INCREMENTAL FALSE NEGATIVE DETECTION FOR CONTRASTIVE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Self-supervised learning has recently shown great potential in vision tasks through contrastive learning which aims to discriminate each image, or instance, in the dataset. However, such instance-level learning ignores the semantic relationship among instances and sometimes undesirably repels the anchor from the semantically similar samples, termed as "false negatives". In this work, we first empirically highlight that the unfavorable effect from false negatives is more significant for the large-scale datasets with more semantic concepts. To address the issue, we propose a novel self-supervised contrastive learning framework that incrementally detects and explicitly removes the false negative samples. Specifically, following the training process, our method dynamically detects increasing high-quality false negatives considering that the encoder gradually improves and the embedding space becomes more semantically structural. Next, we discuss two strategies to explicitly remove the detected false negatives during contrastive learning. Extensive experiments show that our framework outperforms other self-supervised contrastive learning methods on multiple benchmarks within a limited compute.

1 INTRODUCTION

Self-supervised learning of visual representation (Doersch et al., 2015; Pathak et al., 2016; Noroozi & Favaro, 2016; Oord et al., 2018; Gidaris et al., 2018; Chen et al., 2020a) aims to learn a semantic-aware embedding space based on the image data without the supervision of human-labeled annotations. Recently, significant advances have been made by contrastive learning approaches (Hjelm et al., 2019; Tian et al., 2019; Chen et al., 2020b;c; He et al., 2020; Chen et al., 2020d) to reduce the performance gap with supervised counterparts. Most unsupervised contrastive learning methods are developed based on the instance discrimination assumption. These approaches treat each training sample (i.e., instance) individually and learn the representations that discriminate every single sample. Specifically, considering an arbitrary image as an anchor, the only positive sample is generated by applying a different data augmentation to the anchor image, while all other images are treated as negative samples. The training objective is to attract the positive to the anchor while repelling the negative samples from the anchor.

While instance-level contrastive learning has shown impressive performance, these methods do not take the semantic relationship among images into consideration. Although some images share similar semantic concepts with the anchor, they are still considered as negative samples and are equally pushed away from the anchor with all other negatives. We define these samples as "false negatives" in self-supervised contrastive learning, which would adversely affect the representation learning (Saunshi et al., 2019). One might argue that such undesirable effects might be minor for the datasets containing images with diverse semantic concepts since the probabilities of drawing false negatives are relatively low. However, as shown in Figure 1, we empirically find the opposite results that the performance drops due to training with false negative samples are more significant on large-scale datasets with more semantic classes. These results bring up the issue of the capability of instance-level contrastive learning when it is applied to datasets with more complex semantic contents.

To handle the false negative issue, we propose a novel framework to *incrementally detect and explicitly remove false negatives* for self-supervised contrastive learning. During training, we cluster samples in the embedding space and assign the cluster indices as the pseudo labels to the training images. Instances with an identical label to the anchor are then detected as false negatives. However,

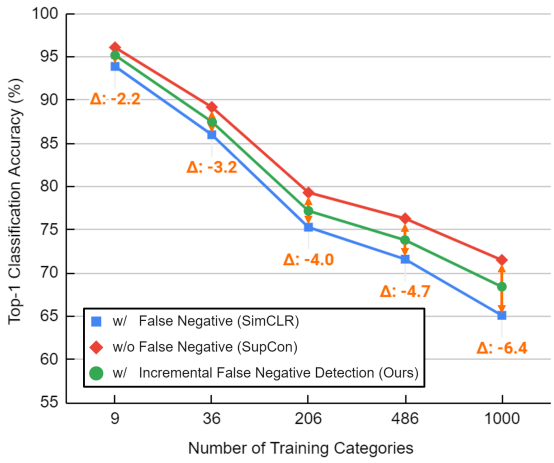

Figure 1: **Effect of false negative for contrastive learning.** To study the effect of false negatives, we compare two frameworks: SimCLR (Chen et al. (2020b), blue) representing instance-level contrastive learning which is trained with false negatives, and SupCon (Khosla et al. (2020), red) benefiting from human-labeled annotations to exclude false negatives. We observe that training with false negatives leads to larger performance gaps (orange) on the datasets with diverse semantic concepts. The proposed approach (green) effectively alleviates such adverse effects by explicitly removing the detected false negatives for self-supervised contrastive learning. See Section 4.3 for the experiment details.

we notice that the pseudo labels generated in the earlier training stages are comparably unreliable to represent the semantic relationship among instances since the semantic structure of the embedding space is still under development. Therefore, we propose a novel strategy to assign the pseudo labels in an incremental manner. In early training stages, we mostly comply with the concept of instance-level learning by only using a small portion of pseudo labels with higher confidence while treating other samples as individual instances. Following the training process, as the encoder is better trained and the clustering algorithm can produce pseudo labels with higher quality, we dynamically include more confident labels to detect and remove false negatives and hence, can benefit semantic-aware representation learning.

Furthermore, we discuss two loss functions (i.e., elimination and attraction) to explicitly address the detected false negatives during contrastive learning. We theoretically show that they are the generalization of triplet loss (Schroff et al., 2015) with multiple weighted negative and positive samples. While both of them can fully avoid the false negatives being pushed apart from the anchor, the attraction further minimizes the distances between the anchor and its false negatives. Empirically, we find that such a more aggressive strategy is less tolerant to the noisy pseudo labels and unsuitable to our self-supervised framework. The main contributions of this paper are summarized as follows:

- This paper shows the first attempt to highlight that in contrastive learning, the adverse effect of false negatives is more severe on large-scale datasets with diverse semantic contents.
- We propose a novel contrastive learning framework with incremental false negative detection (IFND) that smoothly bridges instance-level and semantic-aware contrastive learning. Our approach explicitly detects and removes false negatives in an incremental manner as the embedding space becomes more semantically structural through the training process.
- Our approach performs favorably against the existing self-supervised contrastive learning frameworks on multiple benchmarks within a limited compute. To evaluate the clustering quality, we also introduce an interpretable metric: mean true positive/negative rate.

2 Related Work

Instance-level contrastive learning. Learning semantic-aware visual representation without human supervision is an essential but challenging task. Numerous approaches are developed based on the instance discrimination task (Dosovitskiy et al., 2015; Wu et al., 2018; Ye et al., 2019; Misra & Maaten, 2020), where each instance in the dataset is treated as an individual sample and a representation is learned to classify or contrastively separate samples. Recently, with the contrastive loss (Gutmann & Hyvärinen, 2010) and a set of image transformations or augmentations (Chen et al., 2020b), the instance-level contrastive learning methods (Hjelm et al., 2019; Tian et al., 2019; Chen et al., 2020b;c; He et al., 2020; Chen et al., 2020d) achieve state-of-the-art results and even outperform some supervised pre-training approaches. While significant improvement is achieved, instance-level contrastive learning frameworks typically do not take high-level semantics into ac-

count. For example, any two samples in the dataset would be considered as a negative pair and pushed apart by the contrastive loss although some of them are semantically close and should be nearby in the embedding space. Especially, when the huge amounts of negative samples (8192 for Chen et al. (2020b), 65536 for He et al. (2020)) play a critical role in the success of contrastive learning, these instance-level approaches inevitably draw some "false negatives" and learn less effective representation models as shown by a recent study (Saunshi et al., 2019). In this paper, we further show that the performance gap between the contrastive learning trained with and without false negatives is more significant on the datasets with more semantic classes.

