Supervised Contrastive Learning

Prannay Khosla
Google Research

Piotr Teterwak
Boston University

Chen Wang
Snap Inc.

Aaron Sarna
Google Research

Yonglong Tian
MIT

Phillip Isola
MIT

Aaron Maschinot
Google Research

Ce Liu
Google Research

Dilip Krishnan
Google Research

Presented by: Andrei Arnăutu

Overview

1. Contrastive Learning: introduction
2. Previous work
3. Supervised contrastive learning
    a. Loss objective and desirable properties
    b. Connection to previous work
4. Results
5. Robustness and stability
6. Future work & other applications
7. Personal conclusions
8. Q&A section

What is Contrastive Learning?

- Deep learning technique for supervised or self-supervised low-dimensional representation learning.
- Main components:
  - Positive and Negative samples
  - Loss objective
  - Data augmentation
- Goal:
  - Clusters of similar points are pulled together in the low-dimensional representation. Dissimilar points are pushed apart.
  - Invariance to certain transformations.

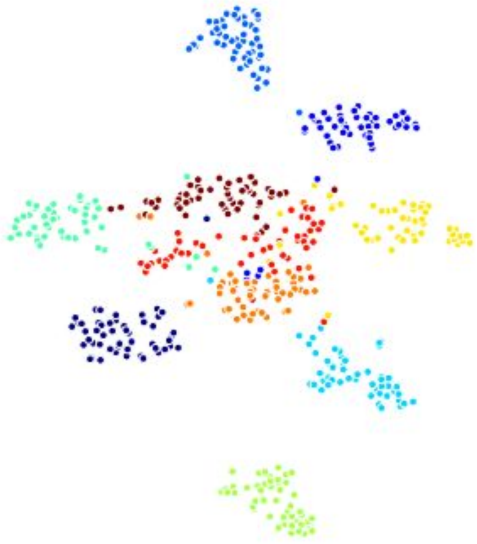
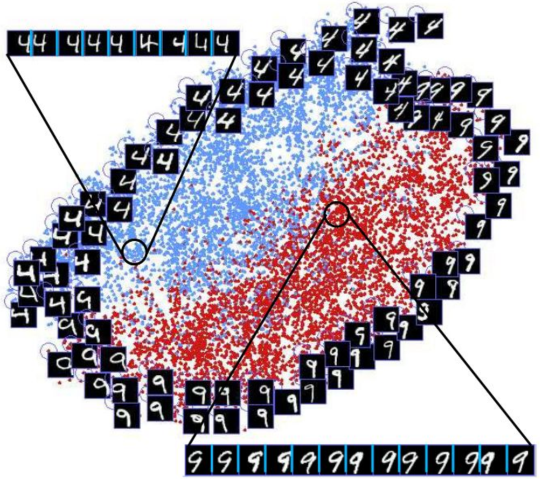


Image from: T. Chen, S. Kornblith, M. Norouzi, G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations", ICML 2020

Positive & Negative samples

- <u>Assumption</u>
  - For each training sample there is a set of other training samples that are are deemed “similar”.
  - This set can be computed via prior knowledge, such as invariance to image distortions.
- <u>Objective</u>
  - A meaningful high to low dimensional mapping <u>maps similar input vectors to nearby points</u> in the feature space and <u>dissimilar input vectors to distant points</u>.

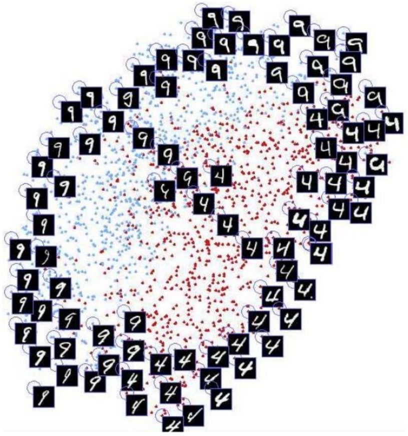


$$L(W, Y, \vec{X}_1, \vec{X}_2) =$$
$$(1-Y)\frac{1}{2}(D_W)^2 + (Y)\frac{1}{2}\{max(0, m - D_W)\}^2$$

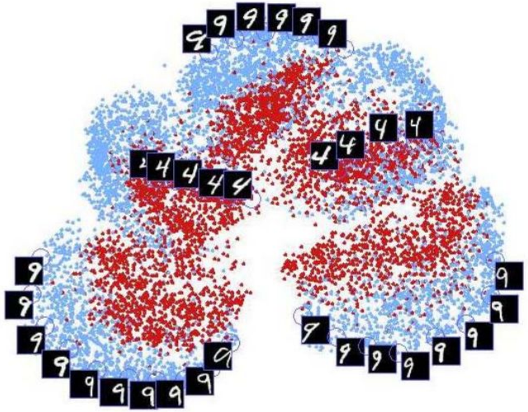
Image and formula from: R. Hadsell, S. Chopra and Y. LeCun, "Dimensionality Reduction by Learning an Invariant Mapping", CVPR 2006

Positive samples via data augmentations

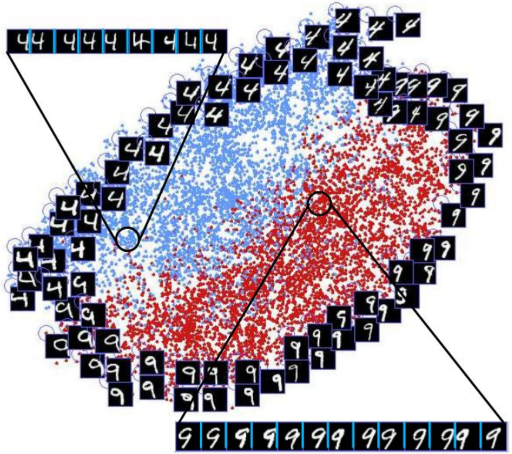
Experiment 1
Similar points: Top 5 NN in image space



Experiment 2
Dataset augmentation: horizontally shifted images
Similar points: Top 5 NN in image space



Experiment 3
Dataset augmentation: horizontally shifted images
Similar points: Top 5 NN in image space + all of the sample's augmentations

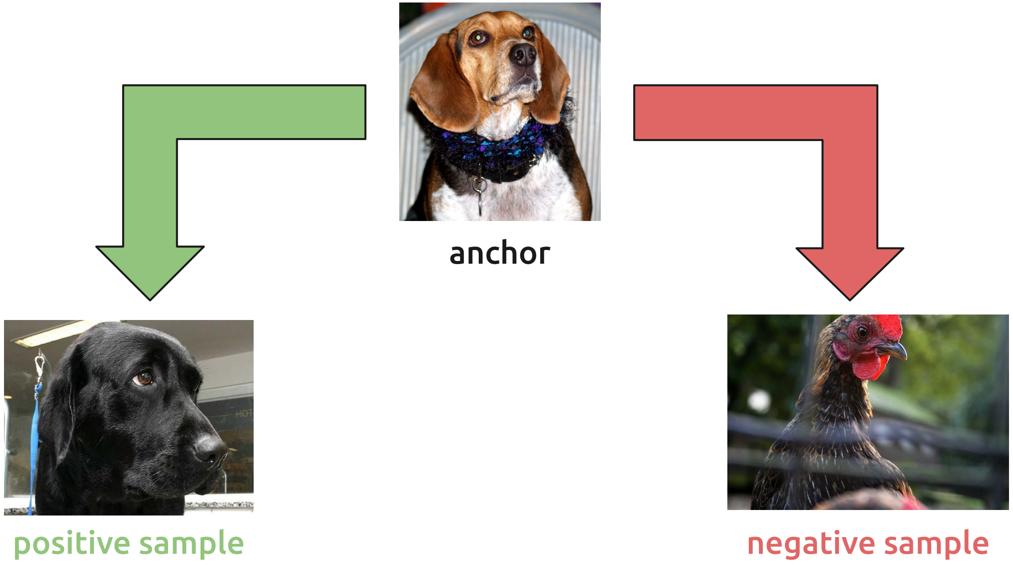


Images taken from: R. Hadsell, S. Chopra and Y. LeCun, "Dimensionality Reduction by Learning an Invariant Mapping", CVPR 2006

Triplet loss (supervised)

- How can we now define the negative samples?

