Recent Progress on Self-Supervised Visual Representation Learning

Han Hu (胡瀚)
Visual Computing Group
Microsoft Research Asia (MSRA)
November 27th, 2020
A Story about **Cake** (in Yann LeCun’s Turing Talk)

- **“Pure” Reinforcement Learning (cherry)**
  - The machine predicts a scalar reward given once in a while.
- **A few bits for some samples**
- **Supervised Learning (icing)**
  - The machine predicts a category or a few numbers for each input
  - Predicting human-supplied data
  - 10→10,000 bits per sample
- **Self-Supervised Learning (cake génoise)**
  - The machine predicts any part of its input for any observed part.
  - Predicts future frames in videos
  - **Millions of bits per sample**

Credit by Yann LeCun
Why Self-Supervised Learning?

- Baby learns how to world works largely by observation

Photos courtesy of Emmanuel Dupoux

Credit by Yann LeCun

Linda Smith, Michael Gasser. The Development of Embodied Cognition: Six Lessons from Babies, 2005
A Story about ImageNet

- AlexNet (NIPS'2012)
A Story about ImageNet
Supervised Pretraining + Finetuning (2014)

• A kind of transfer learning paradigm

Pretraining on ImageNet Classification

Finetuning

Semantic Segmentation

Object Detection

Fine-grained Classification
Two Stories Meet Each Other

• **Unsupervised** Pretraining + Finetuning

**Momentum Contrast for Unsupervised Visual Representation Learning**

Kaiming He  Haoqi Fan  Yuxin Wu  Saining Xie  Ross Girshick

Facebook AI Research (FAIR)
Code: [https://github.com/facebookresearch/moco](https://github.com/facebookresearch/moco)

- For the first time, unsupervised pretraining outperform supervised pretraining on 7 down-stream tasks
The Self-Supervised Learning Era!

- Can utilize unlimited data
- Similar way as that of human baby learning
How Did We Get Here?

Credit mostly by Andrew Zisserman

**Deep Clustering**
- Dosovitskiy et al., NIPS 2014.
- Caron et al, ECCV’2018

**Rotation Prediction**
- Gidaris et al, ICLR’2018

**Image GPT**
- Agrawal et al. ICCV 2015
- Jayaraman et al. ICCV 2015

**Co-Occurrence**
- Isola et al. ICLR Workshop 2016.

**Egomotion**

**Context**
- Noroozi et al 2016
- Pathak et al. CVPR 2016

**Split-brain auto-encoders**
- Zhang et al. CVPR 2017
- Chen et al, ICML’2020
How Did We Get Here?

2014.6
- Exemplar
  - Dosovitskiy et al, NIPS’2014

2018.5
- Memory bank
  - Wu et al, CVPR’2018

2018.12
- Deep metric transfer
  - MSRA

2019.11
- MoCo
- FAIR

• For the first time, unsupervised pretraining outperform supervised pretraining on 7 down-stream tasks

Pre-text task: Image discrimination
Contrastive Learning for Instance Discrimination

Pre-text task: Image discrimination

Input image
	aug. views

features

push

classification

pull
MoCo (CVPR’2020)

- Large dictionary
- Consistent dictionary by momentum encoder

Credit by Kaiming He
After MoCo

- **Large** dictionary
- **Consistent** dictionary by momentum encoder

**MoCo**

Credit by Kaiming He
MoCo Results

- Outperforms supervised methods on 7 down-stream tasks for the first time

<table>
<thead>
<tr>
<th>pre-train</th>
<th>AP&lt;sub&gt;50&lt;/sub&gt;</th>
<th>AP</th>
<th>AP&lt;sub&gt;75&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>random init.</td>
<td>60.2</td>
<td>33.8</td>
<td>33.1</td>
</tr>
<tr>
<td>super. IN-1M</td>
<td>81.3</td>
<td>53.5</td>
<td>58.8</td>
</tr>
<tr>
<td>MoCo IN-1M</td>
<td>81.5 (+0.2)</td>
<td>55.9 (+2.4)</td>
<td>62.6 (+3.8)</td>
</tr>
<tr>
<td>MoCo IG-1B</td>
<td>82.2 (+0.9)</td>
<td>57.2 (+3.7)</td>
<td>63.7 (+4.9)</td>
</tr>
</tbody>
</table>

