Hallucination Improves the Performance of Unsupervised Visual Representation Learning

Jing Wu[1] Jennifer Hobbs[2] Naira Hovakimyan[1]
[1]University of Illinois Urbana-Champaign, [2]Intelinair
{jingwu6, nhovakim}@illinois, jenniferhobbs08+research@gmail.com

Abstract

Contrastive learning models based on Siamese structure have demonstrated remarkable performance in self-supervised learning. Such a success of contrastive learning relies on two conditions, a sufficient number of positive pairs and adequate variations between them. If the conditions are not met, these frameworks will lack semantic contrast and be fragile on overfitting. To address these two issues, we propose Hallucinator that could efficiently generate additional positive samples for further contrast. The Hallucinator is differentiable and creates new data in the feature space. Thus, it is optimized directly with the pre-training task and introduces nearly negligible computation. Moreover, we reduce the mutual information of hallucinated pairs and smooth them through non-linear operations. This process helps avoid over-confident contrastive learning models during the training and achieves more transformation-invariant feature embeddings. Remarkably, we empirically prove that the proposed Hallucinator generalizes well to various contrastive learning models, including MoCoV1&V2, SimCLR and SimSiam. Under the linear classification protocol, a stable accuracy gain is achieved, ranging from 0.3% *to* 3.0% *on CIFAR10&100, Tiny ImageNet, STL-10 and ImageNet. The improvement is also observed in transferring pre-train encoders to the downstream tasks, including object detection and segmentation.*

1. Introduction

In the recent computer vision community, there has been rapid progress in self-supervised learning (SSL), gradually closing the performance gap with supervised learning [23, 29, 10, 7, 51]. Among the diverse approaches of SSL, contrastive learning, such as MoCoV1&V2 [29, 11], SimCLR [10], and SimSiam [12], shows promising results. Generally, contrastive learning treats each image as one class which will be augmented into two separate views. These two views form one positive pair and should ideally be close if mapped to feature space. With sufficient contrast in the feature space, contrastive learning models show a strong capacity to learn transformation-invariant features that are transferable to various downstream tasks, such as classification, object detection, and segmentation [18, 30, 39].

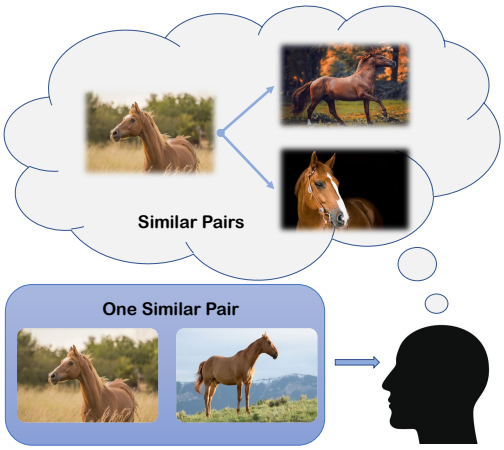

Figure 1: The motivation of the proposed hallucination methods. Given one pair of images with the same semantic meaning, such as a pair of horses, a person can envision further similar pairs by imagining one of the horses in different poses and surroundings. If a contrastive learning model could do such hallucination, it could have additional novel pairs to contrast given the same data. Note that this hallucination process is for illustration only. In the implementation, all the hallucinated samples are computed in the feature space.

To ensure sufficient contrast, researchers from previous work address the issue from two essential practices, either introducing large amounts of positive pairs or adding additional variants&transformation among them. For example, SimCLR uses a batch size that generates thousands of positives to facilitate the convergence of models [10]. Work

from [51, 9] reduces mutual information of positive pairs using stronger data augmentation, i.e., color distortion and jigsaw transformation. Likewise, the work from [44] introduces ContrastiveCrop, and the work from [49] proposes Un-Mix, respectively, to reduce the similar semantic meaning of sample pairs in the original image space. Beyond the data augmentation and image operations, researchers from [68] propose to apply a linear operation to generate hard positive samples in feature space.

Despite the success of prior approaches, we argue that large batch sizes are not always achievable. Meanwhile, all proposed techniques only focus on improving the original pairs. Given one positive pair of positive samples, humans are born with the amazing ability to come up with additional positives by imagining a sample from different surroundings and perspectives without much effort, as demonstrated in Figure 1. This process of self-imagination, in turn, will benefit the human neurological system, improving recognition capacity [24]. Similarly, if we could empower contrastive learning models with the ability to hallucinate or imagine an object to a novel view, additional positive pairs could be provided for the learning tasks.

Unfortunately, exploring feasible methods to hallucinate novel positive pairs is challenging. Firstly, while generative models produce realistic images that could form additional positive views [22, 1, 42, 4], realistic data do not necessarily benefit learning tasks [55]. More importantly, applying these approaches forces us to fall back into a computational dilemma to the previous method. In other words, image-level hallucination still suffers from expensive computation as we still need to encode the hallucinated images into feature space. Lastly, if the generated positive pairs are similar to each other, training a discriminative model would be too trivial, thus showing poor generalization capacity [44, 51, 68].

Therefore, our key insight is that the sample-generation process should aim for three critical elements: (i) feature-space operation (ii) sufficient variance of positive pairs (iii) a differentiable module optimized directly related to the learning task. To achieve this, we propose *Hallucinator* to improve the performance of contrastive learning with Siamese structures. The *Hallucinator* is plugged in after the encoder to manipulate feature vectors and improve the feature-level batch size for further contrast. To ensure adequate variance is introduced, we propose an asymmetric feature extrapolation method inspired by the work from [68]. More importantly, we present a non-linear hallucination process for the extrapolated samples. Such a process is differentiable (i.e. learnable), therefore essentially boosting *Hallucinator* to generate smooth and task-related features.

