Weakly and Self-Supervised Learning

A. Maier, V. Christlein, K. Breininger, Z. Yang, L. Rist, M. Nau, S. Jaganathan, C. Liu, N. Maul, L. Folle, K. Packhäuser, M. Zinnen
Pattern Recognition Lab, Friedrich-Alexander-Universität Erlangen-Nürnberg
April 24, 2023

Outline

Learning with Limited Annotations

Definition

Image-based SSL for Representation Learning

- Generative
- Spatial Context
- Context Similarity
- Contrastive SSL
- Supervised Contrastive Learning
- Bootstrap SSL – A paradigm change

Learning with Limited Annotations

Supervised Learning

So far, we have seen impressive results, achieved with ...

- ... large amounts of training data and
- ... consistent, high-quality annotations

Mask R-CNN [7], image source [2]

The Cost of Annotation

Image-level class labels: ~27 sec [11]

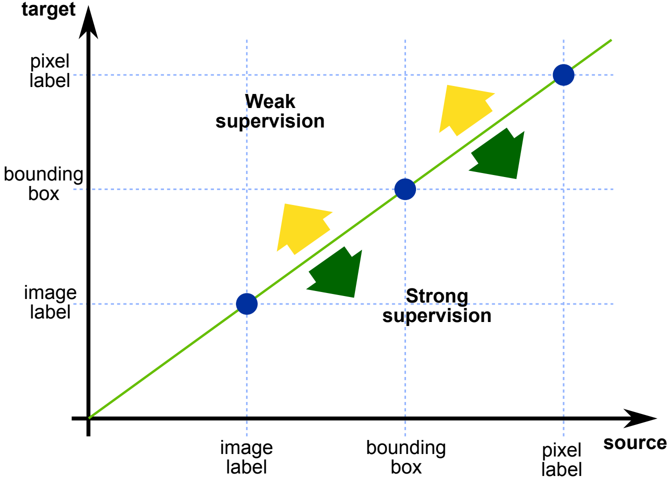
Instance spotting: + 14 sec [11]

Instance Segmentation: + 80 sec [11]

Dense pixel-level annotations: 1.5h [4]

Source: [4], [11]

Strongly vs. Weakly Supervised Learning



Source: Reproduced from CVPR18 Tutorial: Weakly Supervised Learning for Computer Vision

Key Ingredients for Weakly Supervised Learning

Priors: Explicit and Implicit

- Shape + size
- Contrast
- Motion
- Class distribution
- Similarity across images

Hints

- Image labels
- Bounding boxes
- Image caption
- Sparse temporal labels
- Scribbles
- Clicks inside objects

Source: Bearman et al. [3]

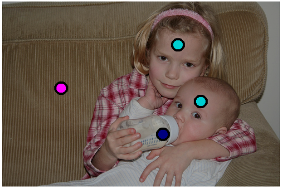
Key Ingredients for Weakly Supervised Learning

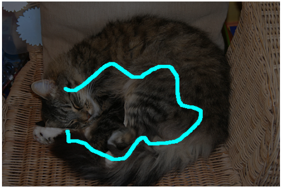
Priors: Explicit and Implicit

- Shape + size
- Contrast
- Motion
- Class distribution
- Similarity across images

Hints

- Image labels
- Bounding boxes
- Image caption
- Sparse temporal labels
- Scribbles
- Clicks inside objects





Source: Bearman et al. [3]

From Labels to Localization

Approach 1: Use a **pretrained** classification network [13]

- How does a change in the input affect the classification?
  ➔ Lecture on Visualization
- Qualitative segmentation map

Source: Simonyan et al. [13]

From Labels to Localization

Approach 1: Use a **pretrained** classification network [13]

- How does a change in the input affect the classification?
  ➔ Lecture on Visualization
- Qualitative segmentation map
- **Problem 1**: Classifier was never trained for localized decisions
- **Problem 2**: Good classifiers don't automatically yield good maps

Source: Simonyan et al. [13]

From Labels to Localization

Approach 2: Use a classification network, but smarter [14]

- Core idea: Use **global average pooling**

From Labels to Localization

Approach 2: Use a classification network, but smarter [14]

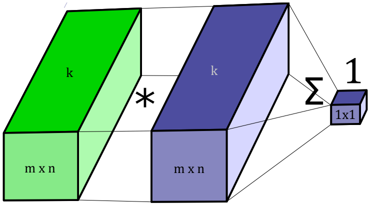
- Core idea: Use **global average pooling**
  ➔ Fully convolutional networks revisited

Fully Convolutional Networks: Revisited

- Fully connected layers fix the input size
  ➔ replace by $m \times n$ convolution

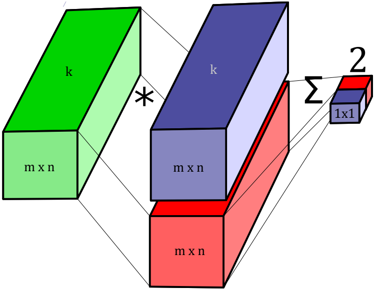
Fully Convolutional Networks: Revisited

- Fully connected layers fix the input size
  ➔ replace by $m \times n$ convolution



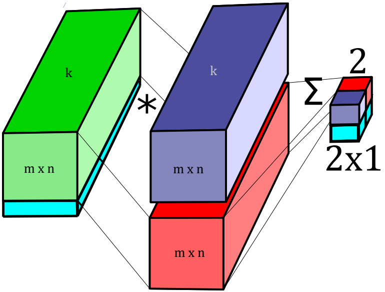
Fully Convolutional Networks: Revisited

- Fully connected layers fix the input size
  ➔ replace by $m \times n$ convolution



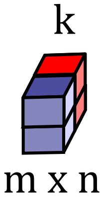
Fully Convolutional Networks: Revisited

- Fully connected layers fix the input size
  ➔ replace by $m \times n$ convolution



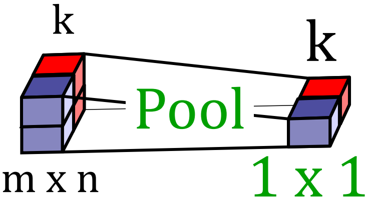
Fully Convolutional Networks: Revisited

- Fully connected layers fix the input size
  ➔ replace by $m \times n$ convolution
- Alternatively, we can also **pool** to the correct size first



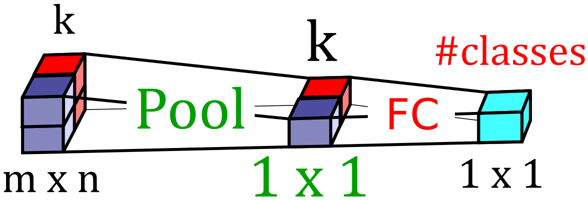
Fully Convolutional Networks: Revisited

- Fully connected layers fix the input size
  ➔ replace by $m \times n$ convolution
- Alternatively, we can also **pool** to the correct size first



