An Introduction to Contrastive Learning

Zhiming Xu
zx2rw@virginia.edu

School of Engineering and Applied Science
University of Virginia

June 4, 2021

- Learning representations $f_{\boldsymbol{\theta}} : \boldsymbol{x} \to \boldsymbol{v}$ via separating positives and negatives (contrast) measured by score.
- Optimize parameters $\boldsymbol{\theta}$
  st. $\mathrm{score}\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}), f_{\boldsymbol{\theta}}(\boldsymbol{x}_+)\right) > \mathrm{score}\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}), f_{\boldsymbol{\theta}}(\boldsymbol{x}_-)\right)$.

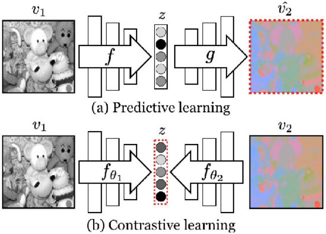


Figure 1: Predictive (autoencoder) v.s. Contrastive [1].

Table of Contents

InsDis

- Non-Parametric Softmax Classifier
- NCE & Regularization
- Image Class Classifier

CMC

- Contrastive Learning with Two Views
- Contrastive Learning with Multiple Views
- Implementation Details

MoCo

- Contrastive Learning as Dictionary Look-up
- Momentum Contrast

SupCon

- Supervised Contrastive Losses
- Choice of $\mathcal{L}^{sup}$

Summary

- Problem formation: non-parametric instance-level classification.
- Learning objective: embedding function $f_{\boldsymbol{\theta}}(\cdot)$ that induces a metric over image space $d_{\boldsymbol{\theta}}(x, y) = \| f_{\boldsymbol{\theta}}(x) - f_{\boldsymbol{\theta}}(y) \|$.
- Novelty: train a non-parametric classifier that distinguishes each image instance as its own class.

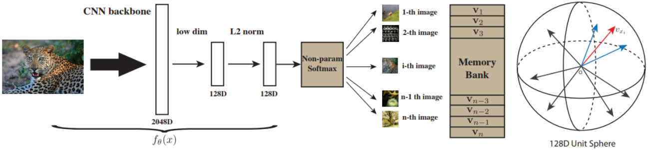


Figure 2: Non-parametric instance discrimination framework.

Table of Contents

InsDis

Non-Parametric Softmax Classifier

NCE & Regularization

Image Class Classifier

CMC

Contrastive Learning with Two Views

Contrastive Learning with Multiple Views

Implementation Details

MoCo

Contrastive Learning as Dictionary Look-up

Momentum Contrast

SupCon

Supervised Contrastive Losses

Choice of $\mathcal{L}^{sup}$

Summary

Parametric & Non-Parametric Classifiers

- Suppose we have n images $\{x_1, x_2, \cdots, x_n\}$ and their features $\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_n\}, \mathbf{v}_i = f_{\boldsymbol{\theta}}(x_i)$.
- *Parametric Classifier*

$$P(i|\mathbf{v}) = \frac{\exp\left(\mathbf{w}_i^\top \mathbf{v}\right)}{\sum_{j=1}^{n} \exp\left(\mathbf{w}_j^\top \mathbf{v}\right)},$$

  $\mathbf{w}_j$ is a weight vector (parameter) for each class j, $\mathbf{w}_j^\top \mathbf{v}$ measures how similar $\mathbf{v}$ matches class (instance) j.
- *Non-Parametric Classifier*

$$P(i|\mathbf{v}) = \frac{\exp\left(\mathbf{v}_i^\top \mathbf{v} / \tau\right)}{\sum_{j=1}^{n} \exp\left(\mathbf{v}_j^\top \mathbf{v} / \tau\right)}, \tag{1}$$

  τ is a temperature parameter controlling the concentration level of distribution, and no training parameters are involved.

Parametric v.s. Non-Parametric

- Weight vectors (parameters) $\{\mathbf{w}_j\}$ are only valid for training classes and do not generalize.
- Non-parametric features $\{\mathbf{v}_i\}$ can be added from new instances.
- Non-parametric formulation eliminates the need for weight vectors and thus reduces computing and storing costs.

