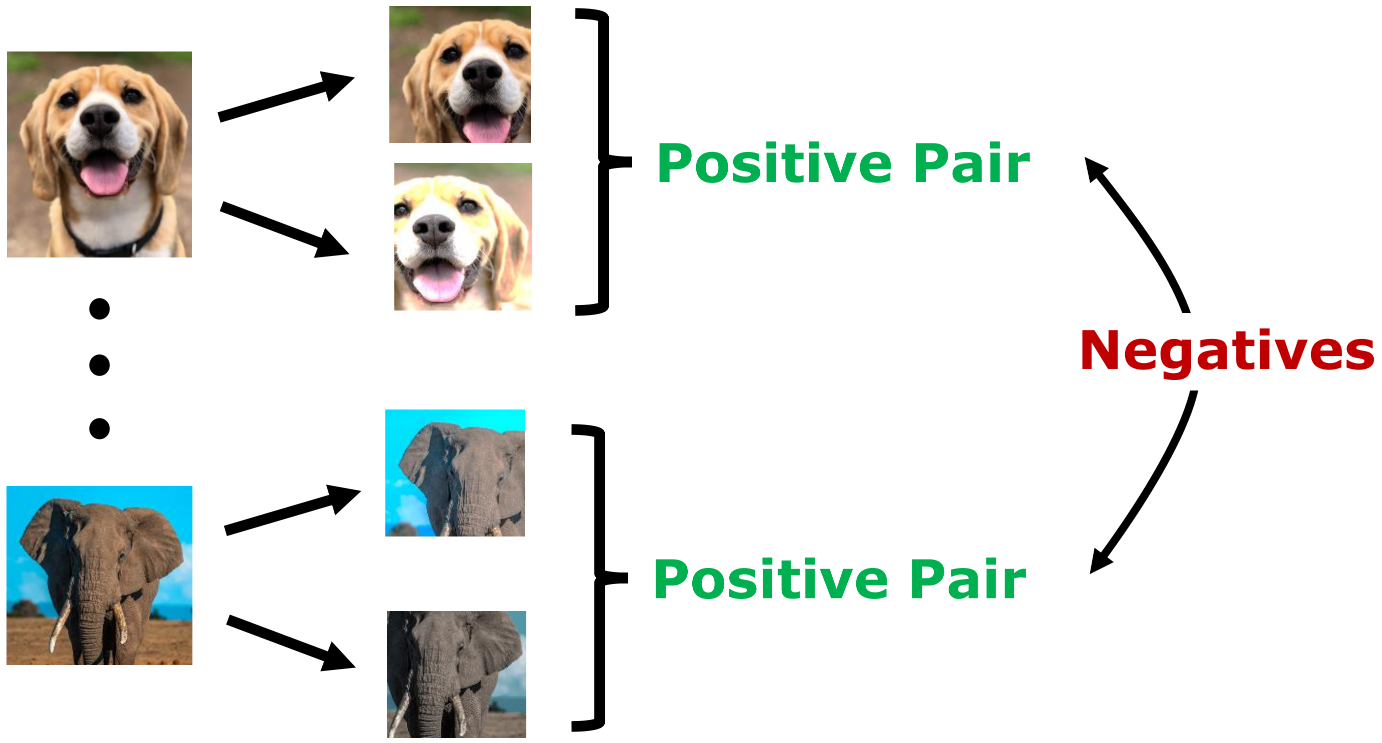
Contrastive Learning



- **Idea:** Learn representations such that similar examples (positives) are closer than representations of different examples (negatives)

Photos from unsplash.com

How to get examples?



- Common way to get positive and negatives is to use random augmentations (e.g., crop, color distortion, etc.)
- Other augmented pairs are negatives

Contrastive Loss

- Given a set of N representations $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}, \mathbf{x}_i \in \mathbb{R}^D$
- Let i_+ be the positive example of the i-th representation.

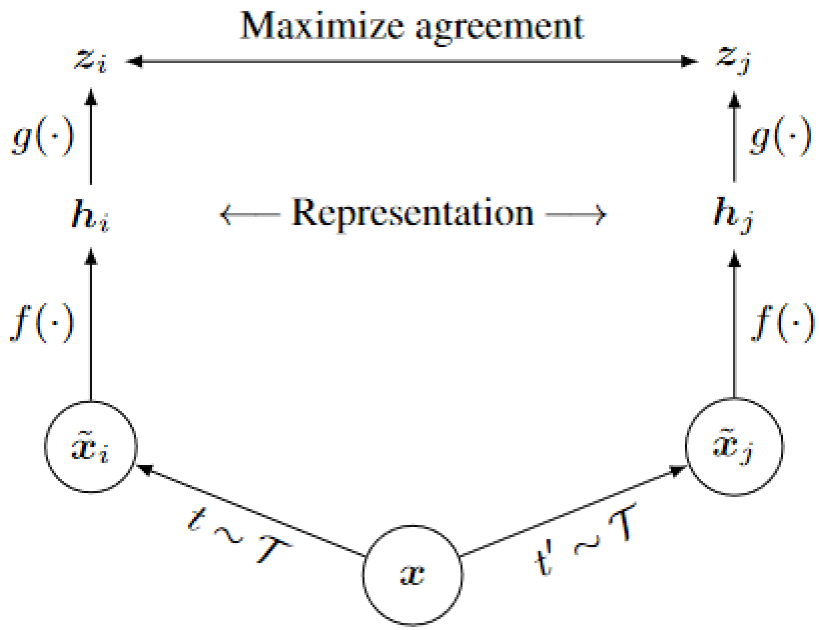
- The (temperature-scaled) **contrastive loss** for the i-th example:

$$\ell_i = -\log \frac{\exp(\mathrm{sim}(\mathbf{z}_i, \mathbf{z}_{i_+})/\tau)}{\sum_{k \neq i} \exp(\mathrm{sim}(\mathbf{z}_i, \mathbf{z}_k)/\tau)}$$

  where τ is a hyperparameter called temperature.

