Leveraging Graded Similarity in Self-Supervised Learning

Master Thesis

Computer Science - University of Twente

Sam Rappange

Examination Committee:

dr. N. Strisciuglio
dr. ir. N. Alachiotis
S. Wang, MSc.

August 21, 2024

Abstract

Self-supervised methods in computer vision have become dominated by contrastive learning approaches, surpassing various pretext tasks such as rotation prediction and colorization. Contrastive methods learn augmentation invariant representations of images by pulling augmented views of the same image to the same representation while pushing views of other images away. Non-contrastive methods reach similar performance without requiring negative views. These methods pull all possible crops of an image to the same representation, regardless of their content, position, or shared information. This can hinder training and force the model to discard valuable information about the views, as view pairs have widely varying amounts of similarity. We propose a novel learning objective utilizing a graded similarity measure to address this limitation. The graded similarity measure uses the overlap of crops as a measure regarding the distance between representations in the latent space. This novel learning objective better encodes the nuanced similarity between views while emphasizing spatial relations. We implement this graded similarity in contrastive methods (SimCLR) and non-contrastive methods (SimSiam). Our results show that pre-trained encoders using our approach reach slightly better performance in transfer learning and up to $1.4\times$ better performance in retrieval tasks. Notably, SimCLR improves significantly from this novel learning objective.

UNIVERSITY OF TWENTE.

1 Introduction

Self-supervised learning (SSL) is a machine learning paradigm that allows models to learn useful representations of data without the need for explicit labeling or supervision. In contrast to supervised learning where data annotation is needed, self-supervised learning uses *pretext* tasks to learn useful representations of the data [29]. An area of machine learning where self-supervised learning finds many applications is computer vision [27]. The pretext tasks used in self-supervised learning force models to learn the semantics and structure of images, such that a strong universal model is obtained. Examples of pretext tasks are image inpainting, rotation prediction, and context prediction [25, 10, 8]. After pre-training, the model can be used to create lower-dimensional representations of the images. These representations can then be directly used or finetuned for various downstream tasks, such as classification, semantic segmentation and object detection, among others [4, 11]. The main advantage of self-supervised learning is that the models can learn from any unlabelled data while obtaining representations that are useful for various downstream tasks [27].

State-of-the-art self-supervised models in computer vision, such as SimCLR [4] and MoCoV2 [6], use a contrastive learning approach. In contrastive learning, the pretext task consists of learning invariance to augmentations applied to images. Examples of these augmentations are random cropping, colour distortion, and blurring [4]. The contrastive learning objective is formulated such that augmented views of the same image are pulled together while pushing away views of other images [33]. Furthermore, non-contrastive methods that do not require negative examples, such as SimSiam [5] and BYOL [11], have been developed that reach similar performance. These methods use a stop-gradient and a predictor MLP to be able to avoid using negative views.

Though these approaches reach state-of-the-art performance, they require long training times, large amounts of data, and large batch sizes or memory buffers [4, 11]. This is partially due to how the learning objective treats view pairs. In (non)-contrastive methods, all random crops of an image are encouraged to have the same representation, regardless of their position, scale, or shared information. This can convolute the learning process, as shown in Figure 2, where different random crops of an image depict completely dissimilar content.

Intuitively, one would expect that the similarity of the representations of different views is related to their shared visual information. However, this is not possible in the discussed (non)-contrastive methods, which do not utilize any information about the view sampling. To that end, we propose a novel graded learning objective to refine contrastive and non-contrastive methods. Taking inspiration from graded learning objectives in Visual Place Recognition (VPR) [17], we present a

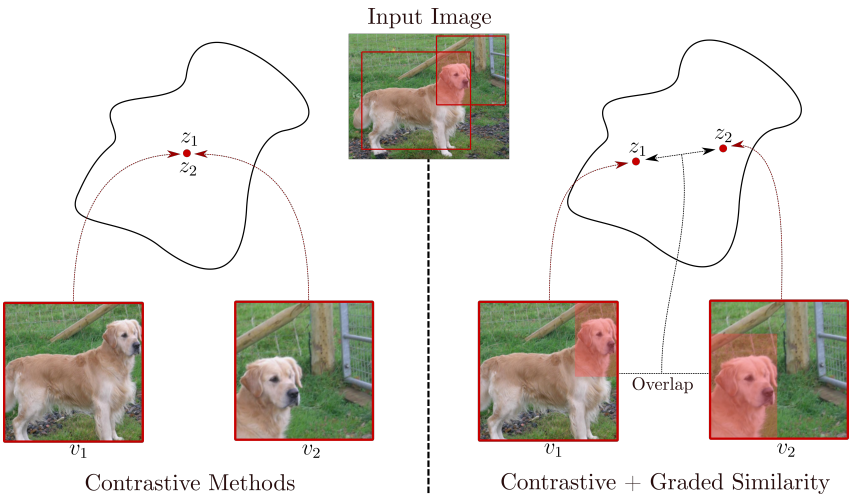


Figure 1: An overview of graded similarity in contrastive learning. Contrastive (and non-contrastive) methods pull all views of an image to the same representation (left), not taking into account that different crops depict widely varying amounts of similarity. To that end, the graded similarity utilizes the shared visual information as a direct measure of the distance between representations (right).

novel learning objective that utilizes shared visual information of image crops. More specifically, we use the overlap of views as a proxy for their shared visual information. Using this overlap, a graded similarity measure is created, which is used in the loss function as a measure of the distance between representations of views. Consequently, the method provides a natural distance metric between representations, accounting for nuanced relationships between views. An overview of the method and its difference with regular (non)-contrastive methods is shown in Figure 1. We demonstrate the method in both contrastive and non-contrastive methods, namely SimCLR and SimSiam [4, 5].

To design and test this novel learning objective, we devise the following research questions:

1. What is a good graded similarity measure between image views used as a measure for the distance between their representations?
   (a) What pixel-level measures are a good proxy for view pair similarity?
   (b) How must the similarity measure be constructed to be directly applied in an augmentation-invariant (non)-contrastive setting?
2. To what extent does incorporating a graded similarity measure improve data utilization and efficiency during pre-training?
3. How does the incorporation of graded view similarity affect the learned image representations?

Figure 2: An example of a false positive view pair generated by random cropping, image from ImageNet [31].

To answer these research questions, we first review relevant background and theory. Following this, we position the proposed method in related work. Next, we describe the methods we use for the experiments. We then conduct extensive experiments to demonstrate the method's performance and explore its interesting properties. Finally, we discuss the results and summarize our key findings by answering the research questions.

2 Background

2.1 Pretext Tasks

Pretext tasks in self-supervised learning are tasks created for the model to solve. These tasks require the model to predict transformations applied to the images. The tasks are designed to force the model to learn the intrinsic structures and semantics of the images, leading to encoders that can be used for various downstream tasks [21].

An example of a pretext task is rotation prediction. Here, images are rotated by $0°, 90°, 180°$ or $270°$ and the network predicts the angle in a multi-class prediction format [10]. Another powerful pretext task is context prediction, where the relative position of two image patches is predicted [8]. Other pretext tasks are colourization [34], image inpainting [25], and solving jigsaw puzzles [23]. Visualizations of these pretext tasks are shown in Figure 3 [1]. The representations learned by performing the pretext tasks are then used for various downstream tasks.

It is important to note that the nature of the pretext task is critical to the performance of the representation on downstream tasks. That is because the learned representations are covariant with the transformations used in the pretext task. Consequently, the generalizability and transferability of the representations are affected [21]. Additionally, while the pretext tasks can give good quality representations [1], the tasks are often handcrafted by utilizing prior domain knowledge about the data for which it is created. This limits the generalization of the task and the representations obtained by the task [4]. Contrastive methods have largely overcome this limitation [21], which we will discuss next.

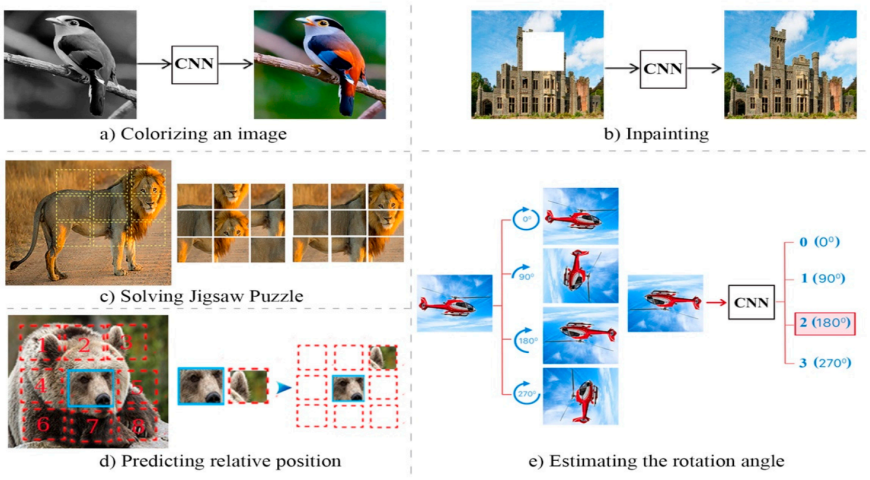


Figure 3: Visualization of several pretext tasks used in self-supervised learning. These pretext tasks allow the model to obtain meaningful representations of the images used for various downstream tasks. Figure from [1].