- The triplet loss provides an extension to the previous idea by selecting a positive sample and a negative sample for each anchor.



anchor

positive sample

negative sample

The original images are taken from ILSVRC2012

Triplet loss (supervised)

- **Main idea**
  - For each sample, we want to construct a triplet by selecting a positive sample (from the same class) and a negative sample (from a different class).
- **Loss formulation**

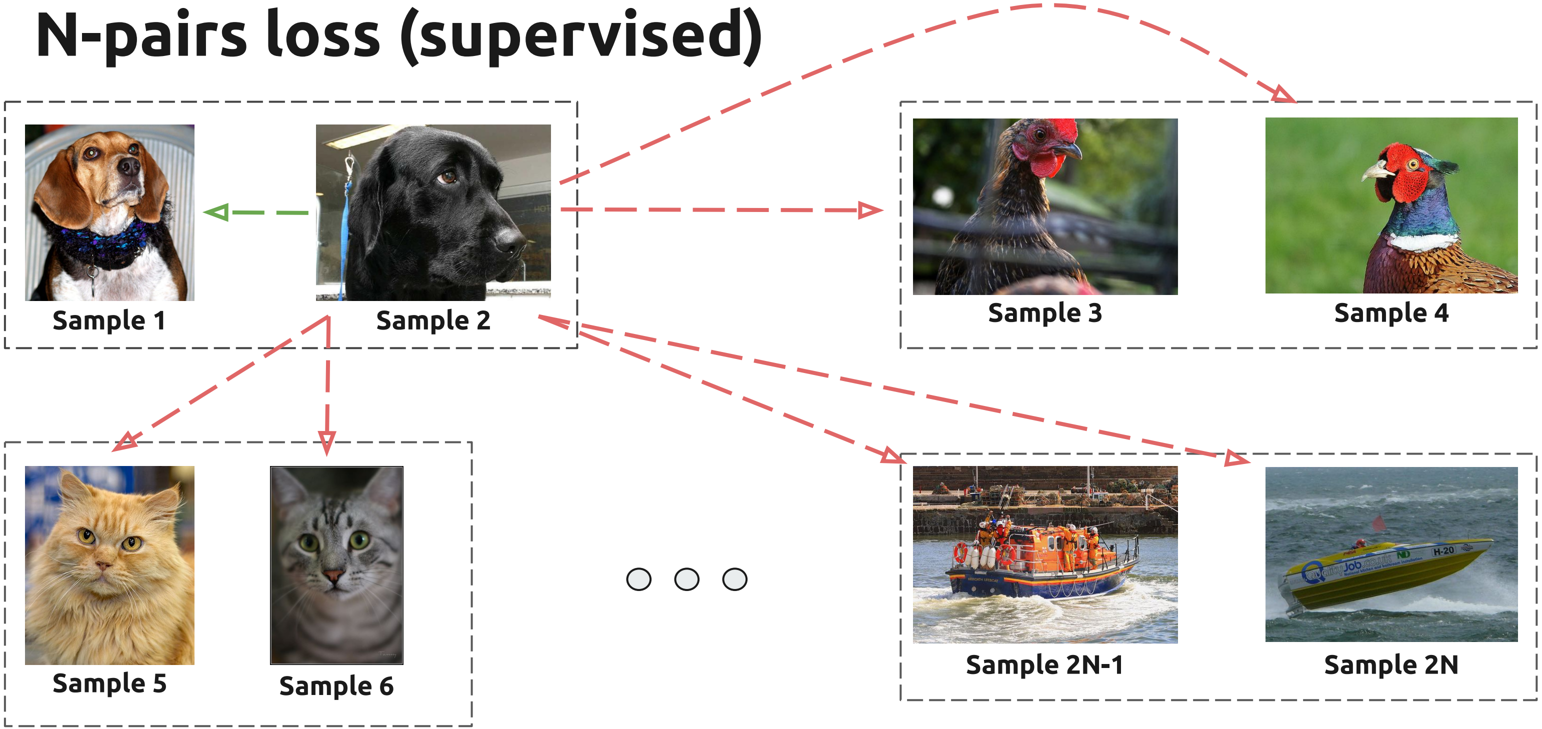
$$\mathcal{L}_{triplet} = \sum_{i=1}^{N} \sum_{\substack{p \in P(i) \\ n \in N(i)}} \max\left(0, \|z_i - z_p\|^2 - \|z_i - z_n\|^2 + \alpha\right)$$

- **Problems**
  - We cannot afford to iterate through all possible pairs of positives and negatives due to computational costs.
  - Even the optimized hard positive and negative mining algorithms are computationally expensive!

Hard positive & negative mining

- For a given anchor, the hard positives and hard negatives are defined in the following way:
  - **Hard positives:** samples that are supposed to be similar to the anchor, but the similarity value between their learned representations is low.
  - **Hard negatives:** samples that are supposed to be very dissimilar, but the similarity value between their learned representations is high.
- Some hard positive and negative mining ideas:
  - **Batch mining**
  - **Online Hard Example Mining (OHEM)**
  - **Distance Weighted Sampling**

N-pairs loss (supervised)



Sample 1

Sample 2

Sample 3

Sample 4

Sample 5

Sample 6

○ ○ ○

Sample 2N-1

Sample 2N

The original images are taken from ILSVRC2012

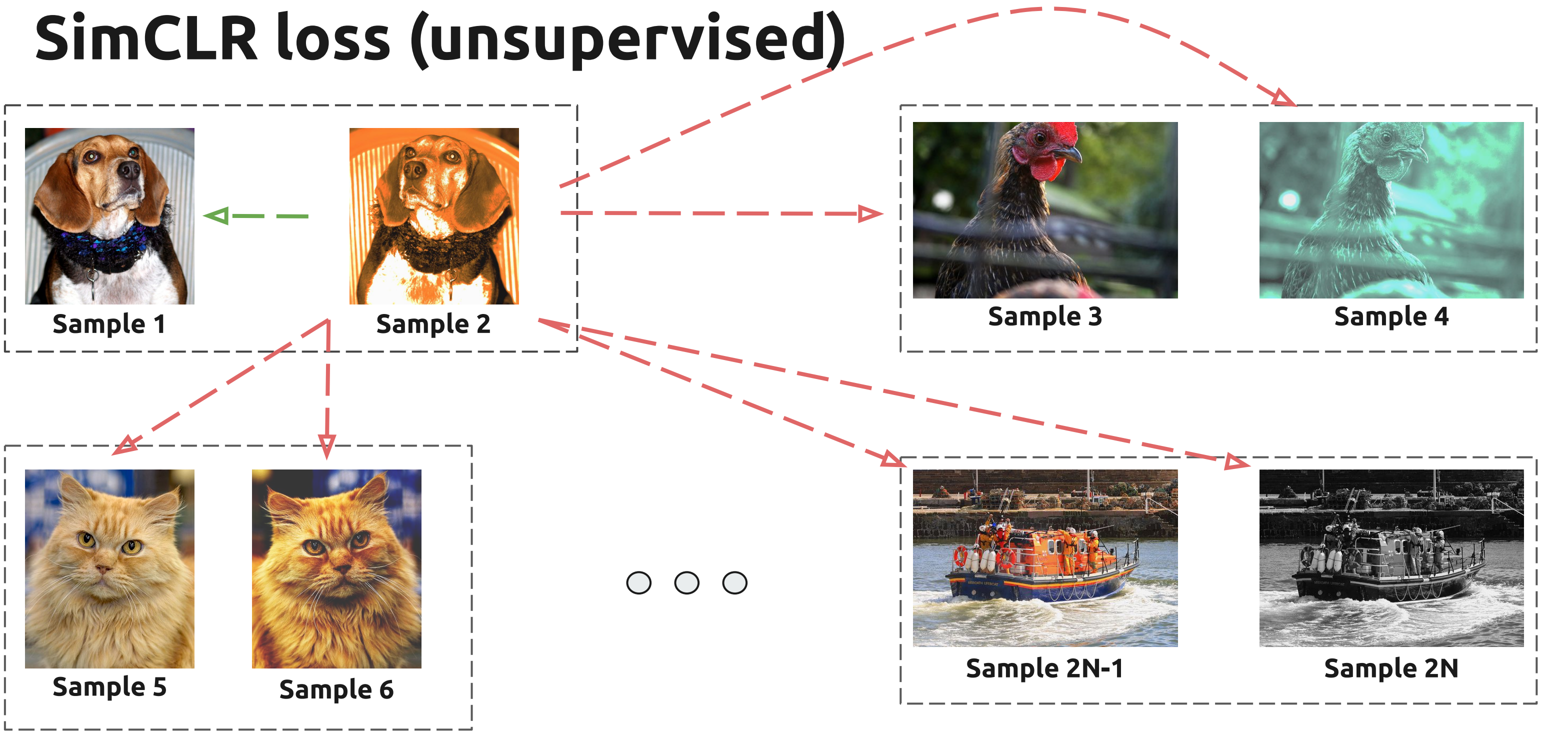
N-pairs loss (supervised)

- **Main idea**
  - Extending the triplet loss to be able to use an arbitrarily large number of negatives.
  - Each batch contains **N pairs of samples**, where each pair contains 2 samples from the same class.
  - **Use the remaining (N - 1) pairs as negative samples.**
- **Loss formulation**

$$\mathcal{L}^{n\text{-}pairs} = -\sum_{i\in I}\log\frac{\exp\left(\boldsymbol{z}_i\cdot\boldsymbol{z}_{k(i)}\right)}{\sum_{a\in A(i)}\exp\left(\boldsymbol{z}_i\cdot\boldsymbol{z}_a\right)}$$

- **Problems**
  - Makes use of only one positive sample.
  - No data augmentations.