Objective Function

- Maximum the joint probability that all instances are classified as themselves: $\prod_{i=1}^{n} P_{\boldsymbol{\theta}}\left(i | f_{\boldsymbol{\theta}}(x_i)\right)$.
- Equivalently, minimize the negative log-likelihood:

$$J(\boldsymbol{\theta}) = -\sum_{i=1}^{n} \log P\left(i | f_{\boldsymbol{\theta}}(x_i)\right). \tag{2}$$

Memory Bank

To compute Eq. 1, all images' features, $\{\mathbf{v}_i\}$ are needed and used multiple times. A *memory bank* V is used to reduce the computing and storing cost.

Memory Bank

Suppose $\mathbf{f}_i = f_{\boldsymbol{\theta}}(x_i)$ is x_i's feature. A memory bank is a set of features $V = \{\boldsymbol{v}_i\}, \forall i$ randomly initialized and updated with $\mathbf{v}_i \leftarrow \mathbf{f}_i$ during each training iteration.

.

Table of Contents

InsDis

Non-Parametric Softmax Classifier

NCE & Regularization

Image Class Classifier

CMC

Contrastive Learning with Two Views

Contrastive Learning with Multiple Views

Implementation Details

MoCo

Contrastive Learning as Dictionary Look-up

Momentum Contrast

SupCon

Supervised Contrastive Losses

Choice of $\mathcal{L}^{sup}$

Summary

Noise-Contrastive Estimation

Due to the large number of classes, n scales up to millions and computing Eq. 2 can be prohibitively expensive. Noise-contrastive estimation (NCE) is used to solve this problem here.

The multi-class classification task before is cast to a binary classification that discriminates *data samples* and *noise samples*. The probability of a representation $\mathbf{v}$ corresponding to the i-th example in V is:

$$P(i|\mathbf{v}) = \frac{\exp\left(\mathbf{v}^{\top}\mathbf{f}_i/\tau\right)}{\boldsymbol{Z}_i}, \tag{3}$$

$$\boldsymbol{Z}_i = \sum_{j=1}^{n} \exp\left(\mathbf{v}_j^{\top}\mathbf{f}_i/\tau\right). \tag{4}$$

Noise-Contrastive Estimation (cont.)

Uniform distribution $U(n)$ is used to formulate *noise distribution*.
Suppose noise samples are m times more frequent than data samples.
The posterior probability of sample i with feature $\mathbf{v}$ from it is:

$$h(i, \mathbf{v}) = \frac{P(i|\mathbf{v})}{P(i|\mathbf{v}) + mP_n(i)}.$$

The objective to minimize negative log-posterior distribution is:

$$J_{NCE}(\boldsymbol{\theta}) = -E_{P_d}\left[\log h(i, \mathbf{v})\right] - m \cdot E_{P_n}\left[\log(1 - h(i, \mathbf{v}'))\right].$$

P_d is the actual data distribution while P_n is the sampled sampled noise distribution. $\mathbf{v}'$ is the feature of another image.

Noise-Contrastive Estimation (cont.)

Compute $\boldsymbol{Z}_i$ from Eq. 4 is expensive. It is approximated via Monte Carlo:

$$\boldsymbol{Z} \simeq \boldsymbol{Z}_i \simeq nE_j \left[\exp\left(\mathbf{v}_j^\top \mathbf{f}_i/\tau\right)\right] = \frac{n}{m}\sum_{k=1}^{m} \exp\left(\mathbf{v}_{j_k}^\top \mathbf{f}_i/\tau\right).$$

With the NCE approximation, the computational complexity reduces from $O(n)$ to $O(1)$ per sample.

Proximal Regularization

During each iteration only one instance per class will be seen, and this causes fluctuation. An additional regularization term $\lambda \left\| \mathbf{v}_i^{(t)} - \mathbf{v}_i^{(t-1)} \right\|$ is added for each positive sample.

