SynCo: Synthetic Hard Negatives for Contrastive Visual Representation Learning

Nikolaos Giakoumoglou Tania Stathaki
Imperial College London
{nikos, t.stathaki}@imperial.ac.uk

Code: https://github.com/giakoumoglou/synco

Abstract

*Contrastive learning has become a dominant approach in self-supervised visual representation learning, but efficiently leveraging hard negatives, which are samples closely resembling the anchor, remains challenging. We introduce SynCo (**Sy**nthetic **n**egatives in **Co**ntrastive learning), a novel approach that improves model performance by generating synthetic hard negatives on the representation space. Building on the MoCo framework, SynCo introduces six strategies for creating diverse synthetic hard negatives "on-the-fly" with minimal computational overhead. SynCo achieves faster training and strong representation learning, surpassing MoCo-v2 by **+0.4%** and MoCHI by **+1.0%** on ImageNet ILSVRC-2012 linear evaluation. It also transfers more effectively to detection tasks achieving strong results on PASCAL VOC detection (57.2% AP) and significantly improving over MoCo-v2 on COCO detection (**+1.0%** AP^{bb}) and instance segmentation (**+0.8%** AP^{msk}). Our synthetic hard negative generation approach significantly enhances visual representations learned through self-supervised contrastive learning.*

1. Introduction

Contrastive learning has emerged as a prominent approach in self-supervised learning, significantly advancing representation learning from unlabeled data through sophisticated feature space optimization techniques and novel architectural innovations. This technique, which distinguishes between similar and dissimilar data pairs, has shown remarkable promise in visual representation tasks across diverse domains and applications. Seminal works such as SimCLR [11] and MoCo [27] established instance discrimination as a pretext task. These methods generate multiple views of the same data point through augmentation, training the model to minimize distance between positive pairs (augmented views of the same instance) while maximizing it for negative pairs (views of different instances).

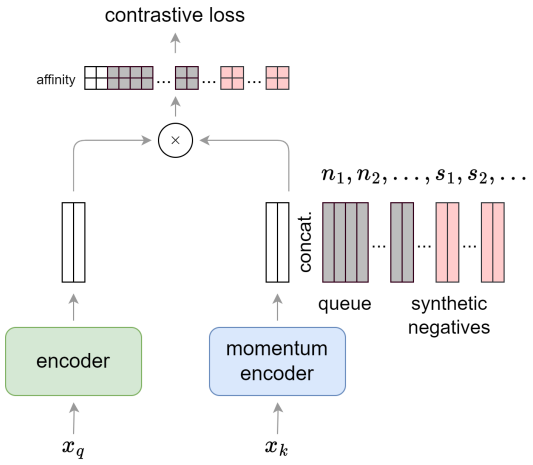


Figure 1. **SynCo extends MoCo** [13, 27] by introducing synthetic hard negatives generated "*on-the-fly*" from a memory queue. The process begins with two augmented views of an image, $\mathbf{x}_q$ and $\mathbf{x}_k$, processed by an encoder and a momentum encoder, respectively, producing feature vectors $\mathbf{q}$ and $\mathbf{k}$. The memory queue holds negative samples $\mathbf{n}_1, \mathbf{n}_2, \ldots$, which are concatenated with synthetic hard negatives $\mathbf{s}_1, \mathbf{s}_2, \ldots$ generated using the SynCo strategies. These combined negatives are used to compute the affinity matrix, which, together with the positive pair (query $\mathbf{q}$ and key $\mathbf{k}$), contributes to the InfoNCE loss calculation.

Despite its effectiveness, instance discrimination faces significant computational and methodological challenges. A key limitation is the need for numerous negative samples, often leading to increased computational costs and memory requirements. For example, SimCLR requires large batch sizes for sufficient negatives [11]. While approaches like MoCo address some issues through dynamic queues and momentum encoders [13, 27], they still face challenges in selecting and maintaining high-quality hard negatives. Some variations, like SimCo [65], take a different approach by removing both the momentum encoder and queue in favor of a dual temperature mechanism that modulates positive and negative sample distances differently in the InfoNCE loss.

Recent studies have highlighted the importance of data augmentations in learning robust representations [4, 11, 17, 22, 41, 44, 47, 53]. These transformations provide diverse, challenging copies of images, increasing the difficulty of the self-supervised task. Moreover, techniques that combine data at the pixel level [63, 66] or feature level [51] have proven effective in helping models learn resilient features. The concept of challenging negative samples has been explored to enhance contrastive learning, with MoCHI [32] showing improvements by incorporating harder negatives.

The concept of challenging negative samples has been explored as a way to enhance contrastive learning models. These samples, which lie close to the decision boundary, are crucial for refining the model's discriminative abilities. Recent work like MoCHI [32] has shown improvements by incorporating harder negatives. However, while the potential of hard negatives is clear, recent trends in AI have shifted focus toward large-scale foundation models [2, 8], leaving this direction relatively unexplored. Yet, as Yann LeCun observed, "*if AI is a cake, self-supervised learning is the bulk of the cake*". We argue that revisiting self-supervised approaches, particularly through innovative hard negative strategies, remains crucial for advancing AI systems.

In this paper, we present SynCo (***Sy***nthetic ***n***egatives in ***Co***ntrastive learning), a novel approach to contrastive learning that leverages synthetic hard negatives to improve the learning process. Building on the foundations of MoCo, SynCo introduces six distinct strategies for generating synthetic hard negatives, each designed to provide diverse and challenging contrasts to the model. Our approach focuses on leveraging only the most challenging negatives (determined by their similarity scores with the query) to create new synthetic negatives that are both difficult and diverse. By incorporating these synthetic samples, SynCo aims to push the boundaries of contrastive learning, improving both the efficiency and effectiveness of the training process.

The main **contributions** of our work are as follows:

- We introduce SynCo, a contrastive learning framework that improves representation learning by leveraging synthetic hard negatives. SynCo enhances model discrimination by generating challenging negatives "*on-the-fly*" from a memory queue using *six* strategies targeting different aspects of the feature space. This process improves performance without significant computational overhead.
- SynCo's strategies extend MoCHI [32] by **(i)** exploring boundaries beyond existing negatives and **(ii)** combining gradient-guided perturbations with controlled stochastic noise, improving uniformity (Figure 3) and inter-/intra-class distances (Figure 4).
- We show improvements across multiple downstream tasks: ImageNet linear evaluation (Tabs. 1 and 2), semi-supervised learning (Tab. 3), and object detection (Tab. 4).