- Commonly: $\mathrm{sim}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u}^\top \mathbf{v}}{\|\mathbf{u}\| \|\mathbf{v}\|}$ (cosine similarity)

SimCLR



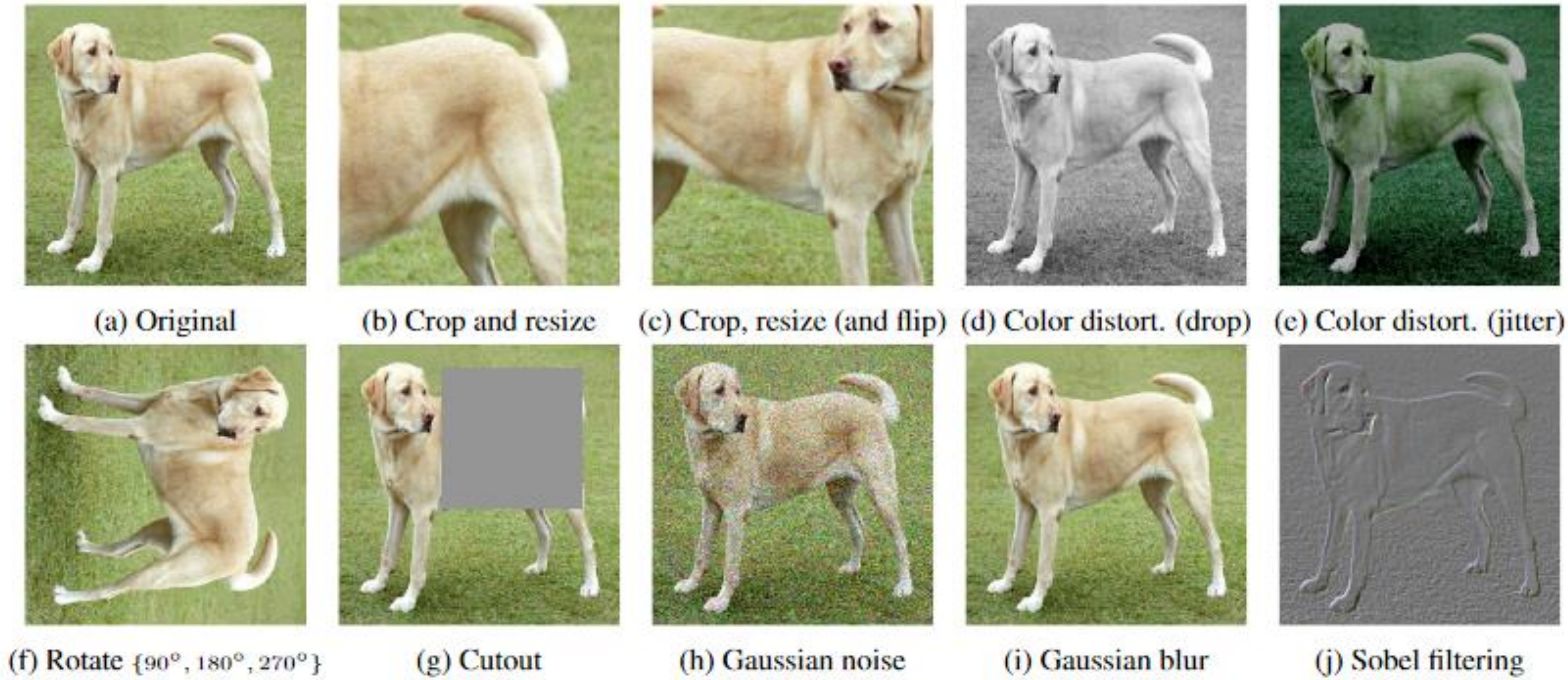
Projection

Augmentation

- **Idea:** Learn representations by finding agreement between *projected* features
- Compute contrastive loss over projections/latents z
- Projection $g(\cdot)$ via FC → ReLU → FC

[Chen, 2020]