SimCLR loss (unsupervised)



The original images are taken from ILSVRC2012

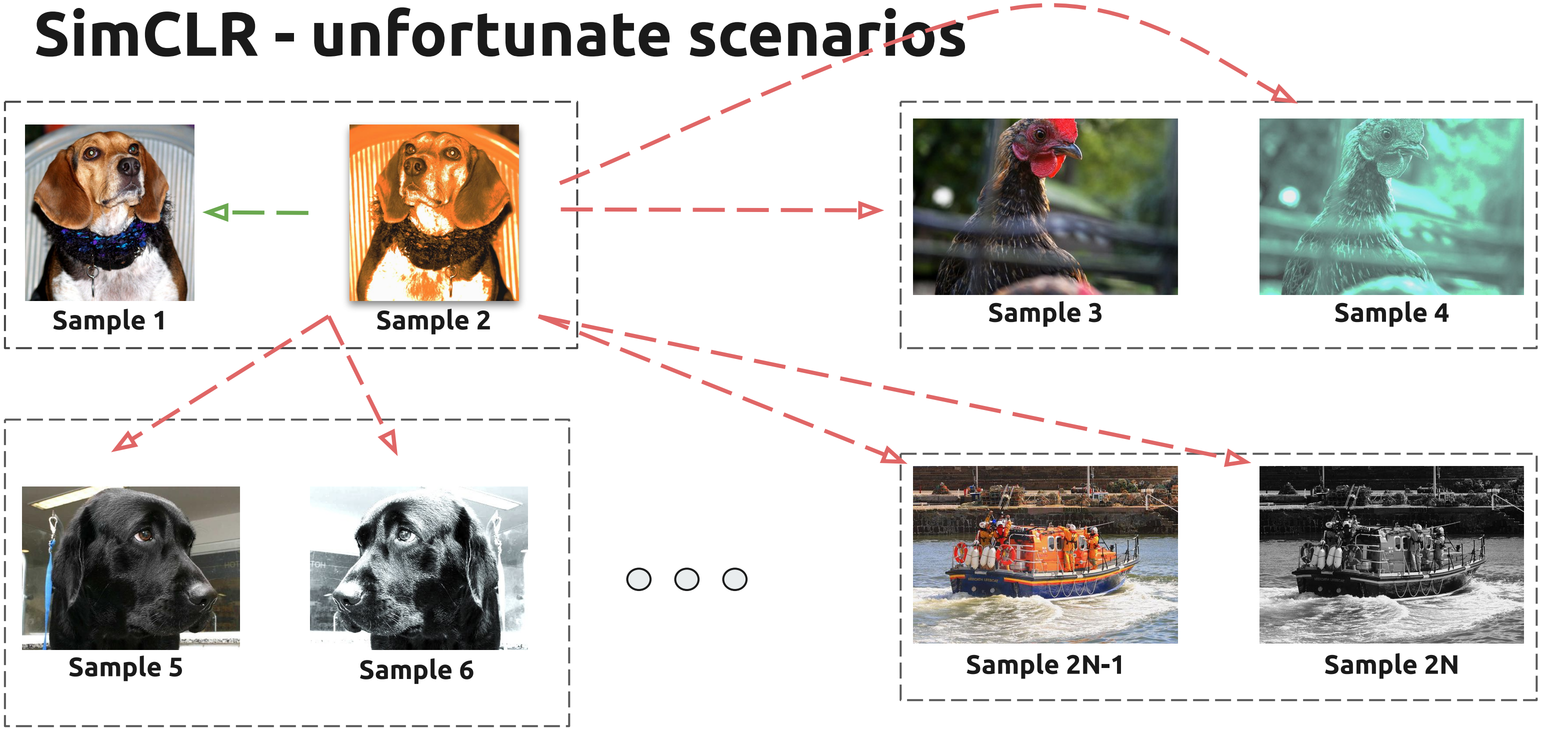
SimCLR (unsupervised)

- Very similar objective to N-pairs loss
  - Employs and **highlights the importance of using data augmentations** for the positive samples.
  - Adds a **temperature parameter** to the loss function.

$$\mathcal{L}_{\text{SimCLR}} = -\sum_{i \in I} \log \frac{\exp\left(\frac{\mathbf{z}_i \cdot \mathbf{z}_{k(i)}}{\tau}\right)}{\sum_{a \in A(i)} \exp\left(\frac{\mathbf{z}_i \cdot \mathbf{z}_a}{\tau}\right)}$$

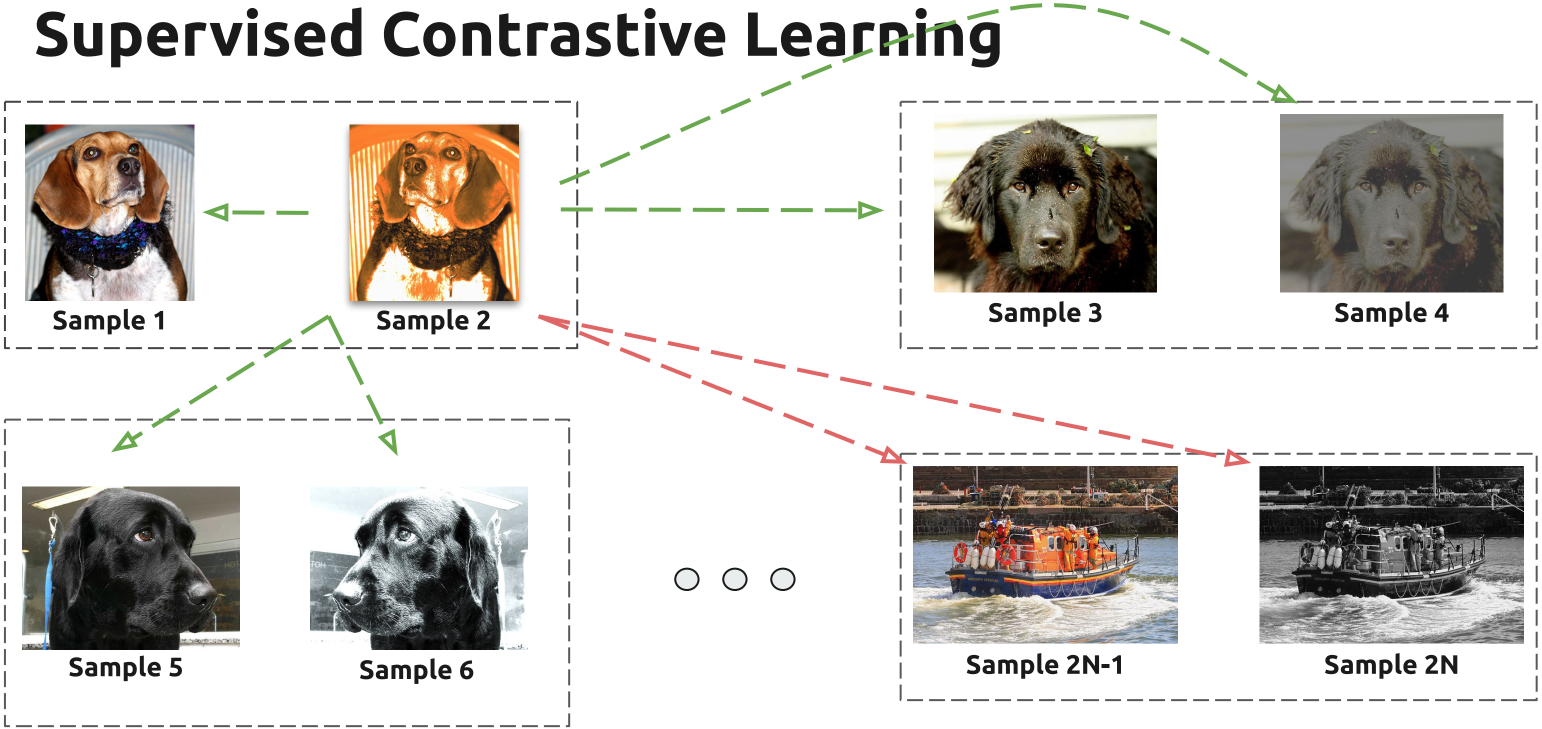
- Contrastive learning strongly benefits from larger training batch sizes
  - Empirical proof that no negative hard mining is needed.