2. Related Work

2.1. Contrastive Learning

Contrastive learning methods focus on instance discrimination as a pretext task, treating each image as its own class [11, 27]. The core principle involves bringing an anchor and a "positive" sample closer in the representation space while pushing the anchor away from "negative" samples [33]. Positive pairs are created through multiple views of each data point [9, 47], using techniques such as color decomposition [46], random augmentation [11, 27], image patches [50], or student-teacher representations [10, 23, 40]. The common training objective, based on InfoNCE [50] or its variants [11, 17, 48, 62], aims to maximize mutual information [3, 28], necessitating numerous negative pairs. While some approaches like SimCLR use large batch sizes [11], others like MoCo [13, 27], PIRL [38], and InstDis [57] employ memory structures. Recent advancements explore strategies such as regularizers [6, 7, 39, 69] or prevent model collapse via redundancy reduction [5, 64]. Some methods eliminate negative samples through asymmetric Siamese structures [10, 12, 23, 40]. While other approaches address the false-negative pair issue [34, 70] and improve representation learning by separating the learning of features and metrics into distinct phases [62].

2.2. Hard Negatives

Hard negatives are critical in contrastive learning as they improve visual representations by helping define the representation space more effectively. These challenging samples are harder to distinguish from the anchor point, enabling the model to better differentiate between similar features. The use of hard negatives involves selecting samples that are similar to positive samples but different enough to aid in learning distinctive features. MoCo [27] maintains challenging negatives via dynamic queues and momentum updates, while SimCLR [11] and InfoMin [47] adjust negative difficulty through augmentation techniques. Recent work has examined optimal sampling strategies [31, 43], semantic negative sampling [21], and broader negative sampling impacts [61]. Building on these ideas, MoCHI [32] proposes interpolation between query and hard negatives, and linear combination of pairs of hard negatives. While these geometric transformations provide improvements, they are limited to exploring the convex hull of existing hard negatives. Our work significantly ***extends*** this foundation in *three* key directions: **(i)** Boundary exploration: Type 2 (extrapolation) pushes beyond the convex hull to explore regions outside existing negatives; **(ii)** Gradient-guided generation: Types 5-6 leverage optimization landscape information for principled perturbations rather than geometric mixing alone; **(iii)** Stochastic robustness: Type 4 introduces controlled noise to prevent overfitting to specific negative patterns.

3. Preliminaries

In this section, we establish the theoretical foundations of contrastive learning and analyze the critical role of hard negatives in representation learning.

3.1. Contrastive Learning

Contrastive learning seeks to differentiate between similar and dissimilar data pairs, often treated as a dictionary look-up where representations are optimized to align positively paired data through contrastive loss in the representation space [27]. Given an image x, and a distribution of image augmentation $\mathcal{T}$, we create two augmented views of the same image using the transformation $t_q, t_k \sim \mathcal{T}$, i.e., $x_q = t_q(x)$ and $x_k = t_k(x)$. Two encoders, f_q and f_k, namely the query and key encoders, generate the vectors $\mathbf{q} = f_q(x_q)$ and $\mathbf{k} = f_k(x_k)$, respectively. The learning objective minimizes a contrastive loss using the InfoNCE criterion [50]:

$$\mathcal{L}(\mathbf{q}, \mathbf{k}, \mathcal{Q}) = -\log \frac{\exp(\mathbf{q}^\top \cdot \mathbf{k}/\tau)}{\exp(\mathbf{q}^\top \cdot \mathbf{k}/\tau) + \sum_{\mathbf{n}\in\mathcal{Q}} \exp(\mathbf{q}^\top \cdot \mathbf{n}/\tau)} \tag{1}$$

Here, $\mathbf{k}$ is f_k's output from the same augmented image as $\mathbf{q}$, and $\mathcal{Q} = \{\mathbf{n}_1, \mathbf{n}_2, \ldots, \mathbf{n}_K\}$ includes outputs from different images, representing negative samples of size K. The temperature parameter τ adjusts scaling for the ℓ_2-normalized vectors $\mathbf{q}$ and $\mathbf{k}$. The key encoder f_k can be updated in two ways. In the synchronized update approach, f_k is updated synchronously with f_q, maintaining identical weights throughout training [11]. Alternatively, a momentum update scheme can be employed, where f_k is updated using: $\theta_k \leftarrow m \cdot \theta_k + (1-m) \cdot \theta_q$ [27]. Here, θ_k and θ_q are the parameters of f_k and f_q, and $m \in [0, 1]$ is the momentum coefficient. This approach allows f_k to evolve slowly, providing consistent negative samples and stabilizing the learning process. The memory bank $\mathcal{Q}$ can be defined as an external memory of all dataset images [38, 46, 57], a queue of recent batches [27], or the current minibatch [11].

The gradient of the contrastive loss in Equation (1) with respect to the query $\mathbf{q}$ is given by:

$$\frac{\partial \mathcal{L}(\mathbf{q}, \mathbf{k}, \mathcal{Q})}{\partial \mathbf{q}} = -\frac{1}{\tau}\left((1 - p_k) \cdot \mathbf{k} - \sum_{\mathbf{n}\in\mathcal{Q}} p_n \cdot \mathbf{n}\right) \tag{2}$$

where

$$p_{z_i} = \frac{\exp(\mathbf{q}^\top \cdot \mathbf{z_i}/\tau)}{\sum_{j\in Z} \exp(\mathbf{q}^\top \cdot \mathbf{z_j}/\tau)} \tag{3}$$

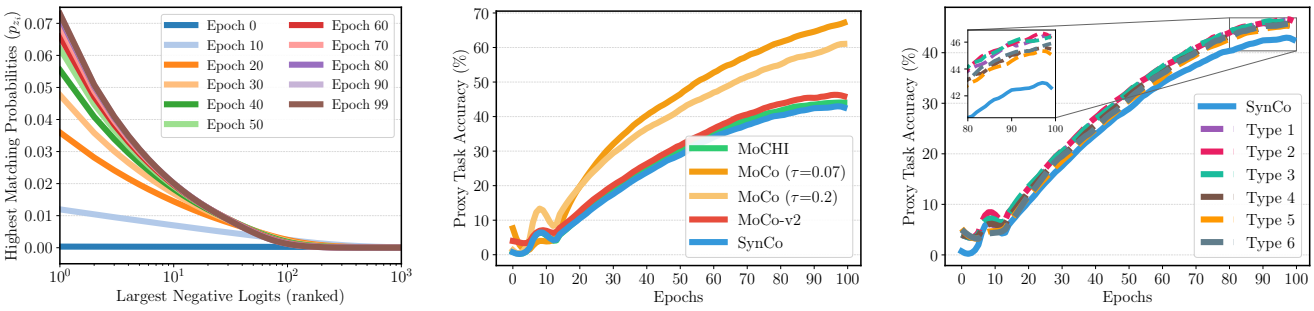
with $\mathbf{z_i}$ being a member of the set $\mathcal{Q} \cup \{\mathbf{k}\}$. The positive and negative logits contribute to the loss similarly to a $(K+1)$-way cross-entropy classification, with the key logit representing the query's latent class [1].

