Residual Relaxation for Multi-view Representation Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Unsupervised representation learning has experienced a renaissance recently due to its rapidly growing performance that could nearly match supervised learning. In this paper, we point out that previous practice can be divided into two regimes, pretext-aware methods and pretext-invariant methods. We notice that the two methodologies actually learn to encode different but both useful image semantics, which motivates us to combine the image semantics of both methods and further improve the representation quality. In this paper, we propose Prelax, a generic method for bridging the gap between the two methodologies with pretext-aware residual relaxation. Experiments show that our method can not only improve existing pretext-invariant methods, but also benefit from stronger image augmentations that will hurt pretext-invariant methods, such as rotation. Remarkably, for the first time, unsupervised learning yields superior performance to supervised learning on CIFAR-10 and ImageNette with our Prelax method. We believe this will encourage the community to explore new mechanics that could combine image semantics learned from different pretext tasks.

1 Introduction

Without access to labels, unsupervised deep learning relies on surrogate objectives to learn data representations, and the chosen surrogate objectives largely determine the quality and property of the learned representations [22, 17]. Although countless many works have been proposed [15, 14], we notice that most unsupervised representation learning methods can be divided into two regimes, the pretext-aware methods and the pretext-invariant methods, and our work draws heavy inspiration from this taxonomy.

The pretext-aware methods learn representations via a manually designed pretext task, which applies a transformation (*e.g.,* image rotation) to the input image and requires the learner to predict properties of the transformation (*e.g.,* the rotation angle) from the transformed images. As a result, the extracted image representations are encouraged to become aware of the applied pretext (*e.g.,* image rotation). Thus, we name this type of method as *pretext-aware methods*. As its alias name Self-Supervised Learning (SSL) suggests, pretext-aware methods are essentially supervised learning (typically classification) with self-generated surrogate labels. Pretext-aware methods mainly differ by the design of the pretext tasks, to name a few, Rotation [9], Jigsaw [19], Relative Path Location [6], Colorization [24]. We refer to [14] for a comprehensive discussion.

Though pretext-aware methods have been popular for a while (roughly 2015-2018), state-of-the-art results are mostly set by the pretext-invariant methods [18], *e.g.,* SimCLR [2, 3], MoCo [11, 4], BYOL [10]. In contrast to pretext-aware methods that aim at predicting pretext labels from the representations, pretext-invariant methods instead try to learn representations that are *invariant* to different predefined augmentations. A typical approach, contrastive learning [20, 13, 21, 2, 11],

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

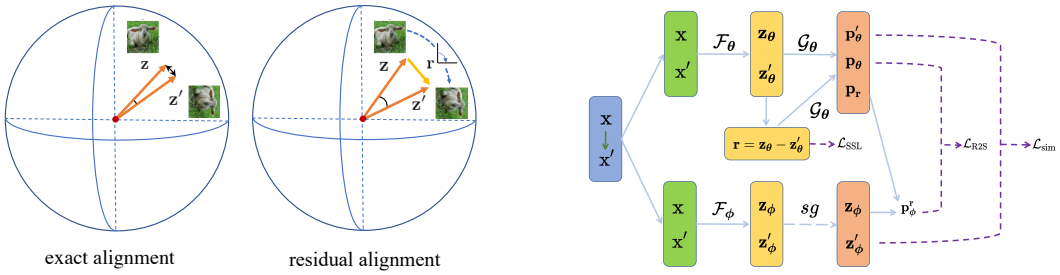

Figure 1: Left: a comparison of the exact alignment (left) of pretext-invariant methods and the residual alignment (right) of our proposed Prelax. As the rotation modifies the image semantics, our Prelax adopts a rotation-aware residual vector $\mathbf{r}$ that relaxes the exact alignment between the two representations, *i.e.,* from $\mathbf{z}' \rightarrow\leftarrow \mathbf{z}$ to $\mathbf{z}' \rightarrow\leftarrow \mathbf{z} + \mathbf{r}$. Right: a diagram of our proposed Prelax (Section 4). An image $\mathbf{x}$ is firstly augmented as $\mathbf{x}'$. Then the positive pair $(\mathbf{x}, \mathbf{x}')$, is processed by the online network $\mathcal{F}_{\boldsymbol{\theta}}$ and the target network $\mathcal{F}_{\boldsymbol{\phi}}$, respectively. Output of the online network is further processed by the target network $\mathcal{G}_{\boldsymbol{\theta}}$, and the gradient of $\mathcal{F}_{\boldsymbol{\phi}}$ is detached, *i.e.,* $\mathrm{stop_grad}$, denoted as sg.

achieves this by maximizes the similarity between two random augmentations of the same image (as positive samples) and minimizes the similarity to other images (as negative samples). Later, some researchers find that the negative samples can be also omitted with certain techniques [10, 5, 1]. Therefore, the core of pretext-invariant methods is to learn representations that are invariant to random augmentations defined by the pretext.

Thus, we can see that pretext-aware and pretext-invariant methods are actually *complementary* to each other. Take the rotation pretext for an example. Pretext-aware methods learn to extract features that are helpful for predicting the rotation angle, while pretext-invariant methods try to avoid these features and extract features that are invariant to different rotation angles, instead. The intriguing thing is that the two methods both learn representations with good performance, indicating that pretext-aware and pretext-invariant representations actually encode complementary but both useful semantics of the same image. The observation motivates us to combine pretext-aware and pretext-invariant methods to obtain better representations. A straightforward approach would be combining pretext-aware and pretext-invariant objectives as multi-task learning [8]. However, as we discuss above, the two objectives are contradictory to each other, one encourages pretext-awareness and one encourages pretext-invariance. Unfortunately, simply combining them would lead to the inevitable tradeoff that degrades the quality of each representation.

Besides, we further note that their choice of pretext tasks, *i.e.,* image augmentations, actually differ a lot. For example, image rotation works well for pretext-aware methods [9], but is rarely used in the pretext-invariant method [2]. We conjecture that it is because rotation is a stronger augmentation that will largely affect the image semantics. Consequently, pretext-invariant learning that enforces two rotation-augmented images to have the same representations will inevitably degrade the representation quality, as the two are semantically very different. This indicates that for strong augmentations that modify useful image semantics, enforcing pretext-invariance will not help but damage the representation quality. In other words, in order to benefit from stronger pretext, we should take the semantic shift caused by augmentations into account.