The proposed *Hallucinator* delivers extra positives and simultaneously enlarges the variance between newly introduced pairs. Moreover, this approach only relies on positive samples. Therefore, it can be easily applied to any Siamese structure by adding it after the encoders as a plug-and-play module. Without the tedious exploration of hyper-parameters and much additional computation, we empirically prove the effectiveness of the proposed *Hallucinator* on popular contrastive learning models, including MoCoV1&V2, SimCLR and SimSiam. We notice a stable improvement ranging from 0.3% to 3.0% under the linear classification protocol, crossing the CIFAR10&100, Tiny ImageNet, STL-10 and ImageNet. We also observe that models trained with *Hallucinator* show better transferability in downstream tasks like object detection and segmentation.

Our contributions can be summarized as follows:

- We investigate a critical yet under-explored aspect of contrastive learning: introducing additional positive pairs with further variation in feature space.
- To the best of our knowledge, this is the first attempt to incorporate the concept of "Hallucination" into contrastive representation learning. We propose *Hallucinator* to realize this idea, which effectively generates smooth and less similar positive feature vectors. The *Hallucinator* is simple, effective, and agnostic to contrastive frameworks.
- We empirically illustrate that the proposed approach significantly benefits the various contrastive learning models within multiple datasets.

2. Related Works

In this section, we introduce related literature on contrastive learning and hallucination techniques.

2.1. Contrastive Learning

The key idea of contrastive learning is to minimize the distance of positive pairs and repulse negative pairs in the feature space [27]. This idea has been successfully applied to unsupervised visual representation tasks, showing promising results in various downstream tasks [10, 29, 11, 23, 12, 2, 32, 17, 54, 41, 43, 50, 57, 60, 63, 61, 8, 37]. One of the breakthroughs of contrastive learning models is SimCLR [10], which introduces a simple but effective visual representation learning method. Without negative pairs, SimCLR learned transformation-invariant representation with a large batch size. Contemporaneous impressive work is MoCo from [29]. To ensure MoCo can be trained smoothly with computational-friendly batch sizes, the authors of MoCo propose a memory bank to store negative features and momentum-updated backbones. After this, SimSiam [12] presents a Siamese-based network that can learn high-quality representation with stop-gradient, successfully avoiding the collapse of contrastive learning models. Other designs of contrastive learning rely on an online network to predict the output of the target network or contrasting cluster assignments [23, 7].

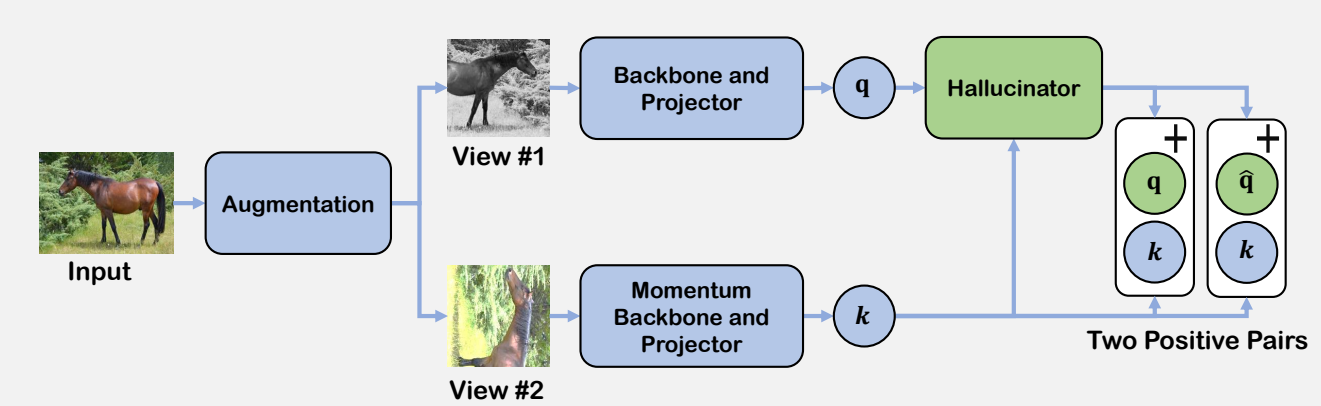

Figure 2: Illustration of contrastive learning (MoCoV2 [11]) with *Hallucinator*. The *Hallucinator* is added after the backbones and projector for feature-level manipulation. While *Hallucinator* feeds original feature vector q forward, it additionally provides hallucinated feature $\hat{q}$ for further contrast.

To further boost the models' performance, researchers explore diverse ways to hard positives and reduce the mutual information of positive samples so that more transformation-invariant embeddings can be learned [51, 48, 45, 13]. For image-level enhancement, typical works focus on improving the data augmentation in pixel space using Mixup, color distortion, or jigsaw transformation [49, 51, 9]. More recent work is presented by [44] using ContrastiveCrop, which creates better positive views through the localization box. For feature-level operation, the authors from [34] prove the effectiveness of hard negative samples. More recent and close work to this paper is [68], which applies symmetric extrapolation to create hard positives but introduces no non-linearity. Importantly, instead of replacing the original positive samples, this paper proposes a differentiable hallucinator to generate an additional positive sample with less mutual information and better smoothness. Such a setting ensures maximal and adaptive contrast during the training, generalizing well to diverse contrastive learning models.