Semantic-aware contrastive learning. To address the limitation of instance-level contrastive learning, several methods are developed to exploit the semantic relationship among images. Chuang et al. (2020) show that the original instance-level sampling strategy includes a sampling bias and propose a debiased contrastive loss. However, this method does not explicitly handle false negative samples. Huynh et al. (2020) determine false negatives of an anchor by finding top k similar samples without referring to global embedding distribution. Khosla et al. (2020) propose a supervised contrastive loss that utilizes label information and treats images of the same class (i.e., false negatives) as positives. However, manually annotated labels cannot be acquired in the unsupervised setting. Closely related to our method, PCL (Li et al., 2021) discovers the representative prototypes for each semantic concept and encourages the image embedding to be closer to its belonging prototype. There are two fundamental differences between PCL and the proposed method: a) PCL uses all assigned prototypes regardless of inferior clustering results in early stages which cannot properly reflect the semantic relationship among images, b) without treating semantically similar samples as false negatives, PCL applies instance-level InfoNCE loss (Oord et al., 2018) as the primary supervision which fails to avoid the images within the same prototype being pushed apart. In contrast, leveraging the progressive detection and explicit removal of false negatives, our framework achieves both greater clustering quality and representation learning. More comparisons are in Section 4.2 and Appendix E.

Clustering for deep unsupervised learning. A few unsupervised learning methods have recently benefited from clustering techniques to construct more effective representation models (Caron et al., 2018; Yang et al., 2016; Xie et al., 2016; Caron et al., 2019; Zhuang et al., 2019; Asano et al., 2020). The main idea is to use cluster indices as pseudo labels and learn visual representations in a supervised manner. However, most of the existing methods require all pseudo labels to optimize the supervised loss, e.g., cross-entropy. In this work, we show that the pseudo labels obtained in earlier training processes are relatively unreliable, and the full adoption of pseudo labels in the early stage would further disrupt the representation learning. To handle this issue, we incorporate instance-level learning with clustering which allows us to only use a set of pseudo labels with sufficient confidence while treating the others as individual instances.

3 Proposed Method

3.1 Instance-Level Contrastive Learning

An instance-level contrastive learning method learns a representation that discriminates one sample from every other. Given M randomly sampled images from a training set $\mathcal{X}$, a contrastive training mini-batch $\mathcal{I}$ consists of $2M$ images obtained by applying two sets of data augmentation on each sampled image. For any anchor image $i \in \mathcal{I}$, the only positive sample is another view (or transformation) of the same image, denoted as i', while the other $2(M-1)$ images jointly constitute the set of negative samples $\mathcal{N}(i)$. The instance-level discrimination is then optimized by the following contrastive loss:

$$\mathcal{L}_{inst} = \sum_{i \in \mathcal{I}} -\log \frac{\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_{i'})}{\sum_{s \in \mathcal{S}(i)} \mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_s)}, \quad \mathcal{S}(i) \equiv \{i', n \mid n \in \mathcal{N}(i)\}, \tag{1}$$

where $\boldsymbol{z}_i = g(f(i))$ is the embedding of the image i from an encoder f and a projection head g, $\mathrm{sim}(\boldsymbol{u}, \boldsymbol{v}) = \exp(\frac{1}{\tau} \frac{(\boldsymbol{u} \cdot \boldsymbol{v})}{\|\boldsymbol{u}\| \|\boldsymbol{v}\|})$ is the similarity of two input vectors, and τ represents a temperature hyper-parameter. For instance-level contrastive learning, the negative sample set $\mathcal{N}(i)$ inevitably includes some samples with similar semantic content as the anchor i, which we term as false negatives, especially when a larger M is commonly used. Consequently, it would separate semantically similar image pairs, which is suboptimal for learning good semantic-aware visual representations.

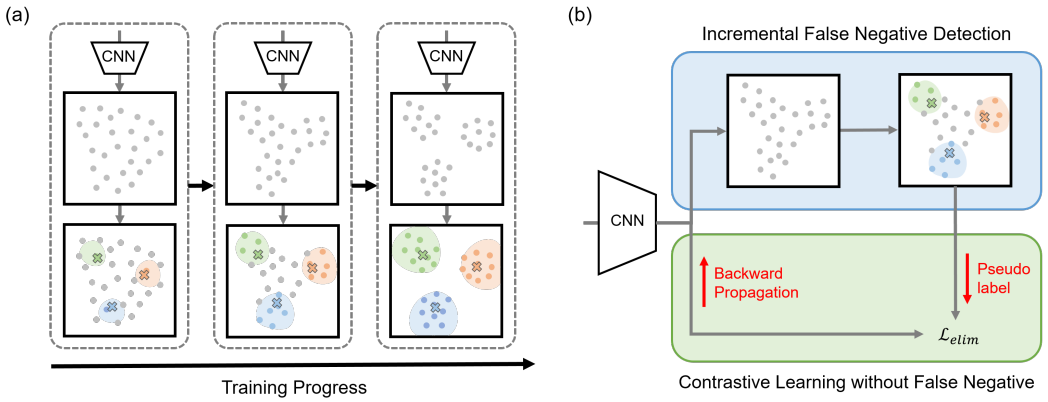

Figure 2: **Algorithmic overview.** (a) Our method uses pseudo labels in an incremental manner with respect to the gradually better-trained encoder and embedding space. (b) The proposed contrastive learning framework uses pseudo labels to explicitly detect and eliminate false negatives from self-supervised contrastive learning.

3.2 Incremental False Negative Detection

To learn an effective semantic-aware representation with self-supervised contrastive learning, our method explicitly detects the false negative samples by performing k-means clustering on the features $f(i)$ of all training images $\{i \in \mathcal{X}\}$ to cluster the features into k groups. We use the centroids to represent the embeddings of the discovered semantic concepts and denote the centroid representation of k-th cluster as $\boldsymbol{c}_k$. The pseudo label y_i of the image i is assigned based on the closest centroid, formally $y_i = \arg\min_k \|f(i) - \boldsymbol{c}_k\|$. In the learning process, two images with an identical assigned pseudo label would be treated as a false negative pair.

Nonetheless, as illustrated in the leftmost of Figure 2(a) (or Figure 3 for a realistic case), we observe that the semantic structure of early learned embedding space is less reliable and could generate noisy pseudo labels which would disrupt the representation learning. We provide a quantitative analysis of this effect in Section 4.4 and Appendix F. To tackle this problem, in early training stages, we only adopt the pseudo labels with higher confidence while treating the other images as individual samples. For an arbitrary image i, the desired confidence κ_i should be larger when the image embedding $f(i)$ is not only closer to the centroid of the belonging group, i.e., $\boldsymbol{c}_{y_i}$, but also farther to the other centroids. To this end, we measure the confidence of an assigned pseudo label by:

$$\kappa_i = \frac{\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{c}_{y_i})}{\sum_{j=1}^{k} \mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{c}_j)}, \tag{2}$$

which is the softmax of cosine similarities between the image representation and every centroid. We then only assign a specific rate (termed as acceptance rate) of pseudo labels with larger confidence. As plotted in the rest part of Figure 2 (or Figure 3), with the training progressing, the encoder is better trained and the semantic structure of embedding space is more clear which can generate more reliable pseudo labels. Hence, we propose to incrementally increase the acceptance rate and leverage semantic-aware representation learning. Empirically, in Section 4.4, we demonstrate that a simple strategy that linearly increases the acceptance rate from 0% to 100% throughout the whole training process can achieve a significant improvement.

Hierarchical semantic definition. In an unsupervised setting, the precise class number of a dataset is usually unknown, and also, a different number of classes can be selected by choosing a different level of granularity. To enhance the flexibility and robustness of our method, we incorporate hierarchical clustering techniques (Johnson, 1967; Corpet, 1988; Murtagh & Legendre, 2014) into our framework. In practice, we perform multiple times of k-means with different cluster numbers to define the false negatives in the different levels of granularity. As such, the objective function (will be introduced in Section 3.3) is revised as the average of the original ones computed by the false negatives defined in different levels of granularity.