2.2 Contrastive Learning

State-of-the-art self-supervised methods can come remarkably close to supervised counterparts on downstream tasks such as classification and object detection without requiring labelled data during training [27]. Most of these state-of-the-art self-supervised learning approaches use a contrastive learning approach. In contrastive learning, the pretext task is to learn augmentation invariance between different views of an image. These contrastive approaches outperform other pretext tasks and generalize better to different datasets [21]. A popular contrastive approach is called Information Noise Contrastive Estimation (InfoNCE) [33].

One prevalent contrastive method that utilizes the InfoNCE learning objective is SimCLR [4]. It consists of a siamese network structure, where both sides consist of an encoder and a projector, as shown in Figure 4. The encoder is a ResNet [12], whereas the projector is an MLP that learns a lower dimensional representation of the encoder's output. The input images are transformed by various augmentations: random cropping, flipping, colour distortion, grayscale, and blurring. The learning objective is formulated such that it pulls views of the same image together while pushing views of other images away. The *NT-Xent* loss is used for this goal, a close adaptation of the InfoNCE loss. Given two augmented views $\mathbf{v}_i$ and $\mathbf{v}_j$ sampled from some image $\mathbf{x} \in \mathbb{R}^{W \times H}$, the loss for this view pair is defined as:

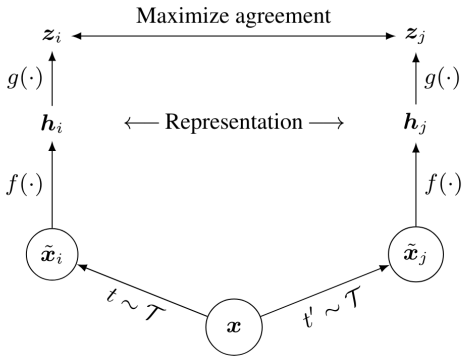


Figure 4: An overview of the Siamese architecture used in SimCLR. The encoder is $f(\cdot)$ that encodes the image, and the projector is $g(\cdot)$. An image $\mathbf{x}$ is augmented twice by two different instances of some set of augmentations $\mathcal{T}$. The representations of views of the same image are pulled together while pushing away representations of other images. Figure from [4].

$$\mathcal{L}_{NTX} = -\log \frac{\exp(\frac{1}{\tau} \cdot \|\bar{\mathbf{z}}_i\| \cdot \|\bar{\mathbf{z}}_j\|)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\frac{1}{\tau} \cdot \|\bar{\mathbf{z}}_i\| \cdot \|\bar{\mathbf{z}}_k\|)}, \tag{1}$$

where $\mathbf{z}_i$ and $\mathbf{z}_j$ are the outputs of the projector, and τ is the temperature parameter controlling randomness. The total loss is calculated over all possible image pairs in the batch, so (i, j) as well as (j, i), among all other pairs. All views of other images in the batch are used as negative examples, as seen in the sum in the denominator. SimCLR uses large batch sizes (4k+) since negative examples are sampled within the batch.

Another popular contrastive method that utilizes the InfoNCE learning objective is Momentum Contrast (MoCo) [13]. Its successor MoCoV2 [6] offers minor improvements over MoCo. This method achieves performance similar to that of SimCLR without requiring large batches. MoCo uses a momentum encoder in one branch of the Siamese setup, meaning the parameters of the encoder are updated by a momentum-based moving average. More importantly, it uses a dictionary-based queue to store negative samples, meaning no large batch sizes are needed but still utilizing much memory [13]. Consequently, this and other contrastive approaches require much memory and long training times for training [13, 4, 11].

2.3 Non-Contrastive Methods

Non-contrastive methods are very similar to contrastive methods, though they differ in using negative views in the learning process. Non-contrastive methods don't require negative examples and thus only use different views of the same image.

An example of these techniques is Bootstrap Your Own Latent (BYOL) [11], which also uses a Siamese network structure. The two networks are called online and target networks. The weights of the target network are an exponential moving average of the online network (as in MoCo), and the target network uses a stop gradient to prevent class collapse. This means that the gradient is not backpropagated in that branch. As in SimCLR [4], it uses two strongly augmented views of an image. A predictor MLP is placed in the online branch to reach a performance similar to that of contrastive methods. This is an additional network that tries to predict the representation of the target branch. The loss is the mean squared error (MSE), equal to the positive part of the contrastive loss used in SimCLR [30]. This approach reaches similar performance without needing negative pairs and large batch sizes.

SimSiam proposes a simpler Siamese architecture that does not require a momentum encoder for the target network [5]. It minimizes the negative cosine similarity, which is the same learning objective as the MSE for L2 normalized representations, see Appendix B. They showed that the stop-gradient operation is the only essential part of these non-contrastive methods to stop class-collapse. However, the predictor and a deeper projector architecture are necessary to reach state-of-the-art performance [5].

The learning objective of these methods is formulated as follows. Suppose there are two views $\mathbf{v}_i$ and $\mathbf{v}_j$ sampled from some image $\mathbf{x} \in \mathbb{R}^{W \times H}$, which are augmented by two different instances of the same set of augmentations. These are fed through the encoder, which outputs a representation, $f_\theta(v_i) = \mathbf{z}_i$. The predictor MLP, $p(\mathbf{z}_i)$ in one branch, then predicts the representation of the other branch, which has a stop-gradient applied to it. The loss is the mean squared error between the two representations, noting that the output dimension of the predictor MLP is the same as that of the projector.

$$L_{MSE} = \|\bar{p}(\mathbf{z}_i) - \text{sg}(\bar{\mathbf{z}}_j)\|_2^2, \tag{2}$$

where $\bar{\mathbf{z}}$ denotes the L2 normalised representation: $\mathbf{z}/\|\mathbf{z}\|_2$, and sg is the stop gradient operation. To symmetrize the loss, both views are fed through both branches such that the final non-contrastive (NC) loss becomes:

$$\mathcal{L}_{NC} = \frac{1}{2} \|\bar{p}(\mathbf{z}_i) - \text{sg}(\bar{\mathbf{z}}_j)\|_2^2 + \frac{1}{2} \|\bar{p}(\mathbf{z}_j) - \text{sg}(\bar{\mathbf{z}}_i)\|_2^2. \tag{3}$$

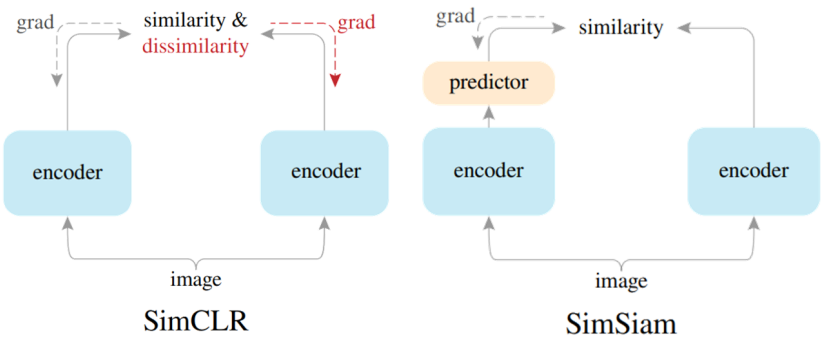


Figure 5: Differences between contrastive (SimCLR) and non-contrastive (SimSiam) methods. SimCLR utilizes negatives, while SimSiam uses an additional predictor and a stop-gradient. Figure from [5].

The total loss is the sum over all view pairs in the batch.
A schematic view of the differences between SimCLR and SimSiam is shown in Figure 5. Though they have a similar Siamese network structure, their difference lies in the use of negative samples. SimCLR uses views of other images as negatives, while SimSiam uses an additional predictor MLP, and a stop-gradient. Most notably, they utilize the same strategy for views of the same image, always pulling them to the same representation.

3 Related Work

The discussed state-of-the-art contrastive and non-contrastive methods reach very good performance. Nevertheless, they need long training times, large amounts of data, and many computational resources to reach this level of performance. Therefore, there has been research into how to improve these methods in terms of data utilization and efficiency.

The data augmentation set used in these state-of-the-art methods consists of various strong augmentations, such as random cropping and resizing, flipping, colour distortion and blurring[4, 11]. These augmentations are essential as they reduce the mutual information between the views fed to the model, which is essential in augmentation-invariant learning objectives [32]. In addition, self-supervised models benefit more from augmentations than supervised methods. This makes them essential in contrastive methods [4].

However, as proposed in Section 1, all random crops of an image are encouraged to have the same representation. This may convolute the learning process and could lead to the model discarding valuable information. This limitation in (non)-contrastive methods has been addressed in many other works [26, 20, 32, 36, 35]. These either improve the cropping strategy or modify the learning objective to overcome this limitation.
We will focus on strategies that modify the learning objective as this is the most relevant to our research. Nevertheless, we discuss the possible impact and synergy of other cropping strategies on our proposed method in Section 7.

One method that improves the learning objective in relation to the random crops is a method that leverages global and local representations [35]. This method discriminates local and global crops and proposes different relations between these crops. For instance, different local crops are pushed away from each other, while local crops are pulled to global crops. It also uses an additional MLP to estimate the similarity between local crops. This method reached better performance and data efficiency in both contrastive and non-contrastive methods.
Another method called LESSL [36] uses five patches of an image, which significantly improves data utilization. A localization task is added on top of the regular contrastive loss to better incorporate spatial relations between views. This method also improves in terms of performance and data efficiency.