Augmentations

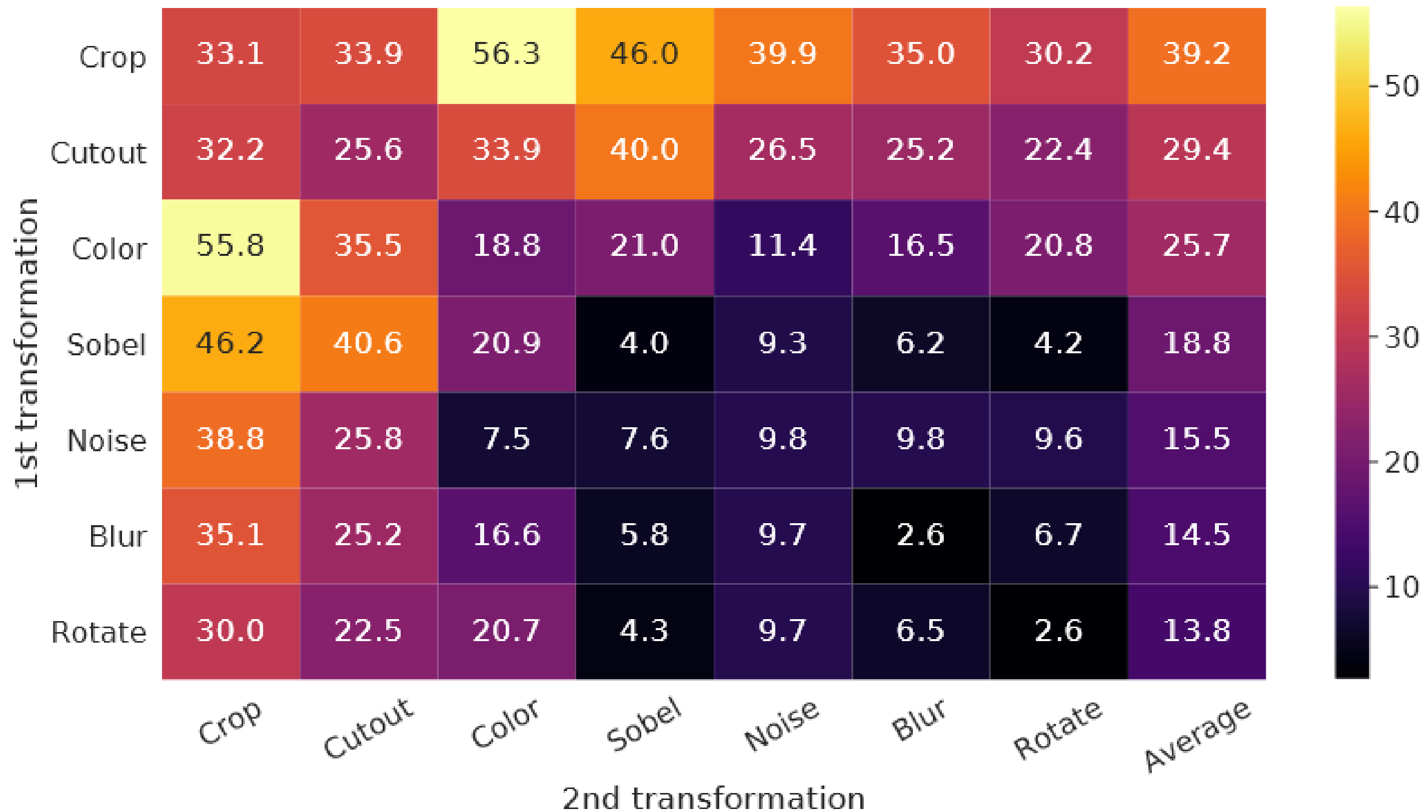


- Various simple image augmentations investigated
- Combinations of multiple augmentations key for good performance

[Chen, 2020]

Figure from [Chen, 2020]

Augmentations matter

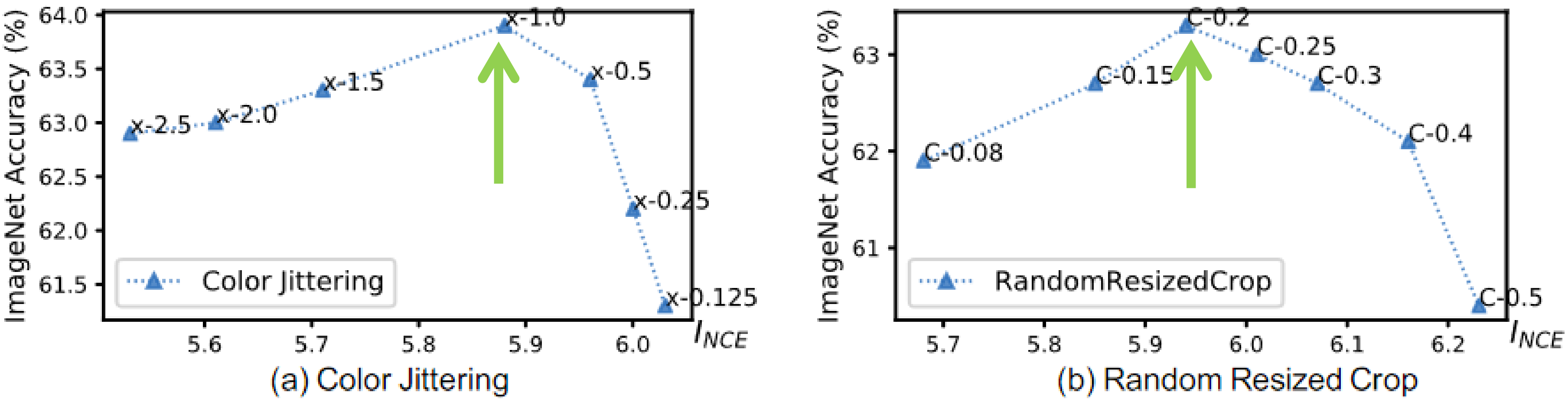


- Augmentations should be large enough to learn good similarity (corresponding to class similarity)
- Crop and color augmentation most important

[Chen, 2020]

Figure from [Chen, 2020]

How much augmentation?