3.2. Understanding Hard Negatives

The effectiveness of contrastive learning approaches hinges critically on the utilization of hard negatives [1, 24, 30, 32, 37, 57]. Current approaches face significant challenges in efficiently leveraging these hard negatives. Sampling from within the same batch necessitates larger batch sizes [11, 14]. Conversely, maintaining a memory bank containing representations of the entire dataset incurs substantial computational overhead in keeping the memory up-to-date [13, 27, 38, 57]. These limitations underscore the need for more efficient strategies to generate and utilize hard negatives in contrastive learning frameworks.

Hardness of negatives. The "hardness" of negative samples, defined by their similarity to positive samples in the representation space, determines how challenging they are for the model to differentiate, directly impacting the effectiveness of the contrastive learning process. Figure 2a illustrates the evolution of negative sample hardness during MoCo-v2 training. Initially, the distribution of these probabilities is relatively uniform. However, as training progresses, a clear trend emerges: fewer negatives contribute significantly to the loss function. This observation suggests that the model rapidly learns to distinguish most negatives, leaving only a small subset that remains challenging. Such a phenomenon underscores the importance of maintaining a diverse pool of hard negatives throughout the training process to sustain effective learning [32].

Difficulty of the proxy task. The difficulty of the proxy task in contrastive learning, typically defined by the self-supervised objective, significantly influences the quality of learned representations. Figure 2b compares the proxy task performance of MoCo and MoCo-v2 on ImageNet-100, measured by the percentage of queries where the key ranks above all negatives. Notably, MoCo-v2, which employs more aggressive augmentations, exhibits lower proxy task performance compared to MoCo, indicating a more challenging learning objective. Paradoxically, this increased difficulty correlates with improved performance on downstream tasks such as linear classification [32]. Additionally, Figure 2c demonstrates how SynCo's performance varies under different configurations, providing insights into the optimal parameter settings for balancing proxy task difficulty and representation quality. The complete SynCo framework consistently outperforms each individual strategy (types 1-6) when applied in isolation. This counterintuitive relationship between proxy task difficulty and downstream performance suggests that more challenging self-supervised objectives can lead to the learning of more robust and transferable representations, motivating the development of strategies to dynamically modulate task difficulty during training.



(a) (b) (c)

Figure 2. **ImageNet-100 experiments.** (a) Histogram of the top 1024 matching probabilities p_{z_i}, $z_i \in \mathcal{Q}$ for MoCo-v2 over various epochs. Logits are organized in descending order, and each line indicates the mean matching probability across all queries. (b) Performance comparison of MoCo, MoCo-v2, MoCHI, SynCo in terms of accuracy on the proxy task (percentage of queries where the key is ranked higher than all negatives). (c) Performance comparison of SynCo under various configurations in terms of accuracy on the proxy task.

4. Synthetic Hard Negatives in Contrastive Learning

In this section, we present an approach for generating synthetic hard negatives in the representation space using **six** distinct strategies. Building on MoCHI, we propose four additional strategies for generating synthetic hard negatives that explore complementary aspects of the representation space (see *supplementary material* for differences from our method). We refer to our proposed approach as SynCo ("***Sy***nthetic ***n***egatives in ***Co***ntrastive learning").

4.1. Generating Synthetic Hard Negatives

Let $\mathbf{q}$ represent the query image, $\mathbf{k}$ its corresponding key, and $\mathbf{n} \in \mathcal{Q}$ denote the negative features from a memory structure of size K. The loss associated with the query is computed using the logits $\ell(\mathbf{z_i}) = \mathbf{q}^\top \cdot \mathbf{z}_i/\tau$, which are processed through a softmax function. We define $\hat{\mathcal{Q}} = \{\mathbf{n}_1, \mathbf{n}_2, \ldots, \mathbf{n}_K\}$ as the ordered set of all negative features, where $\ell(\mathbf{n}_i) > \ell(\mathbf{n}_j)$ for all $i < j$, implying that the negative features are sorted based on decreasing similarity to the query. The most challenging negatives are selected by truncating the ordered set $\hat{\mathcal{Q}}$, retaining only the first $N < K$ elements, denoted as $\hat{\mathcal{Q}}^N$. For the remainder of this section, let $S^{(i)} = \{\mathbf{s}_1^{(i)}, \mathbf{s}_2^{(i)}, \ldots, \mathbf{s}_{N_i}^{(i)}\}$ denote the set of synthetic negatives to be generated for type i, where $i \in \{1, 2, 3, 4, 5, 6\}$ represents the six different synthetic negative generation strategies and N_i denotes the cardinality of each set (*i.e.*, the number of synthetic negatives generated for the i-th strategy). Note that all synthetic hard negatives are ℓ_2-normalized before added to the set of negative logits for the query.

Interpolated synthetic negatives (type 1). Our first strategy creates synthetic negatives through controlled interpolation between samples, similar to MoCHI's type 2. This approach aims to generate features that lie in meaningful regions of the representation space between the query and existing hard negatives. For each query $\mathbf{q}$, we propose to generate N_1 synthetic hard negative features by mixing the query $\mathbf{q}$ with a randomly chosen feature from the N hardest negatives in $\hat{\mathcal{Q}}^N$. Then a synthetic negative feature $\mathbf{s}_k^1 \in S^1$ is given by:

$$\mathbf{s}_k^1 = \alpha_k \cdot \mathbf{q} + (1 - \alpha_k) \cdot \mathbf{n}_i, \quad \alpha_k \in (0, \alpha_{\text{max}}) \tag{4}$$

where $\mathbf{n}_i \in \hat{\mathcal{Q}}^N$ and α_k is randomly sampled from a uniform distribution in the range $(0, \alpha_{\text{max}})$. Interpolation creates a synthetic embedding that lies between the query and the negative in the representation space. We set $\alpha_{\text{max}} = 0.5$ to guarantee that the contribution of the query is always less than that of the negative.

