A Survey of Self-Supervised Learning from Multiple Perspectives: Algorithms, Theory, Applications and Future Trends

Jie Gui, *Senior Member, IEEE,* Tuo Chen, Qiong Cao, Zhenan Sun, *Senior Member, IEEE,* Hao Luo, Dacheng Tao, *Fellow, IEEE*

Abstract—Deep supervised learning algorithms generally require large numbers of labeled examples to attain satisfactory performance. To avoid the expensive cost incurred by collecting and labeling too many examples, as a subset of unsupervised learning, self-supervised learning (SSL) was proposed to learn good features from many unlabeled examples without any human-annotated labels. SSL has recently become a hot research topic, and many related algorithms have been proposed. However, few comprehensive studies have explained the connections among different SSL variants and how they have evolved. In this paper, we attempt to provide a review of the various SSL methods from the perspectives of algorithms, theory, applications, three main trends, and open questions. First, the motivations of most SSL algorithms are introduced in detail, and their commonalities and differences are compared. Second, the theoretical issues associated with SSL are investigated. Third, typical applications of SSL in areas such as image processing and computer vision (CV), as well as natural language processing (NLP), are discussed. Finally, the three main trends of SSL and the open research questions are discussed. A collection of useful materials is available at https://github.com/guijiejie/SSL.

Index Terms—Self-supervised learning, deep learning.

1 Introduction

Deep supervised learning algorithms have achieved satisfactory performance in many areas, such as computer vision (CV) and natural language processing (NLP). Generally, supervised learning algorithms need large numbers of labeled examples to obtain better performance. Models trained on large-scale databases such as ImageNet are widely utilized as pretrained models and then fine-tuned for other tasks (Table 1) due to the following two main reasons. First, the parameters learned on different large-scale databases provide a good starting point. Thus, networks trained on other tasks can converge more quickly. Second, a network trained on large-scale databases has already learned the relevant hierarchy characteristics, which can help lessen overfitting problem during the training processes of other tasks, especially when the numbers of instances in other tasks are small or the training labels are limited.

Unfortunately, in many real data mining and machine learning applications, although many unlabeled training instances can be found, usually only a limited number of labeled training instances are available. Labeled examples are often expensive, difficult, or time-consuming to obtain since they require the efforts of experienced human annotators. For instance, in web user profile analysis, it is easy to collect many web user profiles, but labeling the non-profitable users or profitable users in these data requires inspection, judgment, and even time-consuming tracing tasks to be performed by experienced human assessors, which is very expensive. As another case, in the medical field, unlabeled examples can be easily obtained from routine medical examinations. However, making diagnoses for so many examples in a case-by-case manner imposes a heavy burden on medical experts. For example, to perform breast cancer diagnosis, radiologists must assign labels to every focus in a large number of easily obtained high-resolution mammograms. This process is often very inefficient and time-consuming. Furthermore, supervised learning methods suffer from spurious correlations and generalization errors, and they are vulnerable to adversarial attacks.

To alleviate the two aforementioned limitations of supervised learning, many machine learning paradigms have been proposed, such as active learning, semi-supervised learning and self-supervised learning (SSL). This paper focuses on SSL. SSL algorithms have been proposed to learn good features from a large number of unlabeled instances without using any human annotations.

The general pipeline of SSL is shown in Fig. 1. During the self-supervised pretraining phase, a predefined pretext

- *J. Gui is with the School of Cyber Science and Engineering, Southeast University and with Purple Mountain Laboratories, Nanjing 210000, China (e-mail: guijie@seu.edu.cn).*
- *T. Chen is with the School of Cyber Science and Engineering, Southeast University (e-mail: 230219309@seu.edu.cn).*
- *Q. Cao is with JD Explore Academy (e-mail: mathqiong2012@gmail.com).*
- *Z. Sun is with the Center for Research on Intelligent Perception and Computing, Chinese Academy of Sciences, Beijing 100190, China (e-mail: znsun@nlpr.ia.ac.cn).*
- *H. Luo is with Alibaba Group, Hangzhou 310052, China (e-mail: haoluocsc@zju.edu.cn).*
- *D. Tao is with JD Explore Academy, China and with the School of Computer Science in the University of Sydney, Australia. E-mail: dacheng.tao@gmail.com.*

Manuscript received April 19, 2005; revised August 26, 2015.

Pretraining	Data	Pretraining Tasks	Downstream Tasks
Supervised	extensive labeled data	image categorization	detection / segmentation / pose estimation / depth estimation, etc
		video action categorization	action recognition / object tracking, etc
SSL	extensive unlabeled data	Image: rotation, jigsaw, etc	detection / segmentation / pose estimation / depth estimation, etc
		Video: the order of frames, playing direction, etc	action recognition / object tracking, etc
		NLP: masked language modeling	question answering / textual entailment recognition / natural language inference, etc.

TABLE 1: Contrast between supervised and self-supervised pretraining and fine-tuning.

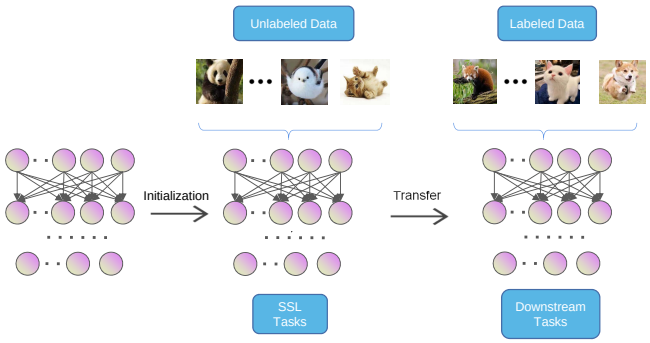

Fig. 1: The general pipeline of SSL.

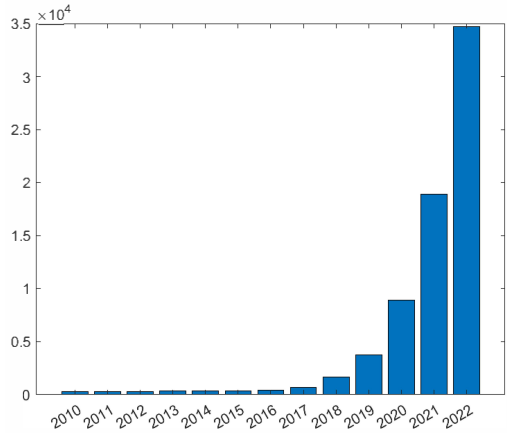

Fig. 2: Google Scholar search results for "self-supervised learning". The vertical and horizontal axes denote the number of SSL publications and the year, respectively.

task is designed for a deep learning algorithm to solve, and the pseudolabels for the pretext task are automatically generated based on certain attributes of the input data. Then, the deep learning algorithm is trained to learn to solve the pretext task. After the self-supervised pretraining process is completed, the learned model can be further transferred to downstream tasks (especially when only a relatively small number of examples are available) as a pretrained model to improve performance and overcome overfitting issues.

Because no human annotations are required to generate pseudolabels during self-supervised training, one main merit of SSL algorithms is that they can make the most of large-scale unlabeled data. Trained with these pseudolabels, self-supervised algorithms have achieved promising results, and the performance gap between self-supervised and supervised algorithms in downstream tasks has decreased. Asano et al. [1] showed that even on only a single image, SSL can surprisingly produce low-level characteristics that generalize well.

SSL [2]–[19] has recently attracted increasing attention (Fig. 2). Yann LeCun, one of the recipients of the 2018 ACM A.M. Turing Award, gave a keynote talk at the Eighth International Conference on Learning Representations (ICLR 2020), and the title of his talk was "The future is self-supervised". Yann LeCun and Yoshua Bengio, who both received the Turing award, said that SSL is key to human-level intelligence [20]. According to Google Scholar, a large number of papers related to SSL have already been published. For example, approximately 18,900 papers related to SSL were published in 2021, constituting approximately 52 papers every day or more than two papers per hour (Fig. 2). To prevent the researchers from becoming lost in so many SSL papers and to collate the latest research findings, we attempt to provide a timely survey of this topic.

Differences From Previous Work: Reviews of SSL are available for specific applications such as recommender systems [21], graphs [22], [23], sequential transfer learning [24], videos [25], adversarial pretraining of self-supervised deep networks [26], and visual feature learning [27]. Liu et al. [18] mainly covered papers written before 2020, and their work did not contain the latest progress. Jaiswal et al. [28] focused on contrastive learning (CL). SSL research breakthroughs in CV have been achieved in recent years. In this work, we therefore mainly include SSL research derived from the CV community in recent years, especially classic and influential research results. The objectives of this review are to explain what SSL is, its categories and subcategories, how it differs and relates to other machine learning paradigms, and its theoretical underpinnings. We present an up-to-date and comprehensive review of the frontiers of visual SSL and divide visual SSL into three parts: context-based, contrastive, and generative SSL, in the hope of sorting the trends for researchers.

The remainder of this paper is organized as follows. Sections 2-7 introduce SSL from the perspectives of its algorithms, theory, applications, three main trends, open questions, and performance comparisons, which can be seen in Table 2. Finally, Section 8 concludes the survey.