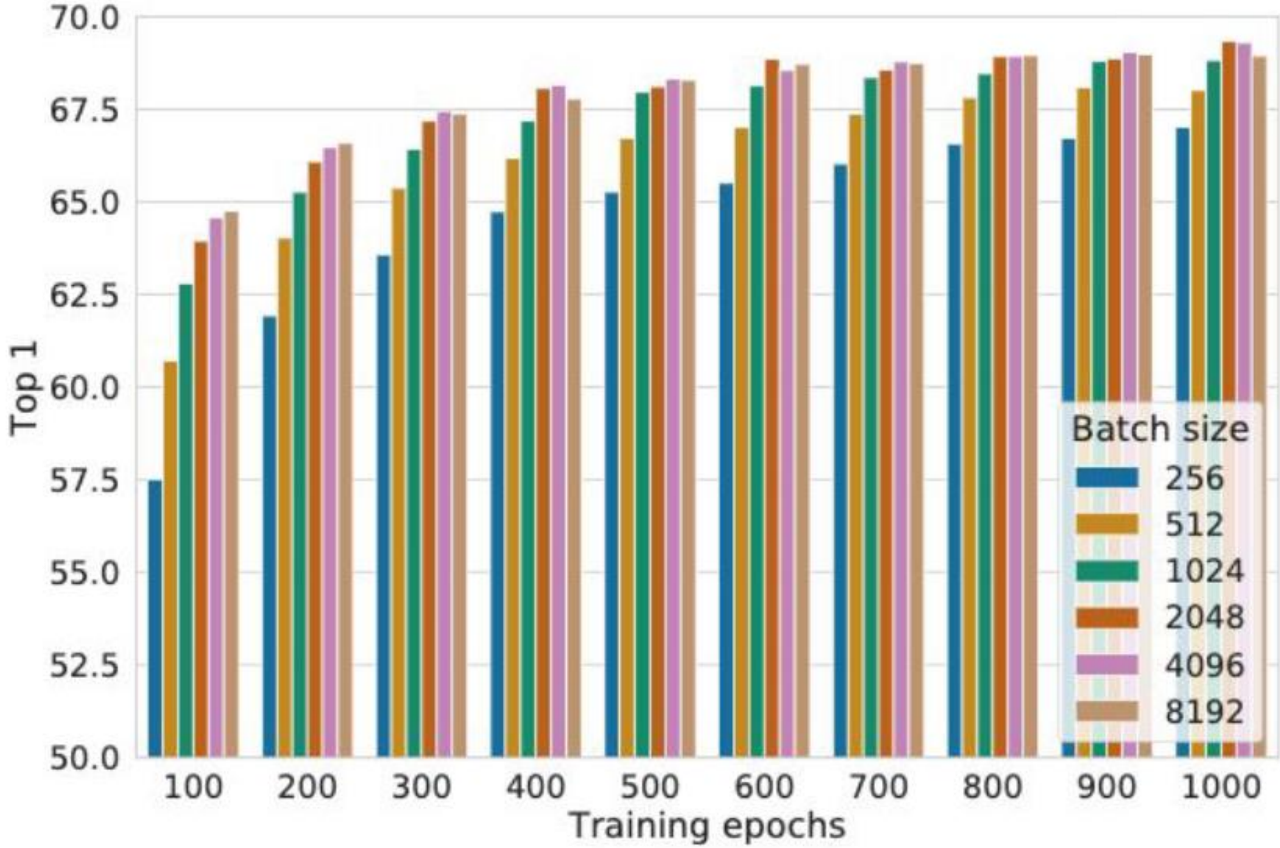


- Right amount of data augmentation crucial for downstream-task
- Too much augmentation removes task-relevant information, too less augmentation keeps irrelevant information

[Tian, 2020]

Figure from [Tiran, 2020]

Batch Size = Negative Examples

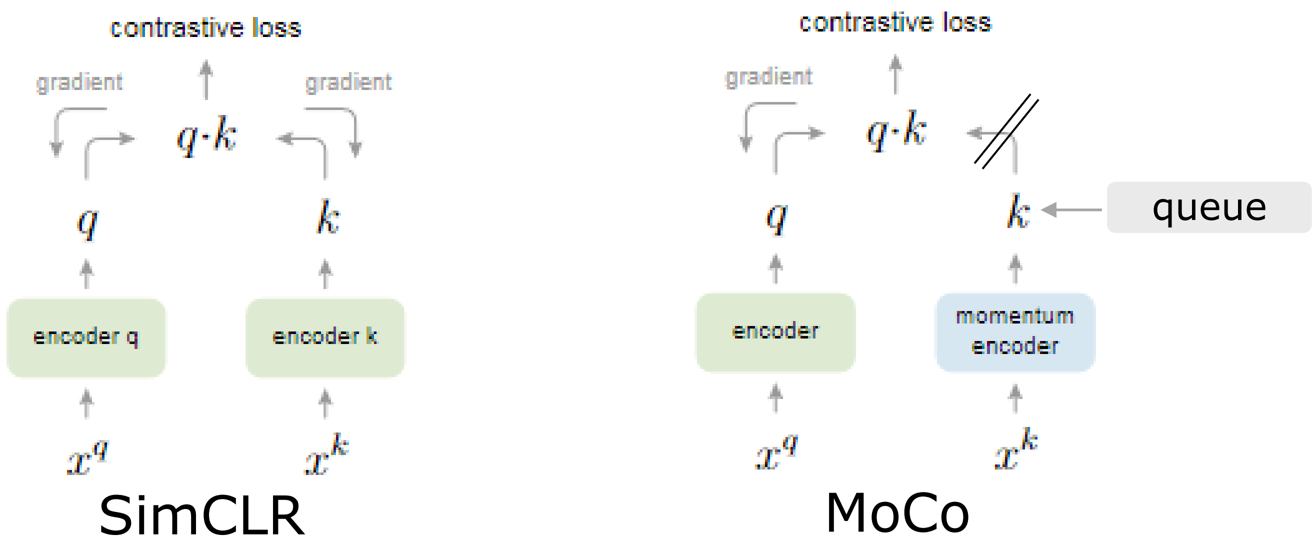


- SimCLR benefits from large batch sizes (e.g., N=4096) and long training (T=1000)

[Chen, 2020]

Figure from [Chen, 2020]

Momentum Contrast (MoCo)

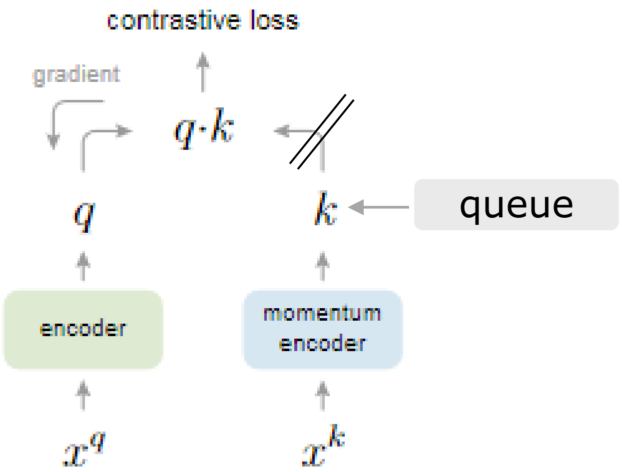


- Large batch sizes might be a problem
- Momentum Contrast (MoCo) solves this by separate encoder and momentum encoder
- Only encoder part is updated via backpropagation!
- Queue of negative examples that can be larger than batch size