3.3 CONTRASTIVE LEARNING WITHOUT FALSE NEGATIVE

Given the assigned pseudo labels, our next goal is to revise the objective of instance-level contrastive learning, i.e., $\mathcal{L}_{inst}$ in Equation 1, to explicitly deal with the detected false negatives. We discuss two strategies to exploit false negatives. First, *false negative elimination* approach straightforwardly excludes the false negatives from the negative sample set:

$$\mathcal{L}_{elim} = \sum_{i \in \mathcal{I}} -\log \frac{\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_{i'})}{\sum_{s \in \mathcal{S}(i)} \text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_s)}, \quad \mathcal{S}(i) \equiv \{i', n \mid n \in \mathcal{N}(i), y_n \neq y_i\}. \tag{3}$$

Second, considering that the false negative samples own the same clustering assignment as the anchor, *false negative attraction* method treats the false negatives as additional positive samples:

$$\mathcal{L}_{attr} = \sum_{i \in \mathcal{I}} \frac{1}{|\mathcal{P}(i)|} \sum_{p \in \mathcal{P}(i)} -\log \frac{\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_p)}{\sum_{s \in \mathcal{S}(i)} \text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_s)}, \begin{cases} \mathcal{S}(i) \equiv \{i', n \mid n \in \mathcal{N}(i)\} \\ \mathcal{P}(i) \equiv \{i', n \mid n \in \mathcal{N}(i), y_n = y_i\}. \end{cases} \tag{4}$$

Behaviors of elimination and attraction strategies. Compared to $\mathcal{L}_{inst}$, $\mathcal{L}_{elim}$ removes the similarities between the anchor and its false negatives from the denominator while on the other hand, $\mathcal{L}_{attr}$ places them on the numerator. They both show a great capability to enlarge the similarities between the anchor and the false negatives when the objective is minimized. To further explore and compare their behaviors, we derive the gradients with regard to the embedding of the anchor $\boldsymbol{z}_i$ as:

$$\frac{\partial \mathcal{L}_{elim,i}}{\partial \boldsymbol{z}_i} = \sum_{\substack{n \in \mathcal{N}(i), \\ y_n \neq y_i}} \frac{\sigma_n^-}{\tau} \boldsymbol{z}_n - \frac{\sigma_{i'}^+}{\tau} \boldsymbol{z}_{i'} \quad \text{and} \quad \frac{\partial \mathcal{L}_{attr,i}}{\partial \boldsymbol{z}_i} = \sum_{\substack{n \in \mathcal{N}(i), \\ y_n \neq y_i}} \frac{\sigma_n^-}{\tau} \boldsymbol{z}_n - \sum_{p \in \mathcal{P}(i)} \frac{\sigma_p^+}{\tau} \boldsymbol{z}_p \tag{5}$$

where σ is the weighted coefficient. The detailed formulation and derivation are in Appendix A. Based on their gradients in Equation 5, $\mathcal{L}_{elim}$ and $\mathcal{L}_{attr}$ are the generalization of triplet loss[†] (Schroff et al., 2015) which would intrinsically repel the negatives and attract the positive(s) to the anchor. While both of them fully eliminate the detected false negatives (i.e., $\{n \mid n \in \mathcal{N}(i), y_n = y_i\}$) from the negative samples and avoid them being pushed apart from the anchor, $\mathcal{L}_{attr}$ further minimizes the distances between the anchor and its false negatives. Such a more aggressive strategy results in less tolerance to noisy pseudo labels which are inevitably generated from an unsupervised clustering. Hence, although SupCon (Khosla et al., 2020) presents the effectiveness of $\mathcal{L}_{attr}$ with manual label annotations, we empirically find $\mathcal{L}_{elim}$ is more stable and suitable to our self-supervised framework.

Finally, we illustrate the framework in Figure 2(b) and provide the pseudo-code in Appendix B.

4 EXPERIMENTS

We validate and analyze our framework on several benchmarks. Due to the limited resource, we train the model on 8 Nvidia V100 GPUs for all experiments. For fair comparisons, we group prior self-supervised algorithms with a similar setting to demonstrate the efficacy of our proposed method.

4.1 IMPLEMENTATION DETAILS

We use ResNet-50 (He et al., 2016) as the encoder, following by a 3-layer MLP as the projection head to obtain 128-D features $\boldsymbol{z}$. We apply the proposed method on ImageNet (Deng et al., 2009) and CIFAR (Krizhevsky et al., 2009). For ImageNet pre-training, we follow the same setting in MoCo v2 (Chen et al., 2020d) and PCL (Li et al., 2021) for fair comparisons. Specifically, we train the encoder for 200 epochs with a mini-batch size of 256 and use the memory-based method (He et al., 2020) to increase the number of samples for contrastive learning in the default setting. We set the temperature parameter τ to 0.25 for all experiments. For pseudo label assignment, we employ k-means clustering implemented by Johnson et al. (2019) to re-assign the pseudo labels after every training epoch and use three cluster numbers $\{1000, 3000, 10000\}$ to hierarchically define semantic concepts. In default, we linearly increase the pseudo label acceptance rate from 0% to 100% through the whole training process and use the elimination loss $\mathcal{L}_{elim}$ as the objective. The training lasts for 60 hours of which 3.5 hours is spent on pseudo label assignment ($\sim 6\%$ of training duration). More details, including the setting for CIFAR pre-training, are provided in Appendix C.

[†]The gradient of triplet loss $\left(\left[\|\boldsymbol{z}_i - \boldsymbol{z}_p\|_2^2 - \|\boldsymbol{z}_i - \boldsymbol{z}_n\|_2^2 + \alpha \right]^+ \right)$ w.r.t $\boldsymbol{z}_i$ is $(2\boldsymbol{z}_n - 2\boldsymbol{z}_p)$

Table 1: **Linear evaluation and transfer learning on three benchmarks.** We evaluate the image representations in terms of effectiveness on ImageNet (Deng et al., 2009) as well as the generality on VOC2007 (Everingham et al., 2010) and Places205 (Zhou et al., 2014) and report the Top-1 classification accuracy (%). The upper group uses a more compact backbone, e.g., AlexNet (Krizhevsky et al., 2012) or ResNet-50 (He et al., 2016), and smaller pre-training batchsize ($\leqslant 256$).

Method	Architecture	Pre-training		Datasets		
		batchsize	epochs	ImageNet	VOC	Places
Jigsaw (Noroozi & Favaro, 2016)	AlexNet	256	-	34.6	67.6	-
Rotation (Gidaris et al., 2018)	AlexNet	128	100	38.7	73.0	35.1
DeepCluster (Caron et al., 2018)	AlexNet	256	500	41.0	73.7	39.8
InstDisc (Wu et al., 2018)	ResNet-50	256	200	54.0	-	45.5
LocalAgg (Zhuang et al., 2019)	ResNet-50	128	200	60.2	-	50.1
CMC (Tian et al., 2019)	ResNet-50	-	200	66.2	-	-
SimCLR (Chen et al., 2020b)	ResNet-50	256	200	64.3	-	-
MoCo (He et al., 2020)	ResNet-50	256	200	60.6	79.2	48.9
MoCo v2 (Chen et al., 2020d)	ResNet-50	256	200	67.5	84.0	50.1
PCL (Li et al., 2021)	ResNet-50	256	200	67.6	85.4	50.3
IFND (Ours)	ResNet-50	256	200	**69.7**	**87.3**	**52.4**
CPC (Oord et al., 2018)	ResNet-101	512	-	48.7	-	-
SeLa (Asano et al., 2020)	ResNet-50	1024	400	61.5	-	-
PIRL (Misra & Maaten, 2020)	ResNet-50	1024	800	63.6	81.8	49.8
SimCLR (Chen et al., 2020b)	ResNet-50	4096	1000	69.3	-	-
BYOL (Grill et al., 2020)	ResNet-50	4096	1000	74.3	-	-
SwAV (Caron et al., 2020)	ResNet-50	4096	800	75.3	88.9	56.7

Table 2: **Semi-supervised learning on ImageNet.** We report the Top-5 classification accuracy (%). All models are pre-trained on the whole ImageNet training set without category labels, then finetuned on 1% or 10% of ImageNet training set using the label information. The numbers of MoCo and MoCo v2 are computed from the pretrained models released in the official repositories.