2 Algorithms

In this section, we first introduce what SSL is. Then, we introduce the pretext tasks of SSL and its combinations with other learning paradigms.

ALGORITHMS	What is SSL?	
	Pretext tasks	Context-based methods
		CL
		Generative algorithms
	Combinations with other learning paradigms	Generative adversarial networks (GANs)
		Semi-supervised learning
		Multi-instance learning
		Multi-view/Multi-modal(ality) learning
		Test time training
THEORY	Generative algorithms	Maximum likelihood estimation (MLE)
		The original GANs
		InfoGAN's disentangling ability
		Denoising autoencoder (DAE)
	Contrastive SSL	Connection to other unsupervised learning algorithms
		Connection to supervised learning
		Connection to metric learning
		Understanding the contrastive loss based on alignment and uniformity
		The relationship between the contrastive loss and mutual information
	Complete collapse and dimensional collapse	
APPLICATIONS	Image processing and computer vision	
	Natural language processing (NLP)	
	Other fields	
THREE MAIN TRENDS	Theoretical analysis of SSL	
	Automatic design of an optimal pretext task	
	A unified SSL paradigm for multiple modalities	
OPEN QUESTIONS	Can SSL benefit from almost unlimited data?	
	What is its relationship with multi-modality learning?	
	Which SSL algorithm is the best/should I use?	
	Do unlabeled data always help?	

TABLE 2: Structure of this paper.

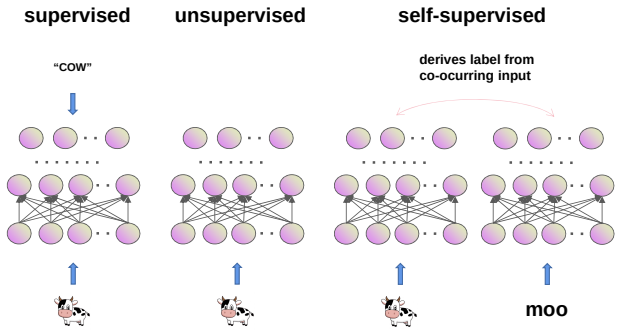

Fig. 3: The differences among supervised learning, unsupervised learning, and SSL. The image is reproduced from [30].

2.1 What is SSL?

Before diving into SSL, we first introduce the concept of unsupervised learning. In unsupervised learning [29], the training data are composed of a set of input vectors x without any corresponding target values. Representative unsupervised learning algorithms include clustering and density estimation.

SSL was possibly first introduced in [30] (Fig. 3). [30] used this structure in natural environments derived from different modalities. For instance, seeing a cow and hearing "mooing" are events often occur together. Thus, although the sight of a cow does not mean that a cow label should be ascribed, it does co-occur with an example of a "moo". The key is to process the cow image to obtain a self-supervised label for the network so that it can process the "moo" sound and vice versa.

Since then, the machine learning community has further developed the SSL idea. **SSL belongs to a subclass of unsupervised learning**. In SSL, output labels can be 'intrinsically' generated from the input data examples by exposing the relations between parts of the data or different views of the data. The output labels are generated from the data examples themselves. From this definition, an autoencoder (AE) may be seen as one kind of SSL algorithms in which the output labels are the data themselves. AEs have been widely used in many areas, such as dimensionality reduction and anomaly detection.

In Yann LeCun's keynote talk at ICLR 2020, SSL was described as equal to filling in the blanks (reconstruction), and he gave several forms of SSL (Fig. 4), which are shown as follows.

1) Predict any part of the input from any other part.
2) Predict the future from the past.
3) Predict the invisible from the visible.
4) Predict any occluded, masked, or corrupted part from all available parts.

In SSL, a part of the input is unknown, and the goal is to predict that part.

Jing et al. [27] further extended the meaning of SSL as follows. If a method does not involve any human-annotated labels, the method falls into SSL. In this way, SSL is equal to unsupervised learning. Therefore, generative adversarial networks (GANs) [31] belong to SSL.

An important concept in the field of SSL is the idea of pretext (also known as surrogate or proxy) tasks. The term "pretext" means that the task being solved is not the true interest but is solved only for the genuine purpose of providing a promising pretrained model. Common pretext tasks include rotation prediction and instance discrimination. To realize different pretext tasks, different loss functions are introduced. As the most important concept in SSL, we first introduce pretext tasks below.

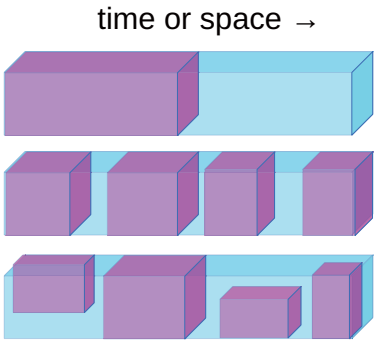

Fig. 4: SSL. This figure is reproduced from Yann LeCun's keynote talk at ICLR 2020. The red part is known, and the other part is unknown.

2.2 Pretext tasks

In this section, we summarize the pretext tasks of SSL. A popular SSL solution is to propose a pretext task for networks to solve, and the networks are trained by learning the objective functions of these pretext tasks. Pretext tasks have two common characteristics, as follows. First, features need to be learned by deep learning methods to solve the pretext tasks. Second, supervised signals are generated from the data themselves (self-supervision).

Existing methods generally utilize three types of pretext tasks: context-based methods, CL, and generative algorithms. Here, generative algorithms generally mean masked image modeling (MIM) methods.

2.2.1 Context-based methods

Context-based methods are usually based on the contextual relationships among the given examples, such as their spatial structures and local and global consistency.

Now, we use rotation as a simple example to demonstrate the concept of context-based pretext tasks [32]. Then, we gradually introduce other tasks.

2.2.1.1 Rotation:

The paradigm that rotation follows involves learning image representations by training neural networks (NNs) to recognize the geometric transformations applied to the original image. For each original image (see "0^o rotation" in Fig. 5), Gidaris et al. [33] created three rotated images with 90^o, 180^o, and 270^o rotations. Each image belonged to one of four classes, 0^o, 90^o, 180^o, or 270^o rotation, which were the output labels generated from the images themselves. More specifically, there is a set of $K = 4$ discrete geometric transformations $G = \{g(\cdot|y)\}_{y=1}^{K}$ where $g(\cdot|y)$ is the operator that applies a geometric transformation with a label of y to image X to produce the transformed image $X^y = g(X|y)$.

Gidaris et al. used a deep convolutional NN (CNN) $F(\cdot)$ to predict rotation; this is a four-class categorization task. The CNN $F(\cdot)$ obtains an input image X^{y^*} (where y^* is unknown to $F(\cdot)$) and produces a probability distribution over all probable geometric transformations:

$$F\left(X^{y^*}|\theta\right) = \left\{F^y\left(X^{y^*}|\theta\right)\right\}_{y=1}^{K}, \tag{1}$$

where $F^y\left(X^{y^*}|\theta\right)$ is the predicted probability for the geometric transformation with a label of y and θ denotes the learnable parameters of $F(\cdot)$.

Intuitively, a good CNN should be able to correctly categorize the $K = 4$ classes of natural images. Thus, given a set of N training instances $D = \{X_i\}_{i=1}^{N}$, the self-supervised training objective of $F(\cdot)$ is

$$\min_{\theta} \frac{1}{N}\sum_{i=1}^{N} loss(X_i, \theta), \tag{2}$$

where the loss function $loss(\cdot)$ is

$$loss(X_i, \theta) = -\frac{1}{K}\sum_{y=1}^{K} \log(F^y(g(X_i|y)|\theta)). \tag{3}$$

In [34], the relative rotation angle was constrained to be within the range $[-30^o, 30^o]$. The rotations were binned into bins of size 3^o, each resulting in a total of 20 classes (or bins).

2.2.1.2 Colorization:

Colorization was first proposed in [35], and [35]–[38] showed that colorization can be a powerful pretext task for SSL. Color prediction has the advantageous characteristic that the training data can be totally free. The L channel of any color image can be used as the input of an NN system, and its corresponding ab color channels in the CIE Lab color space can be used as self-supervised signals. Given an input lightness channel $X \in R^{H\times W\times 1}$, the objective is to predict the ab color channels $Y \in R^{H\times W\times 2}$, where H and W are the height and width dimensionality, respectively. We use Y and $\hat{Y}$ to denote the ground truth and the predicted value, respectively. A natural objective function minimizes the Frobenius norm between Y and $\hat{Y}$:

$$L = \left\|\hat{Y} - Y\right\|_F^2. \tag{4}$$

[35] used the multinomial cross-entropy loss rather than (4) to achieve increased robustness. After the training process is finished, for any grayscale image, we can predict its ab color channels. Then, the L channel and the ab color channels can be concatenated to make the original grayscale image colorful.

2.2.1.3 Jigsaw:

The jigsaw approach uses jigsaw puzzles as proxy tasks. It relies on the intuition that a network accomplishes the proxy tasks by understanding the contextual information contained in the examples. More specifically, it breaks up pictures into discrete patches, then randomly changes their positions and tries to recover the original order. [39] studied the effect of scaling two self-supervised methods (jigsaw [40]–[43] and colorization) along three dimensions: data size, model capacity, and problem complexity. The results [39] showed that transfer performance increases log-linearly with the data size. The representation quality also improves with higher-capacity models and increased problem complexity. Closely related works to [40] include [44], [45].