SimCLR - unfortunate scenarios



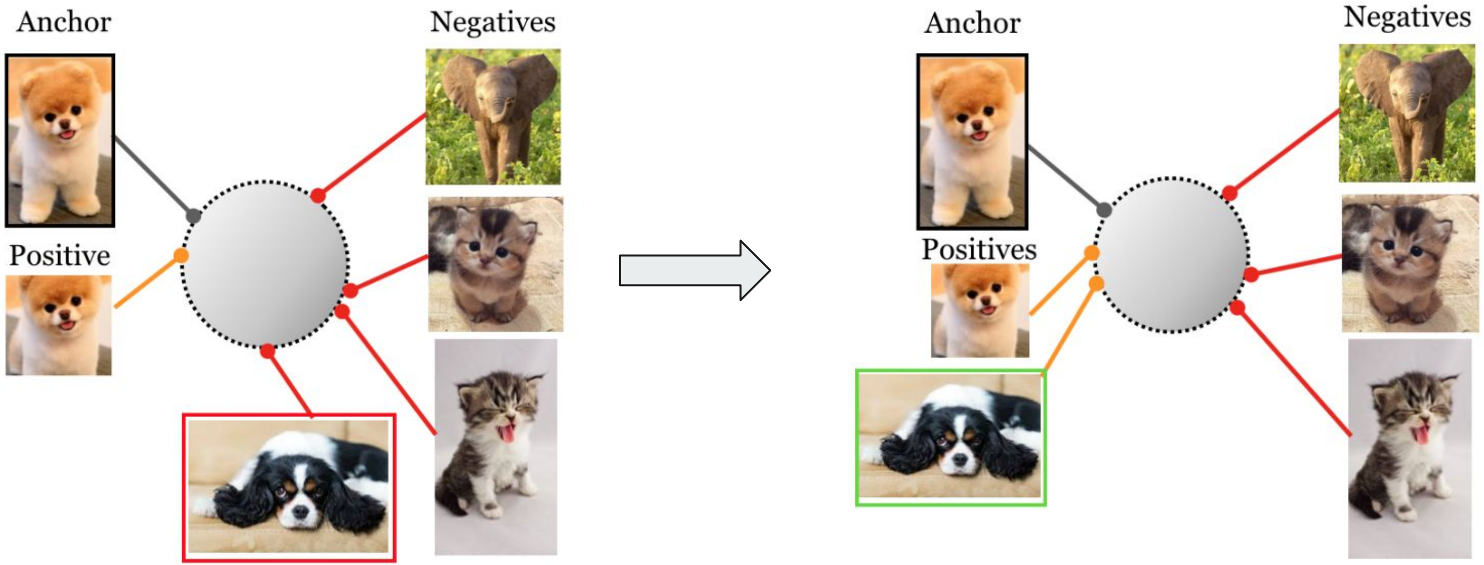
The original images are taken from ILSVRC2012

Supervised Contrastive Learning



The original images are taken from ILSVRC2012

Supervised Contrastive Learning



Self Supervised Contrastive

Supervised Contrastive

Image from: Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. “Supervised Contrastive Learning”, NeurIPS 2020

Paper contributions

1. **Performance boost over the Cross Entropy loss for downstream classification tasks.**
2. Extending contrastive loss: multiple positives per anchor.
3. Analytical proof that the gradient of the loss function performs implicit hard negative mining.
4. Robustness to image corruption
5. Less sensitive to hyperparameter changes compared to the Cross Entropy loss.

Contrastive Loss vs Cross Entropy

- Although widely used in practice, the Cross Entropy loss has a few shortcomings, such as:
  - lack of robustness to noisy labels
  - the possibility of poor margins, which leads to a reduced generalization performance

- The authors argue that the Contrastive Loss yields better results and is more stable to:
  - image corruptions
  - hyperparameter changes (types of augmentations and optimizers, learning rate values)

Network Architecture

- The Contrastive Learning architecture puts more emphasis on learning better discriminative features between samples from different classes.

- The classification head does not propagate gradients back to the encoder.

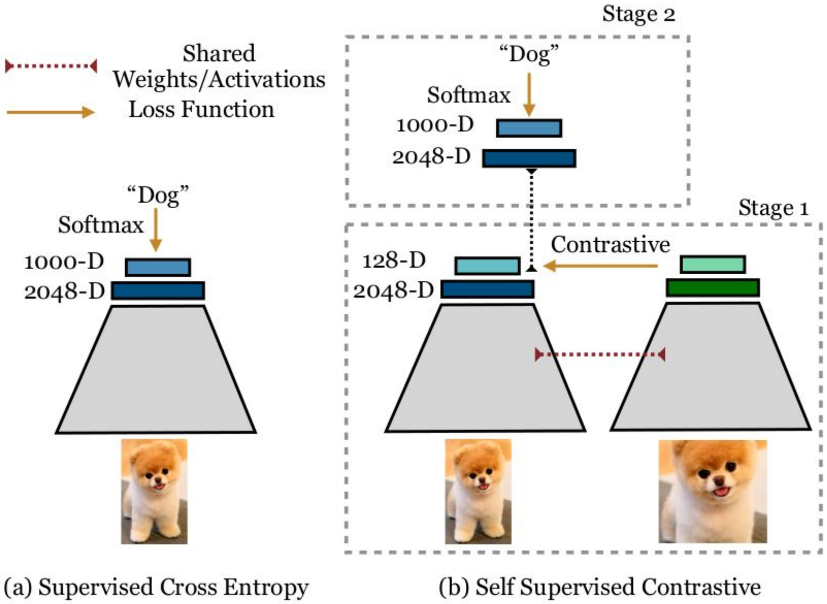


Image from: Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. "Supervised Contrastive Learning", NeurIPS 2020

Network Architecture - why an extra projection layer?

The SimCLR authors conjecture that:

- Using the representation before the projection is due to loss of information induced by the contrastive loss.

- The contrastive representation is trained to be invariant to data transformation. Thus, it can erase some of the information that could be useful for the downstream tasks, such as image color and object orientation.

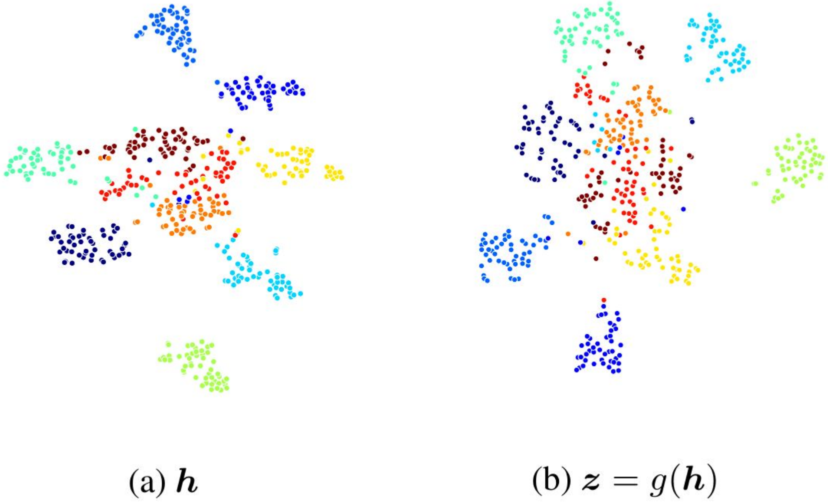
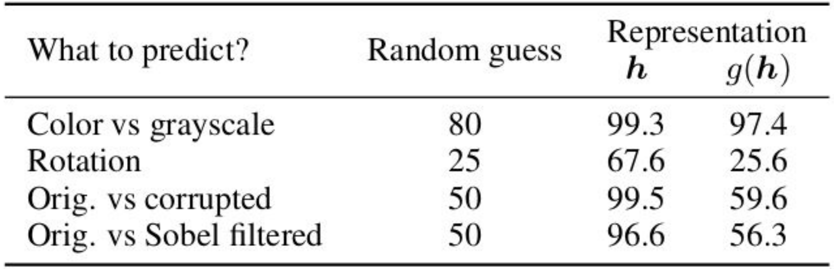
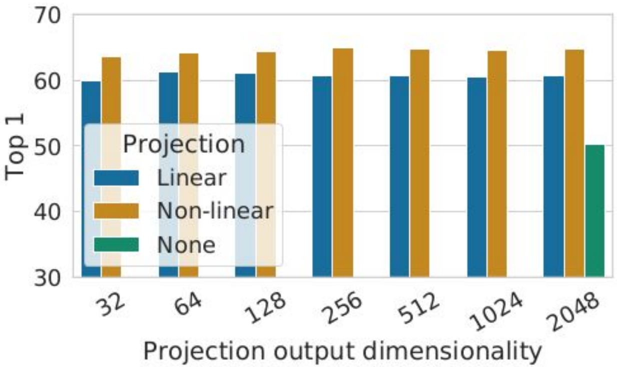


Figure from: T. Chen, S. Kornblith, M. Norouzi, G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations", ICML 2020

Network Architecture - why an extra projection layer?