(b) Faster R-CNN, R50-C4

Table 2. Object detection fine-tuned on PASCAL VOC

<table>
<thead>
<tr>
<th>pre-train</th>
<th>AP&lt;sup&gt;b&lt;/sup&gt;&lt;sub&gt;50&lt;/sub&gt;</th>
<th>AP&lt;sup&gt;b&lt;/sup&gt;&lt;sub&gt;75&lt;/sub&gt;</th>
<th>AP&lt;sup&gt;b&lt;/sup&gt;&lt;sub&gt;50&lt;/sub&gt;</th>
<th>AP&lt;sup&gt;b&lt;/sup&gt;&lt;sub&gt;75&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>random init.</td>
<td>31.0</td>
<td>49.5</td>
<td>33.2</td>
<td>30.4</td>
</tr>
<tr>
<td>super. IN-1M</td>
<td>38.9</td>
<td>59.6</td>
<td>42.7</td>
<td>38.1</td>
</tr>
<tr>
<td>MoCo IN-1M</td>
<td>38.5 (−0.4)</td>
<td>58.9 (−0.7)</td>
<td>42.0 (−0.7)</td>
<td>37.7 (−0.4)</td>
</tr>
<tr>
<td>MoCo IG-1B</td>
<td>38.9 (0.0)</td>
<td>59.4 (−0.2)</td>
<td>42.3 (−0.4)</td>
<td>37.9 (−0.2)</td>
</tr>
</tbody>
</table>

(a) Mask R-CNN, R50-FPN, 1× schedule

Table 5. Object detection and instance segmentation fine-tuned on COCO

<table>
<thead>
<tr>
<th>pre-train</th>
<th>AP&lt;sup&gt;mk&lt;/sup&gt;&lt;sub&gt;50&lt;/sub&gt;</th>
<th>AP&lt;sup&gt;mk&lt;/sup&gt;&lt;sub&gt;75&lt;/sub&gt;</th>
<th>AP&lt;sup&gt;mk&lt;/sup&gt;&lt;sub&gt;50&lt;/sub&gt;</th>
<th>AP&lt;sup&gt;mk&lt;/sup&gt;&lt;sub&gt;75&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>random init.</td>
<td>22.5</td>
<td>34.8</td>
<td>23.8</td>
<td>25.8</td>
</tr>
<tr>
<td>super. IN-1M</td>
<td>24.4</td>
<td>37.8</td>
<td>25.8</td>
<td>25.8</td>
</tr>
<tr>
<td>MoCo IN-1M</td>
<td>24.1 (−0.3)</td>
<td>37.4 (−0.4)</td>
<td>25.5 (−0.3)</td>
<td>25.5 (−0.3)</td>
</tr>
<tr>
<td>MoCo IG-1B</td>
<td>24.9 (+0.5)</td>
<td>38.2 (+0.4)</td>
<td>26.4 (+0.6)</td>
<td>26.4 (+0.6)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>pre-train</th>
<th>Cityscapes instance seg.</th>
<th>Semantic seg. (mIoU)</th>
</tr>
</thead>
<tbody>
<tr>
<td>random init.</td>
<td>25.4</td>
<td>51.1</td>
</tr>
<tr>
<td>super. IN-1M</td>
<td>32.9</td>
<td>59.6</td>
</tr>
<tr>
<td>MoCo IN-1M</td>
<td>32.3 (−0.6)</td>
<td>59.3 (−0.3)</td>
</tr>
<tr>
<td>MoCo IG-1B</td>
<td>32.9 (0.0)</td>
<td>60.3 (+0.7)</td>
</tr>
</tbody>
</table>