The final objective function is:

$$J_{NCE}(\boldsymbol{\theta}) = -E_{P_d}\left[\log h(i, \mathbf{v}) - \lambda \left\| \mathbf{v}_i^{(t)} - \mathbf{v}_i^{(t-1)} \right\|\right] - m \cdot E_{P_n}\left[\log(1 - h(i, \mathbf{v}'))\right].$$

Table of Contents

InsDis

Non-Parametric Softmax Classifier

NCE & Regularization

Image Class Classifier

CMC

Contrastive Learning with Two Views

Contrastive Learning with Multiple Views

Implementation Details

MoCo

Contrastive Learning as Dictionary Look-up

Momentum Contrast

SupCon

Supervised Contrastive Losses

Choice of $\mathcal{L}^{sup}$

Summary

Weighted k-Nearest Neighbor Classifier

To classify test image $\hat{x}$, first compute $\hat{\mathbf{f}} = f_{\boldsymbol{\theta}}(\hat{x})$. Then compare it with all representations in V with cosine similarity and find the k-nearest neighbors $\mathcal{N}_k$. The class with maximum weight from $\mathcal{N}_k$ will be the predicted image class.

Contrastive Multiview Coding [1]

- Problem formation: mutual information maximization.
- Learning objective: a representation that aims to maximize mutual information between different views of the same scene.
- Novelty: contrastive learning with different views.

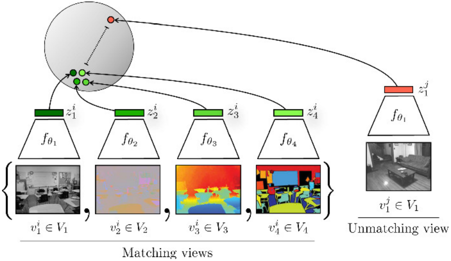


Figure 3: Contrastive multiview coding framework.

Table of Contents

InsDis

- Non-Parametric Softmax Classifier
- NCE & Regularization
- Image Class Classifier

CMC

- Contrastive Learning with Two Views
- Contrastive Learning with Multiple Views
- Implementation Details

MoCo

- Contrastive Learning as Dictionary Look-up
- Momentum Contrast

SupCon

- Supervised Contrastive Losses
- Choice of $\mathcal{L}^{sup}$

Summary

Contrast Between Two Views

The idea is learn a representation that separates samples from different distributions.

Given two views V_1 and V_2 of a collections of samples $\{v_1^i, v_2^i\}_{i=1}^N$, contrast congruent and incongruent pairs $x \sim p(v_1, v_2)$ (*positives*) and $y \sim p(v_1)p(v_2)$ (*negatives*).

A discrimination function $h_{\boldsymbol{\theta}}(\cdot)$ is trained to give high values for positives and low values for negatives. The contrastive loss is thus defined on a sample set $S = \{x, y_1, y_2, \cdots, y_k\}$ as:

$$\mathcal{L}_{contrast} = -E_S \left[\log \frac{h_{\boldsymbol{\theta}}(x)}{h_{\boldsymbol{\theta}}(x) + \sum_{j=1}^{k} h_{\boldsymbol{\theta}}(y_i)} \right].$$

Sample Set and Discrimination Function

To easily construct the sample set S, one view is fixed and the other enumerates positives and negatives:

$$\mathcal{L}_{contrast}^{V_1,V_2} = -E_{\{v_1, v_2^1, \cdots, v_2^{k+1}\}} \left[\log \frac{h_{\boldsymbol{\theta}}(v_1^1, v_2^1)}{\sum_{j=1}^{k+1} h_{\boldsymbol{\theta}}(v_1^1, v_2^j)} \right]. \tag{5}$$

The discrimination function $h_{\boldsymbol{\theta}}$ is a neural network. Specifically, two encoders $f_{\boldsymbol{\theta}_1}$ and $f_{\boldsymbol{\theta}_1}$ is used.

$$h_{\boldsymbol{\theta}}(\{v_1, v_2\}) = \exp\left(\frac{1}{\tau} \cdot \frac{f_{\boldsymbol{\theta}}(v_1) \cdot f_{\boldsymbol{\theta}}(v_2)}{\|f_{\boldsymbol{\theta}}(v_1)\| \cdot \|f_{\boldsymbol{\theta}}(v_2)\|} \right).$$

Two-View Loss

Symmetrically, $\mathcal{L}_{contrast}^{V_2,V_1}$ can be derived from Eq. 5.