2.2. Hallucination

Hallucination is initially proposed to solve the scarcity of data in the classification task [28]. Then, this idea is kept updated and applied in different areas [46, 55, 66, 67, 6, 65, 25, 47], such as object detection, aerial navigation, skeleton-based action recognition, and face generation. While image-level hallucination benefits few-shot recognition by synthesis of novel view [3] or introducing random noises [55], most of the work applies hallucination in the feature space. The work from [28] generates novel class features by transforming shared features in base classes. Authors of [67] build a hallucination framework in the region of interest feature space object to enhance object detection performance. More recent work shows that this hallucination mechanism also benefits 3D human pose estimation by generating novel motion sequences [21]. While hallucination is effective in different learning tasks, to the best of our knowledge, the performance and application of hallucination in SSL are fully unexplored.

2.3. Feature-Level Augmentation

Hallucination relies on effective feature-level augmentation or manipulations. The primary goal of feature augmentation is to extend the limited labeled dataset without relying on expensive computation such as auto-encoder[33], Generative Adversarial Networks[40] or simulation tools[52, 58, 5]. For instance, in the work by [16], a task-agnostic feature augmentation approach is proposed to enrich training data with minimal additional computation. The authors of [36] also explore similar ideas in the domain of few-shot learning. Building upon this, the concept is adapted to sentence representation learning [62, 19] and few-shot learning tasks in remote sensing[56]. More recently, [38] applies feature augmentation based on a meta-learning technique.

3. Method

In this section, we first introduce the overall process of hallucination for contrastive representation learning in Section 3.1. Secondly, we highlight the center cropping method we used, which is crucial to effective hallucination or generation of new samples in Section 3.2. Then, we introduce the *Hallucinator* incorporated into our contrastive models in Section 3.3. Finally, we visualize and discuss the critical properties of the hallucination method from two perspectives in Section 3.4: the similarity of positive samples and the uniformity of the feature distribution.

3.1. The Overall Pipeline

Taking MoCo [11] as an example, we illustrate how a *Hallucinator* can be plugged into a contrastive learning model in Figure 2. Our architecture takes one image x as input. Then, the input x is augmented into two views x_1 and x_2. Each view will be processed by an encoder consisting of a backbone (e.g., ResNet) and a projector (e.g., an MLP head). After the encoders, output vectors q and k are obtained, forming one positive pair (q, k). Then, this positive pair (q, k) is fed to a *Hallucinator*. Notably, *Hallucinator* is only added to one branch of the framework to generate an additional positive feature $\hat{q}$. Together with feature vector k, $\hat{q}$ and k form as an extra positive pair $(\hat{q}, k)$ during the training. Based on different contrastive learning models, the loss functions keep intact, and the average loss of these two positive pairs is computed for back-propagation. The same paradigm could be applied to SimCLR, SiamSiam and other contrastive learning models. We illustrate further details about pipelines and loss functions of other models in this paper in the Supplementary Material (Section 1.1).

3.2. Center Cropping

In contrastive learning, data augmentations aim to ensure the performance of pre-trained representations invariant to nuisances. Among all these methods, random crop plays the most critical role in all the contrastive learning models. Generally, views (cropped tiles) generated by random cropping are diversified, successfully covering all the semantic information over the whole image. However, such a cropping method is likely to generate false positives patches [44]. In other words, patches randomly cropped from the original images do not necessarily share the overlapped pixels and sufficient common information. Therefore, these false positive pairs may be fooling models during training, causing representations to be sub-optimal. Importantly, the issue will be exacerbated if we generate further hallucinated samples based on false positive pairs, which misleads the overall training beyond the sweet spot.

To tackle this issue, we first apply center cropping $\mathbb{C}_{crop}$ to the original image, getting a relatively smaller image $\hat{I}_{x,y}$. Then, random cropping $\mathbb{R}_{crop}$ is applied to $\hat{I}_{x,y}$. More specifically, the center cropping can be formulated as

$$\hat{I}_{x,y} = \mathbb{C}_{crop}(I_{x,y}, p), \tag{1}$$

where $I_{x,y}$ is the input image with (x, y) as the coordinate of the images' center. After center cropping, we keep the center of $\hat{I}_{x,y}$ unchanged. Meanwhile, with the original shape of $I_{x,y}$ defined as (h, w), we define the shape of cropped image $\hat{I}_{x,y}$ as $(\hat{h}, \hat{w})$. The p denotes a ratio of cropped length over the original length, i.e., $p = \frac{\hat{w}}{w} = \frac{\hat{h}}{h}$. Only if particularly mentioned, we set $p = 0.5$ for all experiments in this paper.

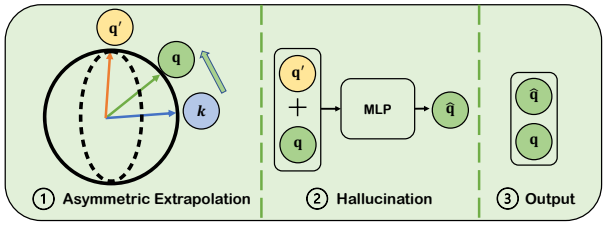

Figure 3: The $Hallucinator$. Stage 1: The original feature vector q is extrapolated to the opposite direction of feature vector k, forming $\hat{q}$ in a linear way. Stage 2: We introduce non-linear transformation to smooth extrapolated features concatenated with q and q'. Stage 3: output original q and hallucinated $\hat{q}$.