Motivated by the observations above, we propose a generic approach, Pretext-aware REsidual ReLAXation (Prelax), to account for the semantic shift and bridge the gap between pretext-aware and pretext-invariant methods. As illustrated in Figure 1, we propose to add a residual vector $\mathbf{r}$, the difference between the representation $\mathbf{z}$ of an image $\mathbf{x}$ and $\mathbf{z}'$ of its augmented version $\mathbf{x}'$, *i.e.,* $\mathbf{r} = \mathbf{z}' - \mathbf{z}$, to the pretext-invariant objective such that they have a certain amount of flexibility to account for the semantic shift introduced by the augmentation. Specifically, pretext-invariant methods enforce them to share the same representations by minimizing the similarity loss $\|\mathbf{z}' - \mathbf{z}\|_2^2$ (informal). Instead, we use the residual $\mathbf{r} = \mathbf{z}' - \mathbf{z}$ to encode the difference in pretext semantics and design a relaxed version of the similarity loss, $\|\mathbf{z}' - (\mathbf{z} + \mathbf{r})\|_2^2$ (informal). This relaxation also enables us to utilize stronger pretext like rotation because we can encode the modified pretext semantics through the residual representation. To ensure that the residual encodes pretext-aware semantics, we add a self-supervised loss for the residual to predict the augmentations of $\mathbf{x}'$. Besides, we also add a

penalty on the similarity loss to make sure the two representations ($\mathbf{z}$ and $\mathbf{z}'$) still stay close and the residual scale is relatively small.

In this way, we can encoder both pretext-aware and pretext-invariant image semantics within the same embedding space and successfully avoid contradiction between the two objectives by encoding the pretext-aware semantics through the residuals between pretext-invariant representations. We implement Prelax on two pretext-invariant backbone, BYOL [10] and SimSiam [5], and improve over the baselines significantly with little computation overhead. We summarize our contributions as follows:

- We propose a generic technique, Residual Relaxation (Prelax), to account for the semantic shift caused by image augmentations. In addition, Prelax enables us to benefit from stronger data augmentations that are harmful in pretext-invariant methods, *e.g.,* rotation.
- With pretext-aware learning of the residual representation, our method successfully bridges the gap between pretext-aware and pretext-invariant representations without causing contradiction. Furthermore, we achieve state-of-the-art performance by combining the two complementary yet useful semantics.
- Experiments show that our methods successfully decouple the pretext-aware and pretext-invariant semantics in the embedding space. ??

2 Related Work

Pretext-aware Methods Pretext-aware methods, also known as self-supervised learning[1], require only unlabeled data in order to formulate a pretext learning task, such as predicting image rotation [9], pixel colors [24], relative patch location [6], or solving jigsaws [19], where a target label can be crafted without supervision. It is widely discussed that the similarity between the pretext task and the downstream task largely effects the performance of the learn representations [14].

Pretext-invariant Methods Although pretext-invariant methods can date back to old techniques such as metric learning and Exemplar [7], its renaissance is due to a mutual information perspective of the contrastive learning of deep representations [20, 13]. Ever since, the interest for contrastive learning has surged rapidly and many effective techniques been devised, *e.g.,* memory bank [18], momentum encoder [11], MLP projecter [2], and obtain state-of-the-art results that nearly matches supervised learning [3]. Recently, several similarity-based works [10, 5, 1] show that we can even learn representations by only matching positive samples without the contrastive negative samples. They instead prevent the embedding space from collapse with implicit mechanisms like $\mathrm{stop_grad}$, predictor, momentum encoder [10], and Sinkhorn iterations [1].

Bridging Pretext-aware and Pretext-invariant Methods Although there are plenty of works on each branch, how to bridge the two complementary methods largely remain under-explored. Previous to our work, only a few have worked in this direction. [8] combines Rotation (pretext-aware) and AMDIM (pretext-invariant) methods for multi-domain representation learning. However, as discussed previously, a direct combination of the two contradictory objectives may harm the final representation. LooC [23] recently proposes to separate the embedding space to several parts, where each subspace learns local invariance *w.r.t.* a specific augmentation instead of global pretext-invariance *w.r.t.* all augmentations. But this is achieved at the cost of limiting the representation flexibility of each pretext to the predefined subspace. Our methods instead adopts residual relaxation for bridging the two methods, which is generic, adaptive, compatible with different pretext.

3 Methodology

In this section, we present the methodology of Prelax, *i.e.,* the residual relaxation, the similarity constraint, and the pretext-aware prediction. To clarify the details of proposed methods, we provide a demonstration based on BYOL[10], including Prelax-std for general pretext-invariant augmentations and Prelax-rot that incorporates stronger pretext-aware augmentations like rotations [9].

[1]The name SSL is also used to refer to pretext-invariant methods in some previous works. Nevertheless, strictly speaking, it is more suitable for pretext-aware methods as they explicitly create pretext labels for self-supervision and train models just like supervised learning.

3.1 Problem Formulation

Given unlabeled data $\{\mathbf{x}_i\}$, unsupervised representation learning aims to learn an encoder network $\mathcal{F}_{\boldsymbol{\theta}}$ that extract meaningful low-dimensional representations $\mathbf{z} \in \mathbb{R}^{d_z}$ from high-dimensional input images $\mathbf{x} \in \mathbb{R}^{d_x}$. We first introduce the general formulation of similarity-based pretext-invariant methods that our method is built upon [10, 5].