Adding $\mathcal{L}_{contrast}^{V_1,V_2}$ and $\mathcal{L}_{contrast}^{V_2,V_1}$, the two-view contrastive loss is:

$$\mathcal{L}(V_1, V_2) = \mathcal{L}_{contrast}^{V_1,V_2} + \mathcal{L}_{contrast}^{V_2,V_1}. \tag{6}$$

Table of Contents

InsDis

- Non-Parametric Softmax Classifier
- NCE & Regularization
- Image Class Classifier

CMC

- Contrastive Learning with Two Views
- Contrastive Learning with Multiple Views
- Implementation Details

MoCo

- Contrastive Learning as Dictionary Look-up
- Momentum Contrast

SupCon

- Supervised Contrastive Losses
- Choice of $\mathcal{L}^{sup}$

Summary

Contrast Between More Than Two Views

Suppose there are M views $V_1, V_2, \cdots, V_M$. The "core" view is the one to optimize. There are two paradigms that can be used.

- *Pair-wise*: $\mathcal{L}_C = \sum_{j=2}^{M} \mathcal{L}(V_1, V_j)$.
- *Full grap*: $\mathcal{L}_F = \sum_{1 \leq i \leq M} \mathcal{L}(V_i, V_j)$.

Table of Contents

InsDis

Non-Parametric Softmax Classifier

NCE & Regularization

Image Class Classifier

CMC

Contrastive Learning with Two Views

Contrastive Learning with Multiple Views

Implementation Details

MoCo

Contrastive Learning as Dictionary Look-up

Momentum Contrast

SupCon

Supervised Contrastive Losses

Choice of $\mathcal{L}^{sup}$

Summary

Negative-sampling & Memory Bank

In the last model, noise-contrast estimation is used to reduce number of classes. CMC uses *negative sampling* and formulate contrastive loss as a $(m+1)$-way softmax classification. The idea of memory bank is also adopted here and it is dynamically updated.

Momentum Contrast [3]

- Problem formation: instance-level classification.
- Learning objective: moving-averaged encoder.
- Novelty: build a large and consistent dictionary on-the-fly that facilitates unsupervised contrastive learning.

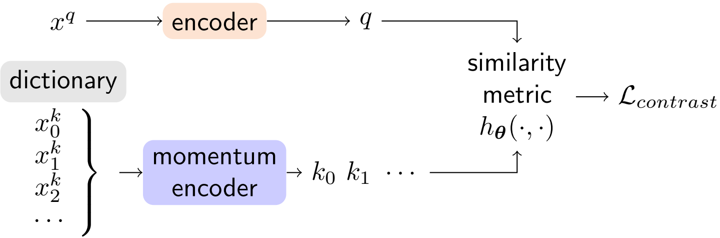


Figure 4: Contrastive as dictionary look-up. The optimization goal is retrieve the positive key(s) from the dictionary.

Table of Contents

InsDis

Non-Parametric Softmax Classifier

NCE & Regularization

Image Class Classifier

CMC

Contrastive Learning with Two Views

Contrastive Learning with Multiple Views

Implementation Details

MoCo

Contrastive Learning as Dictionary Look-up

Momentum Contrast

SupCon

Supervised Contrastive Losses

Choice of $\mathcal{L}^{sup}$

Summary

InfoNCE

Consider an encoded query q and a set of encoded samples (*keys*) $\{k_0, k_1, \cdots\}$. MoCo use the following loss.

InfoNCE

$$\mathcal{L}_q = -\log \frac{\exp(q \cdot k_+/\tau)}{\sum_{j=0}^{K} \exp(q \cdot k_j/\tau)} \tag{7}$$

Intuitively, $\mathcal{L}_q$ is a $K+1$-way softmax classification loss that classifies q as k_+. Two identical or distinct encoders $f_{\boldsymbol{\theta}_q}$ and $f_{\boldsymbol{\theta}_k}$ are used to encode q and k's, respectively.

Table of Contents

InsDis

- Non-Parametric Softmax Classifier
- NCE & Regularization
- Image Class Classifier

CMC

- Contrastive Learning with Two Views
- Contrastive Learning with Multiple Views
- Implementation Details