While center cropping effectively avoids false positive pairs, it reduces the operable region for random cropping and generates positive views with a similar appearance. We, therefore, adopt center-suppressed sampling [44] with a sampling method following a beta distribution $\beta(\alpha, \alpha)$ (i.e., a U-shaped distribution). Concretely, $\beta(\alpha, \alpha)$ assigns a lower probability to the center of the $\hat{I}_{x,y}$ and gives greater probability to its boundary, increasing the variance between views x_1 and x_2. Together, we summarize the process to obtain these two views as

$$\begin{aligned} x_1 &= T(\mathbb{R}_{crop}(\hat{I}_{x,y}|\alpha)), \quad \text{s.t.} \quad \alpha < 1 \\ x_2 &= T(\mathbb{R}_{crop}(I_{x,y}|\alpha)), \quad \text{s.t.} \quad \alpha < 1, \end{aligned} \tag{2}$$

where T denotes data augmentations, including color jittering, random grayscale, Gaussian blur and horizontal flipping. $\mathbb{R}_{crop}(...|\alpha)$ represents random cropping following the $\beta(\alpha, \alpha)$ distribution. α is set to less than 1 to ensure an increasing sampling probability as the pixel's coordinates go beyond the center. A visualization of the center sampling method can be found in Supplementary Material (Section 1.2).

3.3. Hallucinator

Asymmetric Feature Extrapolation. The first objective of this module is to introduce an additional positive pair without introducing extra computations. Therefore, the feature-level operation is preferred compared to image-level operations. To achieve this, *Hallucinator* is plugged in after the views x_1 and x_2 are encoded. As a result, the hallucinated (generated) feature is purely based on the two feature vectors q and k.

Meanwhile, since harder positives improve pre-trained encoders' generalization capacity [51], the hallucinated features in positive pairs are favorable if they share less mutual information. Previous work illustrates that a symmetric

positive extrapolation is effective in generating hard examples for MoCo [68]. Concretely, two positive features are combined with weighted addition, pushing positive features apart. However, the hallucination process is asymmetric, i.e., *Hallucinator* is only added in one of the branches of the model. Then, we propose to apply the singe-side feature extrapolation to the feature vector q, as shown in the first stage of Figure 3. Additionally, we simplify the sampling strategy of weights for extrapolation from a beta distribution to a uniform distribution. To be specific, we summarize the positive extrapolation in our method as follows:

$$q' = (1+\lambda)q - \lambda k \quad \text{s.t.} \quad \lambda \sim U(\beta_1, \beta_2), \tag{3}$$

where λ is sampled from a uniform distribution $U(\beta_1, \beta_2)$. β_1 and β_2, which are the boundary of the uniform distribution, are set to 0 and 0.1 by default.

Hallucination. Positive extrapolation is based on mixup [64], i.e., a linear transformation. While positive extrapolation has been proven beneficial to generating hard examples, this linear feature transformation might have a relatively limited capacity to synthesize new feature vectors. This assumption is based on the more satisfactory performance of the non-linear mixup over the original one [26]. Similarly, if non-linearity is introduced to the feature generation, the generated vector will benefit more from the training and boost the performance of downstream tasks.

To empower our model with the capacity of non-linear fitting, we introduce the hallucination process. Specifically, we first concatenate q' from the equation 3 and the feature vector q together. Then, we use the concatenated feature (q, q') as an input of a non-linear transformation function $H_\theta(.)$ that can be instantiated with n linear layers and a ReLU layer between two successive layers. When $n = 0$, $H_\theta(.)$ is a non-parametric module, forming an identity function. Such a setting performance is relatively sub-optimal, as shown in the ablation study in Table 6. Empirically, we find that $n = \{2, 3\}$ performs well as the *hallucinator* becomes non-linear and more powerful. We set $n = 3$ by default as its results are slightly better. With the transformation function $H_\theta(.)$, the hallucinated feature is defined as

$$\hat{q} = H_\theta(q, q'), \tag{4}$$

where θ is the parameters of $H_\theta(.)$. Notably, $H_\theta(.)$ is differentiable, allowing us to back-propagate the loss of contrastive learning. Therefore, we update not just the parameters of the encoders and projectors but also the parameters θ of the hallucinator. The second stage of Figure 3 illustrates the proposed hallucination process. Following that, we take q and $\hat{q}$ as the output of *Hallucinator*.

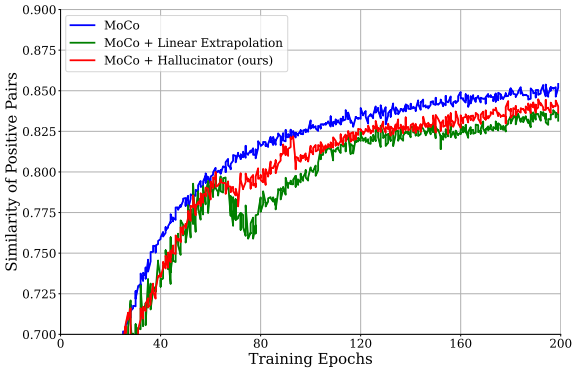

Figure 4: Similarity of positive pairs in training. Smaller values indicate less mutual information and better representation [51, 44, 68]. As *Hallucinator* incorporates non-linearity extrapolation, it guarantees smooth training and harder positive features.

3.4. Discussion and Visualization

To better understand the behavior of *Hallucinator*, we discuss two critical properties that may contribute to and explain its effectiveness. For visualization, we train MoCoV2 [11] with a standard ResNet-18 [31] on Cifar-10 [35].

Similarity of Positive Pair. We first investigate the similarity of positive pairs during the training process. Concretely, we quantize mutual information using the cosine similarity of positive pairs S:

$$S \triangleq \frac{q \cdot k}{\|q\| \, \|k\|}. \tag{5}$$

While hard positives share less mutual information, thus having a smaller cosine similarity value, the performance of downstream tasks will be enhanced [68]. Based on Figure 4, the proposed *Hallucinator* generates harder positives with smaller values of similarity. Consequently, it helps contrastive learning models obtain more nuisances-invariant features. More importantly, different from linear symmetric extrapolation, our training curve is relatively stable without many oscillations introduced. This observation indicates that *Hallucinator* is successfully optimized with the overall framework. Meanwhile, hallucinated features nicely fit into the contrastive learning task.