[He, 2020]

Figure from [He, 2020]

Momentum Encoder



MoCo

- Only updated with weighted average between parameters of encoder θ_q and parameters of momentum encoder θ_k :

$$\theta_k \leftarrow m\theta_k + (1-m)\theta_q$$

- Typically, large values (e.g., m = 0.999) better then smaller values (e.g., m = 0.9)

[He, 2020]

Figure from [He, 2020]

MoCo V2

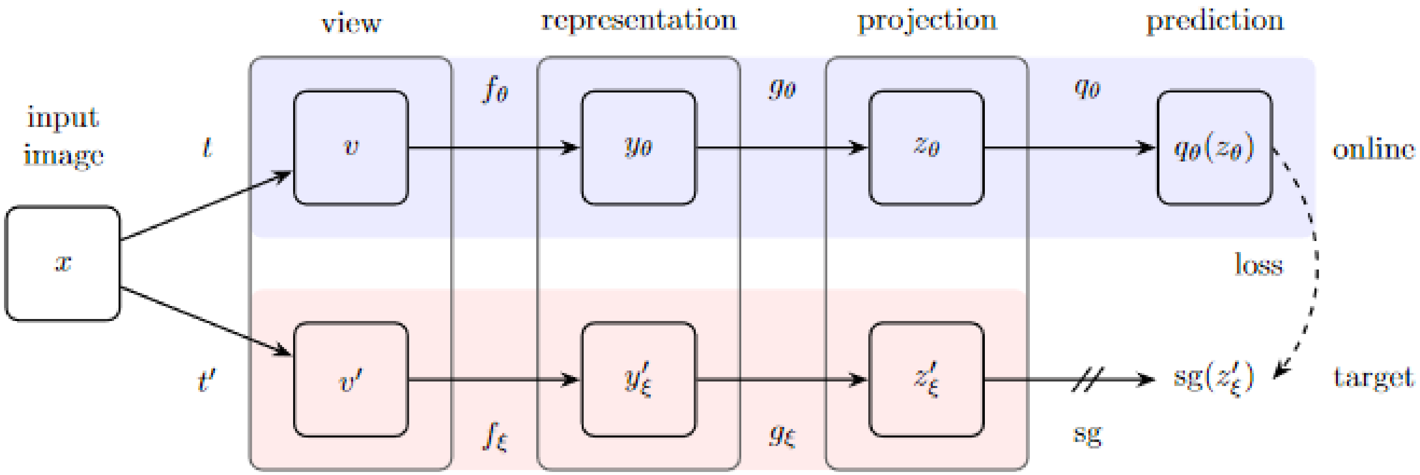
case	MLP	aug+	cos	epochs	batch	ImageNet acc.
MoCo v1 [6]				200	256	60.6
SimCLR [2]	✓	✓	✓	200	256	61.9
SimCLR [2]	✓	✓	✓	200	8192	66.6
MoCo v2	✓	✓	✓	200	256	**67.5**
*results of **longer** unsupervised training follow:*						
SimCLR [2]	✓	✓	✓	1000	4096	69.3
MoCo v2	✓	✓	✓	800	256	**71.1**

- Improvements inspired by of SimCLR:
  1. Use projection head (FC->ReLU->FC)
  2. Stronger data augmentation
  3. Hyperparameter search for temperature

[He, 2020]

Figure from [He, 2020]

Boostrap your own latent (BYOL)



- Augmented views are passed through online and target network
- Online network predicts output of the target network

- Important: There are no negative examples involved!

[Grill, 2020]

Figure from [Grill, 2020]

BYOL training and update

- Loss measures difference between prediction $q(z_\theta)$ and output of target network z'_ξ:

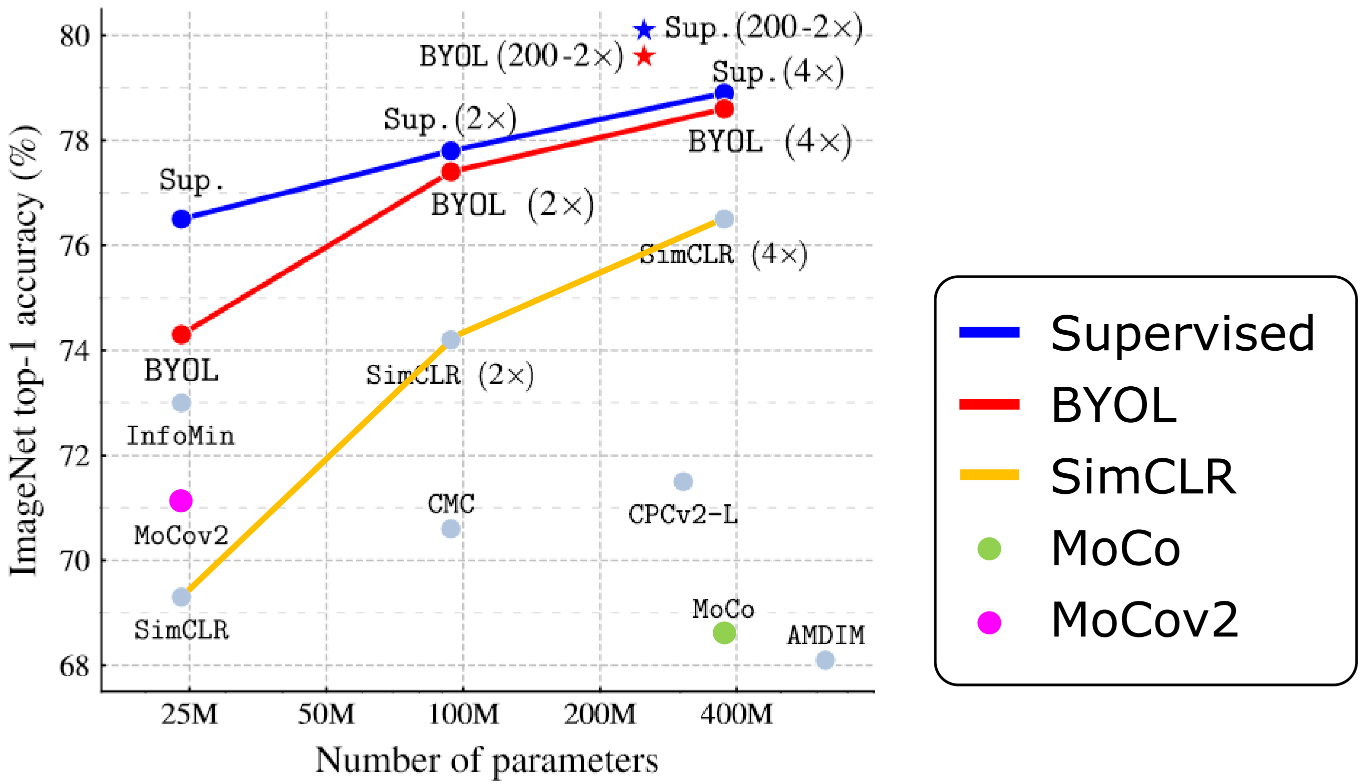
$$\ell = \left\| \frac{q(z_\theta)}{\|q(z_\theta)\|_2} - \frac{z'_\xi}{\|z'_\xi\|_2} \right\|_2^2 = 2 - 2 \cdot \frac{q(z_\theta)^\top z'_\xi}{\|q(z_\theta)\|_2 \|z'_\xi\|_2}$$