These methods show that addressing the limitation regarding the learning objective can significantly improve performance. However, these methods fail to do this in an efficient and natural manner. The methods require additional hyperparameter selection to incorporate the information, which is both method and dataset-dependent [35, 36]. This limits the generalization of the methods and, ultimately, the data efficiency. This is because hyperparameter grid searches must be done before the methods can be successfully implemented. These are also hyperparameters related to balancing the different learning objectives that must be tuned. In addition, they both require additional modules to incorporate the information effectively, also requiring tuning and adding computational overhead.

Our proposed methodology circumvents this by directly using the shared visual information of views in the contrastive learning objective. The overlap of the views is used as a proxy for their shared information, which is used as a direct measure of distance between the representations. This allows for the use of a natural measure of similarity without needing any careful balancing of learning objectives or manual selection of when views are similar. Furthermore, this alleviates the need for hyperparameter tuning across different methods and datasets. Finally, the graded property allows us to capture the full continuous domain of view similarity. To our knowledge, no method has been developed that directly modifies the contrastive learning objective with this graded information.

4 Methods

4.1 Overview

The method uses a graded similarity (GS) measure directly in the learning objective. This similarity measure uses information about the view sampling process, namely the overlap and area of the views. This measure is modified slightly to be used directly in the novel learning objective. An overview of the method is shown in Figure 6.

First, random cropping and resizing is applied, from which the sampling parameters are extracted. Using this, the intersection and area of the views are calculated, which are required for the similarity measure $\psi_{i,j}$. After other augmentations are applied and the views are fed into the encoder, the modified loss is calculated. We discuss how this similarity measure is defined and how it is incorporated into the proposed learning objectives. We consider both contrastive and non-contrastive methods, namely SimCLR and SimSiam.

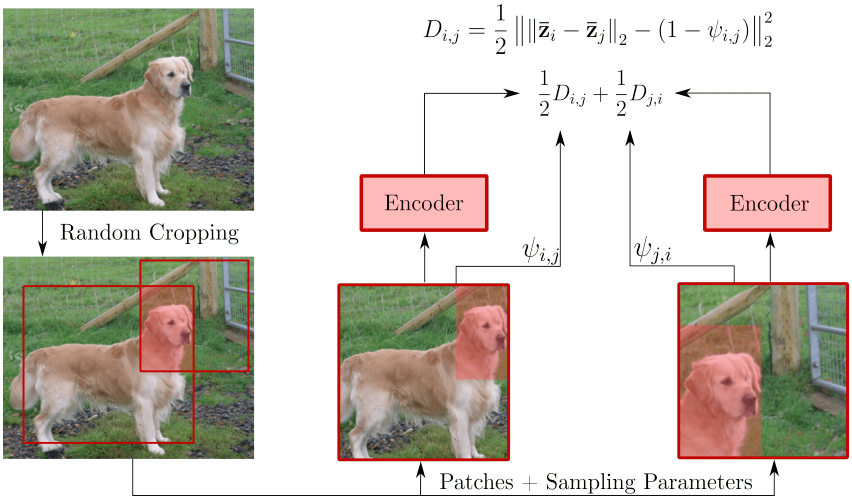


Figure 6: Overview of the pipeline used for the graded similarity for positive view pairs. The graded similarity measure $\psi_{i,j}$ is calculated from the sampling parameters. The symmetrized loss is adjusted to use the graded similarity as a target for regression. We implement the graded similarity in both contrastive and non-contrastive learning methods.

4.2 Similarity Measure

The similarity measure used in this method is based on information about the cropping of the views. Here, we consider uniform random cropping as the view sampling strategy, though any other non-uniform strategy can be used.

We consider the intersection over area (IoA) of a view pair as the graded similarity measure. We show it is superior to the intersection over union (IoU) in Section 6.4. Given two views $\mathbf{v}_i$ and $\mathbf{v}_j$ sampled from an image $\mathbf{x} \in \mathbb{R}^{W \times H}$ by different instances of a random cropping distribution. The IoA of view pair (i, j) is then defined as:

$$\mathrm{IoA}_{i,j} = \frac{\mathrm{Intersection}(\mathbf{v}_i, \mathbf{v}_j)}{\mathrm{Area}(\mathbf{v}_i)} \tag{4}$$

This similarity measure has a few interesting properties. To begin with, it is asymmetric, meaning both views have different values depending on their scale. This has the advantage that views only are regressed to their intersecting content. A small view lying in a larger view has full similarity, while that larger view only has partial similarity based on their intersecting content. This leads to a more natural measure of distance between the representations. Additionally, non-overlapping crops that can have dissimilar content are pushed away.

To optimally incorporate this similarity measure in current state-of-the-art (non)-contrastive methods, a wrapper function is used to obtain the final similarity function. The primary goal of (non)-contrastive methods is to learn augmentation invariance, which requires similar views to be considered equally in the learning objective [4]. Here, we use a threshold of IoA, λ, that sets this threshold for when views are considered equally in the learning objective. Below this threshold, the similarity measure is linearly scaled from 0 to 1.

$$\psi_{i,j} = \begin{cases} \frac{\mathrm{IoA}_{i,j}}{\lambda} & \mathrm{IoA}_{i,j} \leq \lambda \\ 1 & \mathrm{IoA}_{i,j} > \lambda \end{cases}. \tag{5}$$

Note that the complement of this similarity measure is used as the distance measure, i.e., $1 - \psi_{i,j}$. As a base value, $\lambda = 0.5$ is used such that views that share half of their visual content are considered the same in the loss. This is motivated and elaborated on further in Section 6.4.

4.3 Loss Functions

The graded similarity is implemented in the learning objective of both contrastive and non-contrastive methods. It is directly related to the distance between representations in the latent space while conserving the augmentation-invariance learning objective. We will show how the learning objectives are adapted to utilize this graded similarity. First, we consider non-contrastive methods (SimSiam) as this is the base case that only considers positive pairs. We then expand this to SimCLR as it also utilizes negative views.

Non-Contrastive Methods

The loss used in non-contrastive methods such as SimSiam and BYOL is the mean squared error (MSE), given by:

$$D(p(\mathbf{z}_i), \mathbf{z}_j) = \|\bar{p}(\mathbf{z}_i) - \bar{\mathbf{z}}_j\|_2^2, \tag{6}$$

note this only concerns views from the same image, which have a graded similarity value. The loss is adapted to a regression problem where we use the graded similarity as the target during learning. We want to regress the distance between the representations, so we use $1 - \psi_{i,j}$ as the regression target. In addition, the normalized Euclidean distance is used as a distance measure between the representations with this similarity measure. This is motivated by the fact that the gradient dynamics are similar to the baseline (see Appendix C), allowing for the use of similar hyperparameters and optimizers in experiments. The adapted loss D_{GS} then becomes:

$$D_{GS}(\mathbf{z}_i, \mathbf{z}_j; \psi_{i,j}) = \left\| \|\bar{p}(\mathbf{z}_i) - \bar{\mathbf{z}}_j\|_2 - (1 - \psi_{i,j}) \right\|_2^2. \tag{7}$$

Note that this distance function is asymmetric due to the asymmetry in the similarity distance. Furthermore, the stop-gradient is used in the non-predictor branch, and the loss is symmetrized for both views such that the loss becomes:

$$\mathcal{L}_{NC-GS} = \frac{1}{2}\left(D_{GS}(\bar{p}(\mathbf{z}_i), \text{sg}(\mathbf{z}_j); \psi_{i,j}) + D_{GS}(\bar{p}(\mathbf{z}_j), \text{sg}(\mathbf{z}_i); \psi_{j,i})\right), \tag{8}$$

where L_{NC-GS} is the non-contrastive graded-similarity loss used with the SimSiam architecture during experiments.

Contrastive Methods

A similar approach can be taken in a contrastive setting with negatives, such as SimCLR. Here, only the loss of the positive view pairs is modified to utilize the graded similarity. The contrastive loss used in SimCLR is a form of the InfoNCE loss called NT-Xent [4]. The loss for view pair (i, j) is given by:

$$\mathcal{L}_{NTX} = -\log \frac{\exp(\frac{1}{\tau} \cdot \|\bar{\mathbf{z}}_i\| \cdot \|\bar{\mathbf{z}}_j\|)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\frac{1}{\tau} \cdot \|\bar{\mathbf{z}}_i\| \cdot \|\bar{\mathbf{z}}_k\|)}, \tag{9}$$

where N is the batch size, τ is a temperature parameter, and the summation is over all other views in the batch. Note that the total loss is calculated over all pairs in the batch, (i, j) as well as (j, i). This can be rewritten as the following, observing that the first term is the negative cosine similarity for positive view pairs:

$$\mathcal{L}_{NTX} = -\frac{1}{\tau} \|\bar{\mathbf{z}}_i\| \cdot \|\bar{\mathbf{z}}_j\| + \log \sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\frac{1}{\tau} \cdot \|\bar{\mathbf{z}}_i\| \cdot \|\bar{\mathbf{z}}_k\|).$$

The first term is then modified to utilize the graded similarity similarly to non-contrastive methods, where the only changes are the absence of the stop-gradient operation and the addition of temperature. The full derivation can be found in Appendix D.

$$\begin{aligned} \mathcal{L}_{C-GS}(\mathbf{z}_i, \mathbf{z}_j) = &\frac{1}{\tau} D_{GS}(\mathbf{z}_i, \mathbf{z}_j; \psi_{i,j}) \\ &+ \log \sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\frac{1}{\tau} \cdot \|\bar{\mathbf{z}}_i\| \cdot \|\mathbf{z}_k\|), \end{aligned} \tag{10}$$

where $\mathcal{L}_{C-GS}$ is the contrastive graded similarity loss for view pair (i, j). Here, D_{GS} is the expression as defined in equation 7. The total loss is calculated for all positive pairs in the batch; (i, j) as well as (j, i), as with the regular NT-Xent loss. This means the loss is symmetrized, as both $\psi_{i,j}$ and $\psi_{j,i}$ are used.