Random init. COCO keypoint detection

<table>
<thead>
<tr>
<th>pre-train</th>
<th>AP&lt;sup&gt;kp&lt;/sup&gt;&lt;sub&gt;50&lt;/sub&gt;</th>
<th>AP&lt;sup&gt;kp&lt;/sup&gt;&lt;sub&gt;75&lt;/sub&gt;</th>
<th>AP&lt;sup&gt;kp&lt;/sup&gt;&lt;sub&gt;75&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>random init.</td>
<td>65.9</td>
<td>86.5</td>
<td>71.7</td>
</tr>
<tr>
<td>super. IN-1M</td>
<td>65.8</td>
<td>86.9</td>
<td>71.9</td>
</tr>
<tr>
<td>MoCo IN-1M</td>
<td>66.8 (+1.0)</td>
<td>87.4 (+0.5)</td>
<td>72.5 (+0.6)</td>
</tr>
<tr>
<td>MoCo IG-1B</td>
<td>66.9 (+1.1)</td>
<td>87.8 (+0.9)</td>
<td>73.0 (+1.1)</td>
</tr>
</tbody>
</table>

COCO dense pose estimation

<table>
<thead>
<tr>
<th>pre-train</th>
<th>AP&lt;sup&gt;dp&lt;/sup&gt;&lt;sub&gt;50&lt;/sub&gt;</th>
<th>AP&lt;sup&gt;dp&lt;/sup&gt;&lt;sub&gt;75&lt;/sub&gt;</th>
<th>AP&lt;sup&gt;dp&lt;/sup&gt;&lt;sub&gt;75&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>random init.</td>
<td>39.4</td>
<td>78.5</td>
<td>35.1</td>
</tr>
<tr>
<td>super. IN-1M</td>
<td>48.3</td>
<td>85.6</td>
<td>50.6</td>
</tr>
<tr>
<td>MoCo IN-1M</td>
<td>50.1 (+1.8)</td>
<td>86.8 (+1.2)</td>
<td>53.9 (+3.3)</td>
</tr>
<tr>
<td>MoCo IG-1B</td>
<td>50.6 (+2.3)</td>
<td>87.0 (+1.4)</td>
<td>54.3 (+3.7)</td>
</tr>
</tbody>
</table>

LVIS v0.5 instance segmentation

<table>
<thead>
<tr>
<th>pre-train</th>
<th>AP&lt;sup&gt;mk&lt;/sup&gt;&lt;sub&gt;50&lt;/sub&gt;</th>
<th>AP&lt;sup&gt;mk&lt;/sup&gt;&lt;sub&gt;75&lt;/sub&gt;</th>
<th>AP&lt;sup&gt;mk&lt;/sup&gt;&lt;sub&gt;75&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>random init.</td>
<td>22.5</td>
<td>34.8</td>
<td>23.8</td>
</tr>
<tr>
<td>super. IN-1M</td>
<td>24.4</td>
<td>37.8</td>
<td>25.8</td>
</tr>
<tr>
<td>MoCo IN-1M</td>
<td>24.1 (−0.3)</td>
<td>37.4 (−0.4)</td>
<td>25.5 (−0.3)</td>
</tr>
<tr>
<td>MoCo IG-1B</td>
<td>24.9 (+0.5)</td>
<td>38.2 (+0.4)</td>
<td>26.4 (+0.6)</td>
</tr>
</tbody>
</table>

Cityscapes instance seg. | Semantic seg. (mIoU)
MoCo Results

• ImageNet-1K linear evaluation
After MoCo

- SimCLR (ICML’2020)
- NeurIPS’2020 papers
- After NeurIPS’2020
SimCLR (ICML’2020)

- **Simpler**: no momentum, no memory (dictionary)
- **Sufficient distance** between pretext tasks and downstream tasks
  - a linear projection layer \(\rightarrow\) a MLP layer
- **Self-supervised learning** benefit significantly from **longer training**
- Carefully tuning **data augmentation methods**