Method	Architecture	Pre-training		Label fraction	
		batchsize	epochs	1%	10%
InstDisc (Wu et al., 2018)	ResNet-50	256	120	39.2	77.4
MoCo (He et al., 2020)	ResNet-50	256	200	56.9	83.0
MoCo v2 (Chen et al., 2020d)	ResNet-50	256	200	66.3	84.4
PCL (Li et al., 2021)	ResNet-50	256	200	75.3	85.6
IFND (Ours)	ResNet-50	256	200	**77.0**	**87.1**
S4L(MOAM) (Zhai et al., 2019)	ResNet-50 (4×)	256	1000	-	91.2
PIRL (Misra & Maaten, 2020)	ResNet-50	1024	800	57.2	83.8
SimCLR (Chen et al., 2020b)	ResNet-50	4096	1000	75.5	87.8
BYOL (Grill et al., 2020)	ResNet-50	4096	1000	78.4	89.0
SwAV (Caron et al., 2020)	ResNet-50	4096	800	78.5	89.9

Table 3: **Clustering quality on ImageNet.** We compare k-means clustering ($k = 1000$) performance for different algorithms (mostly deep clustering-based methods) using Normalized Mutual Information (Strehl & Ghosh (2002)) (%) as the metric. We run k-means five times and report the average score with the standard deviation. All results are from the official pre-trained models.

Method	NMI
DeepCluster (Caron et al., 2018)	43.2 ± 2.9
MoCo v2 (Chen et al., 2020d)	57.9 ± 2.2
SwAV (Caron et al., 2020)	63.8 ± 1.6
PCL (Li et al., 2021)	65.0 ± 1.9
IFND (Ours)	$\mathbf{67.1} \pm 1.7$

4.2 Comparison with the State-of-the-Arts

Linear evaluation and transfer learning on image classification benchmarks. We first assess the performance of the image representations after pre-training on ImageNet training set. We train

Table 4: **Effect of false negatives on self-supervised learning.** We report the Top-1 classification accuracy (%) and use the results of SupCon as the oracle performance with no effect of false negative. Δ represents the performance drop due to training with false negatives.

Method	Class label	CIFAR 10	CIFAR 100	ImageNet				
				Depth 3	Depth 5	Depth 7	Depth 9	Depth 18
SimCLR	✗	94.2	71.5	93.9	86.0	75.3	71.6	65.1
IFND (Ours)	✗	95.1	74.0	95.2	87.5	77.2	73.8	68.4
SupCon	✓	95.9	76.2	96.1	89.2	79.3	76.3	71.5
Number of categories		10	100	9	36	206	486	1000
Δ(SupCon, SimCLR)		-1.7	-4.7	-2.2	-3.2	-4.0	-4.7	-6.4
Δ(SupCon, IFND)		**-0.8**	**-2.2**	**-0.9**	**-1.7**	**-2.1**	**-2.5**	**-3.1**

a linear classifier with the fixed representations as input on three benchmarks: ImageNet (Deng et al., 2009), VOC2007 (Everingham et al., 2010) and Places205 (Zhou et al., 2014) for both linear evaluation and the validation of transfer learning. We report the results in Table 1. IFND obtains consistent improvements compared to previous self-supervised learning approaches with a similar experiment setting. The direct comparison to PCL (Li et al., 2021) emphasizes the advantages to incrementally detect and explicitly remove the false negatives in contrastive learning.

Semi-supervised learning on ImageNet. Next, we evaluate the embeddings pre-training on ImageNet in the semi-supervised setting. We follow the protocol described in SimCLR (Chen et al., 2020b) and use the same 1% or 10% fraction of ImageNet labeled training set to finetune the pre-trained encoder. The results are shown in Table 2. IFND outperforms the prior approaches within the similar setup and, notably, when using 1% label, it can achieve more superior performance than SimCLR trained with 16 times larger batchsize and 5 times more training epochs.

Clustering quality on ImageNet. We also measure the quality of clustering on the representations pre-training on ImageNet and report the results across five runs of k-means in Table 3. Compared to our baseline framework, MoCo v2 (Chen et al., 2020d), the proposed method leverages semantic-aware learning that encourages the semantically similar samples concentrating in a cluster and shows a significant improvement of 9.2% on NMI. Moreover, IFND achieves the best clustering quality compared to prior deep clustering-based contrastive learning frameworks. In Appendix F, we further analyze the clusterings on the intermediate embedding spaces during the training process to better study our incremental manner to apply the clustering assignments.

4.3 Effect of False Negative Samples

Experimental Setup. To study the effect of false negatives on self-supervised contrastive learning, we compare three representative frameworks, including SimCLR (Chen et al., 2020b) as instance-level learning, SupCon (Khosla et al., 2020) as supervised learning using label information to completely exclude the false negatives, and our method. For a fair comparison, we reproduce SimCLR and SupCon by using the same data augmentations and network architecture (i.e., ResNet-50 with 3-layer MLP). We also apply SimCLR-based implementation for our method without maintaining a momentum feature bank in this section. Considering that the probability of sampling false negatives in a training mini-batch $\mathcal{I}$ is largely determined by the number of semantic classes. We conduct the experiments on CIFAR (Krizhevsky et al., 2009) and ImageNet (Deng et al., 2009) with different numbers of object categories. For ImageNet, we use WordNet (Miller, 1998) lexical hierarchy to re-define the semantic classes in different depths of hierarchy. The dataset defined at a deeper level owns more finer classes. More details can be found in Appendix D. We then train and perform the linear evaluation for three frameworks on each dataset and report the results in Table 4.

Based on the performance drops of SimCLR compared to SupCon (Δ(SupCon, SimCLR) in Table 4), we observe that the adverse effect of false negatives is more significant on the datasets with more object categories, although it is less likely to sample false negatives during contrastive learning. Similar results are observed on both CIFAR and ImageNet. The results clearly show a critical issue of vanilla instance-level contrastive learning when it is applied to the datasets with more complex semantic concepts. With the explicit elimination of the detected false negatives, IFND effectively

Table 5: **Different strategies for pseudo label assignment and false negative removal on CIFAR-100.** We report MTPR (%, ↑) and MTNR (%, ↑) of the k-means clustering assignment ($k = 100$) and the Top-1 classification accuracy (%). Note that, [a] is actually the instance-level learning; [b] and [c] respectively uses the same strategy of pseudo label assignment as in DeepCluster and PCL (Step mode adopts 0% and 100% pseudo labels before and after 100th epoch over 1000 total training epochs); [g] is the final setting of the proposed method.

Index	Acceptance Rate of Pseudo Label			Objective	MTPR	MTNR	Top-1 Acc.
	Scheme	Initial	Final				
[a]	Constant	0%	0%	$\mathcal{L}_{inst}$	0	100	71.5
[b]	Constant	100%	100%	$\mathcal{L}_{elim}$	31.67	99.09	71.3
[c]	Step	0%	100%	$\mathcal{L}_{elim}$	<u>40.02</u>	99.42	73.1
[d]	Linear	0%	25%	$\mathcal{L}_{elim}$	15.25	99.98	72.5
[e]	Linear	0%	50%	$\mathcal{L}_{elim}$	26.16	99.89	73.0
[f]	Linear	0%	75%	$\mathcal{L}_{elim}$	35.96	99.78	<u>73.5</u>
[g]	Linear	0%	100%	$\mathcal{L}_{elim}$	**43.26**	99.65	**74.0**
[h]	Linear	0%	100%	$\mathcal{L}_{attr}$	28.62	98.91	70.2

narrows the gap between self-supervised and supervised frameworks, numerically reducing 51.7% of the performance gap between SimCLR and SupCon on ImageNet with original labels (Depth 18).

4.4 Strategy for False Negative Detection and Removal

In this section, we first introduce a novel and comprehensible evaluation metric for clustering assignment and then present the quantitative study on different strategies to utilize the clustering labels or to remove the detected false negatives. The results on CIFAR-100 are shown in Table 5.

More interpretable metric for clustering performance. In Section 4.2, we use Normalized Mutual Information (Strehl & Ghosh, 2002) to evaluate the clustering results. However, considering that the pseudo labels are essentially used to detect false negatives, in this section, we propose a more interpretable metric for the quality of clustering: Mean True Positive/Negative Rate (MTPR/MTNR). To compute MTPR, we first evaluate the true positive rate (Fawcett, 2006) for each image, describing "the probability that an actual false negative (i.e., an actual positive) will be detected as a false negative sample". MTPR is then defined as the average of true positive rates of all training images. On the other hand, MTNR indicates the average probability that an actual negative is still treated as a negative sample. Take an instance-level contrastive learning framework as a toy example, it does not perform false negative detection, so MTNR is 100% but MTPR is 0%. The goal of the proposed method is to increase MTPR of the clustering while keeping MTNR as high as possible.