For an input image $\mathbf{x} \in \mathbb{R}^{d_x}$, we can generate a different view by data augmentation, $\mathbf{x}' = t(\mathbf{x})$, where $t \in \mathcal{T}$ refers to a randomly drawn augmentation operator from the pretext set $\mathcal{T}$. Then, the transformed input $\mathbf{x}'$ and the original input $\mathbf{x}$ are passed into an online network $\mathcal{F}_{\boldsymbol{\theta}}$ and a target network $\mathcal{F}_{\boldsymbol{\phi}}$, respectively. Additionally, the output of the online network is further processed by an MLP predictor network $\mathcal{G}_{\boldsymbol{\theta}}$, to match the output of the target network. As $\mathbf{x}$ and $\mathbf{x}'$ are two different views of the same image, they should have similar representations. Motivated by this, we can align the two positive samples with the following similarity loss that minimizes $\boldsymbol{\theta}$ *w.r.t.* their ℓ_2 distance,

$$\mathcal{L}_{\text{sim}}(\mathbf{x}', \mathbf{x}; \boldsymbol{\theta}) = \|\mathcal{G}_{\boldsymbol{\theta}}\left(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}')\right) - \mathcal{F}_{\boldsymbol{\phi}}(\mathbf{x})\|_2^2. \tag{1}$$

Each model output $\mathbf{z}$ is typically projected to the unit spherical ball before calculating the distance, *i.e.,* , $\frac{\mathbf{z}}{\|\mathbf{z}\|_2}$, which makes the distance equivalent to the cosine similarity [2].

Besides, contrastive methods [20, 13, 21, 2] further encourages the uniformity of the representations by minimizing the similarity between positive and negative samples

$$\mathcal{R}_{\text{dis}}(\mathbf{x}'; \boldsymbol{\theta}) = \log \sum_{k=1}^{K} \exp(\mathcal{G}_{\boldsymbol{\theta}}(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}'))^\top \mathcal{F}_{\boldsymbol{\phi}}(\mathbf{x}_k^-)), \tag{2}$$

where $\{\mathbf{x}_k^-\}_{k=1}^K$ refers to K independently drawn negative samples. In this paper we mainly focus on similarity-based methods because our proposed method only aims to improve the similarity loss between positive samples. Nevertheless, it can also be easily generalized to contrastive methods by including the dissimilarity regularization above.

3.2 Motivation

As shown in equation 1, pretext-invariant methods pursue the exact alignment between different augmented views of an image, *i.e.,* the positive pairs. However, as discussed previously, the augmentation indeed brings certain amount of semantic shift, as pretext-aware methods could also learn meaningful representation by predicting the assigned augmentation. As a result, exact alignment may hurt the representation learning ability and lead to sub-optimal performance, particularly when the data augmentation is too strong for the positive pairs to be matched exactly. For example, as shown in Table xx, adding image rotation to the pretext set $\mathcal{T}$ hurts the representation ability.

Motivated this, we believe that we need to relax the exact alignment in equation 1 to account for the semantic shift brought by the data augmentation. A straightforward solution is to utilize a margin loss such as

$$\mathcal{L}_{\text{margin}}(\mathbf{x}', \mathbf{x}; \boldsymbol{\theta}) = \max(\|\mathcal{G}_{\boldsymbol{\theta}}\left(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}')\right) - \mathcal{F}_{\boldsymbol{\phi}}(\mathbf{x})\|_2^2 - \alpha, 0), \tag{3}$$

where $\alpha > 0$ is a predefined threshold that controls the tolerance for the mismatch between positive samples. However, as the representation space keeps shifting along the training progress of model parameters $\boldsymbol{\theta}$, it is generally hard to maintain a proper threshold dynamically. Besides, we have no idea whether the relaxed gap between representations is encoded with pretext semantics, or simply caused by insufficient training. As a result, the margin loss is rarely used in pretext-invariant methods.

Nevertheless, the analysis sheds light on how to design the relaxation in terms the following points:

- the relaxation should be flexible and adaptive to the training progress and the difficulty of the image pairs;
- the relaxation should avoid trivial solutions;
- the relaxation should account for the semantic shift introduced by the augmentations.

In this next part, we will present our proposed method that provides a principled solution to all these challenges.

4 Proposed Method

Instead of using a predefined threshold η, we propose to relax the exact alignment with a *pretext-aware residual vector* between the representations, which is flexible, learnable and semantically more meaningful. To achieve this, we propose a residual relaxed version of the similarity loss, and further constrain the relaxation to prevent trivial solutions. At last, we utilize the residual vector to predict the corresponding pretext, making it pretext-aware. In this way, we can combine the representation ability of pretext-invariant methods and pretext-aware methods in this same representation space. In the following, we introduce each component of our method.

Residual Relaxed Similarity (R2S) Loss As discussed previously, if the difference between $\mathbf{x}'$ and $\mathbf{x}$ is large enough, it is improper to align their representations exactly, *i.e.,* $\mathbf{z}' \neq \mathbf{z}$. Instead, we notice that the following *representation identity* will still hold

$$\mathbf{z}' = \mathbf{z} + (\mathbf{z}' - \mathbf{z}), \tag{4}$$

which makes it a more proper objective in general cases.

Inspired by this intuition, we believe that it is better to replace the exact alignment (denoted as $\rightarrow\leftarrow$) in the similarity loss (equation 1), *i.e.,*

$$\mathcal{G}_{\boldsymbol{\theta}}(\mathbf{z}'_{\boldsymbol{\theta}}) \rightarrow\leftarrow \mathbf{z}_{\boldsymbol{\phi}} \tag{5}$$

by aligning the two sides of the identity (equation 4)

$$\mathcal{G}_{\boldsymbol{\theta}}(\mathbf{z}'_{\boldsymbol{\theta}}) \rightarrow\leftarrow \mathbf{z}_{\boldsymbol{\phi}} + \mathcal{G}_{\boldsymbol{\theta}}(\mathbf{z}'_{\boldsymbol{\theta}} - \mathbf{z}_{\boldsymbol{\theta}}), \tag{6}$$

where the residual vector

$$\mathbf{r} \triangleq \mathbf{z}'_{\boldsymbol{\theta}} - \mathbf{z}_{\boldsymbol{\theta}} = \mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}') - \mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}), \tag{7}$$

is further processed by the predictor $\mathcal{G}_{\boldsymbol{\theta}}$ to match the target representation. Here the residual vector $\mathbf{r}$, as the difference between the representations of $\mathbf{x}' = t(\mathbf{x})$ and $\mathbf{x}$, is supposed to encode the semantic shift introduced by the augmentation $t \in \mathcal{T}$. This identity alignment encourages better representation of the image semantics with more flexibility.