Uniformity. We continue to analyze the performance of the proposed model from the perspective of uniformity. Notably, feature vectors from contrastive learning should be roughly uniformly distributed on a unit hyper-sphere, which ensures maximal information is preserved in the feature space [53]. In other words, the closer the feature distribution is to the uniform distribution, the more the feature benefits downstream tasks. To quantize uniformity, we follow previous work [53] to compute the average value of the

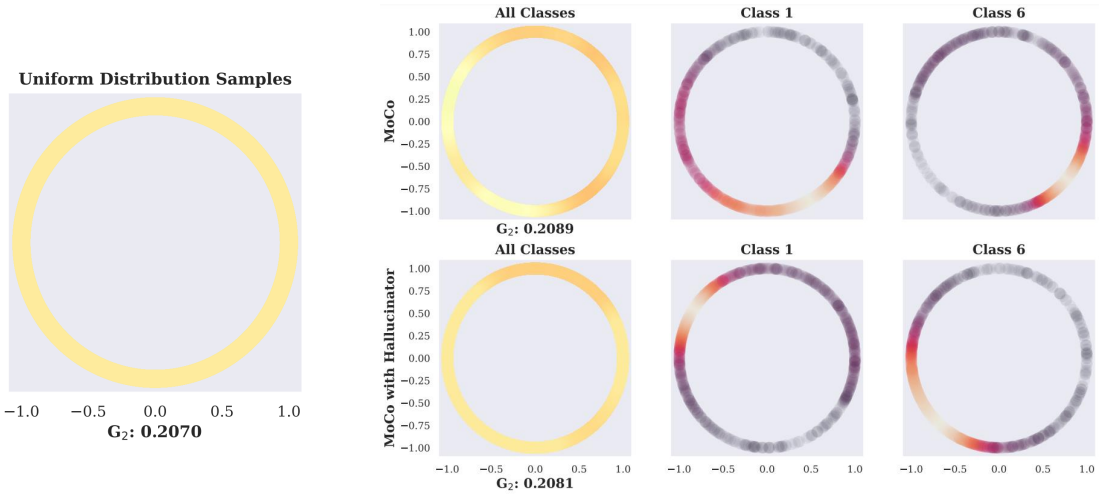

Figure 5: A measure of uniformity based on G_2 potential. We plot 10,000 feature vectors with Gaussian kernel density estimation (KDE) in $\mathbb{R}^2$. The left subplot illustrates the feature vectors from a uniform distribution. The three feature distributions on the right in the first row visualize the features from MoCoV2 [11]. The other three feature distributions in the second row demonstrate MoCoV2 with *Hallucinator*. *Hallucinator* benefits the uniformity with a smaller value of G_2.

Dataset	**CIFAR-10**		**CIFAR-100**		**Tiny ImageNet**		**STL-10**	
Hallucinator	✗	✓	✗	✓	✗	✓	✗	✓
MoCoV1	88.31	**88.94**	60.94	**61.81**	44.65	**45.53**	88.19	**90.09**
MoCoV2	87.21	**89.23**	59.70	**61.26**	47.12	**47.95**	89.32	**90.46**
SimCLR	89.66	**90.11**	60.94	**61.43**	45.22	**46.30**	89.07	**89.98**
SimSiam	90.47	**90.78**	63.39	**64.38**	43.66	**44.96**	87.79	**88.16**

Table 1: Linear classification results for different contrastive methods and datasets in small scales. We adopt ResNet-18 as the backbone and report the classification results with or without *Hallucinator*.

Gaussian potential kernel (i.e., Radial Basis Function kernel) of positive features:

$$G_t(q,k) \triangleq e^{-t\|q-k\|_2^2} \quad \text{s.t.} \quad t>0, \tag{6}$$

where t is a fixed parameter set as 2 for all the experiments. We visualize the feature vectors by mapping them to two-dimension feature space and applying l_2 normalization in Figure 5. $G_2 = 0.2070$ for uniformly distributed samples, whereas features from MoCoV2 have $G_2 = 0.2089$. If we plugin *Hallucinator* into MoCo, *Hallucinator* provides further contrast during the training with extra positives introduced, giving more uniformly distributed features and a decreased G_2 value,i.e., 0.2081. Additionally, we visualize the features of two classes of Cifar-10. Each of these features is well-clustered. With better uniformity, clusters' overlapping decreases, forming more linearly separable features. Therefore, we could observe that the overlapping of feature clusters between class 1 and class 6 in MoCo is larger than the one with *Hallucinator* plugged in.

4. Experiments and Results

In this section, we conduct various experiments on different datasets and contrastive learning models to demonstrate the effectiveness of the *Hallucinator*. Firstly, we introduce the datasets and details in experiments in Section 4.1. Then, we continue to evaluate the performance of the proposed method using linear probing protocol following the paradigm of previous work [10, 11, 12, 23] in Section 4.2. Following this, we conduct ablation studies to understand how each module contributes to the final results in Section 4.3. Lastly, we show the transferability of pre-trained encoders in downstream tasks requiring dense pixel predictions, including object detection and semantic segmentation in Section 4.4.