Strategies for pseudo label assignment. First, we compare three schemes to determine the pseudo label acceptance rate, including constant ([b], as in DeepCluster (Caron et al., 2018)), step ([c], as in PCL (Li et al., 2021)) and linear ([g], *Ours*). The linear strategy achieves both higher MTPR and MTNR compared to the other two strategies. The results validate that using the clustering indices in an incremental manner can find the false negatives more effectively, and further improves the performance of contrastive learning. Next, we analyze the effect of using different final acceptance rates for the linear strategy ([d,e,f,g]). Intuitively, higher acceptance rates for assigning the pseudo labels indicate more false negatives can be detected (MTPR increases) but more true negatives are also improperly found and removed (MTNR decreases). However, we find that using the 100% final acceptance rate ([g]) can averagely detect nearly a half (43.26%) of the actual false negatives with only a slight drop (0.35%) of MTNR compared to the instance-level approach ([a]). Along with the elimination loss, we improve the downstream Top-1 classification accuracy from 71.5% to 74.0%.

Strategies for false negative removal. Finally, we compare the elimination and attraction losses ([g,h]). We find that the learned embeddings through minimizing $\mathcal{L}_{attr}$ are unstable and more rely on the high quality of pseudo labels at the beginning. This observation is consistent with the theoretical analysis in Section 3.3 indicating that $\mathcal{L}_{attr}$ applies a more aggressive scheme which would even pull the instance embeddings with the same given pseudo labels closer and therefore, would be more sensitive to noisy pseudo labels. Empirically, we observe that the model supervised by $\mathcal{L}_{attr}$ performs even worse than the instance-level learning.

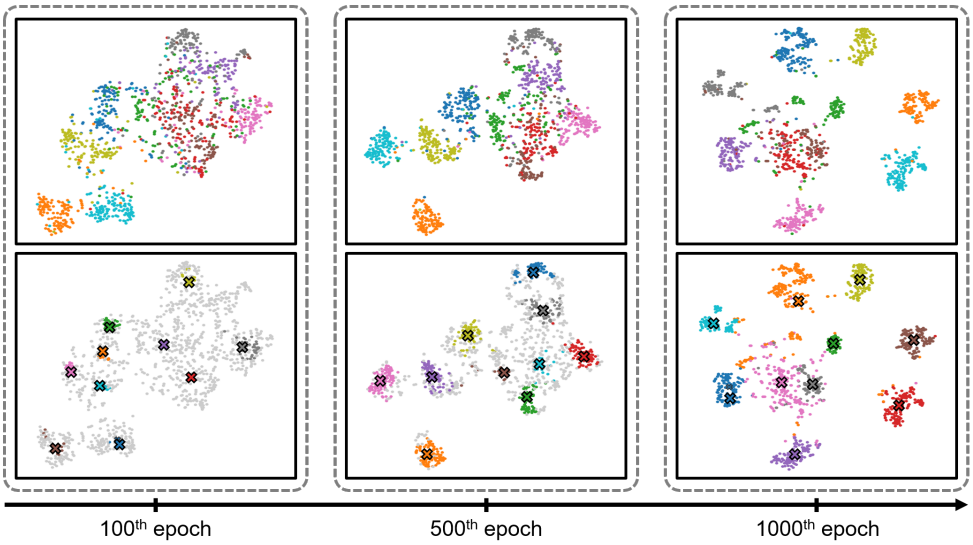

Figure 3: **Visualization of embedding spaces and pseudo label assignments in different training stages on CIFAR-10.** The samples in the upper and lower row are respectively colored with the ground truth labels and assigned pseudo labels. ✖ marks are the cluster centroids and light grey dots represent the instances without assigned pseudo labels.

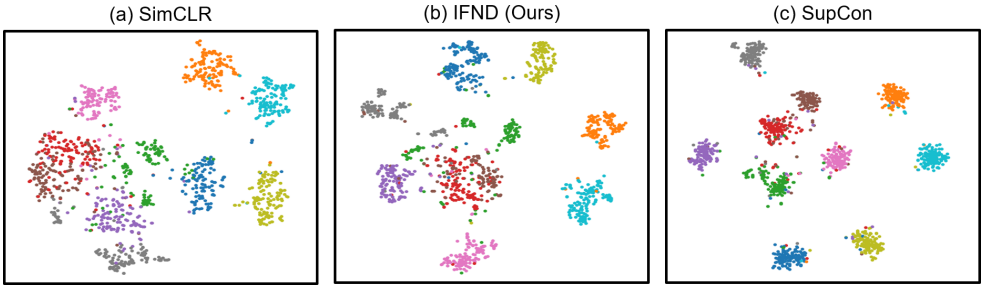

Figure 4: **Visualization of learned embedding spaces by different methods on CIFAR-10.**

4.5 Visualization of Embedding Space

In Figure 3, we visualize the learned embedding spaces and corresponding pseudo label assignments in different training stages using t-SNE (Van der Maaten & Hinton, 2008). In the early stages (e.g., 100th epoch), the semantic structure of the embedding space is still under-developed. Hence, IFND only adopts a little portion of pseudo labels with higher confidence scores. Later in the training process, with the better-trained encoder and more semantically structural embedding space, incrementally including more high-quality pseudo labels can benefit the semantic-aware representation learning. Finally, we compare the embedding spaces learned through SimCLR (Chen et al., 2020b), SupCon (Khosla et al., 2020), and our approach in Figure 4. IFND can learn the embedding space with more compact clusters than the instance-level learning and is also more similar to the supervised framework. We present a quantitative analysis in Appendix F and also provide more visualizations and discussions in Appendix G.

5 Conclusion

In this work, we empirically study the undesired impact of false negatives for self-supervised contrastive learning. To mitigate such adverse effects, we then introduce an incremental false negative detection approach to progressively detect and remove the false negatives through the training stage. Extensive experiment results validate the effectiveness of the proposed framework on several benchmarks and also demonstrate the capability to reduce the gap from the supervised framework.

REFERENCES

Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *International Conference on Learning Representations*, 2020. 3, 6

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *European Conference on Computer Vision*, 2018. 3, 6, 8

Mathilde Caron, Piotr Bojanowski, Julien Mairal, and Armand Joulin. Unsupervised pre-training of image features on non-curated data. In *IEEE International Conference on Computer Vision*, 2019. 3

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*, 2020. 6

Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International Conference on Machine Learning*, 2020a. 1

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, 2020b. 1, 2, 3, 6, 7, 9, 15, 16, 17

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. In *Advances in Neural Information Processing Systems*, 2020c. 1, 2

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020d. 1, 2, 5, 6, 7, 15

Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. In *Advances in Neural Information Processing Systems*, 2020. 3

Florence Corpet. Multiple sequence alignment with hierarchical clustering. *Nucleic acids research*, pp. 10881–10890, 1988. 4

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009. 5, 6, 7, 15

Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *IEEE International Conference on Computer Vision*, 2015. 1

Alexey Dosovitskiy, Philipp Fischer, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1734–1747, 2015. 2

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, pp. 303–338, 2010. 6, 7

Tom Fawcett. An introduction to roc analysis. *Pattern Recognition Letters*, 2006. 8

Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*, 2018. 1, 6

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. In *Advances in Neural Information Processing Systems*, 2020. 6

Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *International Conference on Artificial Intelligence and Statistics*, 2010. 2, 16

Ben Harwood, Vijay Kumar BG, Gustavo Carneiro, Ian Reid, and Tom Drummond. Smart mining for deep metric learning. In *IEEE International Conference on Computer Vision*, 2017. 13

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 5, 6, 15

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1, 2, 3, 5, 6, 15

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. In *International Conference on Learning Representations*, 2019. 1, 2

