Contrastive Self-Supervised Learning: A Survey on Different Architectures

Adnan Khan
Department of Computer Vision
MBZUAI
Abu Dhabi, UAE
adnan.khan@mbzuai.ac.ae

Sarah AlBarri
Department of Machine Learning
MBZUAI
Abu Dhabi, UAE
sarah.albarri@mbzuai.ac.ae

Muhammad Arslan Manzoor
Department of Natural Language Processing
MBZUAI
Abu Dhabi, UAE
muhammad.arslan@mbzuai.ac.ae

2022 2nd International Conference on Artificial Intelligence (ICAI) | 978-1-6654-6896-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICAI55435.2022.9773725

***Abstract*—Self-Supervised Learning (SSL) has enhanced the learning process of semantic representations from images. SSL has reduced the need for annotating or labelling the data by relying less on class labels during the training phase. SSL techniques dependent on Constrative Learning (CL) are acquiring prevalence because of their low dependency on training data labels. Different CL methods are producing state-of-the-art results on datasets which are used as the benchmarks for Supervised Learning. In this survey, we provide a review of CL-based methods including SimCLR, MoCo, BYOL, SwAV, SimTriplet and SimSiam. We compare these pipelines in terms of their accuracy on ImageNet and VOC07 benchmark. BYOL propose basic yet powerful architecture to accomplish 74.30% accuracy score on image classification task. Using clustering approach SwAV outperforms other architectures by achieving 75.30% top-1 ImageNet classification accuracy. In addition, we shed light on the importance of CL approaches which can maximise the use of huge amounts of data available today. At last, we report the impediments of current CL methodologies and emphasize the need of computationally efficient CL pipelines.**

***Index Terms*—Self-Supervised Learning, Contrastive Learning, Image Augmentation, Data Annotation**

I. Introduction

In the previous decade, researchers have put effort in upbringing the performance of AI based systems by training the models on massive amount of labeled data [1]. This approach leads the models highly depend on carefully annotated data. The supervised learning depicts good results with large training datasets, and improved computation resources. Therefore, this technique faces issues in real time scenario where annotation is not possible or dealing with few shot learning problems [2].

Recently the research community shifted to integrated method of generative and contrastive techniques known as Self-Supervised Learning (SSL) [3]. Generally, SSL based approaches mitigate the challenges of traditional approaches by learning feature representation from data itself without expensive annotation. The generative pretext tasks in SSL learn features from input samples with pseudo-labels as representations at intermediate level that helps in downstream tasks [4]. Colorizing pictures, super-resolution, image in-painting, jigsaw puzzles, and audio-visual correlation have shown to be appropriate in learning excellent representations [5]. Contrary, CL is based on the discriminative scheme, which forms the representation that helps to distinguish one object from another. The objective of CL is to learn the representation in a manner where semantically relative features attract and non-relative features repel each other [6]. The SSL based techniques take benefit from the knowledge learnt during pretext phase to perform effectively on any specific downstream task [7]. The downstream tasks include any task or subtask for example classification, detection, and segmentation. However, Semi Supervised Learning requires labeled data for training in the small quantity. On other hand SSL learn the underlying structure itself [8]. The comprehensive review [3] exhibits the comparison of SSL performance with semi supervised learning where SSL surpassed semi-supervised learning with clear margin.

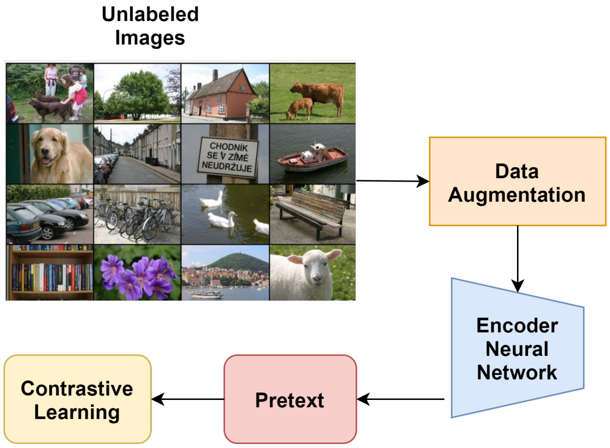


Fig. 1. Pipeline of contrastive learning

In unsupervised learning, neural networks perform better by pretraining without labels and make the representation space meaningful [9]. The random augmentation and different version of same images can be acquired through cropping, flipping, varying the brightness or color. All the images that belong to same original images are treated as “positive” and pulled together. The “negatives” are different images from dataset and pushed apart. Ultimately, the network learns from the random transformations of the same image, get robust, and