4.1. Datasets and Training Details

Datasets and Baseline Models. We first evaluate the performance of *Hallucinator* based on a wide range of datasets crossing different scales. Specifically, these datasets in-

Method	Backbone	Epoch	IN-200	IN-1K
MoCoV1	ResNet-50	100	62.19	57.27
MoCoV1(Ours)	ResNet-50	100	**63.46**	**59.17**
MoCoV2	ResNet-50	100	62.57	64.41
MoCoV2(Ours)	ResNet-50	100	**63.58**	**64.97**
SimCLR	ResNet-50	100	62.22	61.23
SimCLR(Ours)	ResNet-50	100	**63.02**	**61.71**
SimSiam	ResNet-50	100	62.80	63.11
SimSiam(Ours)	ResNet-50	100	**63.52**	**63.55**

Table 2: Linear classification results on IN-200 and IN-1K. Our method plugged in the proposed *Hallucinator* in baseline methods. All the models are pre-trained for 100 epochs and use identical training settings for fair comparisons.

clude CIFAR-10, CIFAR-R100 [35], Tiny ImageNet, STL-10 [14] and ImageNet [15]. Meanwhile, we demonstrate the performance of *Hallucinator* in several popular contrastive learning frameworks, including MoCoV1 [29], MoCoV2 [11], SimCLR [10] and SimSiam [12].

Training and Evaluation Details. Importantly, we strictly adopt the same training settings when evaluating the *Hallucinator*'s performance. While additional gain could be achieved with further exploration of hyper-parameter, it is not the focus of this work. The ultimate goal of the proposed *Hallucinator* is to provide further feature-level contrast for self-supervised learning. With the *Hallucinator* plugin, it benefits diversified self-supervised learning methods regardless of their type of backbones and hyper-parameters of training.

Datasets on relatively small scales include CIFAR-10&100, Tiny ImageNet and STL10. To ensure fair comparisons, we keep training settings the same over all these datasets. We follow the paradigm of previous work for pre-training [44]. Concretely, we pre-train contrastive learning models for 500 epochs with a batch size of 512 and ResNet-18 as the backbone. To optimize the models, we use an SGD optimizer and a cosine-annealed learning rate of 0.5. In the linear classification task, we train models for 100 epochs. With the initial learning set as 10, we divide it by 10 at the 60th and 80th epochs.

As the scale of the dataset expands to ImageNet, we use ResNet-50 as the backbone. For MoCoV1, MoCoV2 and SimSiam, all the pre-training settings follow the original work. We use a batch size of 512 and an SGD optimizer with a cosine-annealed learning rate of 0.5 for SimCLR, the same as the settings in [44]. For the linear classification task, we follow the original work in [29], training the model for 100 epochs. We set the initial learning rate to 30. Then, we divide the learning by 10 at the 60th and 80th with a weight decay of 0.

We set $p = 0.5$ and $\alpha = 0.6$ for center cropping and center-suppressed sampling, respectively. For asymmetric linear extrapolation, we set $\beta_1 = 0$ and $\beta_2 = 0.1$. In the Hallucination process, $n = 3$ for all the results reported in this paper. All the experiments are conducted on a server with 8 GPUs.

Acc.(%)	Center Cropping	Asymmetric Extrapolation	Hallucination
62.57	✗	✗	✗
63.58(+1.01)	✓	✓	✓
62.58(+0.01)	✓		
62.82(+0.25)		✓	
62.84(+0.27)			✓
63.07(+0.50)		✓	✓

Table 3: Ablation of the different modules in the proposed method.

p (%)	90	80	70	60	50	40
Acc.(%)	63.11	63.19	63.33	63.56	63.58	63.21

Table 4: Ablation of classification results w.r.t the p value.

(β_1, β_2)	(-0.5, 0.0)	(0.0, 0.5)	(0.5, 1.0)	(1.0, 1.5)	(0.0, 1.0)
Acc.(%)	62.80	63.32	63.55	63.42	63.58

Table 5: Ablation of classification results w.r.t the extrapolation range.

layer #: n	0	1	2	3	4
Acc.(%)	62.85	63.03	63.57	63.58	63.55

Table 6: Ablation of classification results w.r.t the number of linear layers n.

4.2. Linear Classification Protocol

Following the previous protocol, we first evaluate the proposed method by linear classification of frozen features. For each dataset, we report the top-1 classification accuracy on the validation set.

Results on Small-Scale Datasets. The classification results on Cifar10&100, Tiny ImageNet and STL-10 are reported in Table 1. With *Hallucinator* introduced, we notice a stable improvement over the baselines ranging from 0.31% to 1.98%. Notably, such improvements do not introduce extra computations and generalize well to various models.

Results on ImageNet. For the results of ImageNet, we report the results at two different scales. First, we evaluate its performance on standard IN-1K (i.e. ImageNet-1K), which consists of 1000 classes. Second, we test the proposed method in IN-200 (i.e. ImageNet-200) with 200 randomly selected classes. We report the corresponding results in Table 2. We found that *Hallucinator* essentially benefits

Method	1N-1k	VOC detection			COCO detection			COCO instance segmentation		
	Top-1	AP	AP_{50}	AP_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
Random init	-	33.8	60.2	57.27	26.4	44.0	27.8	29.3	46.9	30.8
Supervised	76.1	53.5	81.3	58.8	38.2	58.2	41.2	33.3	54.7	35.2
InfoMin [51]	70.1	57.6	82.7	64.6	39.0	58.5	42.0	34.1	55.2	36.3
MoCoV1 [29]	60.6	55.9	81.5	62.6	38.5	58.3	41.6	33.6	54.8	35.6
MoCoV1(Ours)	**63.8**	**56.7**	**81.8**	**63.2**	**38.9**	**58.5**	**41.9**	**33.8**	**55.2**	**36.0**
MoCoV2 [11]	67.5	57.0	82.4	63.6	39.0	58.6	41.9	34.2	55.4	36.2
MoCoV2(Ours)	**68.0**	**57.4**	**82.7**	**63.9**	**39.3**	**58.8**	**42.3**	**34.6**	**55.5**	**36.4**

Table 7: Fine-tuning results on object detection tasks on PASCAL VOC and COCO, and instance segmentation on COCO. All models are pre-trained for 200 epochs on ImageNet-1K.