Tri Huynh, Simon Kornblith, Matthew R Walter, Michael Maire, and Maryam Khademi. Boosting contrastive self-supervised learning with false negative cancellation. *arXiv preprint arXiv:2011.11765*, 2020. 3

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 2019. 5

Stephen C Johnson. Hierarchical clustering schemes. *Psychometrika*, pp. 241–254, 1967. 4

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, 2020. 2, 3, 5, 7, 9, 17

Jack Kiefer, Jacob Wolfowitz, et al. Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, 1952. 15

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5, 7, 15, 16

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012. 6

Junnan Li, Pan Zhou, Caiming Xiong, and Steven Hoi. Prototypical contrastive learning of unsupervised representations. In *International Conference on Learning Representations*, 2021. 3, 5, 6, 7, 8, 15, 16

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 15

George A Miller. *WordNet: An electronic lexical database*. MIT press, 1998. 7, 16

Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2, 6

Fionn Murtagh and Pierre Legendre. Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion? *Journal of classification*, pp. 274–295, 2014. 4

Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, 2016. 1, 6

Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 13

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 1, 3, 6

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 1

Nikunj Saunshi, Orestis Plevrakis, Sanjeev Arora, Mikhail Khodak, and Hrishikesh Khandeparkar. A theoretical analysis of contrastive unsupervised representation learning. In *International Conference on Machine Learning*, 2019. 1, 3

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015. 2, 5

Alexander Strehl and Joydeep Ghosh. Cluster ensembles-a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research*, 2002. 6, 8, 17

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019. 1, 2, 6

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 2008. 9

Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krahenbuhl. Sampling matters in deep embedding learning. In *IEEE International Conference on Computer Vision*, 2017. 13

Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2, 6

Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International Conference on Machine Learning*, 2016. 3

Jianwei Yang, Devi Parikh, and Dhruv Batra. Joint unsupervised learning of deep representations and image clusters. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 3

Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2

Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. In *IEEE International Conference on Computer Vision*, 2019. 6

Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in Neural Information Processing Systems*, 2014. 6, 7

Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *IEEE International Conference on Computer Vision*, 2019. 3, 6

A THEORETICAL ANALYSIS FOR ELIMINATION AND ATTRACTION LOSSES

In this section, we show the derivation of the gradients of two discussed objective functions: elimination loss $\mathcal{L}_{elim}$ (Equation 3) and attraction loss $\mathcal{L}_{attr}$ (Equation 4). Furthermore, we demonstrate that they have the inherent capability of hard sample mining (Oh Song et al., 2016; Harwood et al., 2017; Wu et al., 2017). Without loss of generality, we use i to represent an arbitrary anchor image, and re-write the two losses as:

$$\mathcal{L}_{elim,i} = -\log \frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_{i'}/\tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s/\tau)}, \quad \mathcal{S}(i) \equiv \{i', n \mid n \in \mathcal{N}(i), y_n \neq y_i\}, \quad \text{and} \tag{6}$$

$$\mathcal{L}_{attr,i} = \frac{1}{|\mathcal{P}(i)|} \sum_{p \in \mathcal{P}(i)} -\log \frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_p/\tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s/\tau)}, \begin{cases} \mathcal{S}(i) \equiv \{i', n \mid n \in \mathcal{N}(i)\} \\ \mathcal{P}(i) \equiv \{i', n \mid n \in \mathcal{N}(i), y_n = y_i\}. \end{cases} \tag{7}$$

We then derive the gradient of $\mathcal{L}_{elim,i}$ with regard to the embedding of the anchor image $\boldsymbol{z}_i$:

$$\begin{aligned}
\frac{\partial \mathcal{L}_{elim,i}}{\partial \boldsymbol{z}_i} &= \frac{\partial}{\partial \boldsymbol{z}_i} -\log \frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_{i'}/\tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s/\tau)} \\
&= \frac{\partial}{\partial \boldsymbol{z}_i} \log \sum_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s/\tau) - \frac{\partial}{\partial \boldsymbol{z}_i} \log(\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_{i'}/\tau)) \\
&= \sum_{s \in \mathcal{S}(i)} \left(\frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s/\tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s/\tau)} \cdot \frac{\boldsymbol{z}_s}{\tau} \right) - \frac{\boldsymbol{z}_{i'}}{\tau} \\
&= \sum_{\substack{n \in \mathcal{N}(i), \\ y_n \neq y_i}} \left(\frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_n/\tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s/\tau)} \cdot \frac{\boldsymbol{z}_n}{\tau} \right) - \left(1 - \frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_{i'}/\tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s/\tau)} \right) \cdot \frac{\boldsymbol{z}_{i'}}{\tau} \\
&= \sum_{\substack{n \in \mathcal{N}(i), \\ y_n \neq y_i}} \frac{\sigma_n^-}{\tau} \boldsymbol{z}_n - \frac{\sigma_{i'}^+}{\tau} \boldsymbol{z}_{i'},
\end{aligned} \tag{8}$$

where

$$\begin{aligned}
\sigma_x^- &= \frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_x/\tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s/\tau)} = \frac{\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_x)}{\sum\limits_{s \in \mathcal{S}(i)} \text{sim}(\boldsymbol{z}_i \cdot \boldsymbol{z}_s)}, \quad \text{and} \\
\sigma_x^+ &= 1 - \frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_x/\tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s/\tau)} = 1 - \frac{\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_x)}{\sum\limits_{s \in \mathcal{S}(i)} \text{sim}(\boldsymbol{z}_i \cdot \boldsymbol{z}_s)}
\end{aligned} \tag{9}$$

are the coefficient terms respectively for the negative or positive sample x.

Next, the gradient of $\mathcal{L}_{attr,i}$ is derived in a similar way:

$$
\begin{aligned}
\frac{\partial \mathcal{L}_{attr,i}}{\partial \boldsymbol{z}_i} &= \frac{\partial}{\partial \boldsymbol{z}_i} \frac{1}{|\mathcal{P}(i)|} \sum_{p \in \mathcal{P}(i)} -\log \frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_p / \tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s / \tau)} \\
&= \frac{1}{|\mathcal{P}(i)|} \sum_{p \in \mathcal{P}(i)} \left(\frac{\partial}{\partial \boldsymbol{z}_i} \log \sum_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s / \tau) - \frac{\partial}{\partial \boldsymbol{z}_i} \log \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_p / \tau) \right) \\
&= \frac{1}{|\mathcal{P}(i)|} \sum_{p \in \mathcal{P}(i)} \left[\sum_{s \in \mathcal{S}(i)} \left(\frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s / \tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s / \tau)} \cdot \frac{\boldsymbol{z}_s}{\tau} \right) - \frac{\boldsymbol{z}_p}{\tau} \right] \\
&= \sum_{s \in \mathcal{S}(i)} \left(\frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s / \tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s / \tau)} \cdot \frac{\boldsymbol{z}_s}{\tau} \right) - \sum_{p \in \mathcal{P}(i)} \frac{1}{|\mathcal{P}(i)|} \frac{\boldsymbol{z}_p}{\tau} \\
&= \sum_{\substack{n \in \mathcal{N}(i), \\ y_n \neq y_i}} \left(\frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_n / \tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s / \tau)} \cdot \frac{\boldsymbol{z}_n}{\tau} \right) \\
&\qquad - \sum_{p \in \mathcal{P}(i)} \left(\frac{1}{|\mathcal{P}(i)|} - \frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_p / \tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s / \tau)} \right) \cdot \frac{\boldsymbol{z}_p}{\tau} \\
&= \sum_{\substack{n \in \mathcal{N}(i), \\ y_n \neq y_i}} \frac{\sigma_n^-}{\tau} \boldsymbol{z}_n - \sum_{p \in \mathcal{P}(i)} \frac{\sigma_p^+}{\tau} \boldsymbol{z}_p,
\end{aligned}
\tag{10}
$$

where

$$
\begin{aligned}
\sigma_x^- &= \frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_x / \tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s / \tau)} = \frac{\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_x)}{\sum\limits_{s \in \mathcal{S}(i)} \mathrm{sim}(\boldsymbol{z}_i \cdot \boldsymbol{z}_s)}, \quad \text{and} \\
\sigma_x^+ &= \frac{1}{|\mathcal{P}(i)|} - \frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_x / \tau)}{\sum\limits_{s \in \mathcal{S}(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_s / \tau)} = \frac{1}{|\mathcal{P}(i)|} - \frac{\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_x)}{\sum\limits_{s \in \mathcal{S}(i)} \mathrm{sim}(\boldsymbol{z}_i \cdot \boldsymbol{z}_s)}.
\end{aligned}
\tag{11}
$$

Note that the coefficient terms σ_x^+ and σ_x^- in Equation 9 and 11 are determined by $\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_x)$. Larger coefficients are given when negative/positive samples are relatively more similar/dissimilar to the anchor. This observation indicates that these two loss functions have the implicit capability of hard sample mining, i.e., focusing on differentiating the "hard negatives" that are near the anchors or attracting the "hard positives" which are far away from the anchors.