5 Experiments

Baseline Methods We explore the implementation of the graded learning objective in two baseline methods: SimCLR [4] and SimSiam [5], which are contrastive and non-contrastive methods, respectively. We train the baseline and the graded similarity models in identical settings; the same hyperparameters, random seeds, and architectures are used.

Pre-training The pre-training experiments are performed on the following datasets: CIFAR10, STL10 and ImageNet1000. For the CIFAR10 and STL10 pre-training, a ResNet18 [12] is used as the encoder, pre-training for 800 epochs. When pre-training on CIFAR10, the CIFAR ResNet version (smaller conv_1 kernel size) is used. For the ImageNet1000 experiments, a ResNet50 encoder is used pre-training for 100 epochs. For all experiments, $\lambda = 0.5$ is used. The crop ratio of the baseline methods is used, which is $[0.08, 1]$ of the original size for SimCLR and $[0.2, 1]$ for SimSiam.

Further pre-training details, such as augmentation, architecture, and hyperparameter details, are found in Appendix A.

Transfer Learning To assess the quality and transferability of the representations, the transfer learning performance of the pre-trained models is evaluated. The procedure consists of linear evaluation of the ImageNet pre-trained model on a wide variety of datasets, namely CIFAR10/100 [14], Pascal VOC 2007 (for classification) [9], Aircraft [19], Stanford Cars [7], Flowers [22], Food101 [2], and Oxford Pets [24]. The widely used linear evaluation protocol for transfer learning is used, which consists of training a logistic regression classifier on the representations of the frozen (ImageNet) pre-trained encoder.

Image Retrieval To further explore the generalizability of the representations trained using graded similarity, an image retrieval task is performed. The methods are tested on the revisited Oxford5k and Paris6k datasets [28]. Here, the zero-shot performance of the ImageNet pre-trained encoders is evaluated to assess generalizability to this new task.

5.1 Evaluation

Pre-training During pre-training, a KNN-classifier ($k = 201$) is used to track the quality of the learned representations. After pre-training we perform linear-evaluation on the respective pre-training dataset.

Linear Evaluation For linear evaluation on the pre-training dataset, we follow the linear evaluation protocol of the baseline methods [4, 5]. This consists of training a single fully connected layer at the output of the ResNet encoder (pool_5). We train this for 90 epochs using an SGD Nesterov optimizer. For SimCLR, we use a learning rate of $0.1 \times \frac{\text{BatchSize}}{256}$, while for SimSiam, we use a learning rate of $30 \times \frac{\text{BatchSize}}{256}$. We use a batch size of 256 for both. During training, we apply random cropping, resizing, and flipping. Furthermore, we evaluate the images by resizing to 256×256 and center cropping to 224.

Transfer Learning Transfer learning performance is evaluated using the widely used linear evaluation protocol [4]. This consists of training a logistic regression classifier on the representations generated by the frozen encoder. The l_2 regularization parameter is chosen from a grid search of 45 logarithmically spaced values between 10^{-6} and 10^5 on the validation set. As transformations, the images are resized to 224 along the shorter size, after which a 224×224 center-crop is applied.

Image Retrieval The image retrieval task is performed on the revisited Oxford5k and Paris6k datasets [28]. The query images are cropped following the provided bounding box and resized to 224×224, while the database images are only resized. Both images are normalized to ImageNet statistics. The representations of the frozen encoder are L2 normalized, after which the retrieval task is performed. We follow their evaluation protocol, reporting the mean average precision (mAP) and the mean precision @ 10 (mP@10). The mAP gives a good general measure of the performance of the encoder, while the mP@10 provides a realistic indication of ranking performance. Together, these measures give a good overview of the performance of the pre-trained encoders at this new task.

6 Results

6.1 Pre-training & Linear Evaluation

We report a KNN ($k = 201$) accuracy monitor during pre-training. After training, we follow the linear evaluation protocol on the respective datasets. We will discuss the pre-training and linear evaluation performance of the graded similarity in both SimCLR and SimSiam.

SimCLR

The pre-training accuracy monitors (Fig. 7a and 7b) of SimCLR show slightly improved data efficiency on CIFAR10 for the GS implementation while showing no significant difference on

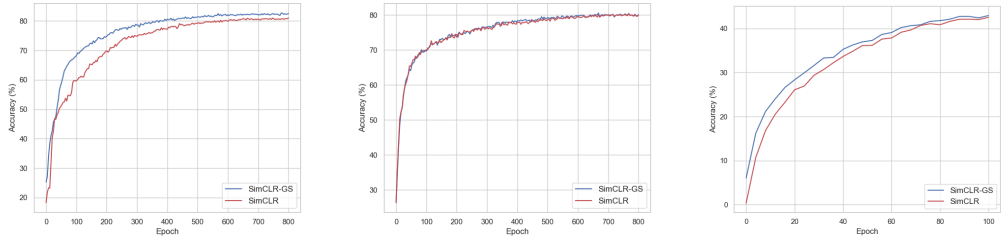
Model *Linear evaluation (%)*	**CIFAR10**	**STL10**	**ImageNet**
SimCLR	86.28	87.98	**63.60**
SimCLR-GS	**87.19**	**88.19**	63.53
SimSiam	88.71	88.57	**65.29**
SimSiam-GS	**89.36**	**88.86**	63.18

Table 1: Pre-training results on CIFAR10, STL10 and ImageNet. Top-1 accuracy is given from the linear evaluation protocol.

STL10. As we see in the linear evaluation accuracy (Table 1), the difference between CIFAR10 and STL10 is not as large as the KNN monitor suggests. When using GS, we see a 0.91% increase on CIFAR10 and a 0.21% increase on STL10. The difference between the KNN accuracy monitor and the linear evaluation may be due to the expressiveness of the classifier or the construction of latent space. As this may be the case, we focus on the linear evaluation accuracy.

The better relative performance on CIFAR may be caused by its lower resolution. Random cropping is used to sample crops, while small views of CIFAR may lose their semantic content due to the low resolution of the crops. Since the graded similarity is more likely to push away these crops, training may be improved due to this. Since this is less of an issue with STL due to the higher resolution, the performance increase may not be as large.

The ImageNet accuracy training monitor (Fig 7c) shows slightly faster training. In contrast, the linear evaluation accuracy (Table 1) shows similar accuracy for both the GS and baseline implementation, namely 0.07% higher for the baseline. The KNN classifier is less expressive, especially at this large dataset, so we focus on the linear evaluation accuracy. A possible explanation for the small relative decrease in performance on ImageNet is that ImageNet does not have square images. This means that the distribution of the overlap of views will be different, and thus, the regression target distribution will be different. Furthermore, the images in ImageNet are not as object-centric, which may synergize with the graded similarity. These effects occur together, among other interactions, which makes it difficult to estimate their exact influence on performance.



(a) CIFAR10 (b) STL10 (c) ImageNet

Figure 7: KNN ($k = 201$) top-1 accuracy monitor of pre-training of SimCLR with the graded similarity implementation on CIFAR10, STL10, and ImageNet.

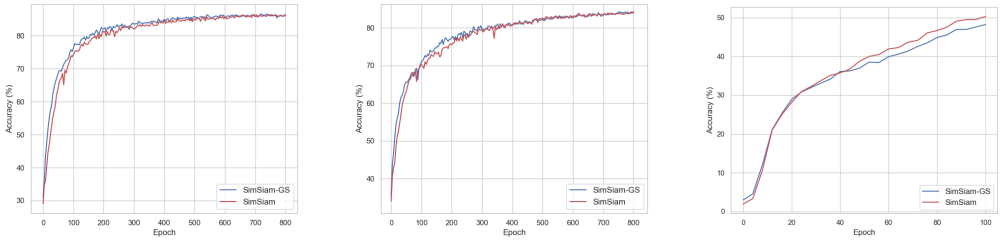
SimSiam

For SimSiam, we see slightly faster training on both CIFAR10 and STL10 (see Fig 8a and 8b), though no large differences in performance are seen on the accuracy monitor during pre-training. The GS implementation sees a linear evaluation accuracy increase of 0.65% CIFAR10 and 0.29% on STL10. The difference between CIFAR10 and STL10 may be due to the same reason, as previously discussed, regarding the small resolution crops losing semantic meaning.
Furthermore, the accuracy monitors on CIFAR10 and STL10 are more consistent with each other

than those of SimCLR. This may be due to the structure of the vector space that is learned by this pre-training method, which gives different results for a KNN classifier than SimCLR. This is supported by the consistent increase in linear evaluation performance compared to that of SimCLR.