Extrapolated synthetic negatives (type 2). As a natural extension of interpolation, we propose extrapolation to explore the "opposite" direction in feature space. While this approach operates further from the decision boundary, we carefully control the exploration through coefficients to balance the difficulty of synthetic negatives, ensuring they provide meaningful learning signals without making the contrastive task intractable. For each query $\mathbf{q}$, we propose to generate N_2 hard negative features by extrapolating beyond the query embedding in the direction of the hardest negative features. Similar to the interpolated method, we use a randomly chosen feature from the N hardest negatives in $\hat{\mathcal{Q}}^N$. Then $\mathbf{s}_k^2 \in S^2$ is given by:

$$\mathbf{s}_k^2 = \mathbf{n}_i + \beta_k \cdot (\mathbf{n}_i - \mathbf{q}), \quad \beta_k \in (1, \beta_{\text{max}}) \tag{5}$$

where $\mathbf{n}_i \in \hat{\mathcal{Q}}^N$ and β_k is randomly sampled from a uniform distribution in the range $(1, \beta_{\text{max}})$. Extrapolation generates a synthetic embedding that lies beyond the query embedding in the direction of the hardest negative. We choose $\beta_{\text{max}} = 1.5$.

Mixup synthetic negatives (type 3). We propose to generate challenging synthetic negatives by combining pairs of hard negative examples, similar to MoCHI's type 1. For each query $\mathbf{q}$, we generate N_3 hard negative features by combining pairs of the N hardest existing negative features in $\hat{\mathcal{Q}}^N$. For $\mathbf{s}_k^3 \in S^3$ we have:

$$\mathbf{s}_k^3 = \gamma_k \cdot \mathbf{n}_i + (1 - \gamma_k) \cdot \mathbf{n}_j, \quad \gamma_k \in (0, 1) \tag{6}$$

where $\mathbf{n}_i, \mathbf{n}_j \in \hat{\mathcal{Q}}^N$ and γ_k is randomly sampled from a uniform distribution in the range $(0, 1)$. Mixup combines pairs of the hardest existing negative features to create a synthetic embedding that represents a blend of challenging cases.

Noise-injected synthetic negatives (type 4). To prevent overfitting to specific negative patterns while maintaining the essential characteristics of hard negatives, we introduce controlled stochasticity through noise injection. For each query $\mathbf{q}$, we propose to generate N_4 hard negative features by adding Gaussian noise to the hardest negative features. Each synthetic negative feature $\mathbf{s}_k^4 \in S^4$ is given by:

$$\mathbf{s}_k^4 = \mathbf{n}_i + \mathcal{N}(\mathbf{0}, \sigma^2 \cdot \mathbf{I}) \tag{7}$$

where $\mathbf{n}_i \in \hat{\mathcal{Q}}^N$ and $\mathcal{N}(\mathbf{0}, \sigma^2 \cdot \mathbf{I})$ represents Gaussian noise with standard deviation σ (where $\mathbf{I}$ is the identity matrix). Noise injection adds Gaussian noise to the hardest negative features, resulting in a synthetic embedding with added randomness.

Perturbed synthetic negatives (type 5). Drawing inspiration from adversarial training [36], we introduce perturbed synthetic negatives that use gradient-based perturbations with variable magnitudes. For each query $\mathbf{q}$, we propose to generate N_5 hard negative features by perturbing the embeddings of the hardest negative features. We formulate each synthetic negative feature $\mathbf{s}_k^5 \in S^5$ as:

$$\mathbf{s}_k^5 = \mathbf{n}_i + \delta \cdot \nabla_{\mathbf{n}_i} \text{sim}(\mathbf{q}, \mathbf{n}_i) \tag{8}$$

where $\mathbf{n}_i \in \hat{\mathcal{Q}}^N$ and $\text{sim}(\cdot, \cdot)$ is the similarity function and δ controls the perturbation magnitude. Perturbation modifies the embeddings of the hardest negative features based on the gradient of the similarity function, creating synthetic negatives that are slightly adjusted to be more challenging for the model. This approach offers greater flexibility than fixed interpolation, as it generalizes to arbitrary similarity functions and can generate negatives of varying hardness.

Adversarial synthetic negatives (type 6). While similar in concept to type 5, adversarial synthetic negatives differ fundamentally in their gradient scaling approach. For each query $\mathbf{q}$, we propose to generate N_6 hard negative features by applying adversarial perturbations to the hardest negative features to maximize their similarity to the query embeddings. Each synthetic negative $\mathbf{s}_k^6 \in S^6$ is defined as:

$$\mathbf{s}_k^6 = \mathbf{n}_i + \eta \cdot \text{sign}(\nabla_{\mathbf{n}_i} \text{sim}(\mathbf{q}, \mathbf{n}_i)) \tag{9}$$

where $\mathbf{n}_i \in \hat{\mathcal{Q}}^N$ and η controls the perturbation magnitude. Adversarial hard negatives apply adversarial perturbations to the hardest negative features, specifically altering them to maximize their similarity to the query embeddings, thereby producing the most challenging contrasts. Where type 5 allows variable perturbation sizes, type 6 enforces unit magnitude through the sign function.

4.2. Integrating Synthetic Hard Negatives into the Contrastive Loss

The synthetic hard negatives generated are integrated into the contrastive learning process by modifying the InfoNCE loss. Let $\mathcal{S} = \bigcup_{i=1}^{6} S^{(i)}$ represent the concatenation of all synthetic hard negatives, where $S^{(i)}$ is the set of synthetic negatives generated by the i-th strategy. This combined set of synthetic negatives augments the original negatives $\mathcal{Q}$, providing a more diverse and challenging set of contrasts for the query. The modified InfoNCE loss is given by:

$$\mathcal{L}(\mathbf{q}, \mathbf{k}, \mathcal{Q}, \mathcal{S}) = -\log \frac{\exp(\mathbf{q}^\top \cdot \mathbf{k}/\tau)}{\exp(\mathbf{q}^\top \cdot \mathbf{k}/\tau) + Z} \tag{10}$$

where Z represents the negative samples:

$$Z = \underbrace{\sum_{\mathbf{n} \in \mathcal{Q}} \exp(\mathbf{q}^\top \cdot \mathbf{n}/\tau)}_{\text{memory-based negatives}} + \underbrace{\sum_{\mathbf{s} \in \mathcal{S}} \exp(\mathbf{q}^\top \cdot \mathbf{s}/\tau)}_{\text{synthetic negatives}}. \tag{11}$$

Here, $\mathcal{Q}$ is the set of original memory-based negatives, and $\mathcal{S}$ is the set of all synthetic hard negatives. By incorporating both real and synthetic negatives, the model is exposed to a wider variety of challenging examples, which encourages learning more robust and generalizable representations. The overall computational overhead of SynCo is roughly equivalent to increasing the queue/memory by $\sum_{i=1}^{6} N_i \ll K$, along with the additional yet negligible cost of generating the synthetic negatives. Since synthetic negatives are generated "*on-the-fly*" during training and can be efficiently computed in parallel with the forward pass, the additional computational cost is marginal compared to the base contrastive learning framework. Moreover, the memory footprint remains manageable as synthetic negatives do not need to be stored persistently in the memory bank. Section 8 of the *supplementary material* provides a detailed description of the SynCo algorithm (see Algorithm 1-7), with a toy illustration in Section 10.12 of the *supplementary*.