Fully Convolutional Networks: Revisited

- Fully connected layers fix the input size
  ➔ replace by $m \times n$ convolution
- Alternatively, we can also **pool** to the correct size first



From Labels to Localization

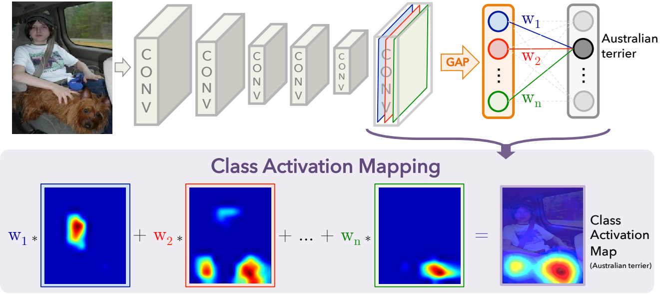
Approach 2: Use a classification network, but smarter [14]

- Core idea: Use **global average pooling**

From Labels to Localization

Approach 2: Use a classification network, but smarter [14]

- Core idea: Use **global average pooling**
- Then look at penultimate layer
- **Class Activation Maps (CAMs)**

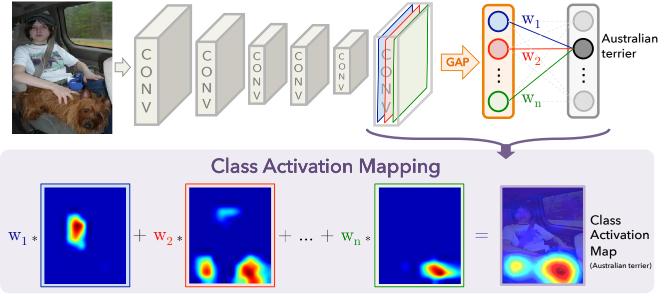


Source: Zhou et al. [14]

From Labels to Localization

Approach 2: Use a classification network, but smarter [14]

- Core idea: Use **global average pooling**
- Then look at penultimate layer
- **Class Activation Maps (CAMs)**
- Generalization: Grad-CAM [12]



Source: Zhou et al. [14]

From Bounding Boxes to Segmentation

Expensively annotated

Fully supervised

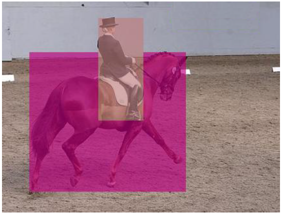
- Manual segmentation is tedious

Source: Khoreva et al. [6]

From Bounding Boxes to Segmentation

Expensively annotated
Fully supervised

Cheaply annotated



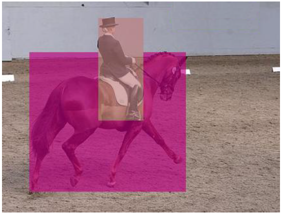
- Manual segmentation is tedious
- Bounding boxes are less tedious

Source: Khoreva et al. [6]

From Bounding Boxes to Segmentation

Expensively annotated
Fully supervised

Cheaply annotated



Cheaply annotated
Weakly supervised

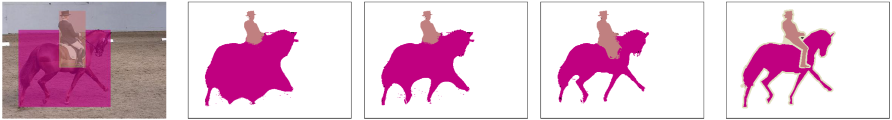
- Manual segmentation is tedious
- Bounding boxes are less tedious

➔ Can we learn segmentation from boxes?

Source: Khoreva et al. [6]

From Bounding Boxes to Segmentation

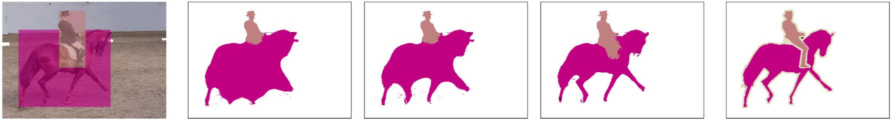
- Observation: Convolutional NNs are somewhat robust to (label-)noise
- Let’s use that: Use bounding boxes as target and recursively estimate better targets [6]



Example input rectangles	Output after 1 training round	After 5 rounds	After 10 rounds	Ground truth

From Bounding Boxes to Segmentation

- Observation: Convolutional NNs are somewhat robust to (label-)noise
- Let's use that: Use bounding boxes as target and recursively estimate better targets [6]



Example input rectangles | Output after 1 training round | After 5 rounds | After 10 rounds | Ground truth

- Problem: Training quickly degrades
- **Postprocess** intermediate predictions

From Bounding Boxes to Segmentation

- **Suppress** detections
  - ...of wrong class
  - ...outside the box
  - ...$<$% box area
  - ...outside of conditional random field boundaries

Source: Khoreva et al. [6]

From Bounding Boxes to Segmentation

- **Suppress** detections
  - ...of wrong class
  - ...outside the box
  - ...$<$% box area
  - ...outside of conditional random field boundaries

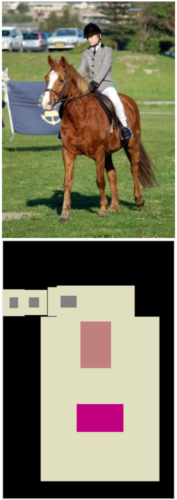
- Additional improvement: smaller boxes
  - Objects are “on average” roundish
  - Corners and edges contain “on average” the least true positives

Source: Khoreva et al. [6]

From Bounding Boxes to Segmentation

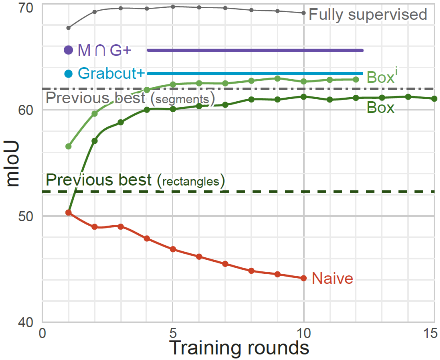
- **Suppress** detections
  - ...of wrong class
  - ...outside the box
  - ...$<$% box area
  - ...outside of conditional random field boundaries

- Additional improvement: smaller boxes
  - Objects are “on average” roundish
  - Corners and edges contain “on average” the least true positives
  - Define “ignore” regions with unknown labels



Source: Khoreva et al. [6]

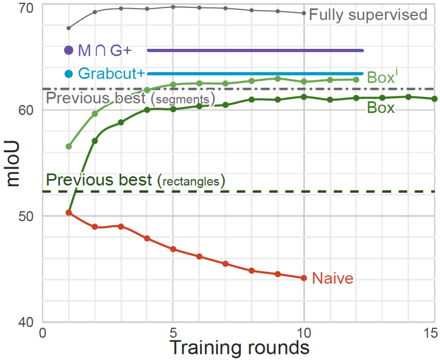
Improved Recursive Training



➔ Shrinking boxes beats state of the art

Source: [6]