To further enable a better tradeoff between the exact alignment (equation 5) and the identity alignment (equation 6), we propose the following unified alignment:

$$\mathcal{G}_{\boldsymbol{\theta}}(\mathbf{z}'_{\boldsymbol{\theta}}) \rightarrow\leftarrow \mathbf{z}_{\boldsymbol{\phi}} + \alpha\mathcal{G}_{\boldsymbol{\theta}}(\mathbf{z}'_{\boldsymbol{\theta}} - \mathbf{z}_{\boldsymbol{\theta}}), \tag{8}$$

where $\alpha \in [0, 1]$ denotes a tradeoff coefficient. When $\alpha = 0$, we recover the exact alignment and when $\alpha = 1$, we recover the identity alignment.

Following the above analysis, we propose the Residual Relaxed Similarity (R2S) loss,

$$\mathcal{L}^{\alpha}_{\mathrm{R2S}}(\mathbf{x}', \mathbf{x}; \boldsymbol{\theta}) = \|\mathcal{G}_{\boldsymbol{\theta}}\left(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}')\right) - \mathcal{F}^{\alpha}_{\boldsymbol{\phi}}(\mathbf{x})\|_2^2. \tag{9}$$

where

$$\mathcal{F}^{\alpha}_{\boldsymbol{\phi}}(\mathbf{x}) = \mathcal{F}_{\boldsymbol{\phi}}(\mathbf{x}) + \alpha\mathcal{G}_{\boldsymbol{\theta}}(\mathbf{r}) = \mathcal{F}_{\boldsymbol{\phi}}(\mathbf{x}) + \alpha\mathcal{G}_{\boldsymbol{\theta}}(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}') - \mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x})). \tag{10}$$

stands for the residual relaxed target. Typically we will detach the dependency of $\mathcal{F}^{\alpha}_{\boldsymbol{\phi}}(\mathbf{x})$ on the online parameters $\boldsymbol{\theta}$, *i.e.,* $\mathrm{stop_grad}$ [10, 5], to avoid representation collapse.

Similarity Constraint Loss Although the identity alignment in equation 6 is more general than the exact alignment and enjoys much more flexibility, it alone has a major drawback for learning representations. That is, because the residual vector is unbounded and can be arbitrary large, the distance between $\mathbf{z}'$ and $\mathbf{z}$ can be arbitrarily large. But this should not be the case as we have the prior knowledge that the positive pairs $\mathbf{x}$ and $\mathbf{x}'$ has very much similar semantics, which is utilized in pretext-invariant methods.

Therefore, we can further utilize this prior knowledge to prevent bad cases of R2S when $\|\mathbf{z}' - \mathbf{z}\|_2^2$ becomes too large. We can achieve this by taking it as a constraint, which leads to the following constrained optimization problem

$$\begin{aligned} &\min_{\boldsymbol{\theta}} \sum_{\mathbf{x}} \|\mathcal{G}_{\boldsymbol{\theta}}\left(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}')\right) - \mathcal{F}^{\alpha}_{\boldsymbol{\phi}}(\mathbf{x})\|_2^2 \\ &\text{such that } \|\mathcal{G}_{\boldsymbol{\theta}}\left(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}')\right) - \mathcal{F}_{\boldsymbol{\phi}}(\mathbf{x})\|_2^2 \leq \varepsilon, \forall \mathbf{x}. \end{aligned} \tag{11}$$

For simplicity, we transform it into an unconstrained Lagrangian objective with a fixed multiplier $\beta \geq 0$, *i.e.,*

$$\mathcal{L}^{\alpha}_{\mathrm{R2S}}(\mathbf{x}', \mathbf{x}; \boldsymbol{\theta}) + \beta \mathcal{L}_{\mathrm{sim}}(\mathbf{x}', \mathbf{x}; \boldsymbol{\theta}). \tag{12}$$

In this way, the former R2S loss encourages residual relaxation with the identity alignment, while the latter similarity loss constrains the relaxation scale at the same time. By tuning the multiplier β, we can flexibly choose the degree of the residual relaxation.

Pretext-aware Loss Till now, there leaves only one of the three requisitions in Section 3.2, that is, how to make sure that the relaxed residual indeed accounts for the semantic shift introduced by data augmentation. Drawing inspirations from the pretext-aware methods, *i.e.,* Self-Supervised Learning (SSL) [9, 19, 6, 14] that learn representations by predicting the pretext, we can also ensure the pretext-awareness of our residual vector by utilizing it to predict the semantic shift between $\mathbf{x}' = t(\mathbf{x})$ and $\mathbf{x}$, *i.e.,* the augmentation operator t.

In practice, we can collect the specific hyper-parameters of the augmentation t, *e.g.,* the rotation angle, the crop size and whether grayscale is applied [2]. These hyper-parameters typically can be divided into two categories, the categorical variables $\mathbf{t}^c$, and the continuous variables $\mathbf{t}^r$. Given the target variables, we pass the residual vector $\mathbf{r}$ (equation 7) through an additional SSL predictor $\mathcal{H}_{\boldsymbol{\theta}}$, and learn pretext-awareness with the Cross Entropy (CE) classification loss and the MSE regression loss, respectively:

$$\mathcal{L}_{\mathrm{SSL}}(\mathbf{x}', \mathbf{x}, \mathbf{t}; \boldsymbol{\theta}) = \mathrm{CE}(\mathcal{H}^c_{\boldsymbol{\theta}}(\mathbf{r}), \mathbf{t}^c) + \|\mathcal{H}^r_{\boldsymbol{\theta}}(\mathbf{r}) - \mathbf{t}^r\|_2^2. \tag{13}$$

Prelax Combing the three objectives, the R2S relaxation loss, the similarity constraint loss, and the pretext-aware SSL loss, we obtain the overall Pretext-aware REsidual ReLAXation (Prelax) objective,

$$\mathcal{L}^{\alpha}_{\mathrm{R2S}}(\mathbf{x}', \mathbf{x}; \boldsymbol{\theta}) + \beta \mathcal{L}_{\mathrm{sim}}(\mathbf{x}', \mathbf{x}; \boldsymbol{\theta}) + \gamma \mathcal{L}_{\mathrm{SSL}}(\mathbf{x}', \mathbf{x}; \boldsymbol{\theta}), \tag{14}$$

where α tradeoffs between the similarity and the identity alignment, β constrains the degree of similarity between positive pairs, and γ adjusts the amount of pretext-awareness of the residual. Further ablation study demonstrates all three components of Prelax are indispensable.