978-1-6654-6896-1/22/$31.00 © 2022 IEEE

generalized [10], [11]. The embedding space based on these transformations produce adaptable representation that leads to significant improvement [12], [13].

The pipeline of the CL and knowledge transfer procedure from pretext to downstream task are visually represented in the Fig. 1. The problem that researchers faced in contrastive approach for unsupervised pretraining is that it pushed apart the samples that should belong to same class (in different orientation and not transformed), made it harder for the classifier to categorize later in the right class or create decision boundaries [14]. Supervised contrastive learning addressed this issue by introducing labels in pretraining objective [15]. The purpose is to classify samples correctly that belong to same class even if the orientation, scenery or features are different. Contrastive pretraining objective functions are better at exploiting the information in the dataset than the cross-entropy loss.
Recent research includes MoCo [16] that considers CL as dynamic dictionary lookup problem. On the ImageNet [17] dataset, SimCLR [18] and SimSiam [19] achieved results that are comparable to the state-of-the-art supervised approach. Likewise, BYOL [20] , and SwAV [21] are some studies that demonstrate the efficacy of the pretext tasks utilized and how they improve model performance.

Several attempts to survey Contrastive SSL are proposed. In [3], the survey provides the knowledge to chose the right pretext task and suitable architecture for different downstream tasks. The architecture pipelines introduced are classified into four parts; namely, memory bank, end-to-end, momentum encoder and clustering. In [22], three methods, which are generation-based, context-based, and free semantic label-based methods, are introduced for image feature SSL. For computer vision, natural language processing, and graph learning application, SSL is analyzed in [2] according to its objective, whether generative, contrastive, or generative-contrastive. The use of geometric transformers as a supervisory signal in SSL is surveyed in [23]. Methods and applications of SSL within the sequential transfer learning framework is reviewed in [24]. The contributions of this paper are to introduce the state-of-the-art architectures of Contrastive SSL, to analyze and compare their performances using ImageNet and VOC07 dataset, and to highlight the importance, potential, and limitations in existing literature for SSL CL.

II. Importance of Self-Supervised Learning

The availability of labeled data today has made it possible to train many task specific machine learning models. Unfortunately these models cannot be generalized to other tasks and can perform better only on the tasks for which they are trained specifically[1]. In humans and animals, common sense plays a significant role to learn and generalize from few examples. We don't need massive amounts of data to identify objects every time. The ability of AI models to generalize to tasks in such a way that humans or animals do is still an open and challenging research area of AI. In realm of easily available huge amounts of data, one of the most promising ways to build such models which can generalize to many tasks is SSL which can approximate the general intelligence in machine learning models. SSL leverage the power of data without relying on the labels and learns those subtle representations and patterns which are difficult for supervised models to learn.

In addition to learning good representations, SSL also mitigates the problems and cost involved in the process of annotating data. Data annotation is a costly process. For instance, the annotation time for assigning semantic class labels can be in the order of hours for a single image where the annotator do pixel-wise labelling and draws boundary lines to segment different instances in images. The accuracy of image classification model can be worsened by the human errors involved in the image labelling in a supervised learning setting. These errors include, fine grained recognition [25], ground truth class unawareness or insufficient training data at the time of annotation.

SSL comes handy in dealing with all the aforementioned problems in data labelling by mitigating the reliance on the class labels of the data. SSL learns the representations in unsupervised fashion [26] as a pretext task which can then be fine-tuned to a downstream task.