Improved Recursive Training



→ Shrinking boxes beats state of the art

- Combine Grabcut and MCG for initial label

  → No need for recursion

Source: [6]

NEXT TIME

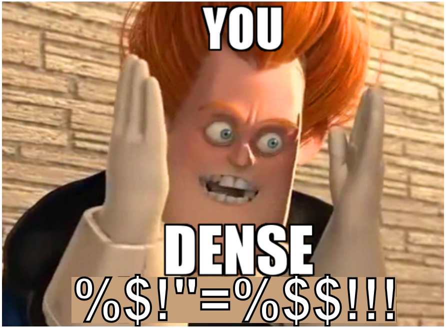
ON DEEP LEARNING

Weakly and Self-Supervised Learning - Part 2

A. Maier, V. Christlein, K. Breininger, Z. Yang, L. Rist, M. Nau, S. Jaganathan, C. Liu, N. Maul, L. Folle, K. Packhäuser, M. Zinnen
Pattern Recognition Lab, Friedrich-Alexander-Universität Erlangen-Nürnberg
April 24, 2023

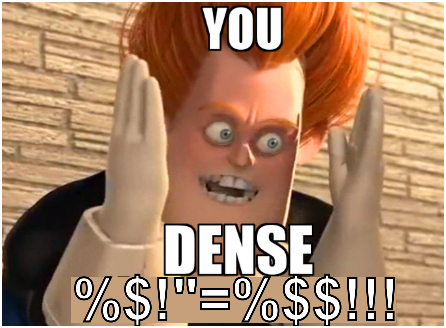
From Sparse Annotations to Dense Segmentations

From Sparse Annotations to Dense Segmentations



Source: Adapted from https://knowyourmeme.com/memes/.

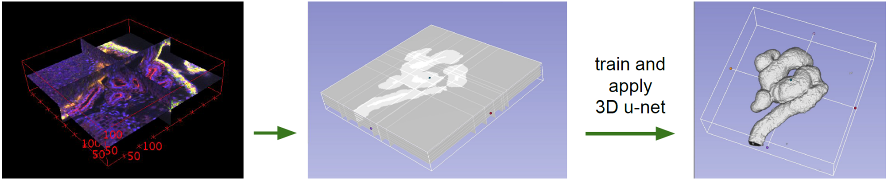
From Sparse Annotations to Dense Segmentations



... not quite

Source: Adapted from https://knowyourmeme.com/memes/.

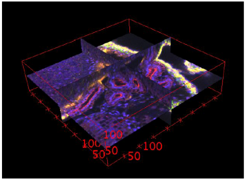
From Sparse Annotations to Dense Segmentations

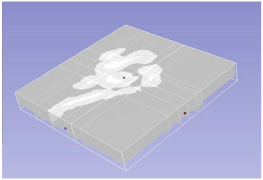


- 3D segmentation is extremely tedious

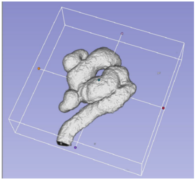
Source: Çiçek et al. [1]

From Sparse Annotations to Dense Segmentations





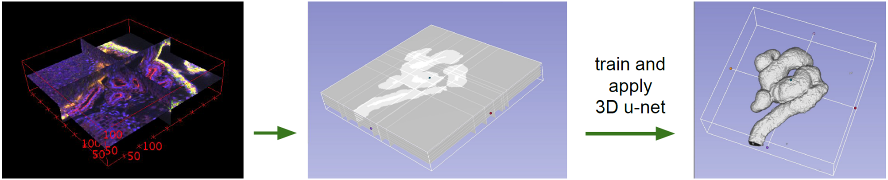
train and apply 3D u-net



- 3D segmentation is extremely tedious
- Obtain only a few labelled 2D slides
- Compute automatic segmentation in 3D

Source: Çiçek et al. [1]

From Sparse Annotations to Dense Segmentations



- 3D segmentation is extremely tedious
- Obtain only a few labelled 2D slides
- Compute automatic segmentation in 3D
- Allows for interactive correction

Source: Çiçek et al. [1]

From Sparse Annotations to Dense Segmentations

Training with sparse labels

- Problem: “One hot” labels $y_{n,i}$ with element i being 1
  ➔ either **true** or **false**

$$L(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \sum_{n} - \log \hat{y}_{n,i} \Big|_{y_{n,i}=1}$$

From Sparse Annotations to Dense Segmentations

Training with sparse labels

- Problem: “One hot” labels $y_{n,i}$ with element i being 1
  ➔ either **true** or **false**

$$L(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \sum_{n} - \log \hat{y}_{n,i} \Big|_{y_{n,i}=1}$$

- Obtain sparse loss by weighted cross entropy [1]

$$L'(\boldsymbol{y}, \hat{\boldsymbol{y}}) = L(\boldsymbol{y}, \hat{\boldsymbol{y}}) \cdot \mathrm{w}(\mathbf{y})$$

where $\mathrm{w}(y_{n,i}) = \begin{cases} 0 & \text{if } y_n \text{ is not labelled} \\ w_{n,i} > 0 & \text{otherwise} \end{cases}$

From Sparse Annotations to Dense Segmentations

Training with sparse labels

- Problem: “One hot” labels $y_{n,i}$ with element i being 1
  ➔ either **true** or **false**

$$L(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \sum_n -\log \hat{y}_{n,i}\Big|_{y_{n,i}=1}$$

- Obtain sparse loss by weighted cross entropy [1]

$$L'(\boldsymbol{y}, \hat{\boldsymbol{y}}) = L(\boldsymbol{y}, \hat{\boldsymbol{y}}) \cdot \mathrm{w}(\mathbf{y})$$

where $\mathrm{w}(y_{n,i}) = \begin{cases} 0 & \text{if } y_n \text{ is not labelled} \\ w_{n,i} > 0 & \text{otherwise} \end{cases}$

- Can be easily extended to interactive segmentation by updating $\boldsymbol{y}$ and $\mathrm{w}(\mathbf{y})$

Take Away: Weakly Supervised Learning

- Fine grained labels are expensive - can we get away with something cheaper?
- Core definition: **Label** has less detail than **target**
- Methods depend on **prior knowledge** and **weak labels** (“hints”)
- Typically inferior to **fully supervised** training
  ➔ but highly relevant in practice

Take Away: Weakly Supervised Learning

- Fine grained labels are expensive - can we get away with something cheaper?
- Core definition: **Label** has less detail than **target**
- Methods depend on **prior knowledge** and **weak labels** (“hints”)
- Typically inferior to **fully supervised** training
  ➔ but highly relevant in practice
- Don’t forget transfer learning (!)
- Related:
  - **Semi-supervised** Learning
  - **Self-supervised** Learning