MoCoV1 with 1.27% and 1.89% improvements for IN-200 and IN-1K accordingly. For MoCoV2, SimSiam and SimCLR, we notice a gain in accuracy ranging from 0.44% to 1.01%. On average, the gains are more salient in IN-200.

4.3. Ablation Studies

In this section of ablation studies, we investigate essential modules in our proposed methods, including center cropping, asymmetric extrapolation and the process of hallucination. We conduct experiments using MoCoV2 with ResNet-50 as the backbone. We report the results of the classification task in IN-200 for all the following experiments.

Contributions of Modules. We first report the results with or without crucial modules introduced in Section 3. According to Table 3, center cropping shows a similar performance to the original cropping method, successfully covering major semantic information of images. However, it successfully avoids false positives in pre-training, which is critical for hallucination. Asymmetric extrapolation benefits the performance of representation learning with reduced mutual information. This observation is consistent with the results shown in the symmetric extrapolation [68]. If we combine asymmetric extrapolation and the hallucination method, the performance of the model could be further boosted. However, it is still sub-optimal because of possible false positives in cropping.

Center Cropping. In this work, we apply center cropping as a critical tool in one branch of contrastive learning to avoid false positives. Therefore, we investigate how the proposed model is influenced by the length ratio p in Table 4. The classification results are favorable when p is in the range (0.5, 0.6). We notice a performance drop when p is less than 0.5. This drop is because the center cropping failed to cover all semantic information in the images.

Extrapolation Range. We continue to report the extrapolation range given its differences in sampling method and single-side architecture compared with the previous method [68]. Based on Table 5, the classification accuracy drops if we set $\beta_1 = -0.5$ and $\beta_2 = 0$, respectively. Such observation is because hallucinated positives share more mutual information, thus less beneficial to training. Generally, extrapolation boosts accuracy by outperforming the baseline by at least 0.75%. However, the improvement decreases when $\beta_1 > 1.0$. We eventually set $\beta_1 = 0.0$ and $\beta_2 = 1.0$, given its best performance.

Hallucination. We investigate how the number of linear layers influences the model's performance. When $n = 0$, the hallucination process is an identity transformation, introducing no additional operation except for extrapolation. When $n = 2$, non-linearity is introduced. Then we notice a salient improvement over the baseline. We set $n = 3$ by default since it gives better performance.

4.4. Transferring Features

The primary goal of representation is to learn transferrable features. We evaluate the transferability of features from the proposed method following the previous protocol [29, 11, 10, 12]. Then, we compare the representation quality by transferring them to downstream tasks, including VOC [18] object detection and COCO [39] object detection and instance segmentation. Notably, we re-implement all these experiments using the same settings in MoCo's detectron2 codebase [59].

Object Detection on PASCAL VOC. Following the paradigm of previous work [29], we use Faster R-CNN[20] as the object detection method using R50-C4 as the detector [30]. We train the model end-to-end on the **trainval2007+2012** and evaluate its performance on **test2007**. As shown in Table 7, we observe a stable gain range from 0.3 to 0.8 under different metrics on MoCoV1 and MoCoV2.

Object Detection and instance segmentation on COCO. We continue to report the detection and segmentation re-

sults on COCO using Mask R-CNN [30]. Similarly, we use the R50-C4 as the backbone. The is model in an end-to-end way on **train2017**. Then, the model is evaluated on **val2017**. Again, the proposed method benefits the object detection and segmentation tasks with various metrics as demonstrated in Table 7.

5. Conclusion

In this work, we propose *Hallucinator*, which generates additional hard positive pairs for contrastive learning models based on Siamese structure. *Hallucinator* generates novel data samples in the feature space to provide the training with further contrast without additional computation. We design an asymmetric feature extrapolation to avoid trivial positive pairs and innovatively introduce non-linear hallucination to smooth the generated samples. We empirically prove the effectiveness and generalization capacity of *Hallucinator* to well-recognized contrastive learning models, including MoCoV1&V2, SimCLR and SimSiam. Finally, we hope this work could bring the concept of "Hallucination" into the SSL domain and unlock future research on sample generations&synthesis in contrastive learning.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.

[2] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *Advances in neural information processing systems*, 32, 2019.

[3] Zhipeng Bao, Yu-Xiong Wang, and Martial Hebert. Bowtie networks: Generative modeling for joint few-shot recognition and novel-view synthesis. *arXiv preprint arXiv:2008.06981*, 2020.

[4] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[5] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[6] Qingxing Cao, Liang Lin, Yukai Shi, Xiaodan Liang, and Guanbin Li. Attention-aware face hallucination via deep reinforcement learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 690–698, 2017.

[7] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.

[8] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[9] Pengguang Chen, Shu Liu, and Jiaya Jia. Jigsaw clustering for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11526–11535, 2021.

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

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

[12] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15750–15758, 2021.

[13] Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. *Advances in neural information processing systems*, 33:8765–8775, 2020.

[14] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on*

artificial intelligence and statistics, pages 215–223. JMLR Workshop and Conference Proceedings, 2011.

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[16] Terrance DeVries and Graham W Taylor. Dataset augmentation in feature space. *arXiv preprint arXiv:1702.05538*, 2017.

[17] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9588–9597, 2021.

[18] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88:303–308, 2009.

[19] Tianyu Gao, Xingcheng Yao, and Danqi Chen. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*, 2021.

[20] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[21] Kehong Gong, Bingbing Li, Jianfeng Zhang, Tao Wang, Jing Huang, Michael Bi Mi, Jiashi Feng, and Xinchao Wang. Posetriplet: co-evolving 3d human pose estimation, imitation, and hallucination under self-supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11017–11027, 2022.