**ImageNet linear evaluation**

<table>
<thead>
<tr>
<th>Model</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>MoCo</td>
<td>60.6</td>
</tr>
<tr>
<td>SimCLR</td>
<td>69.1</td>
</tr>
<tr>
<td>MoCo v2</td>
<td>71.1</td>
</tr>
</tbody>
</table>

+8.5  +2.0
More Insights in SimCLR

• Self-supervised learning benefit more from larger models

• Self-supervised learning benefit significantly for semi-supervised learning

Figure 7. Linear evaluation of models with varied depth and width. Models in blue dots are ours trained for 100 epochs, models in red stars are ours trained for 1000 epochs, and models in green crosses are supervised ResNets trained for 90 epochs\(^7\) (He et al., 2016).

Table 7. ImageNet accuracy of models trained with few labels.

<table>
<thead>
<tr>
<th>Method</th>
<th>Architecture</th>
<th>Label fraction</th>
<th>Top 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supervised baseline</td>
<td>ResNet-50</td>
<td>48.4</td>
<td>80.4</td>
</tr>
<tr>
<td>Methods using other label-propagation:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pseudo-label</td>
<td>ResNet-50</td>
<td>51.6</td>
<td>82.4</td>
</tr>
<tr>
<td>VAT+Entropy Min.</td>
<td>ResNet-50</td>
<td>47.0</td>
<td>83.4</td>
</tr>
<tr>
<td>UDA (w. RandAug)</td>
<td>ResNet-50</td>
<td>-</td>
<td>88.5</td>
</tr>
<tr>
<td>FixMatch (w. RandAug)</td>
<td>ResNet-50</td>
<td>-</td>
<td>89.1</td>
</tr>
<tr>
<td>S4L (Rot+VAT+En. M.)</td>
<td>ResNet-50 (4×)</td>
<td>-</td>
<td>91.2</td>
</tr>
</tbody>
</table>

Methods using representation learning only:

<table>
<thead>
<tr>
<th>Method</th>
<th>Architecture</th>
<th>Label fraction</th>
<th>Top 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>InstDisc</td>
<td>ResNet-50</td>
<td>39.2</td>
<td>77.4</td>
</tr>
<tr>
<td>BigBiGAN</td>
<td>RevNet-50 (4×)</td>
<td>55.2</td>
<td>78.8</td>
</tr>
<tr>
<td>PIRL [Screenshot(Alt + A)]</td>
<td>ResNet-50</td>
<td>57.2</td>
<td>83.8</td>
</tr>
<tr>
<td>CPC v2</td>
<td>ResNet-161(*)</td>
<td>77.9</td>
<td>91.2</td>
</tr>
<tr>
<td>SimCLR (ours)</td>
<td>ResNet-50</td>
<td>75.5</td>
<td>87.8</td>
</tr>
<tr>
<td>SimCLR (ours)</td>
<td>ResNet-50 (2×)</td>
<td>83.0</td>
<td>91.2</td>
</tr>
<tr>
<td>SimCLR (ours)</td>
<td>ResNet-50 (4×)</td>
<td>85.8</td>
<td>92.6</td>
</tr>
</tbody>
</table>

Similar as that of GPT-3 in NLP!
SimCLR v2 (NeurIPS’2020)

• “Big Self-Supervised Models are Strong Semi-Supervised Learners”

Figure 1: Bigger models yield larger gains when fine-tuning with fewer labeled examples.

Figure 2: Top-1 accuracy of previous state-of-the-art (SOTA) methods [1, 2] and our method (SimCLRv2) on ImageNet using only 1% or 10% of the labels. Dashed line denotes fully supervised ResNet-50 trained with 100% of labels. Full comparisons in Table 3.
“Unsupervised” Papers on NeurIPS’2020

• 130 papers by a keyword “unsupervised” (totally about 1,900)

• Representative works
  • BYOL (DeepMind)
  • SwaV (Facebook AI Research)
  • InfoMin (MIT, Google Research)
  • SimCLR v2 (Google Brain)
  • PIC (talk #4 by Zhenda Xie, MSRA)
BYOL