NEXT TIME
ON DEEP LEARNING

Weakly and Self-Supervised Learning - Part 3

A. Maier, V. Christlein, K. Breininger, Z. Yang, L. Rist, M. Nau, S. Jaganathan, C. Liu, N. Maul, L. Folle, K. Packhäuser, M. Zinnen
Pattern Recognition Lab, Friedrich-Alexander-Universität Erlangen-Nürnberg
April 24, 2023

Outline

Learning with Limited Annotations

Definition

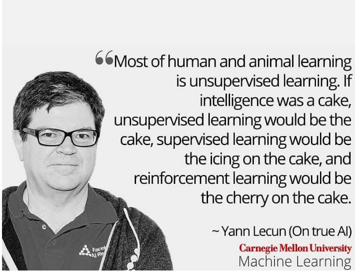
Image-based SSL for Representation Learning

- Generative
- Spatial Context
- Context Similarity
- Contrastive SSL
- Supervised Contrastive Learning
- Bootstrap SSL – A paradigm change

Definition

Motivation

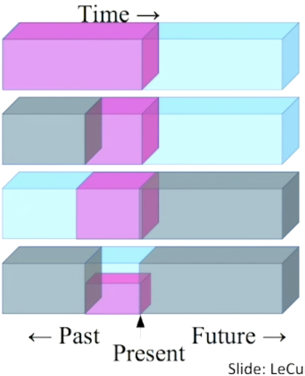
- Jitendra Malik: “Supervision is the opium of the AI researcher”
- Alyosha Efros: “The AI revolution will not be supervised”
- Yann LeCun:



Source: https://www.facebook.com/722677142/posts/10156036317282143/

Idea

- Predict any part of the input from any other part.
- Predict the future from the past.
- Predict the future from the recent past.
- Predict the past from the present.
- Predict the top from the bottom.
- Predict the occluded from the visible
- Pretend there is a part of the input you don't know and predict that.



Slide: LeCun

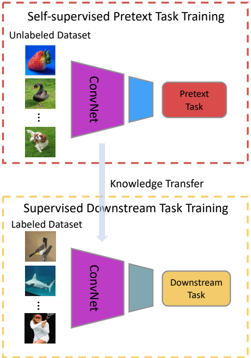
Source: https://www.youtube.com/watch?v=7I0Qt7GALVk

Self-supervised Learning: Definition

Yann LeCun
April 30, 2019 ·
I now call it "self-supervised learning", because "unsupervised" is both a loaded and confusing term.

- Subcategory of unsupervised learning
- Use pretext/surrogate/pseudo tasks in a supervised fashion
  - ➔ Automatically generated labels
  - ➔ Measurement of correctness
- Downstream task: retrieval, supervised or semi-supervised classification, etc.

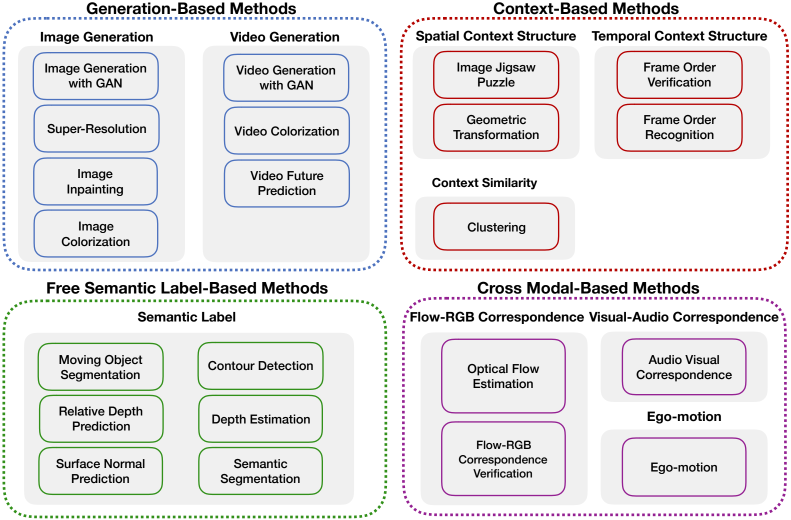
Note: Generative models (e.g. GANs) are also SSL methods



Source: [15]

Source: https://www.facebook.com/722677142/posts/10155934004262143/

Pretext Tasks Overview



Source: [15]

Image-based SSL for Representation Learning

Generative

Image Colorization

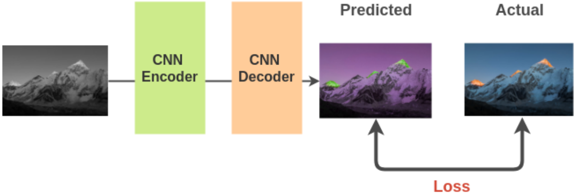
Data generation:

Source: https://amitness.com/2020/02/illustrated-self-supervised-learning/

Image Colorization

Data generation:

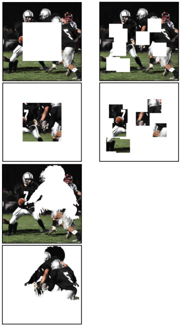
Pretext task: ℓ_2 loss between gray and color version



Source: https://amitness.com/2020/02/illustrated-self-supervised-learning/

Image Inpainting

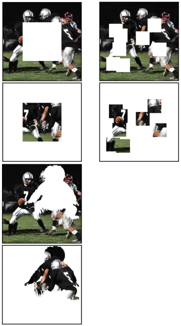
Data generation:



Source: [16]

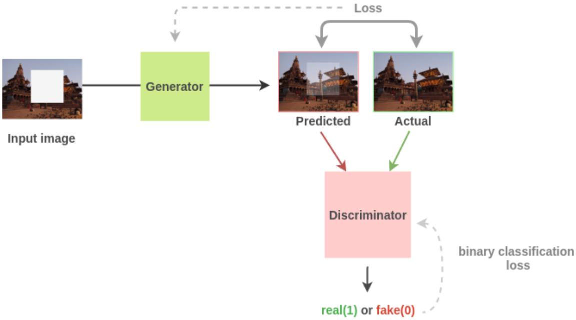
Image Inpainting

Data generation:



Source: [16]

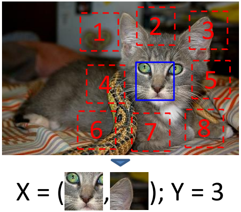
Pretext task:

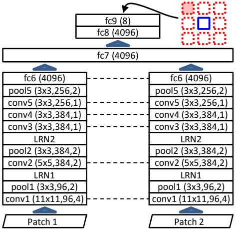


Source: `https://amitness.com/2020/02/illustrated-self-supervised-learning/`

Spatial Context

Solve Jigsaw Puzzle [17]