- Only online network is directly updated via backpropagation
- Target network parameters ξ are updated via momentum:

$$\xi \leftarrow m\xi + (1 - m)\theta$$

[Grill, 2020]

Comparison on ImageNet



- Results for ResNet50 with different widths (=number of channels), e.g., 2x, 4x
- BYOL approaches supervised training

[Grill, 2020]

Figure from [Grill, 2020]

Transfer learning

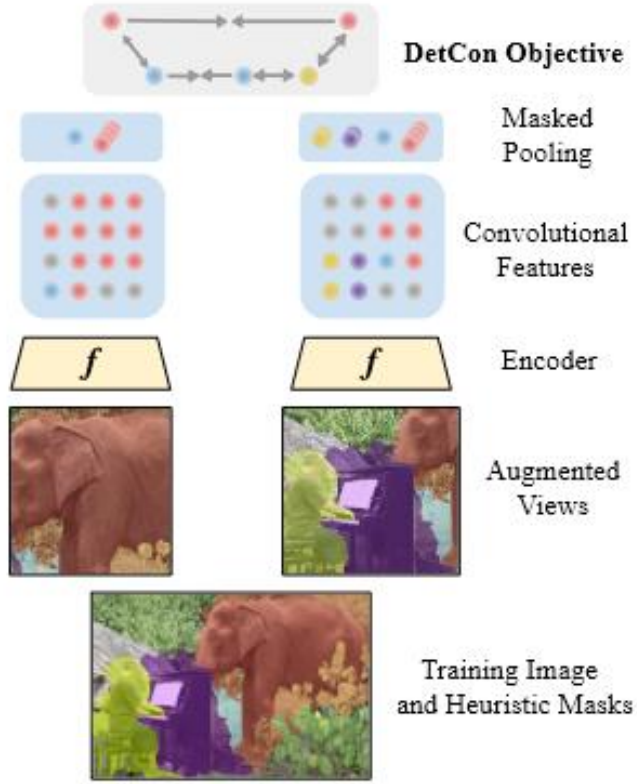
Method	Food101	CIFAR10	CIFAR100	Birdsnap	SUN397	Cars	Aircraft	VOC2007	DTD	Pets	Caltech-101	Flowers
Linear evaluation:												
BYOL (ours)	**75.3**	91.3	**78.4**	**57.2**	**62.2**	**67.8**	60.6	82.5	75.5	90.4	94.2	**96.1**
SimCLR (repro)	72.8	90.5	74.4	42.4	60.6	49.3	49.8	81.4	**75.7**	84.6	89.3	92.6
SimCLR [8]	68.4	90.6	71.6	37.4	58.8	50.3	50.3	80.5	74.5	83.6	90.3	91.2
Supervised-IN [8]	72.3	**93.6**	78.3	53.7	61.9	66.7	**61.0**	**82.8**	74.9	**91.5**	**94.5**	94.7
Fine-tuned:												
BYOL (ours)	**88.5**	**97.8**	86.1	**76.3**	63.7	91.6	**88.1**	**85.4**	**76.2**	91.7	**93.8**	97.0
SimCLR (repro)	87.5	97.4	85.3	75.0	63.9	91.4	87.6	84.5	75.4	89.4	91.7	96.6
SimCLR [8]	88.2	97.7	85.9	75.9	63.5	91.3	88.1	84.1	73.2	89.2	92.1	97.0
Supervised-IN [8]	88.3	97.5	**86.4**	75.8	**64.3**	**92.1**	86.0	85.0	74.6	**92.1**	93.3	**97.6**
Random init [8]	86.9	95.9	80.2	76.1	53.6	91.4	85.9	67.3	64.8	81.5	72.6	92.0

- BYOL provides also strong results on different other datasets
- 7/12 datasets better then supervised pre-training on ImageNet

[Grill, 2020]

Table from [Grill, 2020]