The ImageNet pre-training accuracy monitor of SimSiam is found in Figure 8c. We see a slightly faster start when training with GS, though the accuracy then dips under the baseline. The linear evaluation accuracy for the GS implementation is also significantly lower on ImageNet, namely 2.11%. This is not consistent with the results on CIFAR and STL, nor with the SimCLR results.

The different target distribution of ImageNet caused by the non-square images may have a larger influence on the SimSiam-GS implementation. SimSiam has no negatives, leading to a larger influence of this different distribution than for SimCLR. Furthermore, the SimSiam-GS implementation for ImageNet uses a deeper projector and projection architecture, while that of SimCLR does not. Though this is necessary to reach good performance and circumvent collapsing issues [5], it may not be as essential for the GS implementation. Further considerations regarding this are given in Section 7.



(a) CIFAR10 (b) STL10 (c) ImageNet

Figure 8: KNN ($k = 201$) top-1 accuracy monitor of pre-training of SimSiam with the graded similarity implementation on CIFAR10, STL10, and ImageNet.

6.2 Transfer Learning

We evaluate transfer learning performance on various image classification datasets, of which the results are found in Table 2. Generally, we see better transfer learning performance of SimSiam with respect to SimCLR, which could partially be due to the better performance at ImageNet linear evaluation. Furthermore, it may be due to the structure of the representations learned by this method. However, as this is not the main focus of this research, we will focus on the differences between the baseline and the GS implementations.

We generally see improved transfer learning performance for the SimCLR implementation with graded similarity (Table 2). It improves on all datasets, except VOC2007, while performing similarly to the baseline on ImageNet. Moreover, the relative increase is not constant over all datasets, most notably the largest improvements are seen in the Aircraft and Cars datasets. These results indicate that the features learned by incorporating the graded similarity generalize better to data with

Model	CIFAR10	CIFAR100	VOC2007	Pets	Aircraft	Cars	Flowers102	Food101
Linear evaluation (%)								
SimCLR	83.62	60.40	**67.56**	66.01	40.23	27.04	68.34	60.08
SimCLR-GS	**84.27**	**62.00**	66.78	**66.60**	**43.00**	**29.62**	**69.37**	**60.72**
SimSiam	**90.32**	69.33	**71.46**	**79.74**	49.19	42.31	**83.67**	65.11
SimSiam-GS	89.83	**70.55**	70.80	79.38	**50.90**	**44.1**7	81.59	**65.57**

Table 2: Transfer learning performance of SimCLR and SimSiam with graded similarity (GS) on various natural image classification datasets. The performance is evaluated by the linear evaluation protocol of the 100-epoch ImageNet pre-trained models with both methods.

different characteristics and distributions. These features may be related to spatial structures and relations, as we primarily see performance increases on datasets that rely on this understanding.

SimSiam sees a smaller relative increase in transfer learning performance. This is partly due to the worse performance of SimSiam-GS on ImageNet pre-training with respect to the baseline. Accounting for this, namely, does indicate that the representations learned by the GS objective have slightly better transfer learning performance. The datasets at which performance is improved are consistent with that of the SimCLR implementation, seeing the largest improvements on Aircraft and Cars. This further confirms the hypothesis that the GS objective reinforces spatial understanding.

6.3 Image Retrieval

The results for the image retrieval task are given in Table 3. The zero-shot performance from the frozen pre-trained encoders is evaluated. The graded similarity in SimCLR significantly improves the retrieval performance in both Oxford and Paris. There is no remarkable difference between the performance on the easy and medium tasks. This performance gain is larger than the gain in transfer learning performance on classification tasks. This further indicates that the graded similarity learns a better understanding of spatial structures and relations, which are especially important in retrieval tasks [28]. Moreover, the GS may also learn better viewpoint and occlusion invariant features. These are also important invariances that are tested in the medium retrieval task (Table 3) [28]. Nevertheless, further investigation must be done to confirm these properties.

The SimSiam encoder with graded similarity does not improve on the Oxford dataset while slightly improving on the Paris dataset. We see a slight relative improvement in retrieval performance when taking the linear evaluation accuracy in Table 1 as a baseline.

At this image retrieval task, we see that the graded similarity does not provide the same level of improvement for SimSiam as it does for SimCLR. This difference may be due to the inherent differences between non-contrastive and contrastive methods. Non-contrastive methods already focus on intra-image content and thus possibly capture these spatial relationships within images better. Meanwhile, contrastive methods focus more on contrasting views. This is also motivated by the better baseline image retrieval and transfer learning performance. Therefore, contrastive methods may benefit more from the additional information about spatial relationships. Nevertheless, we foresee that the graded similarity could see better results with different architectures or optimization. This is discussed in more detail in Section 7.2.

Model	**Oxford5k [E]**		**Oxford5k [M]**		**Paris6k [E]**		**Paris6k [M]**	
	mAP	mP@10	mAP	mP@10	mAP	mP@10	mAP	mP@10
zero-shot								
SimCLR	8.85	16.03	7.66	17.86	19.71	62.00	16.60	67.86
SimCLR-GS	**11.11**	**18.24**	**8.97**	**19.43**	**27.15**	**70.29**	**20.28**	**73.29**
SimSiam	**15.78**	20.15	**12.42**	**23.00**	43.07	81.57	30.03	84.57
SimSiam-GS	14.32	**20.59**	11.83	**23.00**	**44.01**	**81.86**	**31.33**	**85.14**

Table 3: Zero-shot image retrieval performance of the ImageNet pre-trained models on the revisited-Oxford5k and Paris6k datasets [28]. The easy [E] and medium [M] tasks are evaluated. The mean average precision (mAP) and mean precision@10 (mP@10) are reported.

6.4 Ablation Studies

As shown by transfer learning and image retrieval performance, the graded similarity can improve the transferability of the learned representations to different tasks and datasets. However, the effects are nuanced, which calls for a more concrete understanding of the properties of the graded similarity. To do this, various ablation studies are done that attempt to highlight different aspects of the graded similarity. We will consider other similarity measures, the role of λ, the removal of augmentations,

and the effect of batch size. If not specified differently, we train SimCLR with the graded similarity for 200 epochs on CIFAR-10 for efficiency reasons, with hyperparameters as specified in Appendix A.

Different Similarity Measures

Different similarity measures were tested to understand better what is critical to the workings of the similarity measure. An example of another interesting measure is the intersection over union, where the union is the area of the combined patches. Note that this is a symmetric measure with the same value for both views. This experiment is done with the implementation of the graded similarity in SimCLR.

In Figure 9, the pre-training KNN accuracy monitor of both similarity measures, including the baseline, is reported. The intersection over union has slightly worse performance, though slightly improving over baseline performance. Hence, we see that the area of the individual views is important to the similarity measure.

This behaviour may have several reasons. First, using the union instead of the area gives a significantly different target distribution in the regression problem that may not be as conducive to learning. Secondly, several unique properties of the similarity measure are lost when using the IoU. For example, when a smaller crop lies in a larger crop, the smaller crop has full similarity, while the larger crop only has their overlap as similarity. This interesting property is lost when using the IoU, which would not consider the small crop to have full similarity. This property is also demonstrated in the method using local and global crops [35]. This method also stresses the importance of this property. As such, we see that the asymmetric properties of the IoA are important to the workings of the similarity measure.

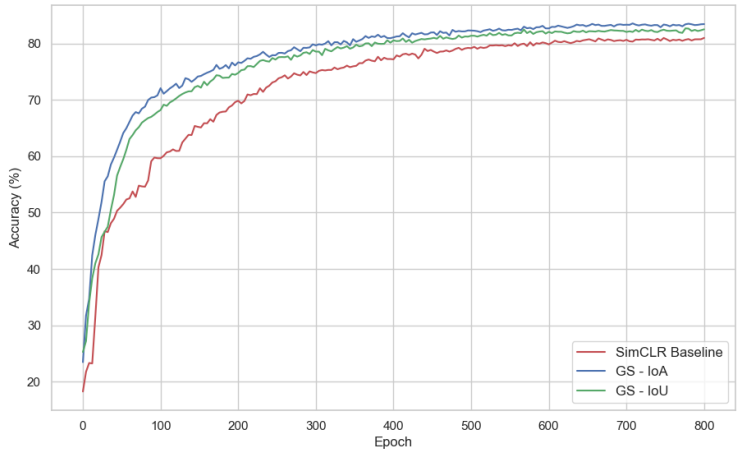


Figure 9: SimCLR baseline and with GS with IoA and IoU as similarity measures. KNN ($k = 201$) accuracy is reported.

Optimal Value of λ

The value of λ in the similarity measure is critical to its performance. To empirically study the behaviour of the threshold, we perform a grid search and evaluate the model for the values of λ. Here, $\lambda \in [0, 1]$ is tested with an interval of 0.1.

We report the accuracy for different values of λ (see Fig. 10). The partially non-linear character of the trend may be caused by the fact that the target (IoA) distribution is non-uniform. This means that some targets have more pairs than others, which affects the performance differently. Furthermore, when λ is raised, the performance also increases, though decreasing after $\lambda = 0.5$. A possible explanation is that in (non)-contrastive self-supervised learning, the heavy augmentations

require treating similar views as hard positives in the loss function. The value of λ allows for the balancing of the two learning objectives, thus showcasing improved performance at different downstream tasks with respect to the regular contrastive task. Therefore, incorporating the graded similarity can enhance the efficiency and performance in contrastive methods, though the heavy augmentations require close views to be considered equally.