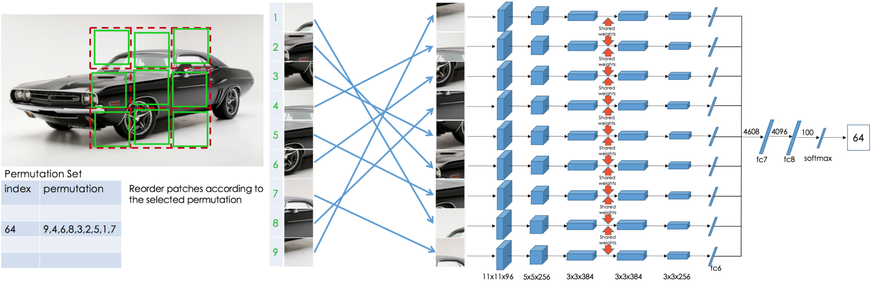


Attention: Trivial solution possible

- boundary patterns, continuing textures ➔ use large enough gaps
- chromatic aberration
  - ➔ Pre-process images by shifting green and magenta toward gray
  - ➔ randomly drop 2 color channels

Source: [17]

Solve Jigsaw Puzzle++ [18]



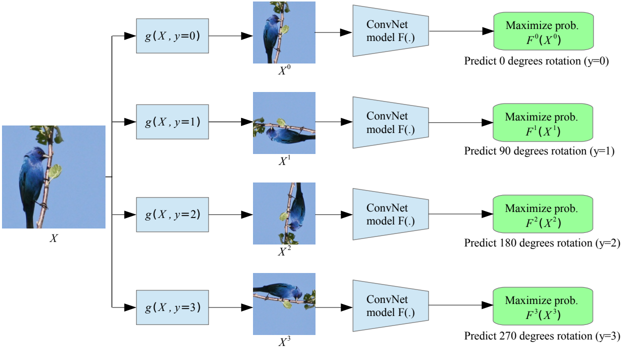
9 tiles ➔ 9! = 362 880 possible permutations

Source: [18]

Solve Jigsaw Puzzle++ [18] (cont.)

Number of permutations	Average hamming distance	Minimum hamming distance	Jigsaw task accuracy	Detection performance
1000	8.00	2	71	**53.2**
1000	6.35	2	62	51.3
1000	3.99	2	54	50.2
100	8.08	2	88	52.6
95	8.08	3	90	52.4
85	8.07	4	91	52.7
71	8.07	5	92	52.8
35	8.13	6	94	52.6
10	8.57	7	97	49.2
7	8.95	8	98	49.6
6	9	9	99	49.7

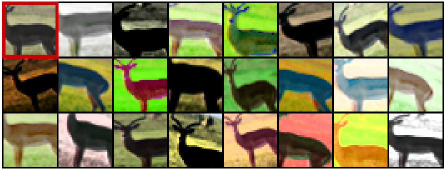
Rotation [19]



Source: [19]

Context Similarity

Distortions [21] (Exemplar-CNN)



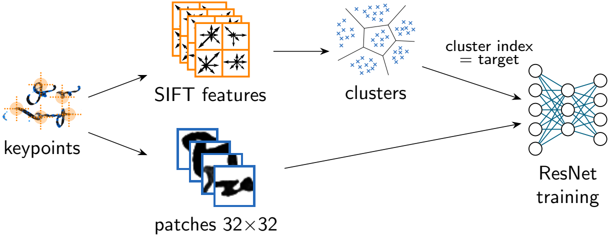
- For each input patch, create N (e.g. $N = 100$) distorted images

➔ All these distorted images form one class

- Discriminate between a set of surrogate classes (e.g. 8000 pseudo-classes)

Source: [21]

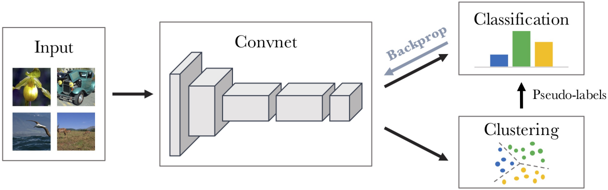
Clustering [22]



- From keypoints in an image extract patches and compute descriptors
- Cluster features of patches using k-means into N clusters ($N = 5000$)
- Use cluster indices as targets for input patches
- Remove features (+patches) in between of two clusters

Source: [22]

Clustering [20] (DeepCluster)

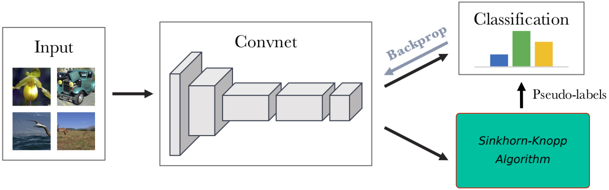


- Alternate between
  - k-means clustering (each epoch) of PCA-whitened ($D = 256$) & ℓ_2-normalized activation features
  - CNN training
- Avoid trivial solutions
  - Re-assign empty clusters
  - Weight contribution of an input by inverse of the size of its assigned cluster

Source: [20]

Clustering [24]

Self-labelling with Optimal Transport



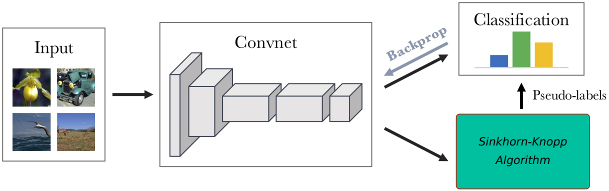
Optimal transport

	Supply	Need	
25 laptops	Warehouse A	Shop 1	25 laptops
25 laptops	Warehouse B	Shop 2	25 laptops

Source: https://amitness.com/2020/04/illustrated-self-labelling/

Clustering [24]

Self-labelling with Optimal Transport



Optimal transport

	Supply	Need	
25 laptops	Warehouse A	Shop 1	25 laptops
25 laptops	Warehouse B	Shop 2	25 laptops

Distance(cost) matrix → Optimal Allocation

	Shop 1	Shop 2
Warehouse A	2km	3km
Warehouse B	2km	1km

	Shop 1	Shop 2
Warehouse A	25	0
Warehouse B	0	25

Source: `https://amitness.com/2020/04/illustrated-self-labelling/`

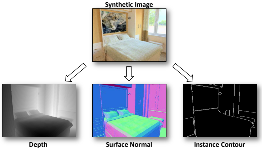
Clustering [24] (cont.)

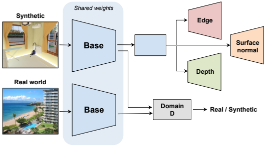
Self-labelling with Optimal Transport

Comparison to DeepCluster

- no separate clustering loss ➔ can lead to degenerate solutions
- clustering approach that minimizes the same cross-entropy loss that the network also optimize.