2.2.1.4 Others:

The pretext task of [46], [47] was a conditional motion propagation problem. Noroozi et al. [48] enforced an addition constraint on the feature representation process: the sum of the feature representations of all image patches should

Fig. 5: Rotation. For each original image ("0^o rotation"), Gidaris et al. [33] created three rotated images: 90^o, 180^o, and 270^o rotations.

be approximately equal to the feature representation of the whole image. Many pretext tasks lead to representations that are covariant with image transformations. [49] argued that semantic representations should be invariant under such transformations, and a pretext-invariant representation learning (PIRL) approach that learns invariant representations based on pretext tasks was developed.

2.2.2 CL

Following simple instance discrimination tasks [50]–[52], many CL-based SSL methods such as momentum contrast (MoCo) v1 [53], MoCo v2 [54], MoCo v3 [55], a simple framework for CL of visual representations (SimCLR) v1 [56], and SimCLR v2 [57] have emerged.

Classic algorithms such as MoCo have pushed the performance of self-supervised pretraining to a level comparable to that of supervised learning, making SSL relevant for large-scale applications for the first time.

Early CL approaches were constructed based on the idea of negative examples. With the development of CL, a number of CL methods that do not use negative examples have emerged. They follow different ideas, such as self-distillation and feature decorrelation, but they all obey the idea of positive example consistency. We describe the different available CL methods below.

2.2.2.1 CL methods based on negative examples: Negative examples based CL follows a similar pretext task: instance discrimination. The basic idea is to make positive examples close to each other and negative examples far from each other in the latent space. The exact way in which positive and negative examples are defined varies according to the given modality and other factors, which can include spatial and temporal consistency in video understanding or the cooccurrence between modalities in multi-modal learning.

MoCo. He et al. [53] viewed CL as a dictionary look-up task. Consider an encoded query q and several encoded examples $\{k_0, k_1, k_2, \cdots\}$, which are the keys of a dictionary. Assume that a single key (denoted as k_+) in the dictionary matches q. A contrastive loss [58] is a function whose value is low if q is similar to its positive key k_+ and dissimilar to all other negative keys. With similarity measured by the dot product, one contrastive loss function form called InfoNCE [59] was considered in MoCo v1 [53]:

$$L_q = -\log \frac{\exp(q \cdot k_+/\tau)}{\sum_{i=0}^{K} \exp(q \cdot k_i/\tau)}, \tag{5}$$

where τ denotes the temperature hyperparameter. The sum is calculated over one positive example and K negative examples. InfoNCE was derived from noise contrastive estimation (NCE) [60], whose objective is

$$L_q = -\log \frac{\exp(q \cdot k_+/\tau)}{\exp(q \cdot k_+/\tau) + \exp(q \cdot k_-/\tau)}, \tag{6}$$

where q is similar to a positive example k_+ and dissimilar to a negative example k_-.

Based on MoCo v1 [53] and SimCLR v1 [56], MoCo v2 [54] uses an multilayer perceptron (MLP) projection head and more data augmentations.

SimCLR. SimCLR v1 [56] randomly samples a minibatch of N instances and defines a contrastive prediction task on pairs of augmented instances from the minibatch, producing $2N$ instances. SimCLR v1 does not explicitly sample negative instances. Instead, given a positive pair, SimCLR v1 treats the other $2(N-1)$ augmented instances in the minibatch as negative instances. Let $sim(u, v) = u^T v/(\|u\| \, \|v\|)$ be the cosine similarity between two instances u and v. Then, the loss function of SimCLR v1 for a positive pair of instances (i, j) is

$$l_{i,j} = -\log \frac{\exp(sim(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} 1_{[k \neq i]} \exp(sim(z_i, z_k)/\tau)}, \tag{7}$$

where $1_{[k \neq i]} \in \{0, 1\}$ is an indicator function equal to 1 iff $k \neq i$ and τ is the temperature hyperparameter. The final loss is computed by all positive pairs, both (i, j) and (j, i), in the mini-batch.

Both MoCo and SimCLR require data augmentation techniques such as cropping, resizing, and color distortion. Other augmentation methods are available [61]. For example, [62] estimated the foreground saliency levels in images and created augmentations by copying and pasting the image foregrounds onto different backgrounds, such as homogeneous grayscale images with random grayscale levels, texture images, and ImageNet images. However, why augmentation helps and how we can perform more effective augmentations are still unclear and require further studies.

2.2.2.2 CL methods based on self-distillation: Bootstrap your own latent (BYOL) [63] is a representative self-distillation algorithm. BYOL was proposed for self-supervised image representation learning without using negative pairs. BYOL uses two NNs, which are called online and target networks. Similar to MoCo [53], BYOL updates the target network with a slow-moving average of the online network.

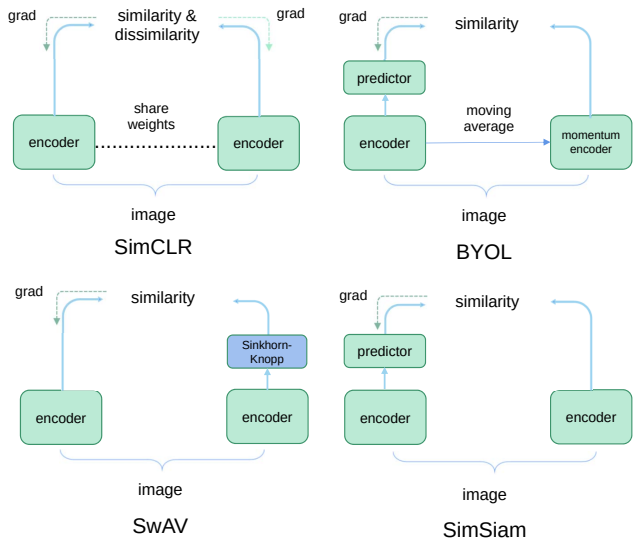

Fig. 6: Comparison among Siamese architectures. The image is reproduced from [65].

Siamese networks such as SimCLR, BYOL, and SwAV [64] have become common structures in various recently developed models for self-supervised visual representation learning. These models maximize the similarity between two augmentations of one image; they are subject to certain conditions to prevent collapsing solutions.

[65] proposed simple Siamese (SimSiam) networks that can learn useful representations without using the following: negative sample pairs, large batches, and momentum encoders. For each data point x, we have two randomly augmented views x_1 and x_2. An encoder f and an MLP prediction head h are used to process the two views. Denoting the two outputs by $p_1 = h\left(f\left(x_1\right)\right)$ and $z_2 = f\left(x_2\right)$, [65] minimized their negative cosine similarity

$$D\left(p_1, z_2\right) = -\frac{p_1}{\left\|p_1\right\|_2} \frac{z_2}{\left\|z_2\right\|_2}, \tag{8}$$

where $\|\|_2$ is the l_2-norm. Similar to [63], [65] defined a symmetric loss as

$$L = \frac{1}{2}\left(D\left(p_1, z_2\right) + D\left(p_2, z_1\right)\right), \tag{9}$$

where this loss is defined based on the example x and the total loss is the average of all examples. More importantly, [65] used a stop-gradient ($stopgrad$) operation by revising (8) as follows:

$$D\left(p_1, stopgrad\left(z_2\right)\right), \tag{10}$$

which means that z_2 is seen as a constant. Analogously, (9) is revised as

$$L = \frac{1}{2}\left(D\left(p_1, stopgrad\left(z_2\right)\right) + D\left(p_2, stopgrad\left(z_1\right)\right)\right). \tag{11}$$

The differences among SimCLR, BYOL, SwAV, and SimSiam are shown in Fig. 6. Since BYOL and SimSiam do not use negative examples, whether they belong to CL is controversial. To be consistent with [66], BYOL and SimSiam belong to CL in this paper.

2.2.2.3 CL methods based on feature decorrelation: Feature decorrelation aims to learn decorrelated features.

Barlow twins. Barlow twins [67] were proposed with a novel loss function; they make the embedding vectors of distorted versions of an example similar while minimizing the redundancy between the components of these vectors. More specifically, similar to other SSL methods [53], [56], Barlow twins produce two views for all images of a batch X sampled from a database and finally produce batches of embeddings Z^A and Z^B, respectively. The objective function of Barlow twins is

$$L_{BT} = \sum_i \left(1 - C_{ii}\right)^2 + \lambda \sum_i \sum_{j \neq i} C_{ij}^2, \tag{12}$$

where λ is a free parameter; C is the cross-correlation matrix computed between the outputs of two equivalent networks along the batch dimension:

$$C_{ij} = \frac{\sum_b z_{b,i}^A z_{b,j}^B}{\sqrt{\sum_b \left(z_{b,i}^A\right)^2} \sqrt{\sum_b \left(z_{b,j}^B\right)^2}}, \tag{13}$$

where b is the batch example index and i, j is the vector dimension index of the network outputs. C is a square matrix with a size equal to the dimensionality of the network output.