What to predict?	Random guess	Representation	
		$\boldsymbol{h}$	$g(\boldsymbol{h})$
Color vs grayscale	80	99.3	97.4
Rotation	25	67.6	25.6
Orig. vs corrupted	50	99.5	59.6
Orig. vs Sobel filtered	50	96.6	56.3



Tables from: T. Chen, S. Kornblith, M. Norouzi, G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations", ICML 2020

Paper contributions

1. Performance boost over the Cross Entropy loss for downstream classification tasks.
2. **Extending contrastive loss: multiple positives per anchor.**
3. Analytical proof that the gradient of the loss function performs implicit hard negative mining.
4. Robustness to image corruption
5. Less sensitive to hyperparameter changes compared to the Cross Entropy loss.

Loss objective

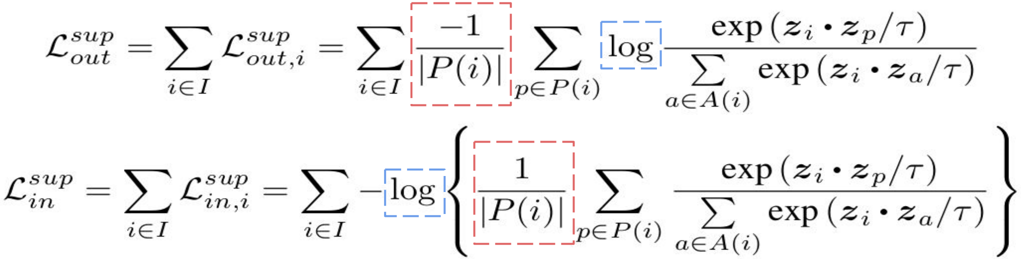
- Generalization to an arbitrarily large number of positives leads to a choice between multiple objective functions.
- The authors propose the “in” and “out” versions (which will be compared later).

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

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

Loss objective: desirable properties

- **Generalization to an arbitrarily large number of positives**
  a. All positives in a multiview batch contribute to the numerator.
  b. For randomly generated batches with size much greater than the number of classes, we will have many positive terms.
  c. The supervised contrastive losses encourage the encoder to give <u>closely aligned representations to all positive samples in a batch</u>.



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

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

Loss objective: desirable properties

- **The contrastive loss is more powerful when we have more negatives**
   a. The ability to discriminate between signal (positives) and noise (negatives) increases.

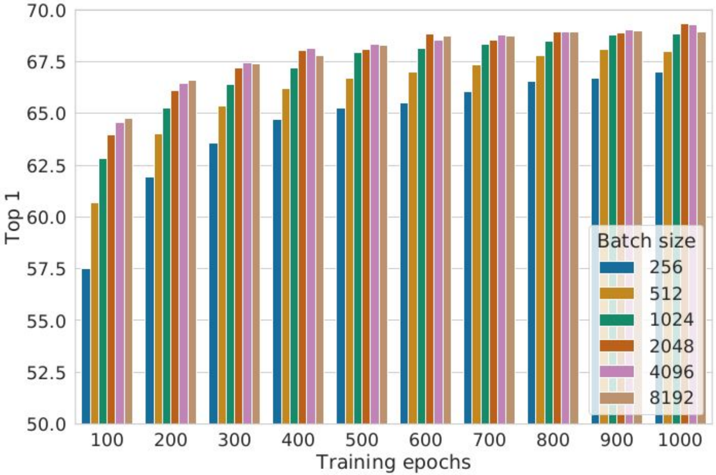


Figure from: T. Chen, S. Kornblith, M. Norouzi, G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations", ICML 2020

Paper contributions

1. Performance boost over the Cross Entropy loss for downstream classification tasks.

2. Extending contrastive loss: multiple positives per anchor.

3. **Analytical proof that the gradient of the loss function performs implicit hard negative mining.**

4. Robustness to image corruption

5. Less sensitive to hyperparameter changes compared to the Cross Entropy loss.

Loss objective: desirable properties

- **Eliminates the need to perform explicit hard positive & negative mining.**
  - Unlike most of the previous methods, we do not have the extra computational cost of searching for hard positive and negative samples.
  - WHY? -> This is due to the gradient structure of the supervised contrastive loss function:
    - Weak Positive/Negative samples —> small gradient contribution
    - Hard Positive/Negative samples —> big gradient contribution

Loss objective: desirable properties

- **Eliminates the need to perform explicit hard positive & negative mining.**

   a. The ability to discriminate between signal (positives) and noise (negatives) increases.

$$\left.\frac{\partial \mathcal{L}_i^{\mathrm{sup}}}{\partial \mathbf{w}_i}\right|_p \propto \left(\mathbf{z}_p - (\mathbf{z}_i \cdot \mathbf{z}_p)\mathbf{z}_i\right)$$

$$\left.\frac{\partial \mathcal{L}_i^{\mathrm{sup}}}{\partial \mathbf{w}_i}\right|_n \propto \left(\mathbf{z}_n - (\mathbf{z}_i \cdot \mathbf{z}_n)\mathbf{z}_i\right)$$

Temperature parameter

- Small vs Large temperature value:
    - Large value: makes the contribution of smaller amplitude gradients more important (smoothens the gradient contribution)
    - Small value: equivalent to optimizing for hard positives/negatives.
    - What works best? —> Authors show that a value of 0.1 (small) yields the best results.

$$\frac{\exp\left(z_i \bullet z_p/\tau\right)}{\sum_{a\in A(i)} \exp\left(z_i \bullet z_a/\tau\right)}$$

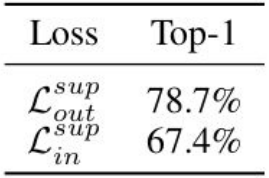
Formula taken from: Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. “Supervised Contrastive Learning”, NeurIPS 2020

Difference between the (in) and (out) losses

- When the normalization factor is **inside the log()**, it only **contributes as an additive constant** in the gradient of the loss.
- When it is **outside of the log()**, it **serves to remove the bias present in the positive samples** in the batch.

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

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

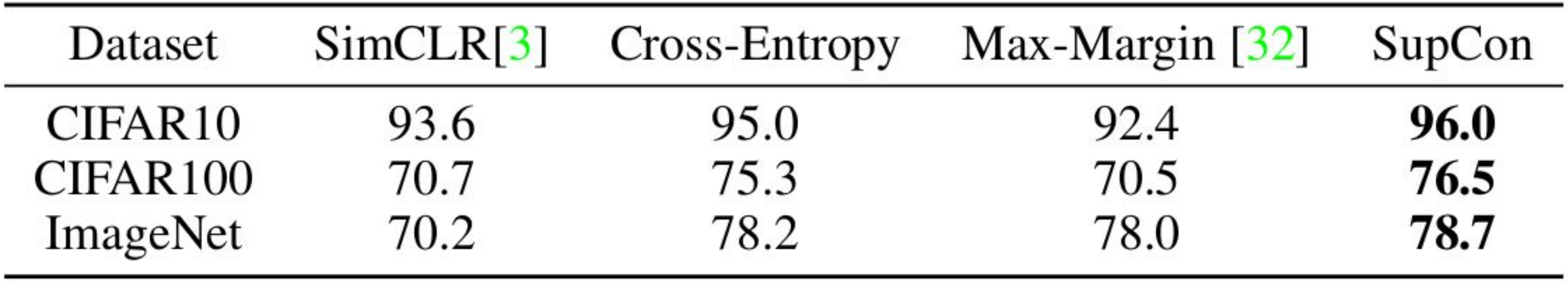


Loss	Top-1
$\mathcal{L}^{sup}_{out}$	78.7%
$\mathcal{L}^{sup}_{in}$	67.4%

Formulas taken from: Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. “Supervised Contrastive Learning”, NeurIPS 2020

Results - classification accuracy

- All methods use the same backbone (ResNet-50). The authors tuned the hyperparameters for each method separately and used the best results for each.
- Main takeaway: the Supervised Contrastive Learning method achieves better results than both Cross-Entropy and the previous best Contrastive Learning approach.