With a combination of the three objectives, we have fulfilled all the three requisitions in Section 3.2, and successfully encode the shifted semantic between $\mathbf{x}'$ and $\mathbf{x}$, *i.e.,* the pretext-relevant semantic, into the residual vector $\mathbf{r}$. As a result, the pretext-invariant representations, *e.g.,* $(\mathbf{z}'_{\boldsymbol{\theta}} + \mathbf{z}_{\boldsymbol{\theta}})/2$, can be decoupled from the pretext-aware representation, *e.g.,* $\mathbf{r} = \mathbf{z}'_{\boldsymbol{\theta}} - \mathbf{z}_{\boldsymbol{\theta}}$. As our learned representation space encodes both pretext-invariant and pretext-aware features, it can be semantically richer than each alone.

5 Practical Implementation

In previous section, we provide an overview of our proposed Prelax with a general formulation. In this part, we instantiate our method with three practical implementations based on existing similarity-based methods [5, 10]: one is built up existing augmentation strategies (Prelax-std), one incorporates a stronger augmentation, image rotation (Prelax-rotation), and the last one combines the previous two strategies (Prelax-all).

5.1 Backbone

BYOL [10] and SimSiam [5] are both similarity-based methods and they only differ in the design of the target network $\mathcal{F}_{\boldsymbol{\phi}}$. BYOL [10] utilizes momentum update of the target parameters $\boldsymbol{\phi}$ from the online parameters $\boldsymbol{\theta}$, *i.e.,* $\boldsymbol{\phi} \leftarrow \tau\boldsymbol{\phi} + (1-\tau)\boldsymbol{\theta}$, where $\tau \in [0, 1]$ is the target decay rate. Instead, SimSiam [5] simply clones the online network as the target network, *i.e.,* $\boldsymbol{\phi} \leftarrow \boldsymbol{\theta}$. Next, we mainly take SimSiam for discussion and our analysis also applies to BYOL.

For a given training image $\boldsymbol{x}$, SimSiam randomly samples two augmentations t_1, t_2 from a standard augmentation list $\mathcal{T}$ designed for pretext-invariant methods, which is composed of a sequence of predefined augmentations [2, 5]. We apply t_1, t_2 to $\mathbf{x}$ and get $\mathbf{x}_1$ and $\mathbf{x}_2$, respectively. Afterwards, SimSiam tries to maximize the similarity of the two augmented views in the representation space. As the two images $(\mathbf{x}_1, \mathbf{x}_2)$ can both serve as the input and the target to each other, SimSiam minimizes the following dual similarity objective for better sample efficiency

$$\|\mathcal{G}_{\boldsymbol{\theta}}(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_1)) - \mathcal{F}_{\boldsymbol{\phi}}(\mathbf{x}_2)\|_2^2 + \|\mathcal{G}_{\boldsymbol{\theta}}(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_2)) - \mathcal{F}_{\boldsymbol{\phi}}(\mathbf{x}_1)\|_2^2. \tag{15}$$

5.2 Prelax-std with Existing Pretext

To begin with, we can directly generalize the baseline method with our Prelax relaxation method under existing augmentation strategies. Specifically, for the same positive pair $(\mathbf{x}_1, \mathbf{x}_2)$, we can calculate their residual vector

$$\mathbf{r}_{12} = \mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_1) - \mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_2), \tag{16}$$

and use it for the relaxation loss (equation 9) that approximately aligns the identity $\mathbf{z}_1 = \mathbf{z}_2 + \mathbf{r}_{12}$ with

$$\mathcal{L}^{\alpha}_{\mathrm{R2S}}(\mathbf{x}_1, \mathbf{x}_2; \boldsymbol{\theta}) = \|\mathcal{G}_{\boldsymbol{\theta}}\left(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_1)\right) - \mathcal{F}^{\alpha}_{\boldsymbol{\phi}}(\mathbf{x}_2)\|_2^2 \tag{17}$$

where

$$\mathcal{F}^{\alpha}_{\boldsymbol{\phi}}(\mathbf{x}_2) = \mathcal{F}_{\boldsymbol{\phi}}(\mathbf{x}_2) + \alpha\mathcal{G}_{\boldsymbol{\theta}}(\mathbf{r}_{12}) = \mathcal{F}_{\boldsymbol{\phi}}(\mathbf{x}_2) + \alpha\mathcal{G}_{\boldsymbol{\theta}}(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_1) - \mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_2)). \tag{18}$$

As for the similarity constraint loss, we simply adopts the similarity loss (equation 1) of the reverse direction,

$$\mathcal{L}_{\mathrm{sim}}(\mathbf{x}_2, \mathbf{x}_1; \boldsymbol{\theta}) = \|\mathcal{G}_{\boldsymbol{\theta}}\left(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_2)\right) - \mathcal{F}^{\alpha}_{\boldsymbol{\phi}}(\mathbf{x}_1)\|_2^2, \tag{19}$$

and we again use the residual $\mathbf{r}_{12}$ for the SSL loss to predict the augmentation t_1 (as $\mathbf{r}_{12}$ directs towards $\mathbf{x}_1$)

$$\mathcal{L}_{\mathrm{SSL}}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{t}_1; \boldsymbol{\theta}) = \mathrm{CE}(\mathcal{H}^c_{\boldsymbol{\theta}}(\mathbf{r}_{12}), \mathbf{t}^c_1) + \|\mathcal{H}^r_{\boldsymbol{\theta}}(\mathbf{r}_{12}) - \mathbf{t}^r_1\|_2^2. \tag{20}$$

Combining the three losses together, we obtain our Prelax-std objective, which incorporates two views $\mathbf{x}_1, \mathbf{x}_2$ generated from the same standard augmentation list $\mathcal{T}$,

$$\mathcal{L}^{\alpha}_{\mathrm{R2S}}(\mathbf{x}_1, \mathbf{x}_2; \boldsymbol{\theta}) + \beta\mathcal{L}_{\mathrm{sim}}(\mathbf{x}_2, \mathbf{x}_1; \boldsymbol{\theta}) + \gamma\mathcal{L}_{\mathrm{SSL}}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{t}_1; \boldsymbol{\theta}). \tag{21}$$

5.3 Prelax-rot: Incorporating Stronger Pretext

As mentioned previously, our relaxation method is supposed to not only help improve existing augmentation strategies, but also allow stronger augmentations that are harmful for pretext-invariant methods, such as Rotation [9] (see Table 2a). Hence, we propose Prelax-rot, a rotation-variant of Prelax that could benefit from the rotation instead.