Variance-Invariance-Covariance Regularization. Similar to Barlow twins [67], variance-invariance-covariance regularization (VICReg) [68] has been proposed for SSL. Barlow twins consider a cross-correlation matrix, while VICReg considers variance, invariance, and covariance. Let d, n, and z_j^A denote the dimensionality of the vectors in Z^A, the batch size, and the vector consisting of every value with dimensionality j among all examples of Z^A, respectively. The variance regularization term v of VICReg is defined as a hinge loss function on the standard deviation of the embeddings along the batch dimension:

$$v\left(Z^A\right) = \frac{1}{d} \sum_{j=1}^{d} max(0, \gamma - S\left(z_j^A, \varepsilon\right)), \tag{14}$$

where S is the regularized standard deviation, which is defined as

$$S(y, \varepsilon) = \sqrt{Var(y) + \varepsilon}, \tag{15}$$

where γ is a constant for the standard deviation, which is set to 1 in the experiments, and ε is a small scalar for preventing numerical instabilities. This criterion encourages the variance inside the current batch to be equal to or greater than γ for every dimension, preventing collapse in cases where all data are mapped to the same vector.

The invariance criterion s of VICReg between Z^A and Z^B is defined as the mean-squared Euclidean distance between each pair of data without any normalization:

$$s\left(Z^A, Z^B\right) = \frac{1}{n} \sum_{b=1}^{n} \left\|z_b^A - z_b^B\right\|_2^2. \tag{16}$$

The covariance criterion $c(Z)$ of VICReg is defined as

$$c\left(Z\right) = \frac{1}{d} \sum_{i \neq j} \left[C(Z)\right]_{i,j}^2, \tag{17}$$

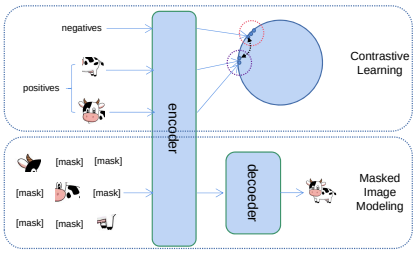

Fig. 7: The broad differences between CL and MIM. Note that the actual differences between their pipelines are not limited to what is shown.

where $C(Z)$ is the covariance matrix of Z.

The overall loss of VICReg is a weighted sum of the variance, invariance, and covariance:

$$l\left(Z^A, Z^B\right) = s\left(Z^A, Z^B\right) + \alpha\left(v\left(Z^A\right) + v\left(Z^B\right)\right) + \beta\left(C\left(Z^A\right) + C\left(Z^B\right)\right), \quad (18)$$

where α and β are two free-tuning parameters.

2.2.2.4 Others:

Along with the works mentioned above, [55], [69] tried to employ vision transformers (ViTs) as the backbone for contrastive SSL. With multi-crop and cross-entropy losses, [69] demonstrated certain distinctive properties that emerge when using the Vision Transformer (ViT) as the backbone for contrastive SSL. Features derived from contrastive self-supervised ViTs are excellent K-nearest neighbors (K-NN) classifiers, and they contain explicit information about the semantic segmentation of an image. These properties motivate certain downstream tasks [70].

In [71], patches taken from the same image were seen as a positive pair, while patches taken from distinct images were seen as a negative pair. The overall objective function of Yang et al. [72] is a weighted combination of CL and MIM for text recognition.

Many other CL-based methods are available [73]–[93]. Note that CL is not only used in SSL but also can be used in supervised learning [94].

2.2.3 Generative algorithms

In this work, we mainly focus on MIM methods among the available generative algorithms. MIM methods [95] (Fig. 7), such as bidirectional encoder representation from image transformers (BEiT) [96], masked AEs (MAEs) [66], context AE (CAE) [97], and a simple framework for MIM (SimMIM) [98], have recently become are very popular and are challenging the dominance of CL. MIM is a generative algorithm that uses the co-occurrence relationships between image patches as the supervision signals.

MIM can be seen as a variant of the denoising AE (DAE) [3]. Its main idea is that a representation should be robust to the noise contained in the input examples. The most famous variant of the DAE model may be bidirectional encoder representations from Transformers (BERT), which has achieved great success in NLP. Due to the excellent achievements of BERT pretraining, researchers want to replicate the success of BERT in CV. However, the success of the BERT pretraining process in NLP is not only derived from its large-scale self-supervised pretraining strategy but also related to the scalable network structure. A key difference between the NLP and CV communities is that they use different primary models, with NLP using transformers and CV using CNNs.

The advent of the original ViT [99] has changed this situation. In the ViT, Alexey Dosovitskiy et al. performed a preliminary exploration of MIM for CV, mimicking the BERT-style masked image prediction process. Their ViT-B/16, a smaller model, achieved 79.9% accuracy on ImageNet through self-supervised pretraining. Although this was a 2% accuracy improvement over that achieved by training from scratch, it still lagged behind the accuracy of supervised pretraining. In addition to ViTs, another early study used context encoders [100], following the same idea as that of MAEs, i.e., image inpainting.

Despite the alignment of their network structures, MAEs were not used in vision research until the advent of BEiT. Before BEiT, another meaningful attempt was image generative pretraining (iGPT) [101], but it did not receive much attention due to its poor accuracy and low computational efficiency. Following BERT in the NLP area, BEiT was used to propose a tailored MIM task for visual pretraining. More specifically, BEiT "tokenizes" the original input image into visual tokens and randomly masks a subset of the image patches. Following BERT, the masked and unmasked image tokens are fed into the ViT, and the pretraining objective is to recover the masked visual tokens according to the unmasked image patches. The authors used a discrete variational AE (dVAE) [102] to build a predefined visual vocabulary to overcome the issues brought by predicting the raw pixels. A noteworthy point is that the downstream task is not specifically [mask] labeled, and the researchers developed different algorithms to mitigate this problem. For example, in the original BERT, the authors used methods such as random words to alleviate the upstream and downstream inconsistency problems. In CV, BEiT and SimMIM follow similar paradigms to that of BERT by feeding special [mask] tokens into the network.

Unlike BEiT, the MAE discards these special mask tokens and regards the task as a regression problem. MAE explains the problems that need to be solved by applying the DAE in CV from three perspectives: the architectures, information density, and decoder. MAE is simple and effective and has become an important baseline in the MIM field.

Here, we define the MIM process in the form shown below:

$$\text{MIM} := \mathcal{L}\left(\mathcal{D}\left(\mathcal{E}\left(\mathcal{T}_1\left(I\right)\right)\right), \mathcal{T}_2\left(I\right)\right),$$

where $\mathcal{E}$ represents the encoder, $\mathcal{D}$ represents the decoder, $\mathcal{T}_1$ represents the transformation of the input before it is fed into the network, and $\mathcal{T}_2$ represents the transformation used to obtain the target label. Note that this is presented for ease of exposition rather than as a strict definition.

The most obvious distinction between BEiT and MAE is the choice of $\mathcal{T}$. BEiT uses the token output from the pretrained tokenizer as its target, while the MAE directly uses the original pixel as its target. BEiT is a two-stage approach, where a tokenizer is trained in the first stage to convert images into visual tokens and then BERT-style training is performed. The MAE is a one-stage end-to-end approach that uses a (nonrequired) decoder to decode the representation derived from the encoder into the original pixels.

BEiT and MAE are two representative MIM approaches that uphold different architectural designs. The architectural designs of subsequently developed MIM methods mostly follow one of these two techniques. Target representation is the core problem of MIM, namely, the choice of $\mathcal{T}_2$. We can organize the different available MIM methods according to $\mathcal{T}_2$, as shown in Table 3.

Methods	Reconstructing Target $\mathcal{T}_2$
Raw Pixels/Low-Level Targets	
ViT	Raw Pixel
MAE	Raw Pixel
MaskFeat	HOG
SimMIM	Raw Pixel
High-Level Targets	
BEiT	VQ-VAE
PeCo	VQ-GAN
CAE	VQ-VAE
Self-Distillation	
data2vec	Self
SdAE	Self
Contrastive / Multi-modal Teacher	
MimCo	MoCo v3
BEiT v2	CLIP

TABLE 3: Classifications of MIM methods based on their reconstruction objectives.

After BEiT and MAE were proposed, a number of variants emerged. iBOT [95] is an "online tokenizer" version of BEiT, as the authors found that the dVAE was still only able to capture low-level semantics within local details. The CAE introduces an alignment constraint, encouraging the representations for masked patches (predicted by a 'latent contextual regressor') to lie in the encoded representation space. This approach decouples the representation learning task and pretext task, which improves the representation learning capacity of the model. Works including [103], [104] used the MAE for videos.

MIM has exhibited massive potential in SSL in the past year. Benefiting from its universal ViT backbone, MIM learns self-supervised visual representations by masking some of the patches of the original image while attempting to recover masked information. Most previous works randomly masked image patches thereby underutilizing the semantic information that is beneficial for visual representation learning. On the other hand, due to the large size of the backbone, most previously developed methods must spend much time on pretraining. Liu et al. [105] proposed an attention-driven masking and throwing strategy (AMT), which can solve both of the above problems.

2.2.4 Summary

As discussed, many pretext tasks of SSL have been constructed; some of the milestone variants are shown in Fig. 8. Note that due to space limitations, only a limited number of variants are shown.