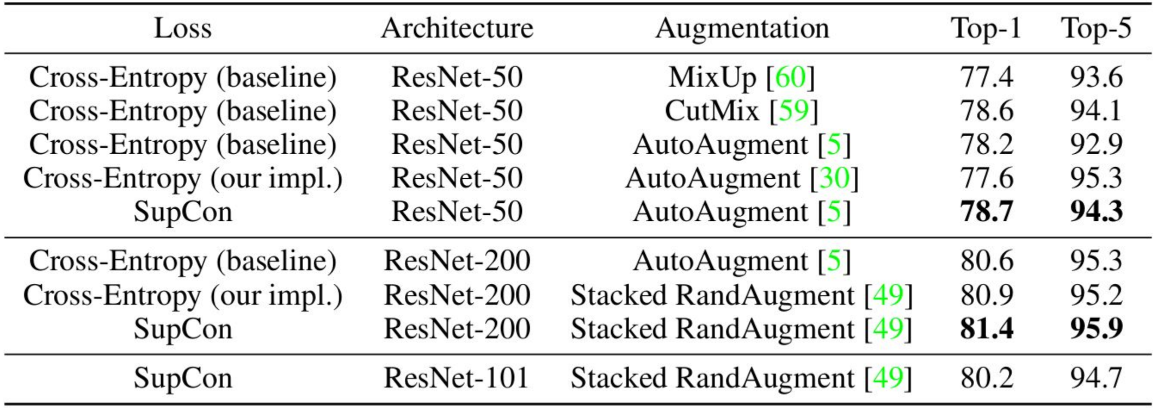


Dataset	SimCLR[3]	Cross-Entropy	Max-Margin [32]	SupCon
CIFAR10	93.6	95.0	92.4	**96.0**
CIFAR100	70.7	75.3	70.5	**76.5**
ImageNet	70.2	78.2	78.0	**78.7**

Table from: Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. “Supervised Contrastive Learning”, NeurIPS 2020

Results - augmentation experiments

- A more capable (deeper) encoder will be able to profit from more complex augmentation processes.
  - Stacked RandAugment performs worse than AutoAugment on ResNet-50.

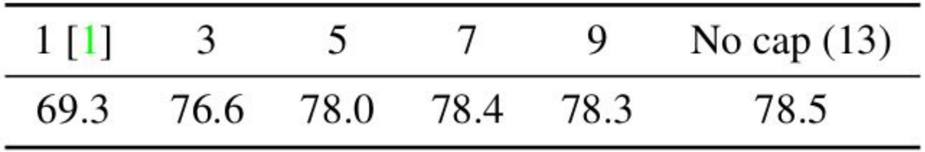


Loss	Architecture	Augmentation	Top-1	Top-5
Cross-Entropy (baseline)	ResNet-50	MixUp [60]	77.4	93.6
Cross-Entropy (baseline)	ResNet-50	CutMix [59]	78.6	94.1
Cross-Entropy (baseline)	ResNet-50	AutoAugment [5]	78.2	92.9
Cross-Entropy (our impl.)	ResNet-50	AutoAugment [30]	77.6	95.3
SupCon	ResNet-50	AutoAugment [5]	**78.7**	**94.3**
Cross-Entropy (baseline)	ResNet-200	AutoAugment [5]	80.6	95.3
Cross-Entropy (our impl.)	ResNet-200	Stacked RandAugment [49]	80.9	95.2
SupCon	ResNet-200	Stacked RandAugment [49]	**81.4**	**95.9**
SupCon	ResNet-101	Stacked RandAugment [49]	80.2	94.7

Table from: Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. “Supervised Contrastive Learning”, NeurIPS 2020

Results - number of positive samples

- Also offers a direct comparison with the self-supervised approach.
- In a similar fashion with the increase of negative examples, the performance gain obtained by using multiple positive samples ends up reaching a plateau region.



1 [1]	3	5	7	9	No cap (13)
69.3	76.6	78.0	78.4	78.3	78.5

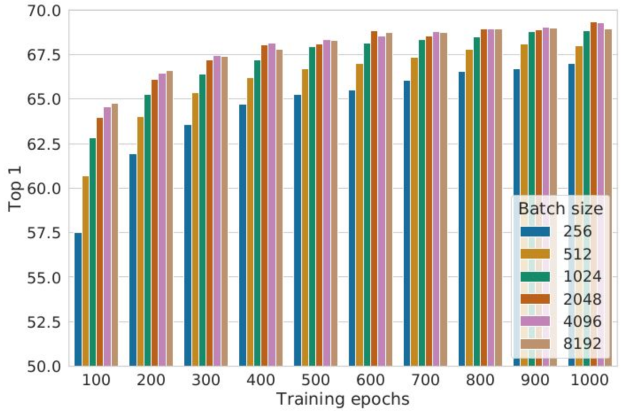
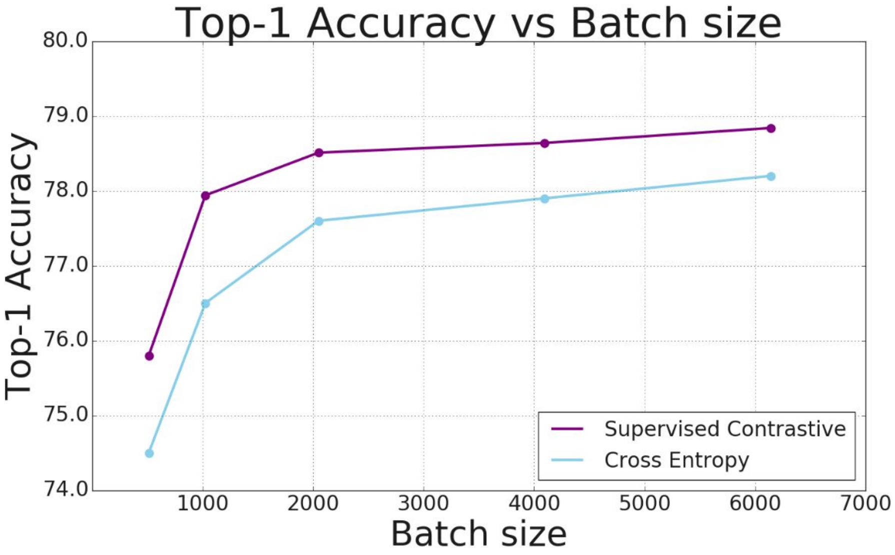
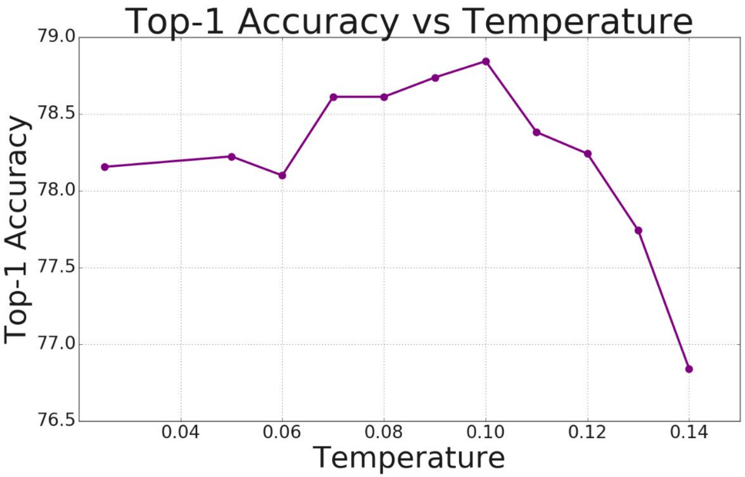


Table from: Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. “Supervised Contrastive Learning”, NeurIPS 2020

Results - batch size and temperature





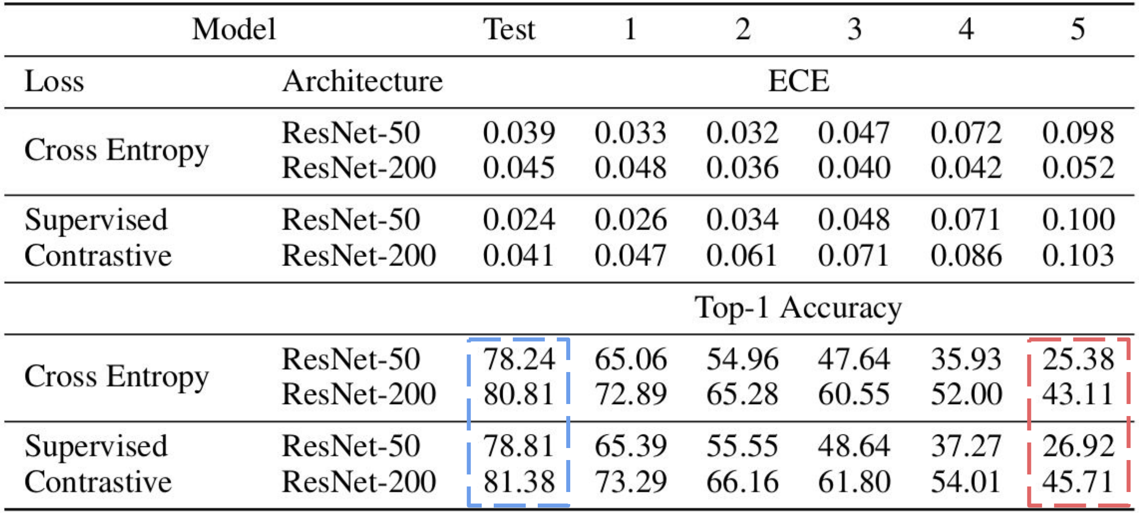
Figures from: Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. “Supervised Contrastive Learning”, NeurIPS 2020

Paper contributions

1. Performance boost over the Cross Entropy loss for downstream classification tasks.
2. Extending contrastive loss: multiple positives per anchor.
3. Analytical proof that the gradient of the loss function performs implicit hard negative mining.
4. **Robustness to image corruption**
5. Less sensitive to hyperparameter changes compared to the Cross Entropy loss.