Multi-task SSL using Synthetic Imagery [23]





- Given: input synthetic RGB image
- Network simultaneously predicts: surface normal, depth, instance contour
- Additionally: minimize feature space domain differences between real and synthetic data

Source: [23]

NEXT TIME

ON DEEP LEARNING

Weakly and Self-Supervised Learning - Part 4

A. Maier, V. Christlein, K. Breininger, Z. Yang, L. Rist, M. Nau, S. Jaganathan, C. Liu, N. Maul, L. Folle, K. Packhäuser, M. Zinnen
Pattern Recognition Lab, Friedrich-Alexander-Universität Erlangen-Nürnberg
April 24, 2023

Contrastive SSL

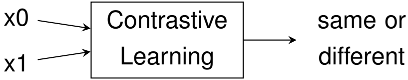
Contrastive Learning

Match the correct animal

Source: https://amitness.com/2020/03/illustrated-simclr/

Contrastive Learning

Match the correct animal



Source: https://amitness.com/2020/03/illustrated-simclr/

Contrastive Learning

Match the correct animal

x0 → Contrastive Learning → same or different
x1 →

Advantages over generative/context models:

- Pixel-level losses could overly focus on pixel-based details, rather than more abstract latent factors
- Pixel-based objectives often assume pixel independence ➔ reduce ability to model correlations or complex structure

Source: `https://amitness.com/2020/03/illustrated-simclr/`

Contrastive Loss

Given: $\mathcal{X} = \{\boldsymbol{x}, \underbrace{\boldsymbol{x}^+}_{\text{positive sample}}, \underbrace{\boldsymbol{x}_1^-, \ldots, \boldsymbol{x}_{N-1}^-}_{\text{negative samples}}\}$; similarity function $s(.)$ (e. g. cosine similarity)

Goal: $s(f(\boldsymbol{x}), f(\boldsymbol{x}^+)) >> s(f(\boldsymbol{x}), f(\boldsymbol{x}^-))$

Contrastive/InfoNCE Loss (aka 'n-pair loss'/'consistency loss'/'ranking-based NCE'):

Cross-entropy loss for (N)-way softmax classifier

$$\mathcal{L}_N = -\mathbb{E}_{\mathcal{X}}\left[\log \frac{\exp\left(s(f(\boldsymbol{x}), f(\boldsymbol{x}^+))\right)}{\exp\left(s(f(\boldsymbol{x}), f(\boldsymbol{x}^+))\right) + \sum_{j=1}^{N-1} \exp\left(s(f(\boldsymbol{x}), f(\boldsymbol{x}_j^-))\right)}\right]$$

$$= -\mathbb{E}_{\mathcal{X}}\left[\log \frac{\exp\left(s(f(\boldsymbol{x}), f(\boldsymbol{x}^+))\right)}{\sum_{j=1}^{N} \exp\left(s(f(\boldsymbol{x}), f(\boldsymbol{x}_j))\right)}\right]$$

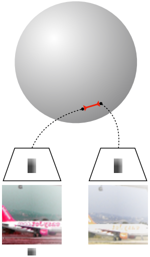
Contrastive Loss (cont.)

Minimizing Contrastive Loss maximizes a lower bound on the mutual information between $f(\boldsymbol{x})$ and $f(\boldsymbol{x}^+)$ [25], [27].

Common Variation: temperature hyperparameter τ

$$\mathcal{L}_N = -\mathbb{E}_{\mathcal{X}} \left[\log \frac{\exp\left(s(f(\boldsymbol{x}), f(\boldsymbol{x}^+))/\tau\right)}{\sum_{j=1}^{N+1} \exp\left(s(f(\boldsymbol{x}), f(\boldsymbol{x}_j))/\tau\right)} \right]$$

Effectivity of Contrastive Loss

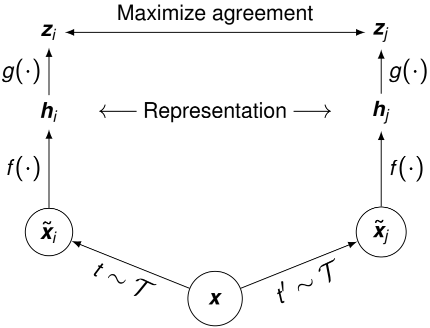


Alignment: Similar samples have similar features

Uniformity: Preserve maximal information.

Source: [35]

Examples: SimCLR [31]

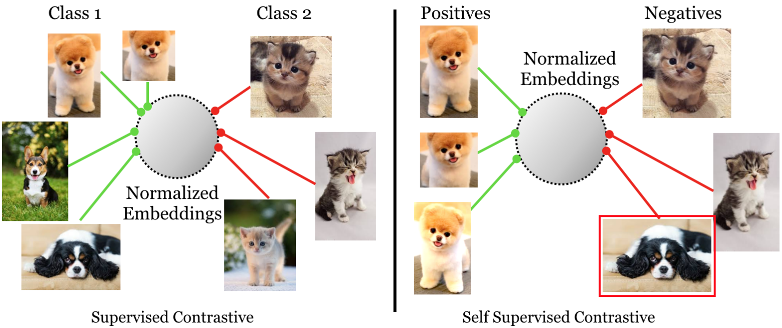


1. Mini-batch of n samples. Each sample is applied with two different data augmentation operations ➔ $2n$ augmented samples: $\tilde{\mathbf{x}}_i = t(\mathbf{x}), \quad \tilde{\mathbf{x}}_j = t'(\mathbf{x}), \quad t, t' \sim \mathcal{T}$
2. One positive pair, $2(n-1)$ negatives. Representation through base encoder f: $\mathbf{h}_i = f(\tilde{\mathbf{x}}_i), \quad \mathbf{h}_j = f(\tilde{\mathbf{x}}_j)$
3. Contrastive loss on top of representation head g:

$$\mathcal{L}_{i,j} = -\log \frac{\exp(s(\mathbf{z}_i, \mathbf{z}_j)/\tau)}{\sum_{k=1}^{2n} \mathbf{1}_{[k \neq i]} \exp(s(\mathbf{z}_i, \mathbf{z}_k)/\tau)}$$

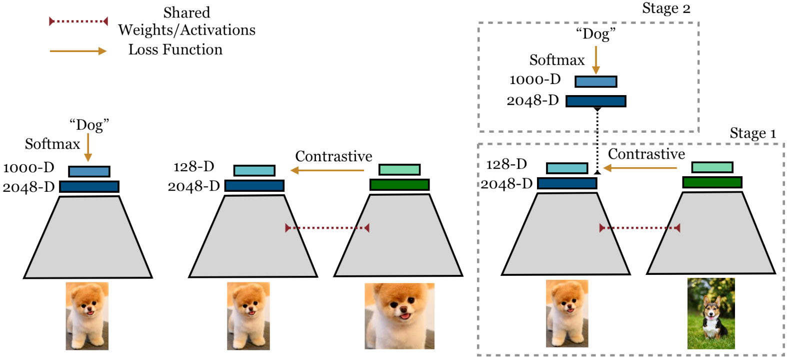
Supervised Contrastive Learning

Supervised Contrastive Learning [33]



Source: [33]

Supervised Contrastive Learning [33]



(a) Supervised Cross Entropy (b) Self Supervised Contrastive (c) Supervised Contrastive

Source: [33]

Supervised Contrastive Loss

- Self-supervised **has no** knowledge about class labels ➔ only one positive example
- Supervised **has** knowledge about class labels ➔ many positives per example
- Compute loss between any sample $\boldsymbol{z}_j$ having the same class as anchor $\boldsymbol{z}_i$ $(\boldsymbol{y}_i = \boldsymbol{y}_j)$