III. Existing Models

A. MoCo

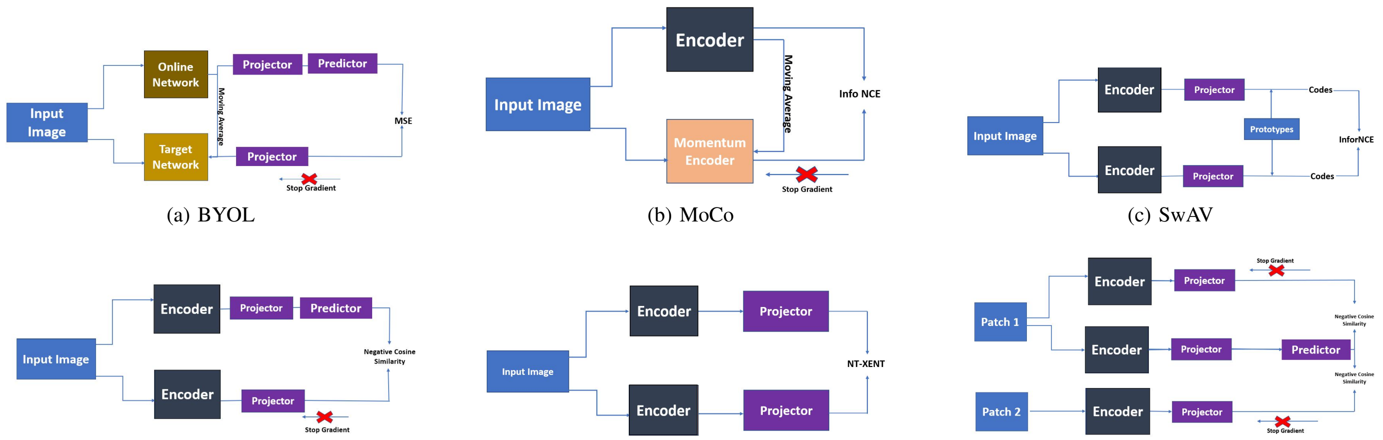
Momentum Contrast (MoCo) [16] views CL [27] from dictionary perspective to match a query with its positive key encoding and make it as dissimilar as possible to the negative key encodings. In the end to end approach, neural network encode query and keys, take the gradients through each query encoding network and key encoding network that is highly memory-inefficient. MoCo proposed that the only need to pass the gradients through the query encoder, the key encoder get updated by adding a momentum update of a query encoder's parameters as represented in the (1) and Fig. 2b.

$$\theta_k \leftarrow m\theta_k + (1-m)\theta_q \tag{1}$$

In MoCo, the loss can be defined as the dictionary lookup problem, as a query is matched to a key in the dictionary [27]. Image can be encoded into query features, where the dictionary is built up by the features of a large set of image samples [28]. Dictionary in computer vision was not readily available, it is computed on the run time dynamically by applying an encoder to set of image samples. The dictionary lookup problem happens in the feature space [29].

MoCo mainly addressed two challenges; i) how to make a large dynamic dictionary , ii) how to make dynamic dictionary consistent when the encoder is being updated (in the context of SGD training). To have a large dictionary in Contrastive SSL framework, features of the previous batch were maintained as a queue. The dictionary consists of current and previous batches

[1] https://ai.facebook.com/blog/self-supervised-learning-the-dark-matter-of-intelligence/



(a) BYOL (b) MoCo (c) SwAV

(d) SimSiam (e) SimCLR (f) SimTriplet

Fig. 2. A comprehensive schematics that summarizes the different architectures of SSL CL. The main blocks are the encoders, stop gradient, projector and momentum encoder.

and is not limited by the batch size. The features in the dictionary from the updated encoder covers multiple batches, which arise the issue of inconsistency, to improve the consistency of the features, they proposed to use the momentum encoder that updates slowly. The momentum encoder is the moving average of the original encoder. Authors found in the ablation experiments that momentum is of central importance.

B. SimCLR

SimCLR [18] came up with the idea of visual representation through CL without demanding specific architecture or huge memory. Their proposed framework consists of following components:

- Random augmentation module, which indiscriminately transforms input data samples into correlated view known as positive pairs.
- A deep network encoder which extracts vector representation from data included augmented versions.
- They added multi-layer perceptron projection head from the representation to the classification layer with contrastive loss function.