Prelax-rot is also built on the existing similarity objective *w.r.t.* $(\mathbf{x}_1, \mathbf{x}_2)$ that are generated with the standard augmentation list $\mathcal{T}$ [2]. Besides, we additionally consider the rotation augmentation with four canonical rotation angles $\mathcal{R} = \{0°, 90°, 180°, 270°\}$. Without loss of generality, we draw a random rotation angle $a \sim \mathcal{R}$ and apply it to rotate $\mathbf{x}_1$ clockwise, leading to the third sample $\mathbf{x}_3 = a(\mathbf{x}_1)$.

Note that $\mathbf{x}_3$ now contains a stronger augmentation that largely modifies the semantics of $\mathbf{x}_1$. Thus, instead of using the exact alignment $\mathbf{z}_3 \rightarrow\leftarrow \mathbf{z}_2$, we propose to incorporate the rotation semantics with the following relaxed alignment:

$$\mathbf{z}_3 \rightarrow\leftarrow \mathbf{z}_2 + \mathbf{r}_{31} \tag{22}$$

where the residual

$$\mathbf{r}_{31} = \mathbf{z}_3 - \mathbf{z}_1 = \mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_3) - \mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_1) \tag{23}$$

encodes the rotation information of $\mathbf{x}_3$, because the only difference between $\mathbf{x}_3$ and $\mathbf{x}_1$ is caused by the rotation r.

Motivated by the analysis above, we propose the Rotation Residual Relaxation Similarity (R3S) loss as

$$\mathcal{L}^{\alpha}_{\mathrm{R3S}}(\mathbf{x}_{1:3}; \boldsymbol{\theta}) = \|\mathcal{G}_{\boldsymbol{\theta}}(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_3)) - \tilde{\mathcal{F}}^{\alpha}_{\boldsymbol{\phi}}(\mathbf{x}_2)\|_2^2, \tag{24}$$

where the rotation relaxed target is

$$\tilde{\mathcal{F}}^{\alpha}_{\boldsymbol{\phi}}(\mathbf{x}_2) = \mathbf{x}_2 + \alpha\mathcal{G}_{\boldsymbol{\theta}}(\mathbf{r}_{31}) = \mathbf{x}_2 + \alpha\mathcal{G}_{\boldsymbol{\theta}}(\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_3) - \mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_1)). \tag{25}$$

Besides, we can utilize the residual $\mathbf{r}_{31}$ to predict the rotation angle $\mathbf{a}$ with the following RotSSL loss for its rotation-awareness

$$\mathcal{L}^{\mathrm{rot}}_{\mathrm{SSL}}(\mathbf{x}_1, \mathbf{x}_3, \mathbf{a}; \boldsymbol{\theta}) = \mathrm{CE}(\mathcal{H}_{\boldsymbol{\theta}}(\mathbf{r}_{31}), \mathbf{a}). \tag{26}$$

In the meantime, we should also controlling the similarity between $\mathbf{x}_1$ and $\mathbf{x}_2$ with the same reverse similarity loss as in equation 19. Combining the three objectives, we obtain the overall objective of Prelax-rot:

$$\mathcal{L}^{\alpha}_{\mathrm{R3S}}(\mathbf{x}_{1:3}; \boldsymbol{\theta}) + \beta\mathcal{L}_{\mathrm{sim}}(\mathbf{x}_2, \mathbf{x}_1; \boldsymbol{\theta}) + \gamma\mathcal{L}^{\mathrm{rot}}_{\mathrm{SSL}}(\mathbf{x}_1, \mathbf{x}_3, \mathbf{a}; \boldsymbol{\theta}). \tag{27}$$

Table 1: (a) Linear evaluation results on CIFAR-10 with ResNet-18 backbone. Scores of previous works are taken from the original paper (or official code) unless specified. (b) Linear evaluation results on ImageNette with ResNet-18 backbone unless specified.

(a) CIFAR-10.

Method	Acc (%)
Supervised [12]	91.3
Supervised (re-implemented) †	93.0
SimCLR [2]*	91.1
BYOL [10] (re-implemented)♯	91.1
SimSiam [5]	91.8
SimSiam + Prelax-std (ours)	92.5
SimSiam + Prelax-rot (ours)	92.4
SimSiam + Prelax-all (ours)	**93.1**

(b) ImageNette.

Method	Acc (%)
Supervised	91.0
Supervised + TTA	92.2
BYOL [10]	91.9
BYOL [10] w/ ResNet-50	92.3
BYOL + Prelax-std (ours)	92.5
BYOL + Prelax-rot (ours)	92.4
BYOL + Prelax-all (ours)	**92.6**

5.4 Prelax-all

At last, we can further combine the two versions, Prelax-std (equation 21) and Prelax-rot (equation 28), named Prelax-all, with the following combined objectives:

$$\frac{1}{2}\left(\mathcal{L}_{\mathrm{R2S}}^{\alpha_1}(\mathbf{x}_1,\mathbf{x}_2;\boldsymbol{\theta})+\mathcal{L}_{\mathrm{R3S}}^{\alpha_2}(\mathbf{x}_{1:3};\boldsymbol{\theta})\right)+\beta\mathcal{L}_{\mathrm{sim}}(\mathbf{x}_2,\mathbf{x}_1;\boldsymbol{\theta})+\frac{\gamma}{2}\left(\mathcal{L}_{\mathrm{SSL}}(\mathbf{x}_1,\mathbf{x}_2,\mathbf{t}_1;\boldsymbol{\theta})+\mathcal{L}_{\mathrm{SSL}}^{\mathrm{rot}}(\mathbf{x}_1,\mathbf{x}_3,\mathbf{a};\boldsymbol{\theta})\right). \tag{28}$$

6 Experiment

In this section, we evaluate our proposed Prelax empirically on benchmark datasets of unsupervised representation learning. We first conduct a thorough study of Prelax on CIFAR-10 and Imagenette, and further validate its effectiveness on large scale datasets, *e.g.,* ImageNet. Due to the limit of computation resource, we mainly conduct comparative study on CIFAR-10, where remarkably, our method attains state-of-the-art performance that is superior its supervised counterpart for the first time.