• Bootstrap Your Own Latent

ImageNet linear evaluation

MoCo 60.6 → SimCLR 69.1 → MoCo v2 71.1 → BYOL 74.3

+8.5 → +2.0 → +3.2
A Finding by BYOL

- MoCo: we need larger dictionary size (more negative pairs)
- BYOL: we do not need negative pairs anymore
  - an asymmetric design
SwaV

- Contrasting Cluster Assignments

ImageNet linear evaluation

MoCo: 60.6
SimCLR: 69.1
MoCo v2: 71.1
BYOL: 74.3
SwaV: 75.3
SwaV

- Deep clustering (ECCV’2018) + contrastive learning
- Additional small patches in view generation
InfoMin: What Makes for Good Views for Contrastive Learning?

• Empirical study on augmentation methods
• Extensive/good results on Pascal VOC and COCO detection
  • Previous methods mostly focus on improving ImageNet linear evaluation accuracy

<table>
<thead>
<tr>
<th>pre-train</th>
<th>AP$_{50}$</th>
<th>AP</th>
<th>AP$_{75}$</th>
<th>ImageNet Acc(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>random init*</td>
<td>60.2</td>
<td>33.8</td>
<td>33.1</td>
<td>-</td>
</tr>
<tr>
<td>supervised*</td>
<td>81.3</td>
<td>53.5</td>
<td>58.8</td>
<td>76.1</td>
</tr>
<tr>
<td>InstDis</td>
<td>80.9</td>
<td>55.2</td>
<td>61.2</td>
<td>59.5</td>
</tr>
<tr>
<td>PIRL</td>
<td>81.0</td>
<td>55.5</td>
<td>61.3</td>
<td>61.7</td>
</tr>
<tr>
<td>MoCo*</td>
<td>81.5</td>
<td>55.9</td>
<td>62.6</td>
<td>60.6</td>
</tr>
<tr>
<td>InfoMin Aug. (ours)</td>
<td>82.7</td>
<td>57.6</td>
<td>64.6</td>
<td>70.1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>pre-train</th>
<th>AP$_{bb}$</th>
<th>AP$_{50}$</th>
<th>AP$_{75}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>random init*</td>
<td>26.4</td>
<td>44.0</td>
<td>27.8</td>
</tr>
<tr>
<td>supervised*</td>
<td>38.2</td>
<td>58.2</td>
<td>41.2</td>
</tr>
<tr>
<td>MoCo*</td>
<td>38.5($\uparrow$0.3)</td>
<td>58.3($\uparrow$0.1)</td>
<td>41.6($\uparrow$0.4)</td>
</tr>
<tr>
<td>InfoMin Aug.</td>
<td>39.0($\uparrow$0.8)</td>
<td>58.5($\uparrow$0.3)</td>
<td>42.0($\uparrow$0.8)</td>
</tr>
</tbody>
</table>
PIC: a Single-Branch Method (Talk #4)

Two-branch methods (almost all previous methods)

- Input image
- Aug. views
- Features

One-branch method (PIC)

- Input image
- Aug. views
- Features

Simpler but the same effective!
Representative Works after NeurIPS’2020

• Higher ImageNet-1K linear evaluation accuracy
  • Contrastive learning with stronger augmentations (CLSA)
  • (ICLR’2021 submission) CLSA 76.2 vs supervised 76.5

• Better understanding
  • What makes instance discrimination good for transfer learning?
  • (ICLR’2021 submission) it is mainly the low-level features that affect!