Results - robustness to corruption

$$ECE = \sum_{m=1}^{M} \frac{|B_m|}{n} |acc(B_m) - conf(B_m)|$$



Model		Test	1	2	3	4	5
Loss	Architecture	ECE					
Cross Entropy	ResNet-50	0.039	0.033	0.032	0.047	0.072	0.098
	ResNet-200	0.045	0.048	0.036	0.040	0.042	0.052
Supervised Contrastive	ResNet-50	0.024	0.026	0.034	0.048	0.071	0.100
	ResNet-200	0.041	0.047	0.061	0.071	0.086	0.103
		Top-1 Accuracy					
Cross Entropy	ResNet-50	78.24	65.06	54.96	47.64	35.93	25.38
	ResNet-200	80.81	72.89	65.28	60.55	52.00	43.11
Supervised Contrastive	ResNet-50	78.81	65.39	55.55	48.64	37.27	26.92
	ResNet-200	81.38	73.29	66.16	61.80	54.01	45.71

Table from: Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. “Supervised Contrastive Learning”, NeurIPS 2020

Paper contributions

1. Performance boost over the Cross Entropy loss for downstream classification tasks.
2. Extending contrastive loss: multiple positives per anchor.
3. Analytical proof that the gradient of the loss function performs implicit hard negative mining.
4. Robustness to image corruption
5. **Less sensitive to hyperparameter changes compared to the Cross Entropy loss.**

Results - hyperparameter stability

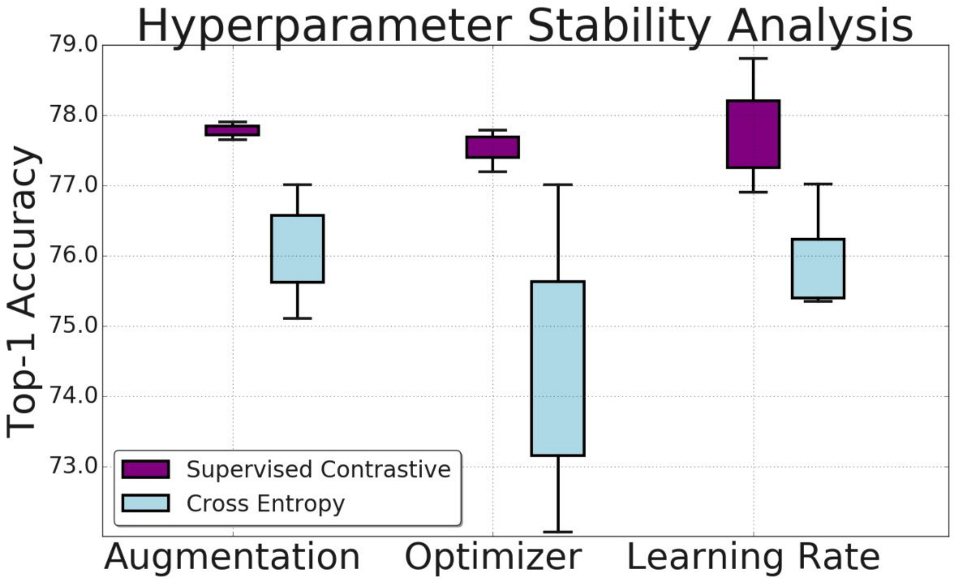
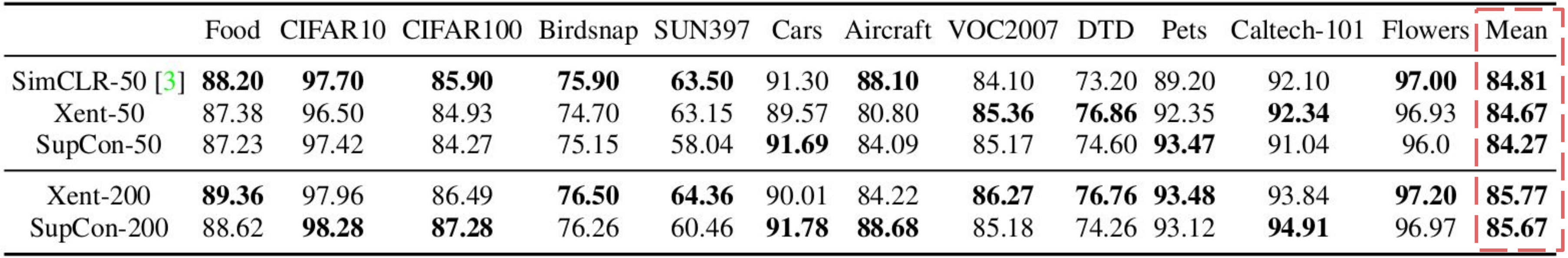


Figure from: Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. “Supervised Contrastive Learning”, NeurIPS 2020

Results - Transfer Learning

- We observe that all of the loss objectives obtain very similar results.
- The authors state that when it comes to transfer learning, the capability of the encoder seems to play a much more important role than the loss objective. However, they decided to leave the connection between the loss objective and the transfer learning capabilities for future work.



	Food	CIFAR10	CIFAR100	Birdsnap	SUN397	Cars	Aircraft	VOC2007	DTD	Pets	Caltech-101	Flowers	Mean
SimCLR-50 [3]	**88.20**	**97.70**	**85.90**	**75.90**	**63.50**	91.30	**88.10**	84.10	73.20	89.20	92.10	**97.00**	**84.81**
Xent-50	87.38	96.50	84.93	74.70	63.15	89.57	80.80	**85.36**	**76.86**	92.35	**92.34**	96.93	**84.67**
SupCon-50	87.23	97.42	84.27	75.15	58.04	**91.69**	84.09	85.17	74.60	**93.47**	91.04	96.0	**84.27**
Xent-200	**89.36**	97.96	86.49	**76.50**	**64.36**	90.01	84.22	**86.27**	**76.76**	**93.48**	93.84	**97.20**	**85.77**
SupCon-200	88.62	**98.28**	**87.28**	76.26	60.46	**91.78**	**88.68**	85.18	74.26	93.12	**94.91**	96.97	**85.67**

Table from: Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. “Supervised Contrastive Learning”, NeurIPS 2020

Future related work

- X. Chen, Y. Liu, Y. Dong et al. “**MobRecon: Mobile-Friendly Hand Mesh Reconstruction from Monocular Image**”
- H. Cha, J. Lee, J. Shin “**Co^{2}L: Contrastive Continual Learning**”
- H. Wang, Y. Zhu, et al. “**MaX-DeepLab: End-to-End Panoptic Segmentation with Mask Transformers**”
- D. Dwibedi, Y. Aytar, et al. “**With a Little Help from My Friends: Nearest-Neighbor Contrastive Learning of Visual Representations**”

Contrastive Continual Learning

- Paper: H. Cha, J. Lee, J. Shin “Co^{2}L: Contrastive Continual Learning”
- Task-Incremental-Learning
  - Example: first distribution: apples vs oranges; second distribution: cats vs dogs
  - In the end, we want the model to be able to perform both types of classifications.
- Domain-Incremental-Learning
  - Example: first distribution: letters written with Font#1; second distribution: letters written with Font#2
- Class-Incremental-Learning
  - Example: first distribution: apples vs oranges; second distribution: cats vs dogs
  - In the end, we want the model to be able to distinguish between all classes!

The power of positive samples

- Paper: X. Chen, Y. Liu, Y. Dong et al. “MobRecon: Mobile-Friendly Hand Mesh Reconstruction from Monocular Image”
- The positive samples can be used to create consistency between different views of the same object.

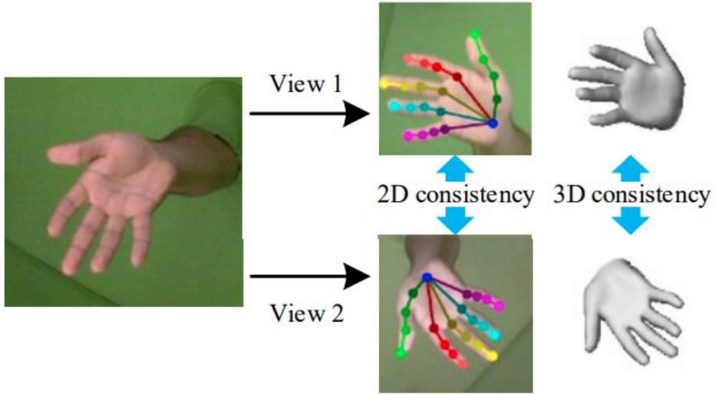


Figure from: X. Chen, Y. Liu, Y. Dong et al. “MobRecon: Mobile-Friendly Hand Mesh Reconstruction from Monocular Image”

Personal conclusions

- **Positive:**
  - Very well structured
  - Comes with a comprehensive supplemental material
  - Provides analytical proofs to support its claims.
  - Comes with a lot of very detailed experiments.