Many other types of pretext tasks are available [106], [107], such as relative patch location [108], noise prediction [109], feature clustering [110]–[112], and cross-channel prediction [113]. The main difference between [111] and [112] is that [111] used agglomerative clustering while [112] used k-means. Recently, methods for combining different cues have been proposed [114].

Context-based approaches have limited applicability due to their inferior performance. The CL and MIM methods are two mainstream types of visual SSL algorithms. Whereas visual CL may suffer from overfitting problems, CL algorithms that combine multi-modality, such as CLIP [115], are more popular.

Kolesnikov et al. [116] revisited previously proposed SSL pretext tasks, conducted a thorough large-scale study, and uncovered multiple crucial insights. [117] evaluated four self-supervised pretraining methods, the variational AE (VAE) [118], rotation [33], contrastive multiview coding (CMC) [119], and augmented multiscale deep infoMax (AMDIM) [120], across a comprehensive set of synthetic datasets and downstream tasks to investigate which factors may play important roles in the utility of these pretraining algorithms for real applications. Kolesnikov et al. [116] found that standard CNN design strategies do not always translate to SSL.

[121] presented an alternative to pretext tasks and showed how can be easily obtained from video games.

2.3 Combinations with other learning paradigms

Note that although SSL has made great strides, the currently popular techniques were not developed in isolation and suddenly exploded. To clearly illustrate the combinations of SSL and existing learning paradigms, we list the relevant learning paradigms in this section.

2.3.1 GANs

GANs are classic unsupervised learning methods and were some of the most successful unsupervised learning methods before SSL methods became popular. GANs can be combined with SSL in many ways, with the self-supervised GANs (SS-GAN) as an example. The objective function of GANs [31], [126] is

$$\begin{aligned}\min_G \max_D \; & V(G,D) = E_{x\sim p_{data}(x)}\left[\log D(x)\right] \\ & + E_{z\sim p_z(z)}\left[\log\left(1 - D\left(G\left(z\right)\right)\right)\right].\end{aligned} \tag{19}$$

The objective function of the SS-GAN [122] is a combination of that of GANs and rotation [33]:

$$\begin{aligned} L_G &= -V(G,D) - \alpha E_{x\sim p_G} E_{r\sim R}[\log Q_D\left(R = r|x^r\right)], \\ L_D &= V(G,D) - \beta E_{x\sim p_{data}} E_{r\sim R}[\log Q_D\left(R = r|x^r\right)], \end{aligned} \tag{20}$$

where $V(G,D)$ is (19) and $r \sim R$ is a rotation selected from a set of possible rotations, which is the same as that in [33]. Let x^r denote an image x rotated by r degrees and $Q\left(R|x^r\right)$ be the discriminator's predictive distribution over the angles of rotation of the example. Rotation [33] is a classic SSL method. The SS-GAN incorporates rotation invariance into the GANs generation process by introducing the rotation prediction task in the GANs training process. Other related works can be found in [122], [127]–[135].

2.3.2 Semi-supervised learning

Both SSL and semi-supervised learning are juxtaposed paradigms and can be used in conjunction with each other. Self-supervised semi-supervised learning (S^4L) [123] is taken as an example. The objective function of S^4L is

$$\min_\theta \; L_l\left(D_l, \theta\right) + wL_u\left(D_u, \theta\right), \tag{21}$$

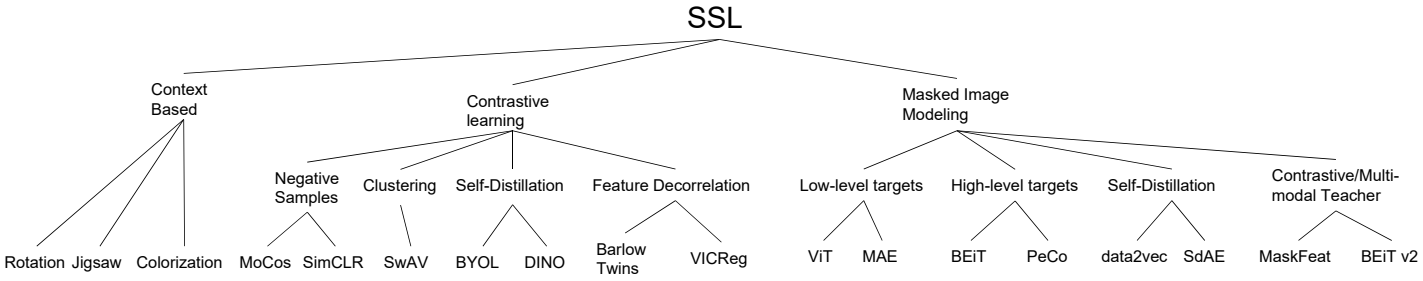

Fig. 8: Pretext tasks of SSL, with the milestone variants shown.

TABLE 4: Different losses of SSL.

Category		Method	Loss	Equation
Pretext	Context-Based	Rotation [33]	$loss(X_i,\theta) = -\frac{1}{K}\sum_{y=1}^{K}\log(F^y(g(X_i\|y)\|\theta))$	(3)
	CL	MoCo v1 [53]	$L_q = -\log\frac{\exp(q\cdot k_+/\tau)}{\sum_{i=0}^{K}\exp(q\cdot k_i/\tau)}$	(5)
		SimCLR v1 [56]	$l_{i,j} = -\log\frac{\exp(sim(z_i,z_j)/\tau)}{\sum_{k=1}^{2N}1_{[k\neq i]}\exp(sim(z_i,z_k)/\tau)}$	(7)
		SimSiam [65]	$L = \frac{1}{2}(D(p_1, stopgrad(z_2)) + D(p_2, stopgrad(z_1)))$	(11)
		Barlow twins [67]	$L_{BT} = \sum_i (1-C_{ii})^2 + \lambda\sum_i\sum_{j\neq i} C_{ij}^2$	(12)
		VICReg [68]	$l(Z^A,Z^B) = s(Z^A,Z^B) + \alpha(v(Z^A) + v(Z^B))$ $+\beta(C(Z^A) + C(Z^B))$,	(18)
Combinations with Other Learning Paradigms		SS-GAN [122]	$L_G = -V(G,D) - \alpha E_{x\sim p_G}E_{r\sim R}[\log Q_D(R=r\|x^r)]$, $L_D = V(G,D) - \beta E_{x\sim p_{data}}E_{r\sim R}[\log Q_D(R=r\|x^r)]$	(20)
		S^4L [123]	$\min_\theta\ L_l(D_l,\theta) + wL_u(D_u,\theta)$	(21)
		SSL improving robustness [124]	$L(x,y,\theta) = L_{CE}(y,p(y\|PGD(x)),\theta) + \lambda L_{SS}(PGD(x),\theta)$	(22)
		unsupervised view learning [125]	$\min_g \max_{f_1,f_2} I_{NCE}^{f_1,f_2}(g(X)_1, g(X)_{2:3})$	(25)
		semi-supervised view learning [125]	$\min_{g,c_1,c_2} \max_{f_1,f_2} I_{NCE}^{f_1,f_2}(g(X)_1, g(X)_{2:3})$ $+L_{ce}(c_1(g(X)_1), y) + L_{ce}(c_2(g(X)_{2:3}), y)$	(26)

where D_l is the labeled training data set, D_u is the unlabeled training data set, L_l is the classification loss defined on all labeled examples, L_u is the self-supervised loss (such as that of the rotation task in (3)) based on both D_l and D_u, w is a free parameter for balancing L_l and L_u, and θ is the parameter of the learning model.

In addition to using SSL as an auxiliary task, there is another classic way to utilize SSL in semi-supervised learning. Following SimCLR, we can execute SSL on unlabeled data and then fine-tune the resulting model on labeled data.

Similarly, to show that self-supervision is robust to adversarial perturbations, the overall loss of [124] is a linear combination of a supervised loss and a self-supervised loss:

$$L(x,y,\theta) = L_{CE}(y,p(y|PGD(x)),\theta) + \lambda L_{SS}(PGD(x),\theta), \quad (22)$$

where x is an example, y is the ground-truth label, θ denotes the model parameters, CE is short for cross-entropy, and PGD is short for projected gradient descent. The first and second terms of (22) are the losses for supervised learning and SSL, respectively.

2.3.3 Multi-instance learning (MIL)

[136] extended the InfoNCE loss (5) to MIL and proposed MIL-NCE:

$$\max_{f,g}\sum_{i=1}^{n}\log\left(\frac{\sum_{(x,y)\in P_i} e^{f(x)^T g(y)}}{\sum_{(x,y)\in P_i} e^{f(x)^T g(y)} + \sum_{(x',y')\in N_i} e^{f(x')^T g(y')}}\right), \quad (23)$$

where x and y denote a video clip and a narration, respectively. f and g are two embedding functions of x and y. Given a specific example (indexed by i), P_i denotes a set of positive video/narration pairs, while N_i is the corresponding set of negative video/narration pairs. [137] also used MIL-NCE (23). Other related works include [138].

2.3.4 Multi-view/multi-modal(ality) learning

Infants can learn about the world through observation. For example, infants can learn the concept of apples via observation and comparison, which differs from supervised algorithms that require large amounts of labeled apple data. This was demonstrated by [12], who collected perceptual data from infants and used an SSL algorithm. That is, babies learn the fruit "apple" through SSL. Furthermore, babies learn about the world via multi-view/multi-modal(ality) learning [115], such as video and audio learning. Therefore, both SSL and multi-view/multi-modal(ality) learning have a natural connection that is used by babies' when learning how the world works.