5. Experiments

In this section, we present comprehensive experiments demonstrating SynCo's effectiveness on ImageNet linear evaluation, semi-supervised learning, and transfer learning to object detection. The *supplementary material* contains: **(i)** implementation details (Sec. 9); **(ii)** Type 1–6 ablations (Sec. 11.1); **(iii)** hyperparameter sensitivity (σ, δ, η, N, N_i, K, Sec. 11.2); **(iv)** vision transformer experiments [16, 49] (Sec. 10.13); **(v)** robustness (Sec. 10.6, 10.7); **(vi)** visualizations, *e.g.*, t-SNE (Sec. 10.8), GradCAM (Sec. 10.9), UMAP (Sec. 10.10), nearest neighbor retrieval (Sec. 10.11).

5.1. Implementation Details

We pretrain SynCo on ImageNet ILSVRC-2012 [15] using a ResNet-50 encoder [25]. Our method builds upon MoCo-v2 [13]; thus, it is only *fair* to compare against other MoCo-based methods [13, 32, 34, 62], which share similar architectures and training setups (see ***bold*** entries in Tables 1 to 4, indicating the best performance among MoCo-based methods-we *underline* the second best). For training we use $K = 65$k. For SynCo, we also have a warm-up of 10 epochs, *i.e.* for the first epochs we do *not* synthesize hard negatives. We empirically set SynCo's hyperparameters σ, δ, η to 0.01. For hard negative generation, we use the top $N = 1024$ hardest negatives, with $N_1 = N_2 = N_3 = 256$ and $N_4 = N_5 = N_6 = 64$. For ImageNet linear evaluation, we train a linear classifier on frozen features for 100 epochs, using a batch size of 256 and a cosine learning rate schedule ($lr = 30.0$). To evaluate transfer learning, we apply SynCo to object detection tasks. For PASCAL VOC [19], we fine-tune a Faster R-CNN [42] on `trainval07+12` and test on `test2007`. For COCO [35], we use a Mask R-CNN [26], fine-tuning on `train2017` and evaluating on `val2017`. We employ Detectron2 [56] and report standard AP metrics, following [27] *without* additional tuning.

5.2. Linear Evaluation on ImageNet

We evaluate SynCo by training a linear classifier on ImageNet-pretrained frozen features. With 200 epochs of pretraining, SynCo obtains 67.9% $\pm$ 0.16% top-1 accuracy and 88.0% $\pm$ 0.05% top-5, showing clear gains over MoCo-based methods (**+0.4%** over MoCo-v2, **+1.0%** over MoCHI, **+0.3%** over PCL-v2 and DCL). While MoCHI's hard negative generation yields lower performance than MoCo-v2, our synthetic hard negatives provide consistent and stable improvements. With 800 epochs, SynCo reaches 70.7% top-1 (**+2.0%** over MoCHI). An extended comparison is provided in Sec. 11.1 of the *supplementary material*. However, at 800 epochs, it still does not surpass MoCo-v2, similar to MoCHI, likely due to an overly hard proxy task. We observe that SynCo's performance plateaus around epoch 400 (see Sec. 10.3, Fig. 5 of the *supplementary*), indicating that continued synthetic negative generation makes the proxy task increasingly challenging in later stages. Motivated by this observation, we stop generating synthetic negatives after epoch 400, allowing the model to consolidate learned features without overwhelming the contrastive objective; this yields 71.6% top-1, a **+0.5%** improvement over MoCo-v2.

Method	Top-1	Top-5
Supervised	76.5	-
PIRL [38]	63.6	-
LA [70]	60.2	-
CMC [46]	60.0	-
SimSiam [12]	68.1	-
ReSSL [67]	62.9	-
AdCo [29]	68.6	-
SimCLR + DCL [62]	65.8	-
MoCo-based		
MoCo [27]	60.7 ↓6.8	-
PCL-v1 [34]	61.5 ↓6.0	-
MoCo-v2 [13] (*baseline*)	67.5 ↑0.0	90.1
MoCHI [32]	66.9 ↓0.6	-
PCL-v2 [34]	67.6 ↑0.1	-
MoCo-v2 + DCL [62]	67.6 ↑0.1	-
MoCo-v2 + NS [21]	<u>67.9</u> ↑0.4	-
SynCo (ours)	**67.9/68.1** ↑0.6	88.0

Table 1. **Linear evaluation on ImageNet ILSVRC-2012.** Top-1 and top-5 accuracies (in %) with 200 epochs of pretraining using ResNet-50. Results for SynCo are given as avg./max over 3 runs.

Method	Epochs	Top-1	Top-5
InfoMin [47]	800	73.0	91.1
SimSiam [12]	800	68.1	-
SimCLR [11]	1000	69.3	-
BYOL [23]	1000	74.3	91.6
DINO [10]	800	75.3	-
Barlow Twins [64]	1000	73.2	91.0
AdCo [29]	800	72.8	-
VICReg [6]	1000	73.2	91.1
CaCo [54]	800	74.1	-
All4One [18]	800	66.6	87.5
MoCo-based			
MoCo-v2 [13]	800	<u>71.1</u> ↑0.0	90.1
MoCHI [32]	800	68.7 ↓2.4	-
SynCo (ours)	800	70.7 ↓0.4	89.8
SynCo† (ours)	800	**71.6** ↑0.5	90.5

Table 2. **Linear evaluation on ImageNet ILSVRC-2012.** Top-1 and top-5 accuracies (in %) for models trained with *extended epochs* using ResNet-50. Results for SynCo are based on 1 run. † We stop generating synthetic negatives at epoch 400.