DetCon

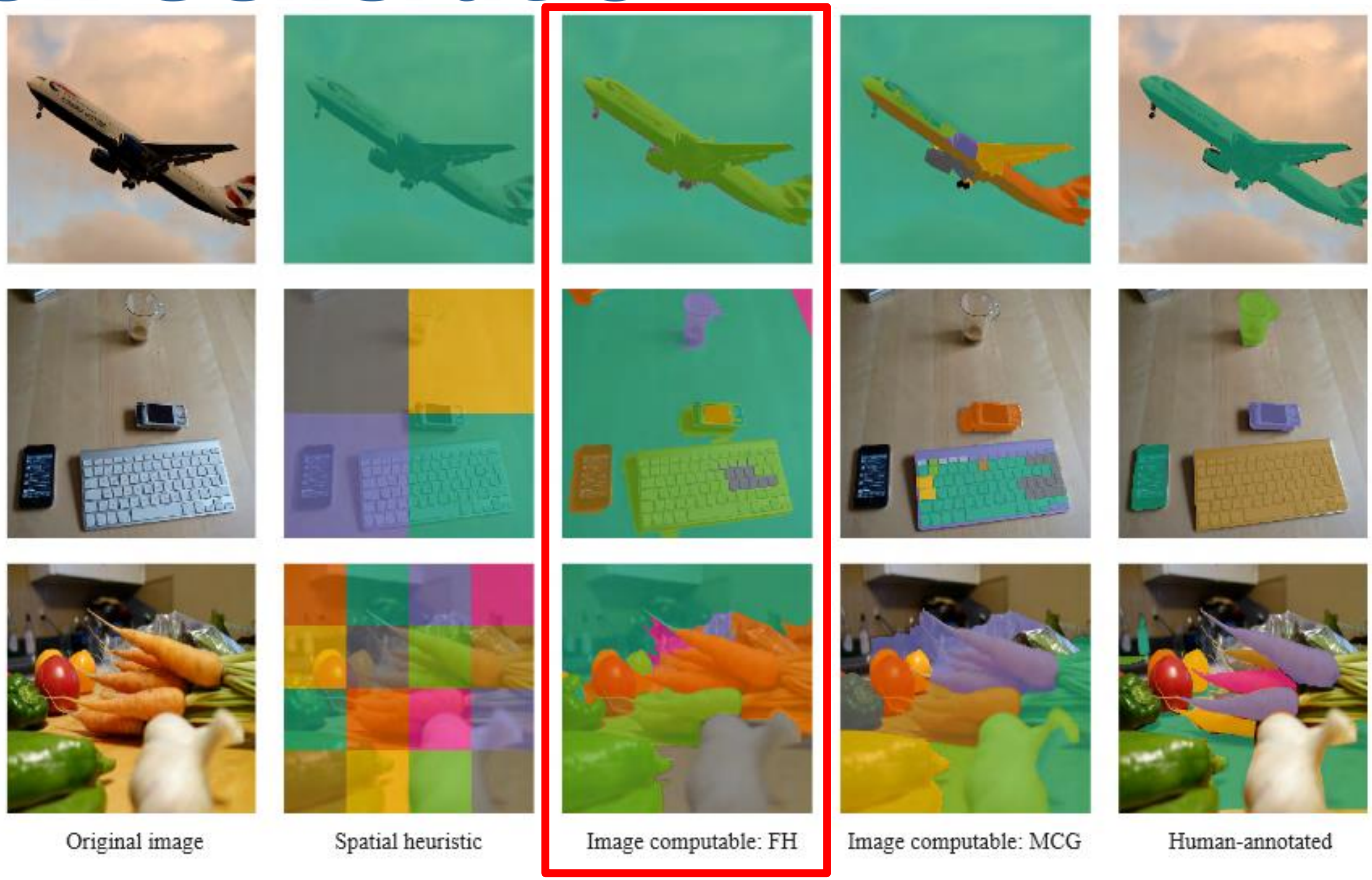


- Contrastive learning targeted specifically at other vision tasks (detection & segmentation)
- **Idea:** By using generated segmentation masks learn object-level features
- Pooled features of same masks are positives, other regions are negatives

[Henaff, 2021]

Figure from [Henaff, 2021]