[22] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[23] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[24] Matthew D Grilli and Elizabeth L Glisky. The self-imagination effect: Benefits of a self-referential encoding strategy on cued recall in memory-impaired individuals with neurological damage. *Journal of the International Neuropsychological Society*, 17(5):929–933, 2011.

[25] Liangke Gui, Adrien Bardes, Ruslan Salakhutdinov, Alexander Hauptmann, Martial Hebert, and Yu-Xiong Wang. Learning to hallucinate examples from extrinsic and intrinsic supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8701–8711, 2021.

[26] Hongyu Guo. Nonlinear mixup: Out-of-manifold data augmentation for text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4044–4051, 2020.

[27] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006.

[28] Bharath Hariharan and Ross Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *Proceedings of the IEEE international conference on computer vision*, pages 3018–3027, 2017.

[29] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[30] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[31] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[32] Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *International conference on machine learning*, pages 4182–4192. PMLR, 2020.

[33] Wei-Ning Hsu, Yu Zhang, and James Glass. Unsupervised domain adaptation for robust speech recognition via variational autoencoder-based data augmentation. In *2017 IEEE automatic speech recognition and understanding workshop (ASRU)*, pages 16–23. IEEE, 2017.

[34] Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. *Advances in Neural Information Processing Systems*, 33:21798–21809, 2020.

[35] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. ., 2009.

[36] Varun Kumar, Hadrien Glaude, Cyprien de Lichy, and William Campbell. A closer look at feature space data augmentation for few-shot intent classification. *arXiv preprint arXiv:1910.04176*, 2019.

[37] Kibok Lee, Yian Zhu, Kihyuk Sohn, Chun-Liang Li, Jinwoo Shin, and Honglak Lee. i-mix: A domain-agnostic strategy for contrastive representation learning. *arXiv preprint arXiv:2010.08887*, 2020.

[38] Jiangmeng Li, Wenwen Qiang, Changwen Zheng, Bing Su, and Hui Xiong. Metaug: Contrastive learning via meta feature augmentation. In *International Conference on Machine Learning*, pages 12964–12978. PMLR, 2022.

[39] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

[40] Giovanni Mariani, Florian Scheidegger, Roxana Istrate, Costas Bekas, and Cristiano Malossi. Bagan: Data augmentation with balancing gan. *arXiv preprint arXiv:1803.09655*, 2018.

[41] 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*, pages 6707–6717, 2020.

[42] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

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

[44] Xiangyu Peng, Kai Wang, Zheng Zhu, Mang Wang, and Yang You. Crafting better contrastive views for siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16031–16040, 2022.

[45] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. *arXiv preprint arXiv:2010.04592*, 2020.

[46] Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Abhishek Kumar, Rogerio Feris, Raja Giryes, and Alex Bronstein. Delta-encoder: an effective sample synthesis method for few-shot object recognition. *Advances in neural information processing systems*, 31, 2018.

[47] Anshul Shah, Aniket Roy, Ketul Shah, Shlok Mishra, David Jacobs, Anoop Cherian, and Rama Chellappa. Halp: Hallucinating latent positives for skeleton-based self-supervised learning of actions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18846–18856, 2023.

[48] Anshul Shah, Suvrit Sra, Rama Chellappa, and Anoop Cherian. Max-margin contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8220–8230, 2022.

[49] Zhiqiang Shen, Zechun Liu, Zhuang Liu, Marios Savvides, Trevor Darrell, and Eric Xing. Un-mix: Rethinking image mixtures for unsupervised visual representation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2216–2224, 2022.

[50] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 776–794. Springer, 2020.

[51] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? *Advances in neural information processing systems*, 33:6827–6839, 2020.

[52] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012.

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

[54] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3024–3033, 2021.

[55] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7278–7286, 2018.

[56] Jing Wu, Naira Hovakimyan, and Jennifer Hobbs. Genco: An auxiliary generator from contrastive learning for enhanced few-shot learning in remote sensing. *arXiv preprint arXiv:2307.14612*, 2023.

[57] Jing Wu, David Pichler, Daniel Marley, David Wilson, Naira Hovakimyan, and Jennifer Hobbs. Extended agriculture-vision: An extension of a large aerial image dataset for agricultural pattern analysis. *arXiv preprint arXiv:2303.02460*, 2023.

[58] Jing Wu, Ran Tao, Pan Zhao, Nicolas F Martin, and Naira Hovakimyan. Optimizing nitrogen management with deep reinforcement learning and crop simulations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1712–1720, 2022.

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

[60] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018.

[61] Zhenda Xie, Yutong Lin, Zheng Zhang, Yue Cao, Stephen Lin, and Han Hu. Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16684–16693, 2021.

[62] Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. Consert: A contrastive framework for self-supervised sentence representation transfer. *arXiv preprint arXiv:2105.11741*, 2021.

[63] Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6210–6219, 2019.

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

[65] Hongguang Zhang, Jing Zhang, and Piotr Koniusz. Few-shot learning via saliency-guided hallucination of samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2770–2779, 2019.

[66] Ruixiang Zhang, Tong Che, Zoubin Ghahramani, Yoshua Bengio, and Yangqiu Song. Metagan: An adversarial approach to few-shot learning. *Advances in neural information processing systems*, 31, 2018.

[67] Weilin Zhang and Yu-Xiong Wang. Hallucination improves few-shot object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13008–13017, 2021.

[68] Rui Zhu, Bingchen Zhao, Jingen Liu, Zhenglong Sun, and Chang Wen Chen. Improving contrastive learning by visualizing feature transformation. In *Proceedings of the*

IEEE/CVF International Conference on Computer Vision, pages 10306–10315, 2021.