2.3.4.1 Multiview CL:
The objective function of standard multiview CL [125] is

$$L_{NCE} = E\left[L_q\right], \tag{24}$$

where L_q is in (5), $L_{NCE} + I_{NCE}(v_1, v_2) = \log(K)$, and v_1 and v_2 are two views of the data point x. Tian et al. [125] studied to find good views for CL and proposed both unsupervised view learning and semi-supervised view learning. For an image X, the operation of splitting over its channels can be represented by $\{X_1, X_{2:3}\}$. Let $\hat{X}$ denote $g\left(X\right)$, i.e., $\hat{X} = g\left(X\right)$, where g is a flow-based model. Tian et al. [125] used adversarial training for both unsupervised view learning and semi-supervised view learning. Two encoders f_1 and f_2 were trained to maximize $I_{NCE}(\hat{X}_1, \hat{X}_{2:3})$, as in (24), and g was trained to minimize $I_{NCE}(\hat{X}_1, \hat{X}_{2:3})$. Formally, the objective function presented in [125] for unsupervised view learning is

$$\min_{g} \max_{f_1, f_2} I_{NCE}^{f_1, f_2}(g(X)_1, g(X)_{2:3}). \tag{25}$$

If several labeled examples are available, the objective function of semi-supervised view learning is

$$\begin{aligned} &\min_{g, c_1, c_2} \max_{f_1, f_2} I_{NCE}^{f_1, f_2}(g(X)_1, g(X)_{2:3}) \\ &+L_{ce}\left(c_1\left(g(X)_1\right), y\right) + L_{ce}\left(c_2\left(g(X)_{2:3}\right), y\right), \end{aligned} \tag{26}$$

where y denotes labels, c_1 and c_2 are classifiers, and L_{ce} is the cross-entropy loss. Other related works can be found in [119], [125], [139]–[143]. Different losses of SSL are shown in Table 4.

2.3.4.2 Images and text:
Given an article, [144] projected its text into the topic probability space via a topic modeling framework and used a semantic-level representation as the self-supervised signal for CNN image training. CLIP [115] utilizes a CL-style pretraining task of predicting which caption goes with which image. Empowered by the CL paradigm, CLIP can train models from scratch on a dataset containing 400 million (image, text) pairs collected from the internet. As a result, CLIP has greatly pushed multi-modal learning into the limelight.

2.3.4.3 Point clouds and other modalities:
[145] proposed an SSL method to jointly learn both 3D point cloud features and 2D image features by exploiting cross-modality and cross-view correspondences based on the triplet and cross-entropy losses. [146] proposed jointly learning view-invariant and mode-invariant characteristics from different modalities, including images, point clouds, and meshes, with heterogeneous networks for 3D data. [147] used SSL for a point cloud data set by employing CL and clustering based on a graph CNN. [148], [149] used an AE for point clouds while [150] used capsule networks for point clouds. Other related works can be found in [151]–[155].

2.3.5 Test time training

Test time training (TTT) with self-supervision [156] was proposed as a general method for improving the performance of predictive models if the given training and test data come from different distributions. TTT turns a single unlabeled test example into an SSL problem, on which TTT updates the model parameters before performing prediction.

Recently, [157] combined TTT with the MAE. If TTT can be viewed as a problem of optimizing a model for each test input, the authors argued that such a one-sample learning problem can be solved with the MAE:

$$h_0 = \arg\min_{h} \frac{1}{n} \sum_{i=1}^{n} l_m\left(h \circ f_0\left(x_i\right), y_i\right), \tag{27}$$

$$f_x, g_x = \arg\min_{f,g} l_s(g \circ f(\text{mask}(x)), x), \tag{28}$$

where f, g, and h denote the MAE, decoder, and main task head, respectively.

The difference from the classic paradigm is that at training time, the main task head uses the features obtained from the MAE rather than the original examples. Correspondingly, a single example is used to train f when performing prediction.

Additionally, this paper provides an intuitive explanation of why TTT is helpful. That is, under distribution shifts, TTT finds a better bias-variance tradeoff. A fixed model is entirely reliant on training data that do not accurately reflect the new test distribution; i.e., the model is biased. The opposite extreme is to train a new model from scratch on each test input, forgetting all training data. This is undesirable because each test input is unbiased by definition but has a high variance due to its singularity.

2.3.6 Summary

The development process of SSL is not isolated and static. By examining the combinations of methods, we can understand the development trend of SSL more clearly. A successful example is CLIP, where CL is combined with multi-modal learning with great success.

SSL has also been combined with other machine learning tasks such as clustering [64], [158]–[160], semi-supervised learning [123], hashing [161]–[165], multi-task learning [166]–[171], transfer learning [172]–[174], graph NNs [141], [175]–[185], reinforcement learning [186]–[188], few-shot learning [189], [190], neural architecture search [191], robust learning [124], [192]–[197], and meta learning [198], [199].

3 THEORY

Several papers have conducted research in theoretical directions [200], [201]. Yang et al. [202] empirically and theoretically demonstrated that class-imbalanced learning can significantly benefit both self-supervised and semi-supervised learning.

3.1 Generative algorithms

Many kinds of SSL methods based on generative algorithms are available, including AEs and GANs, which are based on different but related mathematical principles. Here, we introduce the theories related to DAEs and GANs, which are the theoretical bases for the MIM and GANs approaches that are popular in current CV research.

3.1.1 Maximum likelihood estimation (MLE)

Not all generative algorithms utilize MLE. Several generative methods do not use MLE but can be made to do so (GANs fall into this class). Minimizing the Kullback-Leibler (KL) divergence between $p_{data}(x)$ and $p_g(x)$ can be proven to be equal to maximizing the log-likelihood when the number of instances increases. Refer to [31] for the detailed proof.

MLE is generally formulated as $L_{MLE} = -\sum_x log\, p(x|\theta)$, where x denotes all examples to be modeled and θ represents parameters such as the mean and covariance of a Gaussian distribution. Considering its formula, MLE has two serious problems, which are discussed as follows [18].

First, the distribution is conservative and sensitive. When $p(x|\theta) \to 0$, L_{MLE} is very large, making generative algorithms quite sensitive to infrequent instances. This directly results in a conservative distribution, which has an unsatisfactory performance.

The second issue is low-level abstraction. In MLE, the representation distribution is modeled at the example level, i.e., the pointwise level, such as the words in texts, pixels in pictures, and nodes in graphs. However, most classification tasks aim at high-level abstraction, such as face recognition, machine reading comprehension, and cancer prediction.

In contrast, adversarial learning does not use a pointwise objective. It utilizes distributional matching targets that are more robust and better handle the high-level abstraction challenge. Adversarial learning learns to reconstruct the original data distribution rather than the examples by minimizing the distributional divergence.

3.1.2 The original GANs

To learn the generator distribution p_g over data x, a prior is defined on the input noise variables as $p_z(z)$ [126], where z is the noise variable. Then, the generator denotes a mapping from the noise space to the data space as $G(z, \theta_g)$, where G is an NN with parameters θ_g. The other NN, $D(x, \theta_d)$, is also defined with parameters θ_d, but the output of $D(x)$ is a single scalar. $D(x)$ represents the probability that x is from the real data rather than the generator G. The discriminator D is trained to maximize the probability of assigning correct labels to both real training examples and the fake instances generated by the generator G. Simultaneously, G is trained to minimize $\log(1 - D(G(z)))$.

The objective function of GANs [126] is

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[\log D(x)] + E_{z\sim p_z(z)}[\log(1 - D(G(z)))]. \tag{29}$$

One reason for the success of generative algorithms such as GANs in SSL is their capability to recover the original data distribution without hypotheses for downstream tasks, which enables the wide applications of generative methods.

3.1.3 InfoGAN's disentangling ability

What is a disentangled representation? For instance, for a face database, a useful disentangled representation may allot an independent set of dimensions for each attribute: gender, pose, expression, age, and the label of the corresponding person. Disentangled representations can be helpful for natural tasks that demand knowledge concerning the important features of the input data, which include assignments such as object recognition and face recognition.

An important shortcoming of supervised learning is that it is easily trapped in fake information. A famous instance is that supervised algorithms learn to classify wolves and dogs by whether they are located in snow or grass [203]. This means that the supervised algorithms do not learn the disentangled representations of dogs and grass, which should be mutually independent.

In contrast, InfoGAN [204] is capable of learning disentangled representations in a completely unsupervised manner.

3.1.4 DAE

As the most well-known DAE method, BERT masks a portion of the tokens in the input and replaces the masked position with a special [mask] token. Algorithms based on DAE follow a similar intuition, where the input is usually corrupted, and the algorithm is trained in a way that is robust to corruption.