Mask Generation

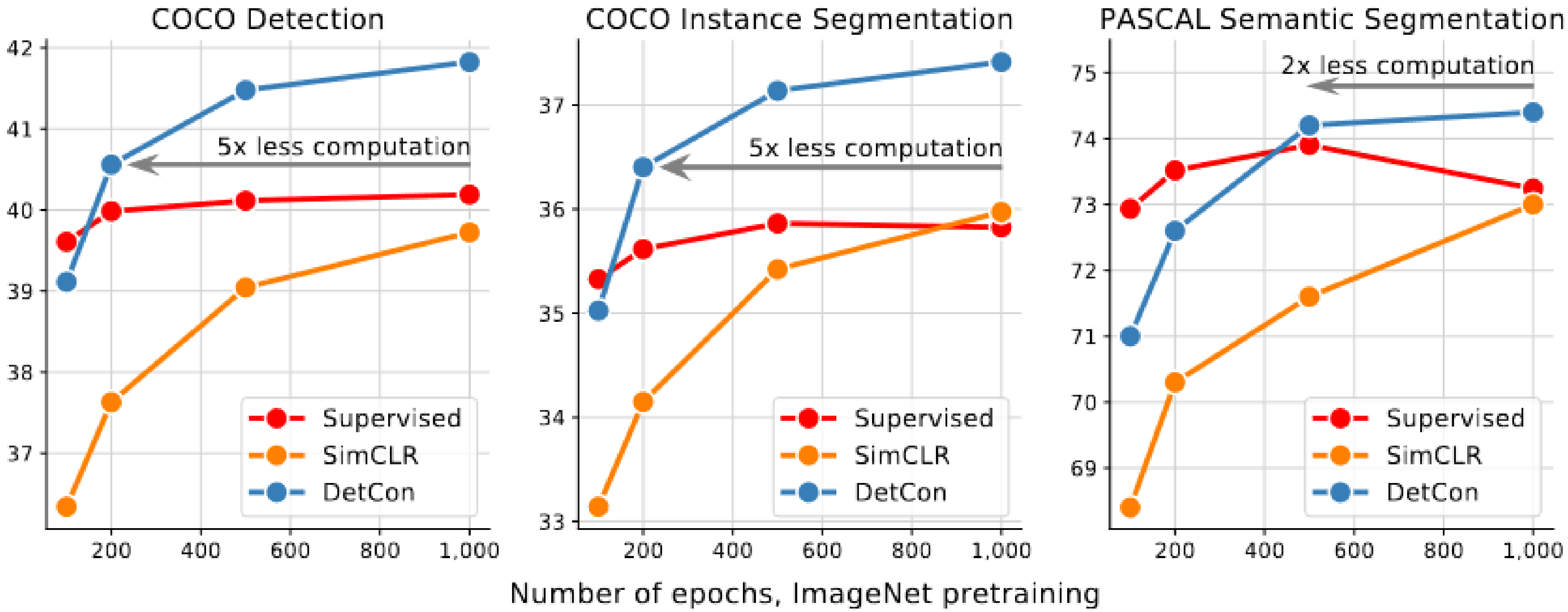


- Different variants investigated
- Off-the-shelf super-pixel segmentation results in good trade-off between compute and quality

[Henaff, 2021]

Figure from [Henaff, 2021]

Results of DetCon

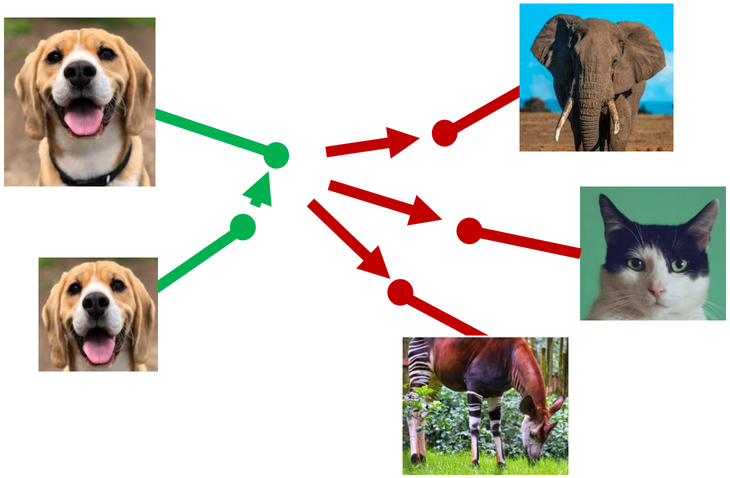


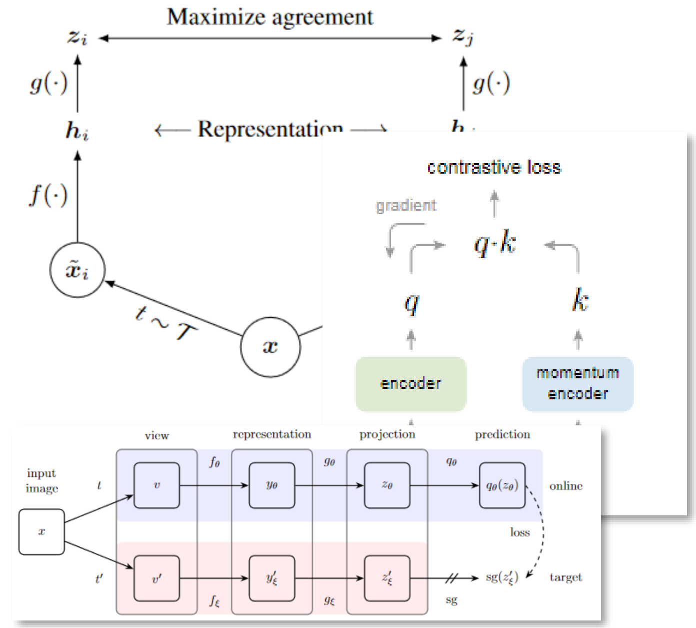
- Surpasses supervised ImageNet pre-training for detection, instance segmentation and semantic segmentation!

[Henaff, 2021]

Figure from [Henaff, 2021]

Summary





- Purely supervised training does not scale
- Using pre-trained models allows to get away with less labels!
- Self-supervised pretraining shows strong performance without any labels!

See you next week!

References

- Doersch et al. Unsupervised Visual Representation Learning by Context Prediction, ICCV, 2015.
- Gidaris et al. Unsupervised Representation Learning by Predicting Image Rotations, ICLR, 2018.
- Grill et al. Bootstrap Your Own LatentA New Approach to Self-Supervised Learning, NeurIPS, 2020.
- He et al. Momentum Contrast for Unsupervised Visual Representation Learning, CVPR, 2020.
- He et al. Rethinking ImageNet Pre-training. ICCV, 2019.
- He et al. Improved Baselines with Momentum Contrastive Learning, arxiv, 2020.
- Henaff et al. Data-Efficient Image Recognition with Contrastive Predictive Coding, ICML, 2020.
- Henaff et al. Efficient Visual Pretraining with Contrastive Detection, arxiv, 2021.
- Kornblith et al. Do Better ImageNet Models Transfer Better?, CVPR, 2019.
- Noroozi et al. Unsupervised Learning of Visual Representation by Solving Jigsaw Puzzles, ECCV, 2016.
- Razavian et al. CNN Features off-the-shelf: an Astonishing Baseline for Recognition, CVPR, 2014.
- Tian et al. What Makes for Good Views for Contrastive Learning? NeurIPS, 2020.
- Van den Oord et al., Representation Learning with Contrastive Predictive Coding, 2018.