The representations in SimCLR in the Fig. 2e is learnt by increasing the agreement between different augmented versions of the same data example via a contrastive loss in the latent space [30]. In the contrastive framework, the cosine similarity is the basic metric for all functions that can be used to compute contrastive loss [11] is given in (2).

$$sim(A,B) = \frac{A.B}{\|A\|\,\|B\|} \tag{2}$$

CL depends on the relative embeddings and the function known as Noise Contrastive Estimation (NCE) as mentioned in the (3).

$$L = -log\frac{exp(sim(q,k_{+})/\tau)}{exp(sim(q,k_{+})/\tau) + exp(sim(q,k_{-})/\tau)} \tag{3}$$

SimCLR introduced some interesting advancements to the contrastive SSL framework which include larger batch size, larger models, stronger data augmentation, adding multi-layer perceptron projection head. SimCLR(4x) model surpasses a supervised learning baseline on the resnet50 model. The author presented an updated version as SimCLR-v2 [31] that exploit deep projection along with memory mechanism from MoCo [16].

C. BYOL

Bootstrap Your Own Latent (BYOL) consists of two neural networks namely an online network and a target network [20]. On high level this CL approach tries to get rid of necessary negative samples involved when doing the contrastive loss for SSL. Two augmented views of the same image are fed to the online and target networks respectively. A representation of the first view is learnt using an encoder in the online network and the updated learnt weights are copied to the target network as an exponential moving average. The objective function is simply the minimization of Mean Squared Error (MSE) [32] between the predictions from online network and target projections shown in (4).

$$L_{\theta},\xi \triangleq \left\|\left(q_{\theta}(z_{\theta}) - z'_{\xi}\right)\right\| \tag{4}$$

The objective function is optimized stochastically to minimize the loss with respect to θ only.

The projection heads are used in online network to reduce the dimensionality of the output of the encoder and generates the low dimensional features for predictor. The predictor is a part of online network which predicts the representation of the target network. The architecture of BYOL is shown in Fig. 2a. Due to the lack of symmetry between the online and target networks and the moving average, BYOL learns to ignore the augmentations of the image.

D. SwAV

Swapping Assignments between Views (SwAV) [21] is an SSL method that exploits the contrastive methods without the need for pairwise comparisons and is therefore not computationally expensive. SwAV clusters' the input images and constrains consistency between assignments of clusters which are constructed for different augmentations (views) of the same input image. In SwAV we learn to predict the cluster assignment of one view from the representation of another view.

In SwAV, two augumented views of the same image are fed to the two feature encoders shown in Fig. 2c. These features are then mapped to its nearest neighbours in a set of clusters which is a finite length discrete codebook. The algorithm predicts the codes of one view from the representation of another view. Swap prediction loss given in (5) is calculated from the codes and features.

$$L(z_t, z_s) = l(z_t, q_s) + l(z_s, z_t) \tag{5}$$

The swap prediction loss shows the fit between the features z and a code q.

E. SimSiam

Simple Siamese (SimSiam) [19] employs Siamese networks in contrastive learning. As noted in Fig. 2d, SimSiam architecture inputs two augmentation of the same image and maximizes the similarities between them. Two identical encoders are used to process the augmented images. The identical encoders share the same weights and backbone, which is ResNet and the Multi-Layer Perceptron (MLP). After one of the encoders, a predictor is added. The predictor function is to transforms the output of the encoder and matches it to the output of the other encoder. Cosine and cross-entropy similarity are used to maximize the similarity between the augmented images. Stop- gradient operation mechanism is introduced in the architecture to prevent the collapsing solution, which is the classification trivial solution. Despite its relatively simple architecture, when SimSiam is tested on a 1000-classes ImageNet set, it reaches competitive results when compared to other state-of-the-art complex architectures.