- **Negative:**
  - The “monologue” of the paper is too focused on the Cross-Entropy comparison.
  - Does not show experiments with “reduced training size"

Q&A Section

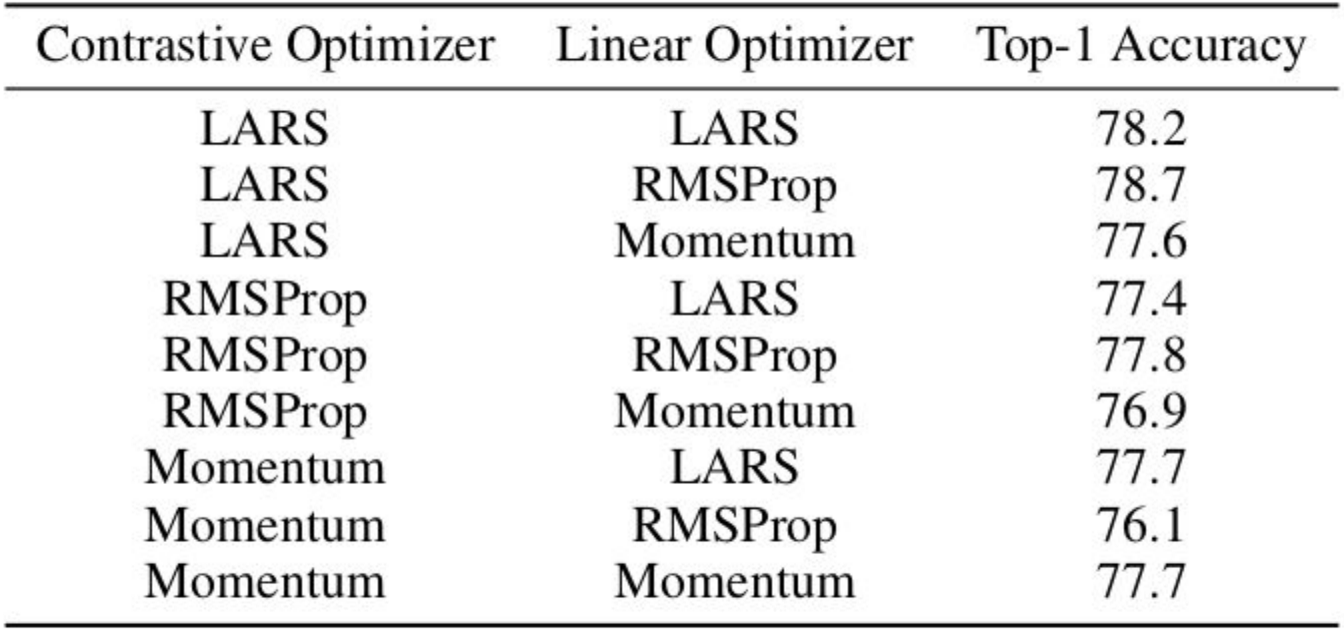
Difference between the (in) and (out) losses

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

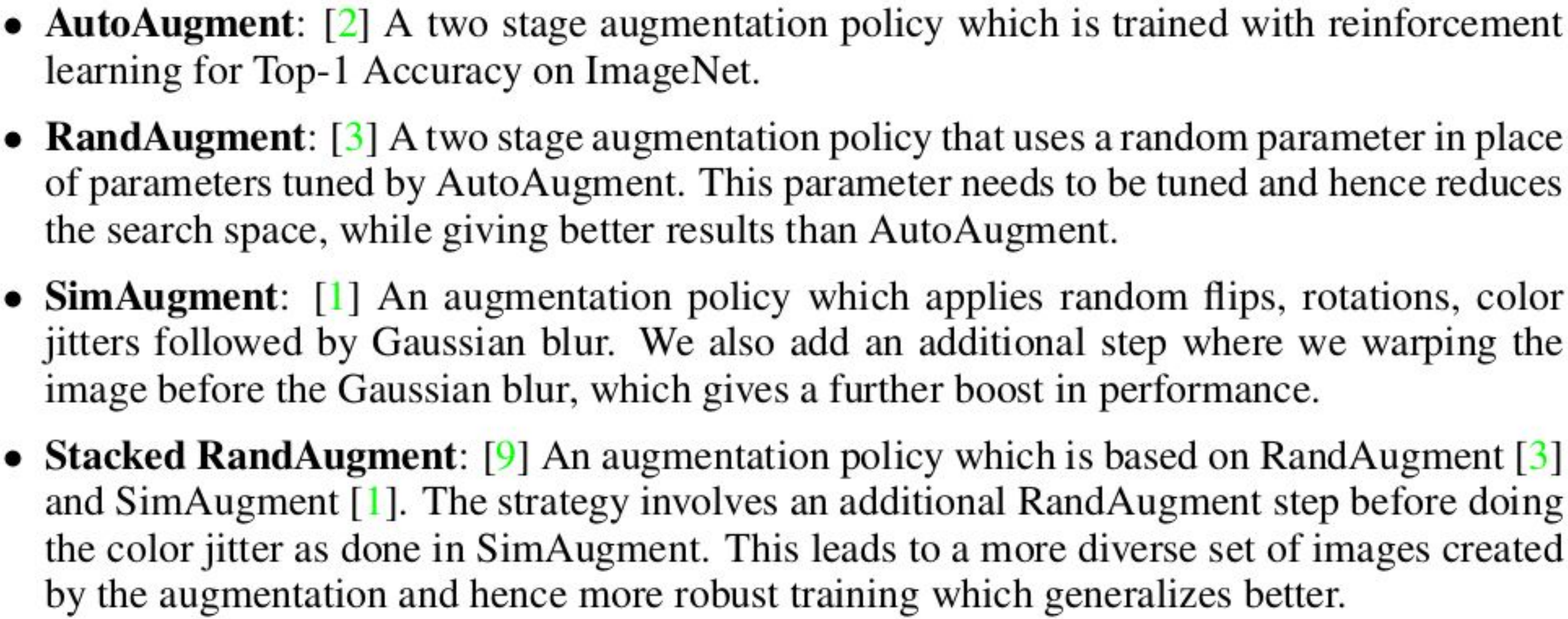
$$X_{ip} \equiv \begin{cases} \dfrac{\exp(\boldsymbol{z}_i \boldsymbol{\cdot} \boldsymbol{z}_p/\tau)}{\sum\limits_{p'\in P(i)} \exp\left(\boldsymbol{z}_i \boldsymbol{\cdot} \boldsymbol{z}_{p'}/\tau\right)} & , \quad \text{if } \mathcal{L}_i^{sup} = \mathcal{L}_{in,i}^{sup} \\ \dfrac{1}{|P(i)|} & , \quad \text{if } \mathcal{L}_i^{sup} = \mathcal{L}_{out,i}^{sup} \end{cases}$$

$$P_{ip} \equiv \frac{\exp\left(\boldsymbol{z}_i \bullet \boldsymbol{z}_p/\tau\right)}{\sum_{a\in A(i)} \exp\left(\boldsymbol{z}_i \bullet \boldsymbol{z}_a/\tau\right)}$$

Formulas taken from: Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. “Supervised Contrastive Learning”, NeurIPS 2020



Contrastive Optimizer	Linear Optimizer	Top-1 Accuracy
LARS	LARS	78.2
LARS	RMSProp	78.7
LARS	Momentum	77.6
RMSProp	LARS	77.4
RMSProp	RMSProp	77.8
RMSProp	Momentum	76.9
Momentum	LARS	77.7
Momentum	RMSProp	76.1
Momentum	Momentum	77.7



- **AutoAugment**: [2] A two stage augmentation policy which is trained with reinforcement learning for Top-1 Accuracy on ImageNet.
- **RandAugment**: [3] A two stage augmentation policy that uses a random parameter in place of parameters tuned by AutoAugment. This parameter needs to be tuned and hence reduces the search space, while giving better results than AutoAugment.
- **SimAugment**: [1] An augmentation policy which applies random flips, rotations, color jitters followed by Gaussian blur. We also add an additional step where we warping the image before the Gaussian blur, which gives a further boost in performance.
- **Stacked RandAugment**: [9] An augmentation policy which is based on RandAugment [3] and SimAugment [1]. The strategy involves an additional RandAugment step before doing the color jitter as done in SimAugment. This leads to a more diverse set of images created by the augmentation and hence more robust training which generalizes better.