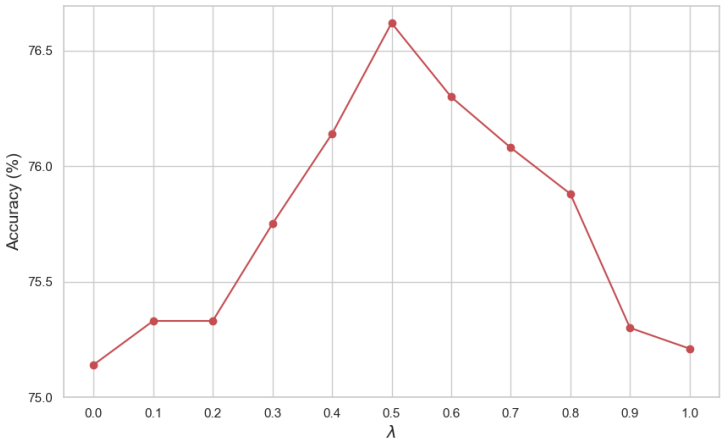
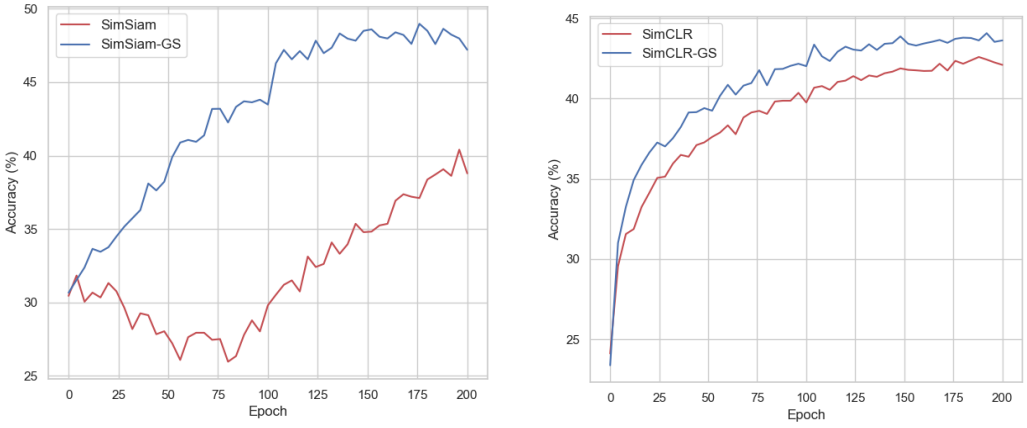


Figure 10: Performance of SimCLR with graded similarity for $\lambda \in [0, 1]$ after pre-training for 200 epochs on CIFAR10. KNN (k=201) top-1 accuracy is reported.

Removing Augmentations

Augmentations are essential in (non)-contrastive methods [4, 16]. In the following study, we remove many augmentations that are essential for reaching good performance to show the effectiveness of the graded similarity. To do this, both SimCLR and SimSiam were trained on CIFAR10 with only random cropping and flipping.

The performance for SimSiam is shown in Figure 11a. The near collapse at the start is avoided when using graded similarity. This means that the additional regularization added by the graded similarity ensures training stability even when augmentations are removed.



(a) SimSiam (b) SimCLR

Figure 11: Performance of SimCLR and SimSiam with only random cropping and flipping on CIFAR10, trained for 200 epochs with settings as in Appendix A.

The results for SimCLR are shown in Figure 11b. We see that SimCLR is not susceptible to class collapse, though including the graded similarity slightly improves the performance. This performance increase is similar to that when using augmentations. Thus, this suggests that in the contrastive setting, there is no significant correlation between the augmentations and the graded similarity. However, a more robust study must be done to verify this.

Thus, we can conclude that in non-contrastive settings, the graded similarity can provide valuable regularization to prevent near-collapse, which is otherwise provided by augmentations. It is important to note that augmentations are not the only factor influencing possible collapse; the stop-gradient and architectural choices also play an essential role. However, including the graded similarity may alleviate the need for deep architectures in the projector and predictor. Moreover, in a contrastive setting with SimCLR, the graded similarity slightly improves over the baseline. This performance increase is similar to when using augmentations, suggesting no significant correlation between augmentations and the graded similarity.

Batch Size

SimCLR generally requires large batch sizes to function well, as the sampling of negatives happens within the batch [4]. In this ablation study, we train SimCLR with and without the GS. Here, we pre-train for 800 epochs as different batch sizes influence training dynamics, thus requiring longer training for accurate results. Note we don't focus on SimSiam as it is more robust to batch size differences and does not require large batch sizes [5].

The top-1 KNN-accuracy for both the baseline and the GS is shown in Figure 12. Generally, there is a trend that the GS performs relatively better at lower batch sizes than the baseline. For both, the performance deteriorates at larger batch sizes, which is specific to CIFAR-10 [4]. The better performance of GS at smaller batch sizes suggests the graded similarity provides information otherwise provided by more negatives within the batch. We also see better stability over different batch sizes, indicating robustness to different training settings.

However, it should be noted that the batch size behaviour is different for each dataset, and the results are dependent on the hyperparameters and optimizer. Furthermore, the KNN-accuracy can give slightly different results than linear evaluation. In an ideal setting, the experiment is run on ImageNet while averaging over different sets of hyperparameters.

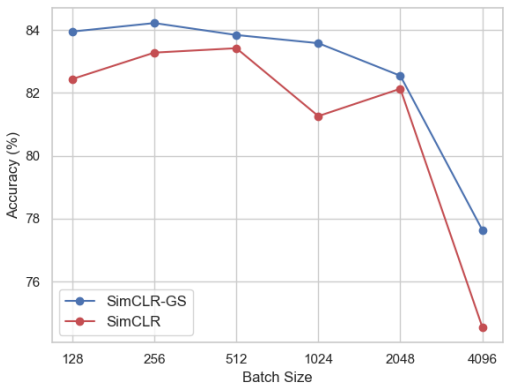


Figure 12: SimCLR trained with and without graded similarity with various batch sizes on CIFAR10. KNN ($k = 201$) top-1 accuracy is reported. Note the batch size scales exponentially.

7 Discussion

7.1 Limitations of Experiments

Computational Resources

For a more thorough analysis of the graded similarity, a full 1000-epoch pre-training on ImageNet would be preferably done. Unfortunately, a 100-epoch pre-training was done due to time and computational resources limitations. Though this limits some results, 100-epoch ImageNet pre-training was done in many of the experiments in the original methods[4, 5], and we have shown that no significant deviations arise at long training times on CIFAR10 and STL10. The only significant change is the lower batch size on SimCLR in ImageNet. However, we have also shown that the performance of using graded similarity does not relatively deviate from the baseline on CIFAR10 and STL10 when using larger batch sizes. Furthermore, we report exact differences with the referenced baselines in Appendix E. Consequently, we may conclude that these are interesting directions for further research, but they do not significantly affect the findings of these experiments.

The ablation studies were performed on CIFAR10, though these were ideally done on ImageNet. Again, this is due to limitations in computational resources and time, as some ablations required many pre-training iterations. We know the difference between performance on ImageNet and CIFAR10, so this knowledge can be partly extrapolated to the ablation studies. However, we cannot say that all results from the ablations generalize to other datasets. Especially studies that rely on dataset properties, such as batch size and the value of λ, may see different outcomes.

Statistical Significance

This research used identical settings for training and evaluation for all experiments, including identical seeding in all randomizers. However, in an ideal case, multiple runs with different seeds were done for all experiments, especially pre-training. Moreover, different sets of hyperparameters would be especially helpful in ablation studies to confirm the findings further. This would lead to stronger statements and statistical bandwidths for the experimental results, though we hope to have mitigated this as much as possible by the measures mentioned earlier.

7.2 Further Research

This research has shown that implementing graded similarity improves the transferability of the pre-trained encoder and can improve data utilization. Though we provide a thorough evaluation of the novel methodology, there remain many interesting directions in which to explore this concept further.

Downstream Tasks

In this research, we have considered two important downstream tasks: classification and image retrieval. As self-supervised learning methods aim to learn a strong general encoder, it is vital to benchmark it on various downstream tasks. We have seen that incorporating the graded similarity can significantly improve the image retrieval performance, suggesting the learning objective better captures spatial structures and relations. Therefore, it is an interesting research direction to further explore other downstream tasks, such as object detection and semantic segmentation. These tasks may benefit from the features learned by the graded similarity. Moreover, non-object-centric datasets, such as OpenImages [15] and CoCo [18] that better represent real-world uncurated data, could be used for pre-training. The graded similarity may transfer better to such datasets than binary learning objectives, as it is influenced less by noisy images.

Graded Similarity

The implementation of the novel graded similarity measure in contrastive learning objectives can improve data efficiency and transferability of the encoder, but there remain further interesting directions to explore. First, the method was tested with the same hyperparameters of the baseline methods for fair testing purposes. However, the gradient of the GS-loss is different (see Appendix C), and there is additional regularization. This means there is probably a different set of hyperparameters that performs better with this method. Furthermore, we saw slightly better performance at smaller

batch sizes in SimCLR on CIFAR10. Hence, it would be interesting to train without a large batch size optimizer like LARS. Regular SGD may work well or better at small batch sizes, removing the need for large ones.