Method	Epochs	Top-1		Top-5	
		1%	10%	1%	10%
Supervised		25.4	56.4	48.4	80.4
InstDis [57]	200	-	-	39.2	77.4
PIRL [38]	800	30.7	60.4	57.2	83.8
SimCLR [11]	1000	48.3	65.6	75.5	87.8
BYOL [23]	1000	53.2	68.8	78.4	89.0
SwAV [9]	800	53.9	70.2	78.5	89.9
Barlow Twins [64]	1000	55.0	69.7	79.2	89.3
VICReg [6]	1000	54.8	69.5	79.4	89.5
All4One [18]	800	39.0	-	60.0	-
MoCo-based					
MoCo-v2 (repr.)	800	48.2	66.1	75.8	87.6
PCL-v1 [34]	200	-	-	75.3	85.6
PCL-v2 [34]	200	-	-	73.9	85.0
MoCHI (repr.)	800	50.4	65.7	76.2	87.2
SynCo (ours)	800	50.8	66.6	77.5	88.0
SynCo† (ours)	800	**51.2**	**67.1**	**78.0**	**88.7**

Table 3. **Semi-supervised learning on ImageNet ILSVRC-2012.** Top-1 and top-5 accuracies with 1% and 10% training examples using ResNet-50. Results for SynCo are averaged over 3 runs.

5.3. Semi-supervised Training on ImageNet

We evaluate SynCo in a semi-supervised setting using 1% and 10% of labeled ImageNet data (and 100% in Sec. 10.4, Fig. 6 of the *supplementary*). Results in Table 3 show that with 1% labels, SynCo achieves 50.8% ± 0.21% top-1 accuracy (**+25.4%** over supervised baseline, **+2.6%** over MoCo-v2, **+2.5%** over SimCLR) and 77.5% ± 0.12% top-5 accuracy. With 10% labels, it reaches 66.6% ± 0.19% top-1 (**+10.2%** over supervised, **+0.5%** over MoCo-v2, **+1.0%** over SimCLR) and 88.0% ± 0.10% top-5 accuracy. Interestingly, when we stop generating synthetic negatives after epoch 400, similar to our observation in linear evaluation, performance improves further to 51.2% ± 0.23% top-1 and 78.0% ± 0.14% top-5 with 1% labels, and 67.1% ± 0.20% top-1 and 88.7% ± 0.11% top-5 with 10% labels.

5.4. Transferring to Detection

We evaluate the SynCo representation, pretrained for 200 epochs, by applying it to detection tasks. Results in Table 4 and Sec. 10.2 (Tab. 9) of the *supplementary* show that on PASCAL VOC, SynCo achieves strong results comparable to MoCHI, while significantly outperforming the supervised baseline (**+3.7** AP). On the more challenging COCO dataset, with 1× schedule, SynCo shows consistent improvements over the supervised baseline (AP^{bb} **+1.7**, AP^{msk} **+1.6**) and MoCo-v2 (AP^{bb} **+1.0**, AP^{msk} **+0.8**). SynCo achieves competitive performance with detection-specific methods, showing comparable results to DetCo [59] and InsLoc [60].

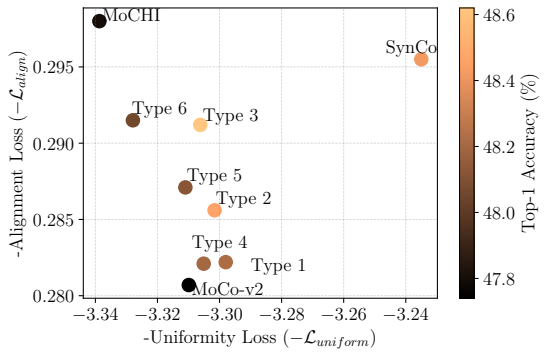


Figure 3. **Alignment and uniformity on ImageNet-100.** Comparison of MoCo-v2, MoCHI, and SynCo (various configurations). The x- and y-axis represent $-\mathcal{L}_{\text{uniform}}$ and $-\mathcal{L}_{\text{align}}$, respectively. The model with the highest performance is located in the upper-right corner of the chart.

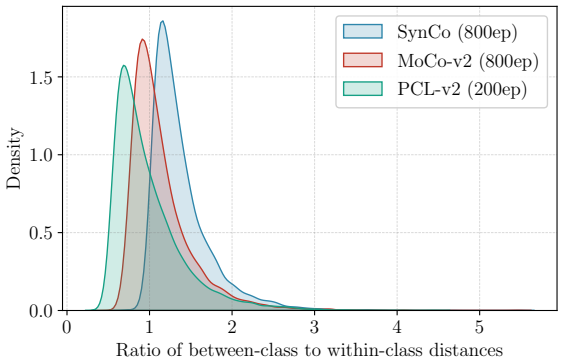


Figure 4. **Distribution of the ratio between inter-class and intra-class distances for MoCo-based methods on ImageNet ILSVRC-2012.** Higher values indicate better class separation. For clarity, we only show MoCo-v2 (800 epochs), PCL-v2 (200 epochs), and SynCo (800 epochs).

6. Discussion

This section examines how synthetic negatives affect proxy task difficulty and shape representation space utilization.

6.1. Is the Proxy Task More Difficult?

We observe that incorporating synthetic negatives leads to faster learning and improved performance. Each synthetic negative type accelerates learning compared to MoCo-v2, with the full SynCo configuration showing the most significant improvement and lowest final proxy task performance, as shown in Figure 2b. This indicates SynCo presents the most challenging proxy task, evidenced by $\max \ell(\mathbf{s}_k^i) > \max \ell(\mathbf{n}_j)$, where $\mathbf{s}_k^i \in S^i$ are synthetic negatives and $\mathbf{n}_j \in \hat{\mathcal{Q}}_N$ are original negatives. Through SynCo, we modulate proxy task difficulty via synthetic negatives, pushing the model to learn more robust features.