6.1 CIFAR-10 and ImageNette

6.1.1 Setup

We conduct our experiments following standard practice of unsupervised learning methods [2, 11, 10, 5]. In particular, we notice that SimCLR [2] and SimSiam [5] report results on CIFAR-10, while the official code of BYOL [10] reports results on the ImageNette dataset[2], a subset of 10 easily classified classes from Imagenet. Therefore, we evaluate our SimSiam variant of Prelax on CIFAR-10, and evaluate the BYOL-variant on Imagette, as detailed below.

CIFAR-10 We implement our SimSiam + Prelax following the same hypeparameters as in the SimSiam paper [5]. Specifically, we use ResNet-18 [12] as the backbone network, followed by a 3-layer projection MLP, whose hidden and output are both 2048-d. The predictor is a 2-layer MLP with 512-d hidden layer and output d=2048. We use SGD for pre-training with learning rate 0.03, momentum 0.9, and cosine decay schedule [16] for 800 epochs, weight decay $5e^{-}4$, and batch size 512. The input image size is 32×32 and the data augmentation follows the practice of existing works [2, 5]. After pre-training, we evaluate the backbone network by fixing its weight and fine-tune a linear classifier on top of its representation with the labeled training data. The linear classifier training uses SGD with linear rate 30.0, momentum 0.9 and batch size 256 for 100 epochs. The learning rate decays by 0.1 at the 60-th and 80-th epochs. We choose $\alpha = 0.1, \beta = 1, \gamma = 0.1$ for Prelax-std, $\alpha = 1, \beta = 2, \gamma = 0.1$ for Prelax-rot, and $\alpha_1 = 0.1, \alpha_2 = 1, \beta = 1, \gamma = 0.1$ for Prelax-all.

[2] https://github.com/fastai/imagenette

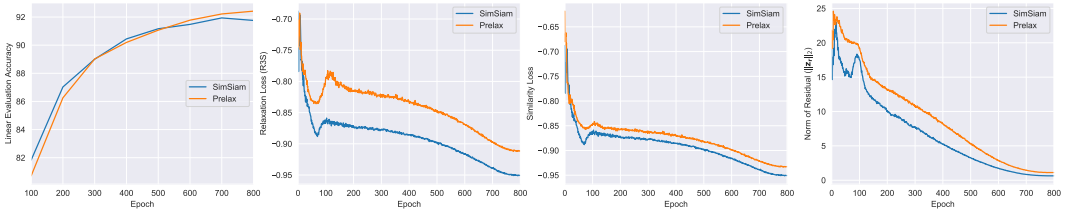

Figure 2: A comparison of learning dynamics between SimSiam [5] and Prelax (ours) on CIFAR-10. From left to right: 1) linear evaluation accuracy (%) on the test set *v.s.* epoch; 2) Residual relaxation loss (R3S) *v.s.* epoch; 3) Similarity loss *v.s.* epoch; 4) Norm of the residual vector (*i.e.,* $\|\mathbf{r}_{31}\|_2$) *v.s.* epoch.

ImageNette Our implementation of BYOL + Prelax on ImageNette mainly takes the same routine as CIFAR-10, and we also takes ResNet-18 as backbone. We utilize the default hyper-parameters of BYOL in the official code. We leave further details to the Appendix. Prelax configs???

Baselines. For a fair comparison with the baselines, we compare with the official result of ResNet-18 if available. Otherwise, we try to compare with the best re-implemented results that we can find till the submission deadline. On CIFAR-10, we include SimCLR [2], BYOL [10] and SimSiam [5] as baselines. For SimCLR, we report the results in the SimSiam paper. BYOL does not report results on CIFAR-10, and we instead adopt an unofficial implementation[3]. Moreover, we include the supervised results [12] as a baseline. As the score in the original paper is relatively low, we also include a higher score by a recent re-implementation[4]. On ImageNette, we report the supervised results with and without TTA (Test Time Augmentation)[5]. As for BYOL, we report the results in its official code[6].

Table 2: (a) Comparison of baseline methods. (b) Ablation study of Prelax. Experiments are conducted on CIFAR-10 with ResNet-18 backbone.

(a) Comparison of SSL methods.

Method	Acc (%)
SimSiam [5]	91.8
SimSiam + margin	91.9
SimSiam + Prelax-std (ours)	**92.5**
Rotation [9]	88.3
SimSiam w/ rotation	87.9
SimSiam + Rotation [9]	91.7
SimSiam + Prelax-rot (ours)	**92.4**
SimSiam + Prelax-all (ours)	**93.1**

(b) Ablation study

Method	Acc (%)
Sim (*i.e.,* SimSiam [5])	91.8
Prelax-std (R2S + Sim + SSL)	**92.5**
Prelax-std w/o SSL	91.5
Prelax-std w/o R2S	92.2
Prelax-rot (R2S + Sim + RotSSL)	**92.4**
Prelax-rot w/o RotSSL	91.9
Prelax-rot w/o R3S	91.1

6.1.2 Results

Comparison with Unsupervised Methods In Table 1a, we compare the top-1 test accuracy of Prelax with previous pretext-invariant methods on CIFAR-10. It can be seen that all three variants of Prelax are superior to previous methods [2, 10, 5] by a large margin (1.3 point at most). In particular, taking SimCLR as a baseline, the improvement of Prelax-all is 2.86 times larger than that of SimSiam. In Table 1b, the BYOL variant of Prelax also demonstrates state-of-the-art performance on ImageNette.

Comparison with Supervised Learning Remarkably, as far as we know, this is the first time that the unsupervised training of ResNet-18 outperforms the supervised training, as shown in Table 1a

[3] https://github.com/DonkeyShot21/essential-BYOL
[4] https://github.com/kuangliu/pytorch-cifar
[5] https://github.com/fastai/imagenette/issues/3
[6] https://github.com/deepmind/deepmind-research/tree/master/byol

& 1b. Therefore, we could claim that the gap between supervised and unsupervised learning has been closed on CIFAR-10 and ImageNette, which has been expected for a long time and serves an important step towards the wider application of unsupervised learning.