B PSEUDO CODE

Algorithm 1: Contrastive Learning with Incremental False Negative Detection

Input: encoder f, projection head g, training samples $\mathcal{X}$, and number of clusters k

```
// instance-level contrastive learning in the beginning
```

Initialize completely different pseudo label y^* for each sample

while *Epoch* $\in$ *Training Epoch* **do**

```
// contrastive learning with false negative elimination
```

  for *mini-batch* $\mathcal{I}$ **do**

    $\boldsymbol{z} = g(f(\{i \sim \mathcal{I}\}))$

    update encoder f and projection head g to minimize $\mathcal{L}_{elim}(\boldsymbol{z}, y^*)$

  end

  if *Epoch* $\in$ *Clustering Epoch* **then** `// pseudo label assignment`

    $\boldsymbol{v} = f(\{x \sim \mathcal{X}\})$

    $\boldsymbol{c}, y = k\text{-means}(\boldsymbol{v}, k)$

    $\kappa = \text{sim}(\boldsymbol{z}, \boldsymbol{c}_y) / \sum_{j=1}^{k} \text{sim}(\boldsymbol{z}, \boldsymbol{c}_j)$

    determine the acceptance rate of the pseudo label as (*Epoch* / *Training Epoch*)

    update pseudo label y^* by a fraction of y with larger κ

  end

end

C PRE-TRAINING AND EVALUATION DETAILS

As described in Section 4.1, we use the proposed method to pre-train ResNet-50 (He et al., 2016) on ImageNet (Deng et al., 2009), CIFAR-10, and CIFAR-100 (Krizhevsky et al., 2009). For ImageNet pre-training, we follow the setting of previous work (He et al., 2020; Chen et al., 2020d) that uses SGD optimizer (Kiefer et al., 1952) with a weight decay of 0.0001 and a momentum of 0.9. The learning rate is set to 0.03 initially which is then multiplied by 0.1 at 120 and 160 epochs. For CIFAR-10 and CIFAR-100 pre-training, we train the encoder for 1000 epochs with a mini-batch size of 1024. Because of the larger mini-batch size, we implement our framework based on SimCLR (Chen et al., 2020b) structure without maintaining a memory bank for the momentum instance features. We also use SGD optimizer with the same weight decay and momentum. But, we set the initial learning rate to 0.5 and use a linear warm-up strategy for the first 10 epochs. In the rest of the training, we decay the learning rate with the cosine annealing schedule (Loshchilov & Hutter, 2016). For pseudo label assignment, we re-assign the pseudo labels after every 20 training epochs (due to more training epochs and fewer training images) for CIFAR and also use three cluster numbers, $\{10, 30, 100\}$ for CIFAR-10 and $\{100, 300, 1000\}$ for CIFAR-100, to hierarchically define semantic concepts.

For linear evaluation on ImageNet, we train the classifier (a fully connected layer followed by softmax) with the fixed pre-trained representations for 100 epochs. Following the the same hyper-parameter setting used in MoCo v2 (Chen et al., 2020d) and PCL (Li et al., 2021), we optimize the model by SGD optimizer with a batch size of 256, an initial learning rate of 10, a momentum of 0.9, and a weight decay of 0. As for the semi-supervised learning, we fine-tune the pre-trained ResNet-50 model and the classifier using the subsets of 1% or 10% ImageNet labeled training data (note that we use the same subsets of ImageNet training data as SimCLR (Chen et al., 2020b)). The model is trained using SGD optimizer for 20 epochs, a batch size of 256, a momentum of 0.9, and a weight decay of 0.0005. The learning rate for ResNet-50 is 0.01 while the one for the linear classifier is 0.1 (for 10% ImageNet) or 1 (for 1% ImageNet). All learning rates are multiplied by 0.2 at 12nd and 16th epochs.

Whole	Whole	Whole	Whole	Whole	Sign
Organism	Organism	Organism	Organism	Instrumentality	Street Sign
Chordate	Chordate	Chordate	Invertebrate	Vehicle	Street Sign
Mammal	Mammal	Aquatic Vertebrate	Gastropod	Warship	Street Sign
French Bulldog	Shetland Sheepdog	Goldfish	Snail	Aircraft Carrier	Street Sign

Figure 5: **Hierarchical semantic concepts for ImageNet.** We show six images and their respective class labels hierarchically defined in Depth $= \{3, 5, 7, 9, 18\}$ from top to bottom. Note that Depth 18 (the bottommost) uses original class labels in ImageNet.

Table 6: **Ablation Comparison with PCL on CIFAR-100.** The step scheme uses 0% and 100% pseudo labels before and after 100th epoch over 1000 total training epochs. The implementation of $\mathcal{L}_{\text{ProtoNCE}}$ is directly from the official repository.

Method	Objective	Pseudo Label Assignment	Top-1 Acc. (%)
SimCLR	$\mathcal{L}_{inst}$	Constant (0%)	71.5
PCL	$\mathcal{L}_{\text{ProtoNCE}}$	Step ($0\% \rightarrow 100\%$)	72.8
	$\mathcal{L}_{elim}$	Step ($0\% \rightarrow 100\%$)	73.1
	$\mathcal{L}_{\text{ProtoNCE}}$	Linear ($0\% \rightarrow 100\%$)	73.4
IFND (Ours)	$\mathcal{L}_{elim}$	Linear ($0\% \rightarrow 100\%$)	**74.0**

D Hierarchical Class Definition for ImageNet

Based on WordNet (Miller, 1998) lexical database, we use the corresponding synset IDs to represent original 1000 classes in ImageNet and recursively trace the "hypernyms" (parent synset) for each class until the coarsest synset, "entity", in WordNet. Then, we construct the tree structure with entity synset as the root, the synsets of original classes as the leaves, and the relations of hypernym or hyponym as the edges. We then re-define the semantic class with different granularities in different depths of the tree. In Section 4.3, we employ the datasets defined in Depth $= \{3, 5, 7, 9, 18\}$. Here, we show some example images with the semantic classes defined in these depths in Figure 5.

E In-Depth Ablation Comparison with PCL

Here, we provide an ablation comparison with PCL (Li et al., 2021). The experimental setup is the same as Section 4.3. Briefly, we re-implement PCL and our IFND with SimCLR (Chen et al., 2020b) based implementation and evaluate the frameworks on CIFAR-100 (Krizhevsky et al., 2009). The results are reported in Table 6.

Compared to SimCLR, PCL applies $\mathcal{L}_{\text{ProtoNCE}}$ as the objective function which also uses instance-level NCE loss (Gutmann & Hyvärinen, 2010) (i.e., $\mathcal{L}_{inst}$) as the primary supervision but auxiliary with prototypical contrastive loss to concentrate the samples within the same prototype. In contrast, IFND applies $\mathcal{L}_{elim}$ which completely eliminates the detected false negatives from negative sample set (as shown in Section 3.3) and performs favorably against $\mathcal{L}_{\text{ProtoNCE}}$. Additionally, for the adoption of the clustering indices, PCL uses the "step" strategy to warm up the framework by instance-level contrastive learning. On the contrary, IFND dynamically includes increasing confident labels which relatively better fits the gradually well-trained encoder and greater clustering quality (as demonstrated in Appendix F). Finally, with both $\mathcal{L}_{elim}$ and linear strategy for pseudo label assignment, IFND outperforms PCL and improves the Top-1 classification accuracy from 72.8% to 74.0%.