F. SimTriplet

Simple Triplet builds on top of the SimSiam architecture, as can be noted in the architecture description and in Fig. 2f. SimTriplet architecture found in [33] utilizes the adjacent patches similarity from pathological images Whole Side Image (WSI). By training on a pair of nearby image patches of the same tissue type, SimTriplet performs two different augmentations on the first image patch and another augmentation of the second image patch. After which SimTriplet eliminates the need for negative images samples by minimizing the intra-sample and inter-sample similarities. Intra-sample is the similarity between the two augmentations of the same image, whereas inter-sample is the similarity between the two adjacent image patches. SimTriplet architecture first step consists of three parallel and identical encoders for the three augmented images. The encoders incorporate ResNet-50 and three-layer multi-layer perceptron (MLP). After one of the encoders, the addition of an MLP predictor attempts to match that encoder output to the output of the two remaining encoders. The two remaining encoders have a stop-gradient mechanism. The total loss, or Triplet loss function, is computed as the summation of losses between inter- and intra-samples, using negative cosine similarity function. As SimTriplet is trained, mixed precision is used as in deep neural network it performs half-precision operations to lessen the training time and required memory. The implication of mixed precision is that SimTriplet can be trained using only a single GPU with 16GB memory. When tested on 1%, 10%, 25%, and 100% annotated training data, SimTriplet outperforms SimSiam network and supervised model based on F1 score and balanced accuracy. For instance, for 10% annotated training data labels, SimTriplet balanced accuracy score is 0.7110 compared to 0.3561 and 0.6864 for supervised model and SimSiam network, respectively [33].

IV. Results

Currently several SSL methods compete with the supervised learning models on different benchmark datasets. Most of these models are evaluated on the ImageNet [17] and Pascal VOC [34] datasets for the downstream tasks such as image classification and object detection, respectively. In this section we compare the performance of these models on different downstream tasks. All the literature that we referred used ResNet as their encoders with not much different set of evaluation metrics. We use these common metrics to compare the results of these models with each other. The widely accepted evaluation metrics are transfer learning with VOC07, linear evaluation [22] and semi-supervised learning on the ImageNet.

A summary of reviewed models is given in Table I. The accuracy of these models in supervised settings is given as a baseline. BYOL, SwAV, and SimSiam are comparable because their architectures are somewhat similar, and therefore, achieve top results (74.3%, 75.3%, and 71.3% respectively) in linear evaluation on ImageNet. The noticeable part is that these results are remarkably close to that of the baseline supervised network. Among the SSL models, the closest model in terms of accuracy to the supervised learning is SwAV which achieves a top performance of 75.3% for the ImageNet linear evaluation. It also shows the best performance for the object detection task with VOC07 dataset, scoring accuracy of 88.9%. The model

TABLE I
Results of ImageNet semi-supervised learning using 10% labels, linear evaluation and transfer learning (TL) on Pascal VOC object detection tasks.

Method	**Top-1 Acc.**	**Top- Acc.**	**Lin. Eval.**	**VOC07**
Supervised	56.40	80.40	76.50	87.50
MoCO [16]	-	-	60.60	-
SimCLR [18]	65.60	87.80	69.30	85.50
BYOL [20]	68.80	89.00	74.30	85.40
SwAV [21]	**70.20**	**89.90**	**75.30**	**88.90**
SimSiam [19]	-	-	71.30	-

is outperforming even the supervised baseline by 1.4%. In addition, the semi-supervised learning with ImageNet, both in top-1 accuracy and top-5 accuracy, SwAV is achieving the highest score of 70.20% and 89.90% respectively. The architecture of SwAV is fairly simple, yet its performance is outstanding on account of online clustering.

V. Discussion

Each of the architectures that is analyzed in our study attempts to mitigate some of the shortcomings found in the other architecture. The shortcomings are noted in Table II as the usage of negative samples, higher computational power, large batches, stop gradient and momentum encoders. MoCo and SimCLR integrate negative samples to avoid the collapsing, or trivial solution. SimCLR outperforms MoCo in linear evaluation by 8.7% despite removing the computationally intensive momentum encoder and stop gradient. BYOL, which employs momentum encoder for accuracy and stop gradient mechanism, removes the need for negative samples. In linear evaluation top-1 accuracy, BYOL performs better than MoCo and SimCLR. Nonetheless, despite its superior performance, it is still lags behind supervised learning and uses high computing power of 512 cloud TPU cores.