Method	Epochs	COCO 1× schedule						COCO 2× schedule						PASCAL VOC		
		AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{msk}	AP^{msk}_{50}	AP^{msk}_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{msk}	AP^{msk}_{50}	AP^{msk}_{75}	AP	AP_{50}	AP_{75}
Supervised	200	38.2	58.2	41.2	33.3	54.7	35.2	40.0	59.9	43.1	34.7	56.5	36.9	53.5	81.3	58.8
Random init	200	26.4	44.0	27.8	29.3	46.9	30.8	35.6	54.6	38.2	31.4	51.5	33.5	33.8	60.2	33.1
SimSiam [12]	200	39.2	59.3	42.1	34.4	56.0	36.7	-	-	-	-	-	-	57.0	82.4	63.7
BYOL [23]	300	-	-	-	-	-	-	40.3	60.5	43.9	35.1	56.8	37.3	51.9	81.0	56.5
SwAV [9]	800	38.4	58.6	41.3	33.8	55.2	35.9	-	-	-	-	-	-	56.1	82.6	62.7
SimCLR [11]	1000	-	-	-	-	-	-	40.3	60.5	43.9	35.1	56.8	37.3	56.3	81.9	62.5
Barlow Twins [64]	1000	39.2	59.0	42.5	34.3	56.0	36.5	-	-	-	-	-	-	56.8	82.6	63.4
Detection-specific																
SoCo [55]	100	40.4	60.4	43.7	34.9	56.8	37.0	41.1	61.0	44.4	35.6	57.5	38.0	59.1	83.4	65.6
InsLoc [60]	200	39.5	59.1	42.7	34.5	56.0	36.8	41.4	60.9	45.0	35.9	57.6	38.4	57.9	82.9	64.9
DetCo [59]	200	39.8	59.7	43.0	34.7	56.3	36.7	41.3	61.2	45.0	35.8	57.9	38.2	57.8	82.6	64.2
ReSim [58]	200	39.7	59.0	43.0	34.6	55.9	37.1	-	-	-	-	-	-	58.7	83.1	66.3
MoCo-based																
MoCo [27]	200	38.5	58.3	41.6	33.6	54.8	35.6	40.7	60.5	44.1	35.4	57.3	37.6	55.9	81.5	62.6
MoCo-v2 [13]	200	38.9	58.4	42.0	34.2	55.2	36.5	40.7	60.5	44.1	35.6	**57.4**	37.1	57.0	82.4	63.6
MoCHI [32]	200	39.2	58.9	42.4	34.3	55.5	36.6	-	-	-	-	-	-	**57.5**	**82.7**	**64.4**
SynCo (ours)	200	**39.9**	**59.6**	**43.3**	**34.9**	**56.5**	**36.9**	**41.0**	**60.6**	**44.8**	**35.7**	**57.4**	**38.1**	57.2	82.6	63.9

Table 4. **Transfer learning results using R50-C4 on COCO with 1× and 2× training schedules and PASCAL VOC07+12.** For COCO, we report AP^{bb} (bounding box detection) and AP^{msk} (instance segmentation). For VOC, we report standard AP metrics. Results for SynCo are averaged over 3 runs.

6.2. Evaluating the Usage of the Representation Space

To assess learned representations, we employ alignment and uniformity metrics [52]. These metrics provide insights into representation space utilization, with alignment quantifying grouping of similar samples and uniformity measuring spread across the hypersphere. Figure 3 presents results for MoCo-based methods. Our findings demonstrate that SynCo significantly improves representation uniformity compared to MoCo-v2 and MoCHI, showing improved utilization of the representation space. Furthermore, incorporating synthetic negatives (types 1 to 6) leads to improved alignment. These results suggest that SynCo's approach yields stronger and more well-distributed feature representations.

6.3. Class Concentration Analysis

To quantify the learned latent space structure, we examine the relationship between within- and between-class distances. Figure 4 (and Sec. 10.5, Fig. 7 of the *supplementary*) shows the distribution of ratios between inter- and intra-class ℓ_2-distances for representations learned by MoCo-based contrastive methods. A higher mean ratio indicates better concentration within classes while maintaining greater separation between classes, reflecting improved linear separability (aligned with Fisher's linear discriminant analysis [20]). After 800 training epochs, SynCo achieves a mean ratio of 1.384, surpassing MoCo-v2 (1.146) and PCL-v2 (0.988).

7. Conclusion

This paper introduces SynCo, a novel approach leveraging synthetic hard negatives to enhance contrastive visual representation learning. By generating diverse negatives "*on-the-fly*" using six complementary strategies, SynCo demonstrates consistent improvements over MoCo-based methods without significant computational overhead. Our experiments show gains on ImageNet linear evaluation, semi-supervised learning, and transfer to detection tasks. The proposed strategies are general and applicable to any contrastive method utilizing InfoNCE loss, *e.g.*, SimCLR [11].

Scope and fair comparison. We deliberately focus on MoCo-based comparisons to ensure fair evaluation under identical training conditions. Recent self-distillation methods like DINO-v2/v3 [40, 45] or iBOT [68] operate in fundamentally different paradigms, *i.e.*, they use teacher–student architectures, multiple crops, and train on billions of samples with larger computational budgets. Comparing SynCo to these methods would conflate the benefits of synthetic negatives with differences in architecture, data scale, and training. Our contribution specifically targets improving contrastive learning through diverse negative sampling. Within this scope, the consistent improvements across downstream tasks validate our approach (Tabs. 1 to 4), since small gains in self-supervised learning often translate to significant benefits when deployed at scale (see Sec. 13).

Acknowledgments

We would like to express our gratitude to Andreas Floros for his valuable feedback, particularly his assistance with equations, notations, and insightful discussions that greatly contributed to this work. We also acknowledge the computational resources and support provided by the Imperial College Research Computing Service (http://doi.org/10.14469/hpc/2232), which enabled our experiments.

References

[1] Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. A theoretical analysis of contrastive unsupervised representation learning, 2019. 3

[2] Muhammad Awais, Muzammal Naseer, Salman Khan, Rao Muhammad Anwer, Hisham Cholakkal, Mubarak Shah, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Foundational models defining a new era in vision: A survey and outlook, 2023. 2

[3] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views, 2019. 2

[4] Randall Balestriero, Mark Ibrahim, Vlad Sobal, Ari Morcos, Shashank Shekhar, Tom Goldstein, Florian Bordes, Adrien Bardes, Gregoire Mialon, Yuandong Tian, Avi Schwarzschild, Andrew Gordon Wilson, Jonas Geiping, Quentin Garrido, Pierre Fernandez, Amir Bar, Hamed Pirsiavash, Yann LeCun, and Micah Goldblum. A cookbook of self-supervised learning, 2023. 2

[5] Wele Gedara Chaminda Bandara, Celso M. De Melo, and Vishal M. Patel. Guarding barlow twins against overfitting with mixed samples, 2023. 2

[6] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning, 2022. 2, 6, 7

[7] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicregl: Self-supervised learning of local visual features, 2022. 2

[8] Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, S. Buch, Dallas Card, Rodrigo Castellon, Niladri S. Chatterji, Annie S. Chen, Kathleen A. Creel, Jared Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren E. Gillespie, Karan Goel, Noah D. Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas F. Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, O. Khattab, Pang Wei Koh, Mark S. Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir P. Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Benjamin Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, J. F. Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Robert Reich, Hongyu Ren, Frieda Rong, Yusuf H. Roohani, Camilo Ruiz, Jack Ryan, Christopher R'e, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishna Parasuram Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei A. Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. On the opportunities and risks of foundation models. *ArXiv*, 2021. 2

[9] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances in Neural Information Processing Systems*, pages 9912–9924. Curran Associates, Inc., 2020. 2, 7, 8