MoCo

- Contrastive Learning as Dictionary Look-up
- Momentum Contrast

SupCon

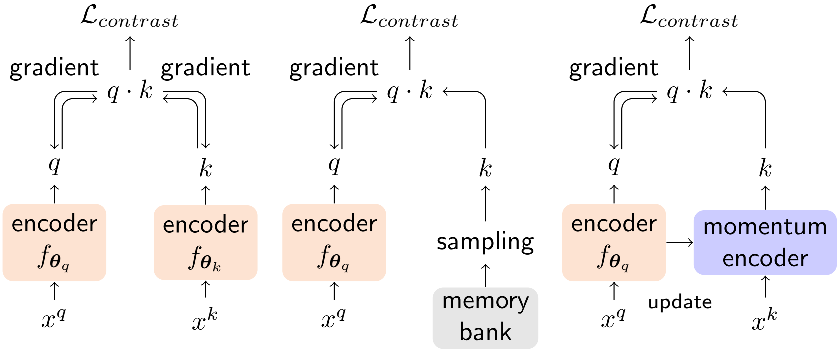
- Supervised Contrastive Losses
- Choice of $\mathcal{L}^{sup}$

Summary

Momentum Contrast

- *Dictionary as a queue*: maintains dictionary as a queue (first in, first out), decouples its size from batch size, and progressively updated with new batches and iterations.
- *Momentum update*: By maintaining a queue, the encoder $f_{\boldsymbol{\theta}_k}$ can not be updated via back-propagation. Momentum update is used instead to update it consistently: $\boldsymbol{\theta}_k \leftarrow m \cdot \boldsymbol{\theta}_k + (1-m) \cdot \boldsymbol{\theta}_q$.

Comparison



(a) end-to-end

(b) memory bank

instance-only: InsDic

multiview: CMC

(c) MoCo

Supervised Contrastive Learning [4]

- Problem formation: supervised image classification
- Learning objective: a representation from supervised class labels.
- Novelty: extend self-supervised contrast to fully-supervised scenarios.

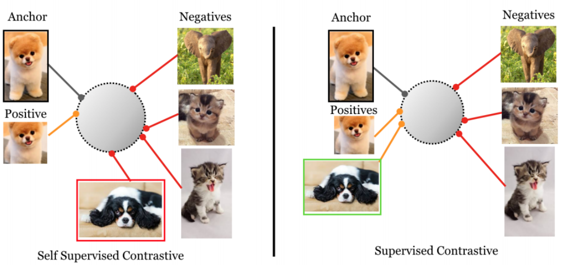


Figure 5: Self-supervised v.s. Supervised Contrast.

Table of Contents

InsDis

Non-Parametric Softmax Classifier

NCE & Regularization

Image Class Classifier

CMC

Contrastive Learning with Two Views

Contrastive Learning with Multiple Views

Implementation Details

MoCo

Contrastive Learning as Dictionary Look-up

Momentum Contrast

SupCon

Supervised Contrastive Losses

Choice of $\mathcal{L}^{sup}$

Summary

Supervised Contrastive Losses

Self supervised contrastive losses 2, 6 and 7 do not consider additional label information. To include it, two kinds of supervised contrastive losses are proposed:

$$\mathcal{L}_{out}^{sup} = \sum_{i \in I} \mathcal{L}_{out,i}^{sup} = \sum_{i \in I} -\frac{1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_p / \tau\right)}{\sum_{a \in A(i)} \exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_a / \tau\right)}$$

$$\mathcal{L}_{in}^{sup} = \sum_{i \in I} \mathcal{L}_{in,i}^{sup} = \sum_{i \in I} -\log\left\{\frac{1}{|P(i)|} \sum_{p \in P(i)} \frac{\exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_p / \tau\right)}{\sum_{a \in A(i)} \exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_a / \tau\right)}\right\}$$

- $A(i)$ is all the samples except i itself.
- $P(i)$ is all the samples in $A(i)$ that has the same class label as i.

Properties of Supervised Contrastive Losses

- Generalize to an arbitrary number of positives.
- Capability increases with more negatives.
- Ability to perform hard positive/negative mining.