Methods for Bridging Pretext-aware and Pretext-invariant Representations In Table 2a, we compare different strategies for bridging Rotation [9] (pretext-aware) and SimSiam [5] (pretext-invariant). Simsiam w/ rotation refers to we add the random rotation augmentation in to the data augmentation pipeline of SimSiam, while SimSiam + Rotation refers to that we further add the Rotation loss [9] to SimSiam w/ rotation. From the table, we can see that as a strong rotation, adding rotation augmentation will degrade the performance of SimSiam, and simply adding the SimSiam and Rotation objectives will not help alleviate this issue. Instead, our Prelax-based methods can benefit from the rotation transformation and further improve over both baselines, making it a promising approach to bridge the pretext-aware and pretext-invariant representations.

6.1.3 Analysis

Ablation Study We perform ablation study of each componet of the Prelax objectives on CIFAR-10. From Table 2b, we can see that the residual SSL loss slightly improves the accuracy, while the R2S alone leads to worse performance because the residual is not pretext-aware. Combining all the three objectives we can further improve the performance by a large margin. Similar phenomenon also appears in the ablation study of Prelax-rot. Further combining the five objectives, Sim + R2S + R3S + SSL + RotSSL, we can attain the best accuracy with Prelax-all.

Learning Dynamics In Figure 2, we compare SimSiam with Prelax-rot in terms of the learning dynamics. We can see that with our residual relaxation technique, both the relaxation loss and the similarity loss become larger than SimSiam. In particular, the size of the residual indeed converges to a large value with Prelax (1.1) than with SimSiam (0.6). As for the downstream classification accuracy, we notice that Prelax-rot starts with a lower accuracy, but converges to a large accuracy at last. This indicates that Prelax-rot learns to encode more image semantics, which may be harder to learn at first, but will finally outperform the baseline with better representation ability.

7 Conclusion

In this paper, we have proposed a generic method, Prelax (Pretext-aware Residual Relaxation), to account for the semantic shift caused by image augmentations. With pretext-aware learning of the residual relaxation, our method successfully bridges the gap between pretext-aware and pretext-invariant representations without causing contradiction. Experiments show that we have achieved state-of-the-art performance on CIFAR-10 and ImageNette by combining the two complementary yet useful semantics.

References

[1] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020. 2, 3

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *ICML*, 2020. 1, 2, 3, 4, 6, 7, 8, 9

[3] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020. 1, 3

[4] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 1

[5] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*, 2020. 2, 3, 4, 5, 6, 8, 9, 10

[6] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. *ICCV*, 2015. 1, 3, 6

[7] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. *NeurIPS*, 2014. 3

[8] Zeyu Feng, Chang Xu, and Dacheng Tao. Self-supervised representation learning from multi-domain data. *ICCV*, 2019. 2, 3

[9] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *ICLR*, 2018. 1, 2, 3, 6, 7, 9, 10

[10] Jean-Bastien Grill, Florian Strub, Florent Altché, C. Tallec, Pierre H. Richemond, Elena Buchatskaya, C. Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, B. Piot, K. Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. *NeurIPS*, 2020. 1, 2, 3, 4, 5, 6, 8, 9

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

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CVPR*, 2016. 8, 9

[13] 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. *ICLR*, 2019. 1, 3, 4

[14] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. 2019. 1, 3, 6

[15] Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. Contrastive representation learning: A framework and review. *IEEE Access*, 2020. 1

[16] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *ICLR*, 2017. 8

[17] Sean Metzger, Aravind Srinivas, Trevor Darrell, and Kurt Keutzer. Evaluating self-supervised pretraining without using labels. *arXiv preprint arXiv:2009.07724*, 2020. 1

[18] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. *CVPR*, 2020. 1, 3

[19] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. *ECCV*, 2016. 1, 3, 6

[20] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 1, 3, 4

[21] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019. 1, 4

[22] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. *NeurIPS*, 2020. 1

[23] Tete Xiao, Xiaolong Wang, Alexei A Efros, and Trevor Darrell. What should not be contrastive in contrastive learning. *ICLR*, 2021. 3

[24] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. *ECCV*, 2016. 1, 3

A Appendix

Proof of Lemma 1. For the data generation process of $\mathbf{X}_1, \mathbf{X}_2$, we have the following Markov chain:

$$\mathbf{Z}_1 \leftarrow \mathbf{X}_1 \leftarrow \mathbf{U} \rightarrow \mathbf{X}_2 \rightarrow \mathbf{Z}_2. \tag{29}$$

According to data processing inequality (DIP), the above inequalities hold. Moreover, because $I(\mathbf{Z}_1; \mathbf{Z}_2|\mathbf{U}) = 0$, we also have

$$I(\mathbf{Z}_1; \mathbf{Z}_2) = I(\mathbf{Z}_1; \mathbf{Z}_2|\mathbf{U}) + I(\mathbf{Z}_1; \mathbf{Z}_2; \mathbf{U}) = I(\mathbf{Z}_1; \mathbf{Z}_2; \mathbf{U}) = I(\mathbf{Z}_1; \mathbf{U}) - I(\mathbf{Z}_1; \mathbf{U}|\mathbf{Z}_2). \tag{30}$$

Thus, if $I(\mathbf{Z}_1; \mathbf{U}|\mathbf{Z}_2) = 0$, we have $I(\mathbf{Z}_1; \mathbf{Z}_2) = I(\mathbf{Z}_1; \mathbf{U}) = I(\mathbf{Z}; \mathbf{U})$. □

Proof of Lemma 2. The lemma holds according to the following equality:

$$\begin{aligned} -H(\mathbf{S}|\mathbf{Z}) &= I(\mathbf{Z}; \mathbf{S}) - H(\mathbf{S}) = \mathbb{E}_{P_d(\mathbf{S})P_d(\mathbf{X}|\mathbf{S})} \log P(\mathbf{S}|\mathbf{Z}) \\ &= \mathbb{E}_{P_d(\mathbf{S})P_d(\mathbf{X}|\mathbf{S})} \log Q_{\boldsymbol{\pi}}(\mathbf{S}|\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{X})) + \mathrm{KL}(P(\mathbf{S}|\mathbf{Z}) \| Q_{\boldsymbol{\pi}}(\mathbf{S}|\mathbf{Z})). \end{aligned} \tag{31}$$

□