[10] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers, 2021. 2, 6

[11] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020. 1, 2, 3, 6, 7, 8

[12] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning, 2020. 2, 6, 8

[13] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning, 2020. 1, 2, 3, 6, 8

[14] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers, 2021. 3

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 6

[16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021. 6

[17] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations, 2021. 2

[18] Imanol G. Estepa, Ignacio Sarasúa, Bhalaji Nagarajan, and Petia Radeva. All4one: Symbiotic neighbour contrastive learning via self-attention and redundancy reduction, 2023. 6, 7

[19] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2009. 6

[20] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, second edition, 2009. 8

[21] Songwei Ge, Shlok Mishra, Haohan Wang, Chun-Liang Li, and David Jacobs. Robust contrastive learning using negative samples with diminished semantics, 2022. 2, 6

[22] Nikolaos Giakoumoglou, Tania Stathaki, and Athanasios Gkelias. A review on discriminative self-supervised learning methods in computer vision, 2025. 2

[23] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning, 2020. 2, 6, 7, 8

[24] 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)*, pages 1735–1742, 2006. 3

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 6

[26] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn, 2018. 6

[27] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning, 2020. 1, 2, 3, 6, 8

[28] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization, 2019. 2

[29] Qianjiang Hu, Xiao Wang, Wei Hu, and Guo-Jun Qi. Adco: Adversarial contrast for efficient learning of unsupervised representations from self-trained negative adversaries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1074–1083, 2021. 6

[30] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Mining on manifolds: Metric learning without labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7842–7851, 2018. 3

[31] Ruijie Jiang, Thuan Nguyen, Shuchin Aeron, and Prakash Ishwar. Hard-negative sampling for contrastive learning: Optimal representation geometry and neural- vs dimensional-collapse. *Transactions on Machine Learning Research*, 2025. 2

[32] Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning, 2020. 2, 3, 6, 8

[33] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning, 2021. 2

[34] Junnan Li, Pan Zhou, Caiming Xiong, and Steven C. H. Hoi. Prototypical contrastive learning of unsupervised representations, 2021. 2, 6, 7

[35] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015. 6

[36] Mohammad Mehrabi, Adel Javanmard, Ryan A. Rossi, Anup Rao, and Tung Mai. Fundamental tradeoffs in distributionally adversarial training. In *Proceedings of the 38th International Conference on Machine Learning*, pages 7544–7554. PMLR, 2021. 5

[37] Anastasiya Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Working hard to know your neighbor's margins: Local descriptor learning loss. In *Advances in Neural Information Processing Systems*, pages 4826–4837, 2017. 3

[38] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations, 2019. 2, 3, 6, 7

[39] Jovana Mitrovic, Brian McWilliams, Jacob Walker, Lars Buesing, and Charles Blundell. Representation learning via invariant causal mechanisms, 2020. 2

[40] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023. 2, 8

[41] Colorado J Reed, Sean Metzger, Aravind Srinivas, Trevor Darrell, and Kurt Keutzer. Selfaugment: Automatic augmentation policies for self-supervised learning, 2021. 2

[42] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks, 2016. 6

[43] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples, 2021. 2

[44] Renan A. Rojas-Gomez, Karan Singhal, Ali Etemad, Alex Bijamov, Warren R. Morningstar, and Philip Andrew Mansfield. Sassl: Enhancing self-supervised learning via neural style transfer, 2024. 2

[45] Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. Dinov3, 2025. 8

[46] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding, 2020. 2, 3, 6

[47] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? In *Advances in Neural Information Processing Systems*, pages 6827–6839. Curran Associates, Inc., 2020. 2, 6

[48] Nenad Tomasev, Ioana Bica, Brian McWilliams, Lars Buesing, Razvan Pascanu, Charles Blundell, and Jovana Mitrovic. Pushing the limits of self-supervised resnets: Can we outperform supervised learning without labels on imagenet?, 2022. 2

[49] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention, 2021. 6

[50] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2019. 2, 3

[51] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. *arXiv preprint arXiv:1806.05236*, 2018. 2

[52] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020. 8

[53] Xiao Wang and Guo-Jun Qi. Contrastive learning with stronger augmentations, 2022. 2

[54] Xiao Wang, Yuhang Huang, Dan Zeng, and Guo-Jun Qi. Caco: Both positive and negative samples are directly learnable via cooperative-adversarial contrastive learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 6

[55] Fangyun Wei, Yue Gao, Zhirong Wu, Han Hu, and Stephen Lin. Aligning pretraining for detection via object-level contrastive learning, 2021. 8

[56] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 6

[57] Zhirong Wu, Yuanjun Xiong, Stella Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance-level discrimination, 2018. 2, 3, 7

[58] Tete Xiao, Colorado J Reed, Xiaolong Wang, Kurt Keutzer, and Trevor Darrell. Region similarity representation learning, 2021. 8

[59] Enze Xie, Jian Ding, Wenhai Wang, Xiaohang Zhan, Hang Xu, Peize Sun, Zhenguo Li, and Ping Luo. Detco: Unsupervised contrastive learning for object detection, 2021. 7, 8

[60] Ceyuan Yang, Zhirong Wu, Bolei Zhou, and Stephen Lin. Instance localization for self-supervised detection pretraining, 2021. 7, 8

[61] Zhen Yang, Ming Ding, Tinglin Huang, Yukuo Cen, Junshuai Song, Bin Xu, Yuxiao Dong, and Jie Tang. Does negative sampling matter? a review with insights into its theory and applications, 2024. 2

[62] Chun-Hsiao Yeh, Cheng-Yao Hong, Yen-Chi Hsu, Tyng-Luh Liu, Yubei Chen, and Yann LeCun. Decoupled contrastive learning, 2022. 2, 6

[63] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019. 2

[64] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction, 2021. 2, 6, 7, 8

[65] Chaoning Zhang, Kang Zhang, Trung X. Pham, Axi Niu, Zhinan Qiao, Chang D. Yoo, and In So Kweon. Dual temperature helps contrastive learning without many negative samples: Towards understanding and simplifying moco, 2022. 1

[66] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. Mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 2

[67] Mingkai Zheng, Shan You, Fei Wang, Chen Qian, Changshui Zhang, Xiaogang Wang, and Chang Xu. Ressl: Relational self-supervised learning with weak augmentation, 2021. 6

[68] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer, 2022. 8

[69] Jiachen Zhu, Rafael M. Moraes, Serkan Karakulak, Vlad Sobol, Alfredo Canziani, and Yann LeCun. Tico: Transformation invariance and covariance contrast for self-supervised visual representation learning, 2022. 2

[70] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings, 2019. 2, 6