Both SwAV and SimSiam architectures removes the need for negative samples, large batches, and momentum encoders. In SwAV, online clustering build a computationally inexpensive architecture by creating a list of feature and then using simple algorithms to match these features, such as kth nearest neighbor. As SimSiam architecture doesn't include online clustering, stop gradient mechanism is introduced. In all of the downstream tasks introduced in Table I, SwAV has the highest scores compared to the other architectures. SimTriplet was not tested neither on ImageNet nor on VOC07 datasets, as it relies on the fact that the two adjacent patches of the image are similar. SimTriplet is tested on WSI dataset, and for that application it outperformed SimSiam as noted in [33].

Finally, it is worth noting that many of these architectures have further enhanced modifications and versions with stronger linear evaluation top-1 accuracy, and object detection rates. However, the modified versions are much more computationally exhaustive. For instance, SimCLR version 2 outperforms supervised learning method by using 400 million parameters, compared to the 25 million parameters for standard SimCLR introduced in [18]. Despite Self-Supervised CL techniques being comparatively new in literature, they are showing potential to perform better than supervised learning with labeling of only 1% or 10% of the total data. This justifies intensive research which could perhaps optimize the architectures, expand on the strength of each, and introduce hybrid models.

A. Limitations and Future Work

The CL shows promising results on the benchmark datasets, however, there is a need for more theoretical analysis to justify these results. According to [4] the SSL methods MoCo [27] and PIRL [35] does not capture the viewpoint and category instance invariance which is of important consideration in recognition tasks. The architecture design and sampling methods greatly affect the performance of contrastive objective function [36]. The authors in [37] show that SSL is greatly dependent on the pretext task chosen for its training. They further elaborate that SSL is better at extracting the task-specific features from the data and completely ignores the task-irrelevant features and therefore more theoretical analysis is required to understand the design pipeline of the contrastive methodology. In SSL the representations are learnt using the self-supervised objectives which are greatly influenced by the underlying data [3]. With the increasing sizes of the datasets, the associated biases are hard to mitigate. In order to train the SSL models which are based on contrastive loss, the learning is greatly affected when easy negatives examples are encountered during the training. Due to less difference in the positive and the negative examples, the ability of the model of converging quickly is limited. The authors of [18] tackle the issue by increasing the batch sizes while [16] uses huge memory banks. Both of these methods are difficult to replicate without having massive computing power.

TABLE II
Comparison of different CL SSL methods based on architecture components and resources in pretext task.

Method	-ve samp.	Compute	Large Batches	Stop Grad	Mom. Enc.
MoCo [16]	✔	8 GPU	✕	✔	✔
SimCLR [18]	✔	-	✔	✕	✕
BYOL [20]	✕	512 TPUs	✕	✔	✔
SwAV [21]	✕	64 GPUs	✕	✕	✕
SimSiam [19]	✕	-	✕	✔	✕
SimTriplet [33]	✕	1 GPU	✕	✔	✕

VI. Conclusion

This study examines several current high-performing SSL approaches for acquiring visual representation based on contrastive learning. We discussed the augmentation strategies for each model, the architectural design, the phenomenon to build pretext tasks that finally lead to learn representations for downstream tasks. Additionally, we reviewed the CL-based methods namely SimCLR, MoCo, BYOL, SwAV, SimTriplet and SimSiam. The pretraining of the network on unlabeled data using CL has produced promising results for different vision tasks including classification, detection, and segmentation. The analysis was reported for all architecures and compared the accuracies on ImageNet benchmark. SwAV outperformed all other CL approaches that we reviewed.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016. 1

[2] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE Transactions on Knowledge and Data Engineering*, 2021. 1, 2

[3] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, p. 2, 2020. 1, 2, 5

[4] S. Purushwalkam and A. Gupta, "Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases," *Advances in Neural Information Processing Systems*, vol. 33, pp. 3407–3418, 2020. 1, 5

[5] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, and G. Brain, "Time-contrastive networks: Self-supervised learning from video," in *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 1134–1141, IEEE, 2018. 1

[6] M. U. Gutmann and A. Hyvärinen, "Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics.," *Journal of Machine Learning Research*, vol. 13, no. 2, 2012. 1