• More study on BYOL why it does not collapse
  • BYOL (Arxiv v3)
  • Exploring Simple Siamese Representation Learning (tech report)

• Pixel-level pretext tasks
  • PixPro, for more spatially fine-grained representation learning
Motivation of PixPro

- Improvements on ImageNet-1K linear evaluation

Totally 15.6% absolute improvements in 1 year!
PixPro

- Improvements on Pascal VOC object detection (C4)

```plaintext
55.9  +0.4  →  56.3  +0.7  →  57.6  +0  →  57.6  +2.6  →  60.2
MoCo  SimCLR  MoCo v2  InfoMin  PixPro
```

Totally 1.1% absolute improvements in 1 year!
PixPro Results

- VOC detection (+2.6 mAP)
- COCO FPN detection (+0.8 mAP) COCO C4 (+1.0 mAP)
- Cityscape segmentation (+1.0 mIoU)

<table>
<thead>
<tr>
<th>Method</th>
<th>#. Epoch</th>
<th>Pascal VOC (R50-C4)</th>
<th>COCO (R50-FPN)</th>
<th>COCO (R50-C4)</th>
<th>Cityscapes (R50)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>AP</td>
<td>AP₅₀</td>
<td>AP₇₅</td>
<td>mAP</td>
</tr>
<tr>
<td>scratch</td>
<td>-</td>
<td>33.8</td>
<td>60.2</td>
<td>33.1</td>
<td>32.8</td>
</tr>
<tr>
<td>supervised</td>
<td>100</td>
<td>53.5</td>
<td>81.3</td>
<td>58.8</td>
<td>39.7</td>
</tr>
<tr>
<td>MoCo [18]</td>
<td>200</td>
<td>55.9</td>
<td>81.5</td>
<td>62.6</td>
<td>39.4</td>
</tr>
<tr>
<td>SimCLR [8]</td>
<td>1000</td>
<td>56.3</td>
<td>81.9</td>
<td>62.5</td>
<td>39.8</td>
</tr>
<tr>
<td>MoCo v2 [9]</td>
<td>800</td>
<td>57.6</td>
<td>82.7</td>
<td>64.4</td>
<td>40.4</td>
</tr>
<tr>
<td>InfoMin [30]</td>
<td>200</td>
<td><strong>57.6</strong></td>
<td>82.7</td>
<td>64.6</td>
<td><strong>40.6</strong></td>
</tr>
<tr>
<td>InfoMin [30]</td>
<td>800</td>
<td>57.5</td>
<td>82.5</td>
<td>64.0</td>
<td>40.4</td>
</tr>
<tr>
<td>PixPro (ours)</td>
<td>100</td>
<td>58.8</td>
<td>83.0</td>
<td>66.5</td>
<td>41.3</td>
</tr>
<tr>
<td>PixPro (ours)</td>
<td>400</td>
<td><strong>60.2</strong></td>
<td><strong>83.8</strong></td>
<td><strong>67.7</strong></td>
<td><strong>41.4</strong></td>
</tr>
</tbody>
</table>
From Instance-Level to Pixel-Level Learning

Previous pre-text tasks: instance discrimination

Memory bank, MoCo, SimCLR, BYOL, SwaV, PIC, ...

pixel-level pretext task
Pixel-Level Contrastive Learning

An image

view #1

view #2

pull

push

pixel discrimination
Pixel-to-Propagation Consistency

An image

Pixel-to-propagation consistency
Pixel-to-Propagation Consistency

- **Pixel contrast**: spatial sensitivity
- **Propagation**: spatial smoothness

Figure 2. Architecture of the *PixContrast* and *PixPro* methods.
Aligning Pre-Training to Downstream Networks

- Using the same architecture as in downstream tasks
Beyond Image-based Unsupervised Pre-training

• Video based pre-training
  • Representative researchers
    • Andrew Zisserman, Weidi Xie, Xiaolong Wang, Alexei Efros et al

• Multi-modality pre-training
Take-Home Message

- Enjoy the “cake”

- **“Pure” Reinforcement Learning (cherry)**
  - The machine predicts a scalar reward given once in a while.
  - A few bits for some samples

- **Supervised Learning (icing)**
  - The machine predicts a category or a few numbers for each input
  - Predicting human-supplied data
  - 10→10,000 bits per sample

- **Self-Supervised Learning (cake génoise)**
  - The machine predicts any part of its input for any observed part.
  - Predicts future frames in videos
  - Millions of bits per sample
Reference