Table of Contents

InsDis

Non-Parametric Softmax Classifier

NCE & Regularization

Image Class Classifier

CMC

Contrastive Learning with Two Views

Contrastive Learning with Multiple Views

Implementation Details

MoCo

Contrastive Learning as Dictionary Look-up

Momentum Contrast

SupCon

Supervised Contrastive Losses

Choice of $\mathcal{L}^{sup}$

Summary

Gradient Analysis

The gradients for both $\mathcal{L}^{sup}_{out,i}$ and $\mathcal{L}^{sup}_{in,i}$ w.r.t. $\boldsymbol{z}_i$ have the same form:

$$\frac{\partial \mathcal{L}^{sup}_{\cdot,i}}{\partial \boldsymbol{z}_i} = \frac{1}{\tau}\left\{\sum_{p\in P(i)} \boldsymbol{z}_p\left(P_{ip} - X_{ip}\right) + \sum_{n\in N(i)} \boldsymbol{z}_n P_{in}\right\}$$

- $N(i)$ is the complement of $P(i)$ w.r.t. $A(i)$.
- $P(ix) \equiv \dfrac{\exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_x/\tau\right)}{\sum_{a\in A(i)} \exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_a/\tau\right)},\ x \in \{p, n\}$.
- $X_{ip} = \begin{cases} \dfrac{\exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_p/\tau\right)}{\sum_{p'\in P(i)} \exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_{p'}/\tau\right)}, & \text{if } \mathcal{L}^{sup}_{\cdot,i} = \mathcal{L}^{sup}_{in,i} \\ \dfrac{1}{|P(i)|}, & \text{if } \mathcal{L}^{sup}_{\cdot,i} = \mathcal{L}^{sup}_{out,i} \end{cases}$.

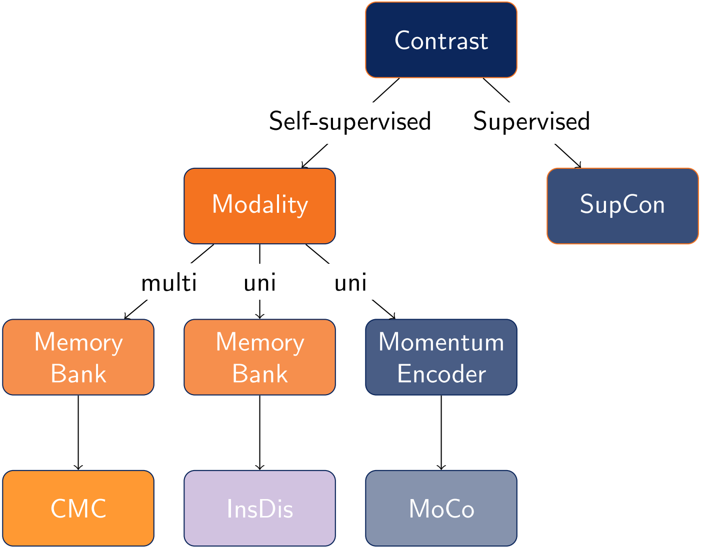
Stabilization

If $\boldsymbol{z}_p$ is substituted with the less biased mean representation $\bar{\boldsymbol{z}}$, X_{ip}^{in} will reduce to X_{ip}^{out}. This will lead to more stability in training and reduce impact of a single positive sample.

Supervised contrastive loss $\mathcal{L}^{sup}$

$$\mathcal{L}_{out}^{sup} = \sum_{i \in I} \mathcal{L}_{out,i}^{sup} = \sum_{i \in I} - \frac{1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_p / \tau)}{\sum_{a \in A(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_a / \tau)}$$

Contrastive Cookbook



Thank You for Your Attention

Q & A

References

[1] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XI*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds., ser. Lecture Notes in Computer Science, vol. 12356, Springer, 2020, pp. 776–794. DOI: 10.1007/978-3-030-58621-8_45. [Online]. Available: https://doi.org/10.1007/978-3-030-58621-8%5C_45.

[2] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, IEEE Computer Society, 2018, pp. 3733–3742. DOI: 10.1109/CVPR.2018.00393. [Online]. Available: http://openaccess.thecvf.com/content%5C_cvpr%5C_2018/html/Wu%5C_Unsupervised%5C_Feature%5C_Learning%5C_CVPR%5C_2018%5C_paper.html.

References (cont.)

[3] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick, "Momentum contrast for unsupervised visual representation learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, IEEE, 2020, pp. 9726–9735. DOI: 10.1109/CVPR42600.2020.00975. [Online]. Available: https://doi.org/10.1109/CVPR42600.2020.00975.

[4] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M.-F. Balcan, and H.-T. Lin, Eds., 2020. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/d89a66c7c80a29b1bdbab0f2a1a94af8-Abstract.html.