The DAE [3] is executed by first corrupting the original input X to obtain a partially destroyed version $\widetilde{X}$ by means of a stochastic mapping $q_{\mathcal{D}}(\widetilde{X} \mid X)$. The corrupted input $\widetilde{X}$ is then mapped to a hidden representation $\mathbf{y} = f_\theta(\tilde{\mathbf{x}})$ from which DAE reconstructs a $\mathbf{z} = g_{\theta'}(\mathbf{y})$. The NN is trained to minimize the average reconstruction error $L_{\mathbf{H}}(\mathrm{x}, \mathrm{z}) = \mathbb{H}(\mathcal{B}_{\mathbf{x}} \| \mathcal{B}_{\mathbf{z}})$.

The DAE defines the joint distribution

$$q^0(X, \widetilde{X}, Y) = q^0(X) q_{\mathcal{D}}(\widetilde{X} \mid X) \delta_{f_\theta(\widetilde{X})}(Y), \tag{30}$$

where $\delta_u(v)$ selects a mass of 0 when $u \neq v$. Thus, Y is a deterministic function of $\widetilde{X} \cdot q^0(X, \widetilde{X}, Y)$ parameterized by θ. The objective function minimized by stochastic gradient descent becomes

$$\arg\min_{\theta,\theta'} \mathbb{E}_{q^0(X,\widetilde{X})}\left[L_{\mathbf{H}}\left(X, g_{\theta'}\left(f_\theta(\widetilde{X})\right)\right)\right]. \tag{31}$$

From an information-theoretic perspective, the denoising self-encoder maximizes a lower bound imposed on the mutual information of the reconstructed target and the original example. As a result, the NN mines the rich intrinsic structural information while reconstructing the original examples, which benefits large models with high capacities.

The corresponding theoretical analysis of MLE, GANs, InfoGAN, and DAE may inspire that of SSL.

3.2 Contrastive SSL

Although contrastive SSL has achieved great success, the reason why it works remains somewhat unclear and poorly understood. Several papers worked toward this research direction. [205]–[207] theoretically proved that the feature representations learned through CL are helpful for downstream tasks.

3.2.1 Connection to other unsupervised learning algorithms

3.2.1.1 Connection to principal component analysis:

Tian [208] demonstrated that CL under a set of loss functions such as InfoNCE has a max-min formula, in which the

max function seeks feature representations for maximizing contrast, and the min function imposes weights on pairs of examples with similar representations. Tian [208] demonstrated that the max function that conducts representation learning is equivalent to principal component analysis (PCA) for a deep linear network, and nearly all local minima are global, restoring the optimal PCA solutions. Experiments indicated that this formula has comparable or better performance on STL-10 and CIFAR10 when extending beyond InfoNCE, yielding new contrastive losses. In addition, Tian [208] extended his theoretical analysis to 2-layer rectified linear unit (ReLU) networks, demonstrating the significant distinction between linear and nonlinear situations and the roles that data augmentation plays during the training process.

As we know, the covariance matrix S_t of PCA is

$$S_t = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^T \tag{32}$$

where n, x_i, and $\bar{x}$ are the number of examples, the i-th example denoted as a column vector, and the mean example, respectively. We have the following theorem for PCA.

Theorem 1. The covariance matrix of S_t of PCA is equivalent to $\frac{1}{2n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(x_i - x_j)(x_i - x_j)^T$.

Proof 1. Since

$$\sum_i \bar{x}x_i^\top = n\bar{x}\bar{x}^\top,$$
$$\sum_i x_i\bar{x}^\top = n\bar{x}\bar{x}^\top,$$

we have

$$\begin{aligned}
S_t &= \frac{1}{n}\left(\sum_{i=1}^{n} x_i x_i^\top - 2\sum_{i=1}^{n} x_i\bar{x}^\top + n\bar{x}\bar{x}^\top\right)\\
&= \frac{1}{n}\left(\sum_{i=1}^{n} x_i x_i^\top - n\bar{x}\bar{x}^\top\right)\\
&= \frac{1}{2n^2}\left(2n\sum_{i=1}^{n} x_i x_i^\top - 2\sum_{i=1}^{n} x_i\sum_{j=1}^{n} x_j^\top\right)\\
&= \frac{1}{2n^2}\left(n\sum_{i=1}^{n} x_i x_i^\top + n\sum_{j=1}^{n} x_j x_j^\top - 2\sum_{i=1}^{n} x_i\sum_{j=1}^{n} x_j^\top\right)\\
&= \frac{1}{2n^2}\sum_{i=1}^{n}(x_i - x_j)(x_i - x_j)^\top.
\end{aligned}$$

This completes the proof.

Therefore, PCA aims to maximize the distance between every pair of examples in the low-dimensional subspace, and PCA may be seen as a special case of instance discrimination.

3.2.1.2 Connection to spectral clustering:
Chen et al. [209] showed that representations obtained through CL are spectral clustering embeddings of a positive pair graph. Generally, they defined the population augmentation graph. The nodes of the graph are the augmented data in the population distribution, and the existence of an edge between two nodes is determined by whether they are augmentations of the same original example. Their key assumption is that different underlying classes are connected by very few edges. Therefore, the sparsest partition for such a graph is large. This characteristic results from the continuity of the population data inside the same class, which has already been empirically demonstrated [210].

The authors formed a matrix obtained by performing spectral decomposition on the adjacency matrix, and they interpreted each row of the matrix as the representation of an example. With some linear transformation, they showed that the corresponding feature extractor could be recovered by minimizing the following atypical contrastive loss:

$$\begin{aligned}
&\mathcal{L}(f) = -2\cdot\mathbb{E}_{x,x^+}\left[f(x)^\top f(x^+)\right]\\
&+\mathbb{E}_{x,x'}\left[\left(f(x)^\top f(x')\right)^2\right].
\end{aligned} \tag{33}$$

Furthermore, under the condition that the dimensionality of the representation is larger than the maximum number of disconnected subgraphs, linear classification with learned representations is guaranteed to induce a small error. The researchers further discussed their ideas under the assumption of limited data.

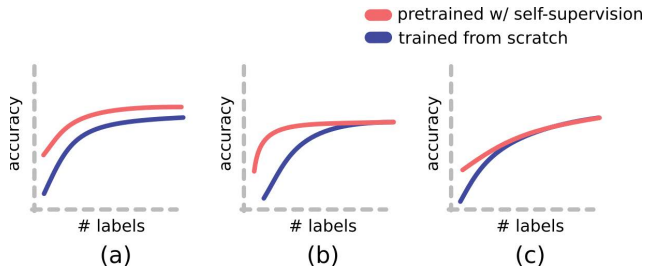

Fig. 9: Three probable hypotheses about conducting supervised training from scratch vs. self-supervised pretraining. This figure is directly taken from [117].

3.2.2 Connection to supervised learning

It has been shown that CL-based self-supervised pretraining is especially effective when the downstream task is classification but not as effective when downstream tasks are other types. It is interesting how contrastive pretraining is good for supervised learning, especially concerning whether SSL can learn more than supervised learning, at least in terms of accuracy.

Newell et al. [117] checked the three probable hypotheses in Fig. 9 stating that pretraining: (a) always improves, (b) has higher accuracy when fewer labels are available but plateaus at the same accuracy as that of the baseline, and (c) converges to the baseline performance before its accuracy plateaus. Experiments were conducted on the synthetic COCO dataset with rendering, which could supply as many labels as possible, and the authors found that self-supervised pretraining follows the assumption in (c), showing that SSL cannot learn more than supervised learning but can succeed with fewer labels.

3.2.3 Connection to metric learning

[211] supplied analyses by connecting InfoNCE to the triplet (k-plet) loss in the deep learning field. The InfoNCE

loss can be rewritten as

$$\begin{aligned} &L_{NCE} = E[\frac{1}{K}\sum_{i=1}^{K} log\frac{e^{f(x_i,\ y_i)}}{\frac{1}{K}\sum_{j=1}^{K} e^{f(x_i,\ y_j)}}] \\ &= logK \\ &-E\left[\frac{1}{K}\sum_{i=1}^{K} log\left(1+\sum_{j\neq i} e^{f(x_i,\ y_j)-f(x_i,\ y_i)}\right)\right]. \end{aligned} \tag{34}$$

Note that (34) has only a constant difference from (24). To be consistent with the original paper [211], we use (34). In the special situation where x and y have values in the same space and f is constrained to be $f(x,y) = \phi(x)^T \phi(y)$, for a certain function ϕ, this is the same as the expectation of the multi-class K-pair loss (up to a sign change and a constant difference), as shown in (7) in [212]:

$$\begin{aligned} &L_{K-pair-mc}\left(\{(x_i,y_i)\}_{i=1}^{K},\phi\right) \\ &= \frac{1}{K}\sum_{i=1}^{K} log\left(1+\sum_{j\neq i} e^{\phi(x_i)^T\phi(y_j)-\phi(x_i)^T\phi(y_i)}\right). \end{aligned} \tag{35}$$

Performing representation learning by maximizing L_{NCE} utilizing a symmetric separable function $f(x,y) = \phi(x)^T \phi(y)$ and an encoder $g = g_1 = g_2$ shared across views is therefore equal to conducting metric learning according to (35).

Chen et al. [213] provided a new perspective on the relationship between CL and metric learning. They argued that metric learning can be viewed as a distance polarization algorithm when we can access supervised information. Therefore, they developed a corresponding self-supervised version. By encouraging distance polarization during contrastive SSL, the structure of the representation space becomes more differentiated.