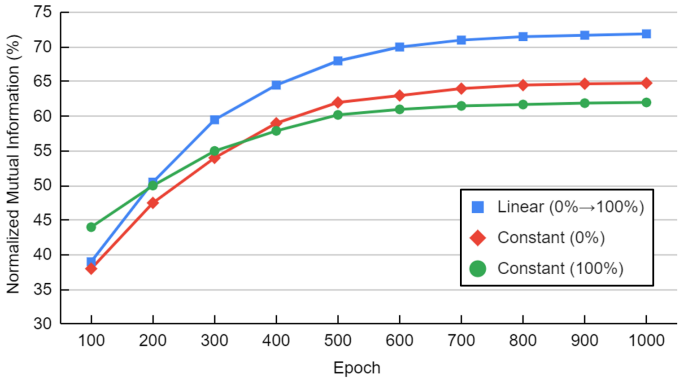

Figure 6: **Clustering quality in different training stages on CIFAR-100.** We train the model with three different strategies to determine the acceptance rate of pseudo labels and measure the quality of k-means clusterings ($k = 100$) on the intermediate embedding spaces during the training process.

F Clustering Quality through Training Process

In this section, we measure the clustering results in different training stages to first verify the claim that the clusterings are comparably unreliable in the early training stages, and also study the practical effect of our incremental manner to adopt clustering indices. We compute Normalized Mutual Information (Strehl & Ghosh, 2002) between the ground truth labels and the clustering indices of all training images and plot the results in Figure 6.

Following the training, we observe that the clustering is gradually improved and becomes better to reflect the semantic relationship among images. Take this effect into consideration, we dynamically adopt increasing high-quality labels which can benefit semantic-aware learning and further lead to a greater clustering in the following stages. As for the model that consistently applies all assigned pseudo labels, although it can achieve a better clustering in the early stages, however, the earlier inferior labels (compared to the ones generated in the latter stages) instead disturb the training and finally result in both worse representation learning and clustering as in Table 5 and Figure 6.

G Visualization of Embedding Space through Training Process

In this section, we visualize and compare the embedding spaces learned by four contrastive learning frameworks, including (a) SimCLR (Chen et al., 2020b), (b) IFND (trained by $\mathcal{L}_{attr}$), (c) IFND (trained by $\mathcal{L}_{elim}$), and (d) SupCon (Khosla et al., 2020), through the training process in Figure 7.

Compared to IFND trained by $\mathcal{L}_{elim}$, the one trained by $\mathcal{L}_{attr}$ learns the embedding spaces with more concentrated clusters which are more similar to SupCon due to using similar attraction objective function. However, during training, the unsupervised clustering would inevitably produce some noisy pseudo labels (as supported by the MTNR in Table 5 being less than 100%). Consequently, the attraction loss may inappropriately pull the true negative samples closer to the anchor. We can then find that although the learned embedding space in the rightmost of Figure 7(b) shows more concentrated clusters, nonetheless, a cluster is not actually constituted by the image embeddings representing the same semantic content. Also, as shown in Table 5, the attraction loss leads to inferior representation performance compared to the elimination objective.

As for IFND trained by $\mathcal{L}_{elim}$, which is the final proposed method, it learns a similar embedding space as SimCLR in the early stage, since the acceptance rate for assigning the pseudo label is still low and the proposed framework behaves like the instance-level contrastive learning (e.g., SimCLR) in earlier processes. However, with more reliable pseudo labels generated in later stages, $\mathcal{L}_{elim}$ avoids more detected false negatives being pushed apart from the anchor. Finally, the model effectively bridges the instance-level and semantic-aware representation learning and learns a more favorable embedding space against SimCLR.

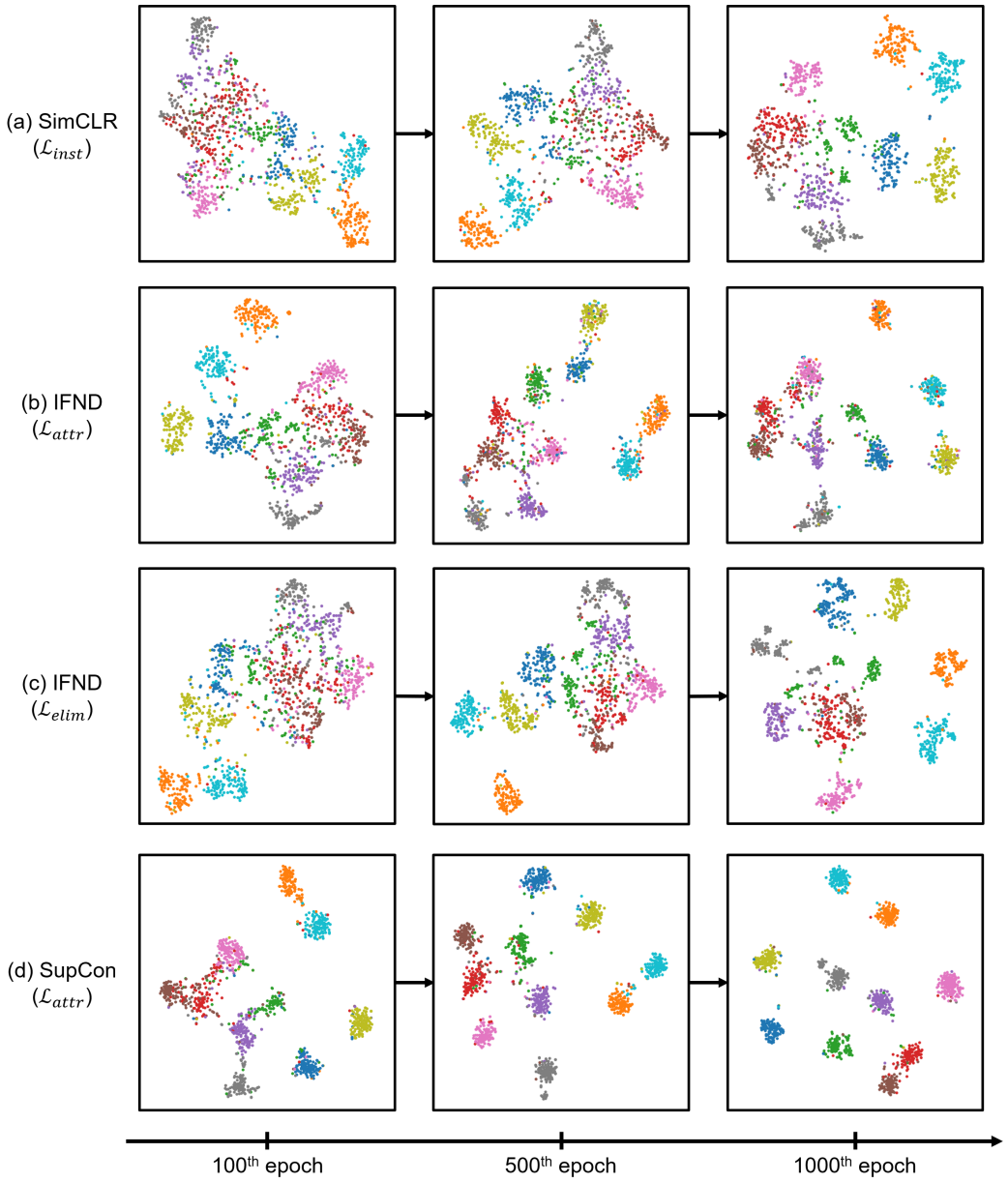

Figure 7: **Visualization of embedding spaces on CIFAR-10.** We compare four contrastive learning frameworks based on their learned embedding spaces through the training process. We list their respective objective functions in the bracket.