Supervised Contrastive Loss

- Self-supervised **has no** knowledge about class labels ➔ only one positive example
- Supervised **has** knowledge about class labels ➔ many positives per example
- Compute loss between any sample $\boldsymbol{z}_j$ having the same class as anchor $\boldsymbol{z}_i$ $(\boldsymbol{y}_i = \boldsymbol{y}_j)$

$$L_{\text{sup}} = \sum_{i=1}^{2N} - \dots \sum_{j=1}^{2N} \mathbb{1}_{i \neq j} \cdot \mathbb{1}_{\boldsymbol{y}_i = \boldsymbol{y}_j} \cdot \log \frac{\exp\left(\boldsymbol{z}_i^\top \boldsymbol{z}_j / \tau\right)}{\sum_{k=1}^{2N} \mathbb{1}_{i \neq k} \cdot \exp\left(\boldsymbol{z}_i^\top \boldsymbol{z}_k / \tau\right)}$$

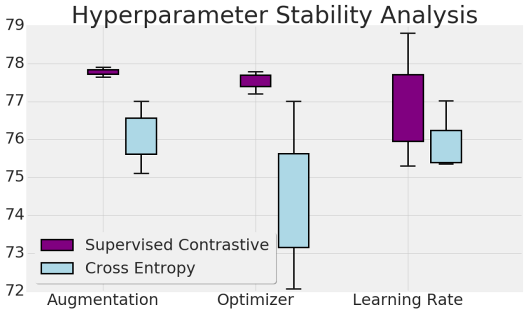
Supervised Contrastive Loss (cont.)

- Vectors $\boldsymbol{z}$ need to be normalized, i.e. $\boldsymbol{z} = \boldsymbol{w}/\|\boldsymbol{w}\|$, where $\boldsymbol{w}$ is the output of the projection network
- Gradient w.r.t. to $\boldsymbol{w}$ is high for hard positives and negatives and small otherwise ➔ "built-in" focal loss
- For one positive and one negative it turns out that

$$L_{\text{sup}} \propto \|\boldsymbol{z}_a - \boldsymbol{z}_p\|^2 - \|\boldsymbol{z}_a - \boldsymbol{z}_n\|^2 + 2\tau$$

➔ Common contrastive loss in siamese networks

Hyperparameter stability



- Increased stability w.r.t. to non-optimal hyperparameters

Source: [33]

What else?

- Training about 50% slower than CE
- Improves over training with SOTA data augmentation (CutMix)
- Enables unsupervised clustering in latent space ➔ correction of label-noise, new possibilities for semi-supervised, . . .

Bootstrap SSL – A paradigm change

Bootstrap Your Own Latent (BYOL) [34] Overview

So far: Contrastive loss between exemplar, positive and negatives

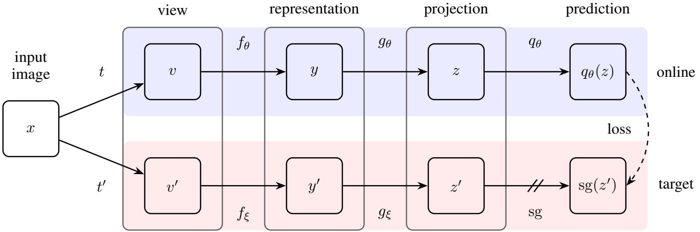
➔ Negative pairs critical (often: large batchsizes, memory banks, custom mining strategies)

➔ Right augmentation strategy critical

BYOL:

- No negative pair
- No contrastive loss
- More resilient to changes in batch size and set of image augmentations compared to its contrastive counterparts

BYOL [34] Method



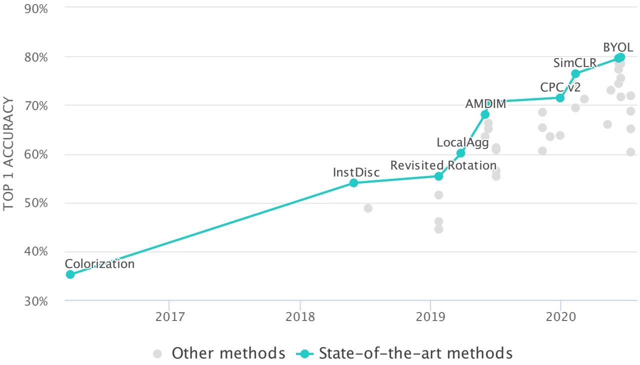
- Two networks: **online** and **target** network ➔ interact and learn from each other
- In theory: trivial solution possible (e.g. zero for all images)

➔ use slow-moving average of the online network as target network

- Loss: MSE of ℓ^2-normalized predictions (proportional to cosine distance)

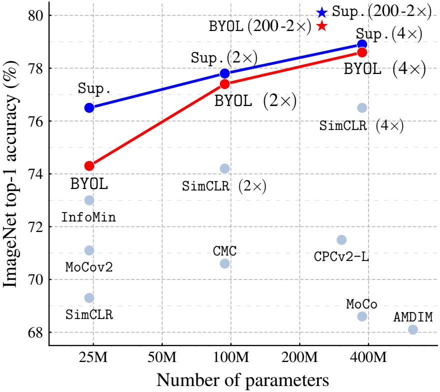
Source: [34]

SSL State of the Art



Source: https://paperswithcode.com/sota/self-supervised-image-classification-on

SSL State of the Art (cont.)



Source: [34]

Further Reading

Blogs:

- https://lilianweng.github.io/lil-log/2019/11/10/self-supervised-learning.html
- https://amitness.com/2020/02/illustrated-self-supervised-learning/
- https://ankeshanand.com/blog/2020/01/26/contrative-self-supervised-learning.html

Others:

- https://github.com/jason718/awesome-self-supervised-learning
- https://www.youtube.com/watch?v=7I0Qt7GALVk

NEXT TIME

ON DEEP LEARNING

Next Time: Emerging Methods

- Can we process graphs using deep networks?
- Do we really have to learn everything form scratch?
- Let's see whether we can re-use prior knowledge in deep learning...

References

References I

[1] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, et al. “3d u-net: learning dense volumetric segmentation from sparse annotation”. In: MICCAI. Springer. 2016, pp. 424–432.

[2] Waleed Abdulla. Mask R-CNN for object detection and instance segmentation on Keras and Ten Accessed: 27.01.2020. 2017.

[3] Olga Russakovsky, Amy L. Bearman, Vittorio Ferrari, et al. “What’s the point: Semantic segmentation with point supervision”. In: CoRR abs/1506.02106 (2015). arXiv: 1506.02106.

[4] Marius Cordts, Mohamed Omran, Sebastian Ramos, et al. “The Cityscapes Dataset for Semantic Urban Scene Understanding”. In: CoRR abs/1604.01685 (2016). arXiv: 1604.01685.

References II

[5] Richard O. Duda, Peter E. Hart, and David G. Stork. Pattern classification. 2nd ed. New York: Wiley-Interscience, Nov. 2000.