In addition, the architectures were not changed for the GS implementation. Especially for SimSiam, a less deep projection and prediction layer may lead to better results. This deep architecture is necessary to avoid collapse and reach good performance [5]. However, we show better regularization and stability when removing augmentation. Therefore, a downscale of the projector and predictor architectures may benefit training with the graded similarity.

Furthermore, though the data efficiency is slightly improved on some datasets, there may be further potential that can be exploited by a different cropping strategy, such as object-aware cropping [20] or a multi-crop strategy [36]. These cropping strategies may synergize better with the graded similarity with respect to regular contrastive methods. Multi-crop strategies decrease the performance of these contrastive methods in baseline settings [3]. However, as the asymmetric graded similarity provides additional information about view relations during training, it may synergize well with these multi-crop strategies. This may also encourage further exploration of a different construction of the loss function or different crop ratios.

Finally, it would be interesting to incorporate the graded similarity in other InfoNCE-based methods, such as MoCoV2, or utilize it with transformer architectures [3]. Implementing these different methods may further the understanding of the graded similarity learning objective and provide exciting results.

8 Conclusions

In this research, we have proposed a novel learning objective for (non)-contrastive learning methods that utilizes a graded similarity measure based on shared visual information. These methods consider all views from the same image equally, even though the views may have widely varying amounts of similarity. To address this, the overlap of views was used as a direct measure of distance between their representations. The method has been demonstrated in both contrastive and non-contrastive methods, namely SimCLR and SimSiam. To summarize our key findings, we answer the research questions as proposed in Section 1.

1. What is a good graded similarity measure between image views used as a measure for the distance between their representations?
We have considered the intersection over area (IoA) and the intersection over union (IoU) as similarity measures for view pairs. Our results show the IoA is superior to the IoU as a similarity measure, which is caused by the interesting asymmetric properties of the IoA.
Furthermore, to optimally incorporate the graded similarity with heavy augmentations in (non)-contrastive methods, we have proposed a threshold at which views are considered equivalent. The inclusion of this threshold allows for significant performance increases. We empirically show that the best performance is reached when views that share at least half of their information are pulled to the same representation.

2. To what extent does incorporating a graded similarity measure improve data utilization and efficiency during pre-training?
Our method slightly improves data efficiency and performance on lower-resolution datasets such as CIFAR10 and STL10 for both methods. The implementation in SimCLR also improves training efficiency on ImageNet, but this is not the case for SimSiam. This may be due to different target distributions for ImageNet or network architectures present in SimSiam. Furthermore, including the graded similarity provides interesting regularization properties in SimSiam. SimSiam can avoid near-collapse when removing augmentations when using the graded similarity objective.
Finally, our results suggest that SimCLR with graded similarity does not require as large batch sizes. This reduces memory and computational power requirements since the method better utilizes the data during training. We anticipate that an improved cropping strategy, such as object-aware cropping or a multi-view cropping strategy, may further aid the data utilization of the method due to synergy with the graded similarity.

3. How does the incorporation of graded view similarity affect the learned image representations?
The performance on the respective pre-training dataset does not change significantly when using the graded similarity. More importantly, the representations obtained by the graded similarity objective transfer better to other datasets, as shown by slightly improved transfer learning performance. This is further emphasized by a significant performance increase at an unseen retrieval task, increasing performance up to $1.4\times$. These results suggest a better understanding of spatial structures and relations. We foresee that a larger variety of downstream tasks may further highlight the effectiveness of the graded similarity in enhancing the transferability of the learned image representations.

References

[1] Saleh Albelwi. "Survey on self-supervised learning: auxiliary pretext tasks and contrastive learning methods in imaging". In: *Entropy* 24.4 (2022), p. 551.

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. "Food-101 – Mining Discriminative Components with Random Forests". In: *European Conference on Computer Vision*. 2014.

[3] Mathilde Caron et al. "Emerging properties in self-supervised vision transformers". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 9650–9660.

[4] Ting Chen et al. "A Simple Framework for Contrastive Learning of Visual Representations". In: *Proceedings of the 37th International Conference on Machine Learning*. 2020, pp. 1597–1607.

[5] Xinlei Chen and Kaiming He. "Exploring Simple Siamese Representation Learning". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 15750–15758.

[6] Xinlei Chen et al. "Improved baselines with momentum contrastive learning". In: *arXiv preprint arXiv:2003.04297* (2020).

[7] Afshin Dehghan et al. "View independent vehicle make, model and color recognition using convolutional neural network". In: *arXiv preprint arXiv:1702.01721* (2017).

[8] Carl Doersch, Abhinav Gupta, and Alexei A Efros. "Unsupervised visual representation learning by context prediction". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1422–1430.

[9] Mark Everingham et al. "The pascal visual object classes (voc) challenge". In: *International journal of computer vision* 88 (2010), pp. 303–338.

[10] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. "Unsupervised representation learning by predicting image rotations". In: *arXiv preprint arXiv:1803.07728* (2018).

[11] Jean-Bastien Grill et al. "Bootstrap Your Own Latent - A New Approach to Self-Supervised Learning". In: *Advances in Neural Information Processing Systems*. Vol. 33. 2020, pp. 21271–21284.

[12] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[13] Kaiming He et al. "Momentum contrast for unsupervised visual representation learning". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 9729–9738.

[14] Alex Krizhevsky and Geoffrey Hinton. *Learning multiple layers of features from tiny images*. Tech. rep. Toronto, Ontario: University of Toronto, 2009.

[15] Alina Kuznetsova et al. "The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale". In: *International journal of computer vision* 128.7 (2020), pp. 1956–1981.

[16] Hankook Lee et al. "Improving transferability of representations via augmentation-aware self-supervision". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 17710–17722.

[17] María Leyva-Vallina, Nicola Strisciuglio, and Nicolai Petkov. "Regressing Transformers for Data-efficient Visual Place Recognition". In: *arXiv preprint arXiv:2401.16304* (2024).

[18] Tsung-Yi Lin et al. "Microsoft coco: Common objects in context". In: *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer. 2014, pp. 740–755.

[19] S. Maji et al. *Fine-Grained Visual Classification of Aircraft*. Tech. rep. 2013. arXiv: 1306.5151 [cs-cv].

[20] Shlok Kumar Mishra et al. "Object-aware Cropping for Self-Supervised Learning". In: *Transactions on Machine Learning Research* (2022). ISSN: 2835-8856.

[21] Ishan Misra and Laurens van der Maaten. "Self-supervised learning of pretext-invariant representations". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 6707–6717.

[22] M-E. Nilsback and A. Zisserman. "Automated Flower Classification over a Large Number of Classes". In: *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*. Dec. 2008.

[23] Mehdi Noroozi and Paolo Favaro. "Unsupervised learning of visual representations by solving jigsaw puzzles". In: *European conference on computer vision*. Springer. 2016, pp. 69–84.

[24] Omkar M Parkhi et al. "Cats and dogs". In: *2012 IEEE conference on computer vision and pattern recognition*. IEEE. 2012, pp. 3498–3505.

[25] Deepak Pathak et al. "Context encoders: Feature learning by inpainting". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2536–2544.

[26] Xiangyu Peng et al. "Crafting better contrastive views for Siamese representation learning". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2022). DOI: 10.1109/cvpr52688.2022.01556.

[27] Senthil Purushwalkam and Abhinav Gupta. "Demystifying Contrastive Self-Supervised Learning: Invariances, Augmentations and Dataset Biases". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 3407–3418.

[28] Filip Radenović et al. "Revisiting oxford and paris: Large-scale image retrieval benchmarking". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 5706–5715.

[29] Veenu Rani et al. "Self-supervised learning: A succinct review". In: *Archives of Computational Methods in Engineering* 30.4 (2023), pp. 2761–2775.

[30] Pierre H Richemond et al. "BYOL works even without batch statistics". In: *arXiv preprint arXiv:2010.10241* (2020).

[31] Olga Russakovsky et al. "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision* 115 (2015), pp. 211–252.

[32] Yonglong Tian et al. "What makes for good views for contrastive learning?" In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 6827–6839.

[33] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. "Representation Learning with Contrastive Predictive Coding". In: *arXiv e-prints*, arXiv:1807.03748 (July 2018). DOI: 10.48550/arXiv.1807.03748. arXiv: 1807.03748 [cs.LG].

[34] Richard Zhang, Phillip Isola, and Alexei A Efros. "Colorful image colorization". In: *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*. Springer. 2016, pp. 649–666.

[35] Tong Zhang et al. "Leverage your local and global representations: A new self-supervised learning strategy". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2022). DOI: 10.1109/cvpr52688.2022.01608.

[36] Wenyi Zhao et al. "LESSL: Can LEGO sampling and collaborative optimization contribute to self-supervised learning?" In: *Information Sciences* 615 (2022), pp. 475–490. ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2022.10.058.

A Pre-Training Details

We use PyTorch for the implementation of the models. For all randomized we use seed 5. We pre-train on a single NVIDIA L40 GPU.

Augmentations
We follow the augmentations as reported in the baseline papers [4, 5].