[7] Y. Tian, X. Chen, and S. Ganguli, "Understanding self-supervised learning dynamics without contrastive pairs," in *International Conference on Machine Learning*, pp. 10268–10278, PMLR, 2021. 1

[8] W. Falcon and K. Cho, "A framework for contrastive self-supervised learning and designing a new approach," *arXiv preprint arXiv:2009.00104*, 2020. 1

[9] A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with exemplar convolutional neural networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 9, pp. 1734–1747, 2015. 1

[10] O. Henaff, "Data-efficient image recognition with contrastive predictive coding," in *International Conference on Machine Learning*, pp. 4182–4192, PMLR, 2020. 2

[11] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018. 2, 3

[12] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," *arXiv preprint arXiv:1803.07728*, 2018. 2

[13] C. Zhuang, A. L. Zhai, and D. Yamins, "Local aggregation for unsupervised learning of visual embeddings," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6002–6012, 2019. 2

[14] P. H. Le-Khac, G. Healy, and A. F. Smeaton, "Contrastive representation learning: A framework and review," *IEEE Access*, vol. 8, pp. 193907–193934, 2020. 2

[15] Y. M. Asano, C. Rupprecht, and A. Vedaldi, "Self-labelling via simultaneous clustering and representation learning," *arXiv preprint arXiv:1911.05371*, 2019. 2

[16] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020. 2, 3, 4, 5

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009. 2, 4

[18] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*, pp. 1597–1607, PMLR, 2020. 2, 3, 4, 5

[19] X. Chen and K. He, "Exploring simple siamese representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15750–15758, 2021. 2, 4, 5

[20] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, *et al.*, "Bootstrap your own latent: A new approach to self-supervised learning," *arXiv preprint arXiv:2006.07733*, 2020. 2, 3, 4, 5

[21] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9912–9924, 2020. 2, 3, 4, 5

[22] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE transactions on pattern analysis and machine intelligence*, 2020. 2, 4

[23] M. Ali and S. Hashim, "Survey on self-supervised representation learning using image transformations," *CoRR*, 2022. 2

[24] H. H. Mao, "A survey on self-supervised pre-training for sequential transfer learning in neural networks," *CoRR*, vol. abs/2007.00800, 2020. 2

[25] D. Lin, X. Shen, C. Lu, and J. Jia, "Deep lac: Deep localization, alignment and classification for fine-grained recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2

[26] A. Tendle and M. R. Hasan, "A study of the generalizability of self-supervised representations," *Machine Learning with Applications*, vol. 6, p. 100124, Dec 2021. 2

[27] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, pp. 1735–1742, IEEE, 2006. 2, 5

[28] M. Ye, X. Zhang, P. C. Yuen, and S.-F. Chang, "Unsupervised embedding learning via invariant and spreading instance feature," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6210–6219, 2019. 2

[29] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492–1500, 2017. 2

[30] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3733–3742, 2018. 3

[31] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. E. Hinton, "Big self-supervised models are strong semi-supervised learners," *Advances in neural information processing systems*, vol. 33, pp. 22243–22255, 2020. 3

[32] C. Sammut and G. I. Webb, "Mean squared error," in *Encyclopedia of Machine Learning*, p. 653, Springer, 2010. 3

[33] Q. Liu, P. C. Louis, Y. Lu, A. Jha, M. Zhao, R. Deng, T. Yao, J. T. Roland, H. Yang, S. Zhao, *et al.*, "Simtriplet: Simple triplet representation learning with a single gpu," *arXiv preprint arXiv:2103.05585*, 2021. 4, 5

[34] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010. 4

[35] I. Misra and L. van der Maaten, "Self-supervised learning of pretext-invariant representations," 2019. 5

[36] S. Arora, H. Khandeparkar, M. Khodak, O. Plevrakis, and N. Saunshi, "A theoretical analysis of contrastive unsupervised representation learning," *arXiv preprint arXiv:1902.09229*, 2019. 5

[37] Y.-H. H. Tsai, Y. Wu, R. Salakhutdinov, and L.-P. Morency, "Self-supervised learning from a multi-view perspective," *arXiv preprint arXiv:2006.05576*, 2020. 5