[6] Anna Khoreva, Rodrigo Benenson, Jan Hosang, et al. “Simple Does It: Weakly Supervised Instance and Semantic Segmentation”. In: arXiv preprint arXiv:1603.07485 (2016).

[7] Kaiming He, Georgia Gkioxari, Piotr Dollár, et al. “Mask R-CNN”. In: CoRR abs/1703.06870 (2017). arXiv: 1703.06870.

[8] Sangheum Hwang and Hyo-Eun Kim. “Self-Transfer Learning for Weakly Supervised Lesion Localization”. In: MICCAI. Springer. 2016, pp. 239–246.

[9] Maxime Oquab, Léon Bottou, Ivan Laptev, et al. “Is object localization for free? weakly-supervised learning with convolutional neural networks”. In: Proc. CVPR. 2015, pp. 685–694.

References III

[10] Alexander Kolesnikov and Christoph H. Lampert. “Seed, Expand and Constrain: Three Principles for Weakly-Supervised Image Segmentation”. In: CoRR abs/1603.06098 (2016). arXiv: 1603.06098.

[11] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, et al. “Microsoft COCO: Common Objects in Context”. In: CoRR abs/1405.0312 (2014). arXiv: 1405.0312.

[12] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, et al. “Grad-CAM: Why did you say that? Visual Explanations from Deep Networks via Gradient-based Localization”. In: CoRR abs/1610.02391 (2016). arXiv: 1610.02391.

[13] K. Simonyan, A. Vedaldi, and A. Zisserman. “Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps”. In: Proc. ICLR (workshop track). 2014.

References IV

[14] Bolei Zhou, Aditya Khosla, Agata Lapedriza, et al. "Learning deep features for discriminative localization". In: Proc. CVPR. 2016, pp. 2921–2929.

[15] Longlong Jing and Yingli Tian. "Self-supervised Visual Feature Learning with Deep Neural Networks: A Survey". In: arXiv e-prints, arXiv:1902.06162 (Feb. 2019). arXiv: 1902.06162 [cs.CV].

[16] D. Pathak, P. Krähenbühl, J. Donahue, et al. "Context Encoders: Feature Learning by Inpainting". In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016, pp. 2536–2544.

[17] C. Doersch, A. Gupta, and A. A. Efros. "Unsupervised Visual Representation Learning by Context Prediction". In: 2015 IEEE International Conference on Computer Vision (ICCV). Dec. 2015, pp. 1422–1430.

References V

[18] Mehdi Noroozi and Paolo Favaro. “Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles”. In: Computer Vision – ECCV 2016. Cham: Springer International Publishing, 2016, pp. 69–84.

[19] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. “Unsupervised Representation Learning by Predicting Image Rotations”. In: International Conference on Learning Representations. 2018.

[20] Mathilde Caron, Piotr Bojanowski, Armand Joulin, et al. “Deep Clustering for Unsupervised Learning of Visual Features”. In: Computer Vision – ECCV 2018. Cham: Springer International Publishing, 2018, pp. 139–156.

References VI

[21] A. Dosovitskiy, P. Fischer, J. T. Springenberg, et al. “Discriminative Unsupervised Feature Learning with Exemplar Convolutional Neural Networks”. In: IEEE Transactions on Pattern Analysis and Machine Intelligence 38.9 (Sept. 2016), pp. 1734–1747.

[22] V. Christlein, M. Gropp, S. Fiel, et al. “Unsupervised Feature Learning for Writer Identification and Writer Retrieval”. In: 2017 14th IAPR International Conference on Document Analysis and Recognit Vol. 01. Nov. 2017, pp. 991–997.

[23] Z. Ren and Y. J. Lee. “Cross-Domain Self-Supervised Multi-task Feature Learning Using Synthetic Imagery”. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. June 2018, pp. 762–771.

References VII

[24] Asano YM., Rupprecht C., and Vedaldi A. "Self-labelling via simultaneous clustering and representation learning". In: International Conference on Learning Representations. 2020.

[25] Ben Poole, Sherjil Ozair, Aaron Van Den Oord, et al. "On Variational Bounds of Mutual Information". In: Proceedings of the 36th International Conference on Machine Learning. Vol. 97. Proceedings of Machine Learning Research. Long Beach, California, USA: PMLR, Sept. 2019, pp. 5171–5180.

[26] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, et al. "Learning deep representations by mutual information estimation and maximization". In: International Conference on Learning Representations. 2019.

References VIII

[27] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. “Representation Learning with Contrastive Predictive Coding”. In: arXiv e-prints, arXiv:1807.03748 (July 2018). arXiv: 1807.03748 [cs.LG].

[28] Philip Bachman, R Devon Hjelm, and William Buchwalter. “Learning Representations by Maximizing Mutual Information Across Views”. In: Advances in Neural Information Processing Systems 32. Curran Associates, Inc., 2019, pp. 15535–15545.

[29] Yonglong Tian, Dilip Krishnan, and Phillip Isola. “Contrastive Multiview Coding”. In: arXiv e-prints, arXiv:1906.05849 (June 2019), arXiv:1906.05849. arXiv: 1906.05849 [cs.CV].

[30] Kaiming He, Haoqi Fan, Yuxin Wu, et al. “Momentum Contrast for Unsupervised Visual Representation Learning”. In: arXiv e-prints, arXiv:1911.05722 (Nov. 2019). arXiv: 1911.05722 [cs.CV].

References IX

[31] Ting Chen, Simon Kornblith, Mohammad Norouzi, et al. "A Simple Framework for Contrastive Learning of Visual Representations". In: arXiv e-prints, arXiv:2002.05709 (Feb. 2020), arXiv:2002.05709. arXiv: 2002.05709 [cs.LG].

[32] Ishan Misra and Laurens van der Maaten. "Self-Supervised Learning of Pretext-Invariant Representations". In: arXiv e-prints, arXiv:1912.01991 (Dec. 2019). arXiv: 1912.01991 [cs.CV].

[33] Prannay Khosla, Piotr Teterwak, Chen Wang, et al. "Supervised Contrastive Learning". In: arXiv e-prints, arXiv:2004.11362 (Apr. 2020). arXiv: 2004.11362 [cs.LG].

References X

[34] Jean-Bastien Grill, Florian Strub, Florent Altché, et al. “Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning”. In: arXiv e-prints, arXiv:2006.07733 (June 2020), arXiv:2006.07733. arXiv: 2006.07733 [cs.LG].

[35] Tongzhou Wang and Phillip Isola. “Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere”. In: arXiv e-prints, arXiv:2005.10242 (May 2020), arXiv:2005.10242. arXiv: 2005.10242 [cs.LG].

[36] Junnan Li, Pan Zhou, Caiming Xiong, et al. “Prototypical Contrastive Learning of Unsupervised Representations”. In: arXiv e-prints, arXiv:2005.04966 (May 2020), arXiv:2005.04966. arXiv: 2005.04966 [cs.CV].