- **Random cropping, resizing and flipping.** For SimCLR we use a crop ratio of $[0.08, 1]$, while for SimSiam we use $[0.2, 1]$ of the original size. After this, resizing is performed to $[3/4, 4/3]$ of the original aspect ratio. Random flipping is performed with probability 0.5.
- **Color distortion**. We perform color jittering and color dropping. For SimCLR, color jittering is done with probability 0.8, with strength 0.8 for brightness, contrast and saturation, while 0.2 strength is used for hue. Color dropping is used with probability 0.2. SimSiam uses the same settings, but half strength for color jittering.
- **Blurring**. Finally, blurring is done with $\sigma \in [0.1, 2.0]$ with probability 0.2. Blurring is not done for pre-training on CIFAR10.

The baseline hyperparameters and architectures for both methods are used, which we will now detail.

SimCLR For all datasets, a 2 layer projection head is used with hidden dimension 512 and output dimension 128. We use a temperature of 0.5, which is optimal for lower batch size and shorter (< 300 epoch) pre-training [4]. For ImageNet pre-training the following settings hold. A LARS optimizer with learning rate 0.3 and weight decay 10^{-6} is used with a batch size of 256. The learning rate is scaled by a cosine decay scheduler with warm-up of 10 epochs. We pre-train for 100 epochs. For CIFAR10 and STL-10 pre-training we use a 1024 batch size and pre-train for 800 epochs.

SimSiam For ImageNet pre-training, a 3-layer projection head is used in the architecture, with hidden dimension 2048. The prediction MLP has 2 layers with hidden dimension 512. In addition, a SGD optimizer with learning rate 0.05, weight decay 10^{-4} and momentum 0.9 is used. For CIFAR10 and STL10 pre-training, a 2 layer projection head with dimension 2048 is used, while the predictor remains the same. Furthermore, a lower learning rate of 0.03 and higher weight decay of $5 \cdot 10^{-4}$ is used. Pre-training duration for the different datasets is the same as for SimCLR. We use a batch size of 256 for all experiments as this gives optimal performance [5]. Finally, a cosine decay without warm-up learning rate scheduler is used for all experiments.

B Relation Between MSE and Cosine Similarity

Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, and $\|\cdot\|_2$ be the L2 norm. Then $\bar{\mathbf{x}} = \frac{\mathbf{x}}{\|\mathbf{x}\|_2}$ such that $\|\bar{\mathbf{x}}\|_2^2 = 1$. Rewriting the mean squared error (MSE):

$$\begin{aligned}
\text{MSE} &= \|\bar{\mathbf{x}} - \bar{\mathbf{y}}\|_2^2 \\
&= \|\bar{\mathbf{x}}\|_2^2 + \|\bar{\mathbf{y}}\|_2^2 - 2\left(\bar{\mathbf{x}} \cdot \bar{\mathbf{y}}\right) \\
&= 2 - 2\left(\bar{\mathbf{x}} \cdot \bar{\mathbf{y}}\right) \\
&= 2 - 2\cos(\theta) \\
&= 2 \cdot (1 - \cos(\theta))
\end{aligned}$$

Gives the negative cosine similarity, apart from a constant and offset. Therefore, minimizing the MSE for L2 normalized vectors is the same learning objective as minimizing the negative cosine similarity.

C Gradient of D_{GS}

$$D_{GS}(\mathbf{z}_i, \mathbf{z}_j; \psi_{i,j}) = \left\| \|\bar{\mathbf{z}}_i - \bar{\mathbf{z}}_j\|_2 - (1 - \psi_{i,j}) \right\|_2^2$$

We define $d = \|\bar{p}(\mathbf{z}_i) - \bar{\mathbf{z}}_j\|_2$, such that:

$$D_{GS} = \|d - (1 - \psi_{i,j})\|_2^2$$

We then compute the gradient as:

$$\begin{aligned}\frac{\partial D_{GS}}{\partial(\mathbf{z}_i)} &= \frac{\partial D_{GS}}{\partial d} \cdot \frac{\partial d}{\partial \bar{\mathbf{z}}_i} \\ &= 2\left(d - (1 - \psi_{i,j})\right) \cdot \frac{\bar{\mathbf{z}}_i - \bar{\mathbf{z}}_j}{\left\|\bar{\mathbf{z}}_i - \bar{\mathbf{z}}_j\right\|_2} \\ &= 2\left(\left\|\bar{\mathbf{z}}_i - \bar{\mathbf{z}}_j\right\|_2 - (1 - \psi_{i,j})\right) \cdot \frac{\bar{\mathbf{z}}_i - \bar{\mathbf{z}}_j}{\left\|\bar{\mathbf{z}}_i - \bar{\mathbf{z}}_j\right\|_2}\end{aligned}$$

Which differs from the gradient of the regular MSE:

$$\frac{\partial \mathcal{L}_{MSE}}{\partial(\mathbf{z}_i)} = 2\left(\bar{\mathbf{z}}_i - \bar{\mathbf{z}}_j\right)$$

The difference in the gradient is then:

$$\frac{\partial \mathcal{L}_{MSE}}{\partial \mathbf{z}_i} - \frac{\partial D_{GS}}{\partial(\mathbf{z}_i)} = 2\left(1 - \psi_{i,j}\right) \cdot \frac{\bar{\mathbf{z}}_i - \bar{\mathbf{z}}_j}{\left\|\bar{\mathbf{z}}_i - \bar{\mathbf{z}}_j\right\|_2}$$

This means that the direction of the gradient is corrected by a factor that is determined by their similarity measure. Note this correction is proportional to the angle between the representations. When $\psi_{i,j} = 1$ (overlapping views, depending on λ), the gradient is that of the regular MSE. With this construction of D_{GS}, we conserve the magnitude of the gradient, making it possible to compare it with the baseline under identical hyperparameter settings.

D Derivation of $\mathcal{L}_{C-GS}$

The contrastive loss used in contrastive methods such as SimCLR is a form of the InfoNCE loss called NT-Xent [4], defined as:

$$\mathcal{L}_{NTX} = -\log \frac{\exp(\frac{1}{\tau} \cdot \|\bar{\mathbf{z}}_i\| \cdot \|\bar{\mathbf{z}}_j\|)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\frac{1}{\tau} \cdot \|\bar{\mathbf{z}}_i\| \cdot \|\bar{\mathbf{z}}_k\|)}$$

Where N is the batch size, τ is a temperature parameter and the summation is over different augmented views of images in the batch. This can be rewritten as the following, observing that the first term is the negative cosine similarity:

$$\mathcal{L}_{NTX} = -\frac{1}{\tau} \cdot \|\bar{\mathbf{z}}_i\| \cdot \|\bar{\mathbf{z}}_j\| + \log \sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\frac{1}{\tau} \cdot \|\bar{\mathbf{z}}_i\| \cdot \|\bar{\mathbf{z}}_k\|)$$

We can then rewrite the negative cosine similarity as the mean squared error. This does not change the learning objective, see Appendix B, but allows the problem to be changed into regression with the similarity measure as a target. Again using the notation $\bar{\mathbf{z}} = \mathbf{z}/\left\|\mathbf{z}\right\|_2$ for the L2 normalised representation, we obtain:

$$\mathcal{L} = \frac{1}{\tau}\left(\left\|\bar{\mathbf{z}}_i - \bar{\mathbf{z}}_j\right\|_2^2 - 1\right) + \log \sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\frac{1}{\tau} \cdot \|\bar{\mathbf{z}}_i\| \cdot \|\bar{\mathbf{z}}_k\|)$$

We want to minimize the MSE between the normalized euclidean distance between the representations and the similarity distance. Doing this and removing constant terms as they do not alter the learning objective, we obtain:

$$\begin{aligned}\mathcal{L}_{C-GS} =& \frac{1}{\tau}\left(\left\|\left\|\bar{\mathbf{z}}_i - \bar{\mathbf{z}}_j\right\|_2 - (1 - \psi_{i,j})\right\|_2^2\right) \\ &+ \log \sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\frac{\mathbf{z}_i \cdot \mathbf{z}_k}{\tau \cdot \|\mathbf{z_i}\| \cdot \|\mathbf{z_k}\|})\end{aligned}$$

After substituting the expression with D_{GS}:

$$\begin{aligned}\mathcal{L}_{C-GS}(\mathbf{z}_i, \mathbf{z}_j) =& \frac{1}{\tau} D_{GS}(\mathbf{z}_i, \mathbf{z}_j; \psi_{i,j}) \\ &+ \log \sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\frac{1}{\tau} \cdot \|\bar{\mathbf{z}}_i\| \cdot \|\mathbf{z}_k\|)\end{aligned}$$

It is important to note that the loss is calculated over all positive pairs; (i, j) as well as (j, i).

E Comparison With State-of-the-Art

As we have slightly different pre-training settings and durations, we can compare our results to that of state-of-the-art referenced in the papers [4, 5].
Though we train the baseline SimCLR with a lower batch size of 256, we outperform their results with 0.6%, even when they use improved learning rate scaling, which we do not use. On transfer learning, we get slightly lower performance. This averages about 10%, but this depends on the dataset. This can be accounted to the lower pre-training duration, where they train $10\times$ as long.
Our SimSiam implementation reaches slightly lower linear evaluation accuracy, namely a decrease of 2.8%. This may be due to the linear evaluation protocol we use, as they mention the SGD with lower batch size gives slightly lower accuracy [5]. Furthermore, it may be due to differences in augmentations used during linear evaluation, which they do not report. Consequently, we used the same as in SimCLR [4]. They do not report transfer learning classification performance, so this cannot be compared.