3.2.4 Understanding the contrastive loss from alignment and uniformity perspectives

The contrastive loss aims to make learned feature representations for positive pairs similar while pushing features derived from negative pairs far from each other. Conventional wisdom says that learned feature representations should extract the most shared information between positive pairs and remain unchanged with respect to other noise factors [119]. Thus, the loss should prefer the following two characteristics.

- Alignment: two examples in a positive pair ought to be mapped to close features and therefore be (mostly) unchanged relative to unneeded noise factors.
- Uniformity: features should preserve as much of the information contained in the data as possible and be roughly uniformly distributed on the unit hypersphere.

In [214], interesting theoretical analyses were conducted on the contrastive loss function, and it was split into two terms. From (5) and (24), we have

$$\begin{aligned} &L_{contrastive} = E\left[-\log\frac{\exp(q\cdot k_+/\tau)}{\sum_{i=0}^{K}\exp(q\cdot k_i/\tau)}\right] \\ &= E\left[-\log\frac{\exp(q\cdot k_+/\tau)}{\exp(q\cdot k_+/\tau)+\sum_{i=1}^{K}\exp(q\cdot k_-^i/\tau)}\right] \\ &= E[-q\cdot k_+/\tau] \\ &+E[\log(\exp(q\cdot k_+/\tau)+\sum_{i=1}^{K}\exp(q\cdot k_-^i/\tau))], \end{aligned} \tag{36}$$

where k_-^i is the ith negative example, the first term aims at "alignment", and the second term aims at "uniformity".

For further analysis, the authors proposed the alignment and uniformity losses as follows.

$$L_{align}(f;\alpha) = \underset{(x,y)\sim p_{pos}}{\mathbb{E}}[||f(x)-f(y)||_2^{\alpha}], \alpha > 0. \tag{37}$$

$$L_{uniform}(f;t) = \log \underset{(x,y)\sim p_{data}}{\mathbb{E}}[e^{-t||f(x)-f(y)||_2^2}]. \tag{38}$$

However, it is unclear whether alignment and uniformity should be in the formula of the above two losses because BYOL and SimSiam have no negative examples but achieve good performances. This inspires us to attain uniformity through other methods, such as regularization, the exponential moving average, batch normalization, and the stop-gradient operation.

3.2.5 The relationship between the contrastive loss and mutual information

Mutual information is a fundamental concept in statistics. Now, consider the joint distribution between two groups of variables x and y given by $p(x,y)$ [29]. If the groups of variables are independent, their joint distribution is the product of their marginal distributions $p(x,y) = p(x)p(y)$. If the variables are not independent, we can obtain some idea of whether they are 'close' to being independent by considering the KL divergence between the joint distribution and the product of their marginal distributions, which is given by

$$\begin{aligned} &I[x,y] = KL(p(x,y) \parallel p(x)p(y)) \\ &= -\iint p(x,y)\, ln(\frac{p(x)p(y)}{p(x,y)})dxdy. \end{aligned} \tag{39}$$

From (24) and [59], we have

$$\begin{aligned} &L_{contrastive} = E\left[-\log\frac{\exp(q\cdot k_+/\tau)}{\sum_{i=0}^{K}\exp(q\cdot k_i/\tau)}\right] \\ &\geq -I(q,k_+) + \log(K+1). \end{aligned} \tag{40}$$

That is, the negative mutual information $-I(q,k_+)$ plus $\log(K+1)$ is the lower bound of the contrastive loss.

3.3 Complete collapse and dimensional collapse

SSL algorithms learn helpful feature representations by minimizing the distances between the feature representations produced from augmented views (Fig. 10 a). This yields a collapsed solution where the learned feature representations are constant (Fig. 10 b). Contrastive algorithms prevent complete collapse through a negative example term that pushes the embedding feature vectors of different inputs away from each other. Tian et al. [215] showed why algorithms without negative examples such as SimSiam [65] and BYOL [63] work: the dynamics of the eigenspace alignment between the predictor and its input correlation matrix play a vital role in preventing complete collapse.

Jing et al. [87] showed that although CL-based algorithms prevent complete collapse, they still experience dimensional collapse where the embedding feature representations lie in a lower-dimensional subspace than their dimension (Fig. 10 c). [87] supplied theoretical analyses of this case and demonstrated that two mechanisms, including implicit regularization and strong augmentation, cause dimensional collapse. Inspired by their theory, a new CL-based SSL algorithm named DirectCLR was proposed to

directly optimize the feature representation space without depending on a trainable projector. DirectCLR performs better than SimCLR on ImageNet with a linear projector.

For more theoretical analyses and details such as those concerning generalization, refer to [18].

4 Applications

SSL was first introduced for vowel class recognition [216] and then was further extended for object extraction [217]. SSL has been widely used in many areas, such as NLP, speech [218], image processing and CV.

4.1 Image processing and CV

[219] proposed a fully convolutional volumetric AE to learn deep embeddings of object shapes in an unsupervised way. Furthermore, SSL has been applied to many areas of image processing and CV, such as image inpainting [100], human parsing [220], [221], scene deocclusion [222], semantic image segmentation [223]–[230], monocular vision [231]–[233], person reidentification (re-ID) [234]–[236], visual odometry [237], [238], scene flow estimation [239], knowledge distillation [240], optical flow prediction [241], vision-language navigation (VLN) [242], faces [243]–[246], physiological signal estimation [247], [248], image denoising [249], [250], object detection [251]–[253], super-resolution [254], [255], voxel prediction from 2D images [256], ego-motion [257]–[261], and mask prediction [262].

4.1.1 SSL models for videos

SSL has been widely used for applications such as video representation learning [263]–[267] and video retrieval [268].

Utilizing the HowTo100M dataset [269], [136] proposed MIL-NCE, which inherits qualities from MIL and NCE, to learn the video representations in narrated videos.

Wang et al. [270] utilized hundreds of thousands of unlabeled videos from the web to learn visual representations. The key idea is that visual tracking provides a self-supervised signal. That is, two patches connected by a track ought to have similar visual representations because they are probably the same object or belong to the same object part.

In [271], two self-supervised models (a long short-term memory (LSTM) AE model and an LSTM-based future prediction model) were combined to create a composite model that could both reconstruct the input and predict the future.

We can use the following four kinds of information in videos[1] to define a proxy loss: the temporal information in videos, motions of objects such as optical flows, multi-modal data (e.g., RGB, audio, and narration data), and the spatial-temporal coherence of objects such as colors and shapes.

The first three kinds of information belong to paradigm 1, which tests transferability. First, the proxy task is trained to obtain a representation. In the downstream task, fine-tuning is used to test whether this representation can be effectively used to solve the downstream task.

The last kind of information belongs to paradigm 2, in which the representation learned from the proxy task is directly used in the downstream task without fine-tuning.

1. https://www.bilibili.com/video/BV1N5411j76F

4.1.1.1 Temporal information in videos:

Several kinds of temporal information in videos can be used, such as the order of the frames, the video playing direction, the video playing speed, and future prediction information [272], [273].

- The order of the frames. [274] presented a method for learning a visual representation from the raw spatiotemporal signals in videos and determining whether a sequence of frames obtained from a video is in the correct temporal order. Fernando et al. [275] proposed a novel self-supervised CNN pretraining method based on a new auxiliary task called odd-one-out learning, where the aim is to identify the unrelated or odd element in a set of related elements. The unrelated or odd element denotes an odd video subsequence sampled such that it has the wrong temporal frame order while the related elements have the correct temporal order. Lee et al. [276] used temporally shuffled frames (i.e., in non-chronological order) as inputs and trained a CNN to predict the order of the shuffled sequences. That is, [276] used temporal coherence as a self-supervised signal. Based on [276], Xu et al. [277] utilized temporally shuffled clips instead of frames as inputs and trained 3D CNNs to sort the shuffled clips.
- Video playing direction. [278] learned to see the arrow of time to tell whether a video sequence was playing forward or backward.
- Video playing speed. [279] predicted the speeds of moving objects in videos (whether they moved faster or slower than normal speed). Yao et al. [280] used playback rates and their corresponding video content as self-supervision signals for video representation learning. [281] addressed the problem of self-supervised video representation learning from a video pace prediction perspective.

4.1.1.2 Motions of objects in videos:

[282] was interested in the SSL of the motions in videos via the utilization of dynamic motion filters for producing a better motion representation to boost human action recognition in particular. The idea of SSL with videos (CoCLR) [137] is similar to that of SimCLR [56]. Other related works include [283]–[289].

4.1.1.3 Multi-modal(ality) data in videos:

A natural connection exists between the auditory and visual elements of a video. [290] used this correlation to learn general and effective models for both video and audio analysis from the self-supervised temporal synchronization perspective. Other methods [291]–[295] are also based on both video and audio modalities. [296]–[300] used both video and text modalities. [301] leveraged three modalities in videos: vision, audio, and language. [302] proposed a self-supervised method for learning representations and robotic behaviors from unlabeled videos recorded from different viewpoints. Other related works can be found in [303]–[307].

4.1.1.4 Spatial-temporal coherence of objects in videos:

Wang et al. [308] introduced a self-supervised algorithm for learning visual correspondence from unlabeled videos. The main idea is to use the cycle consistency in time as a self-

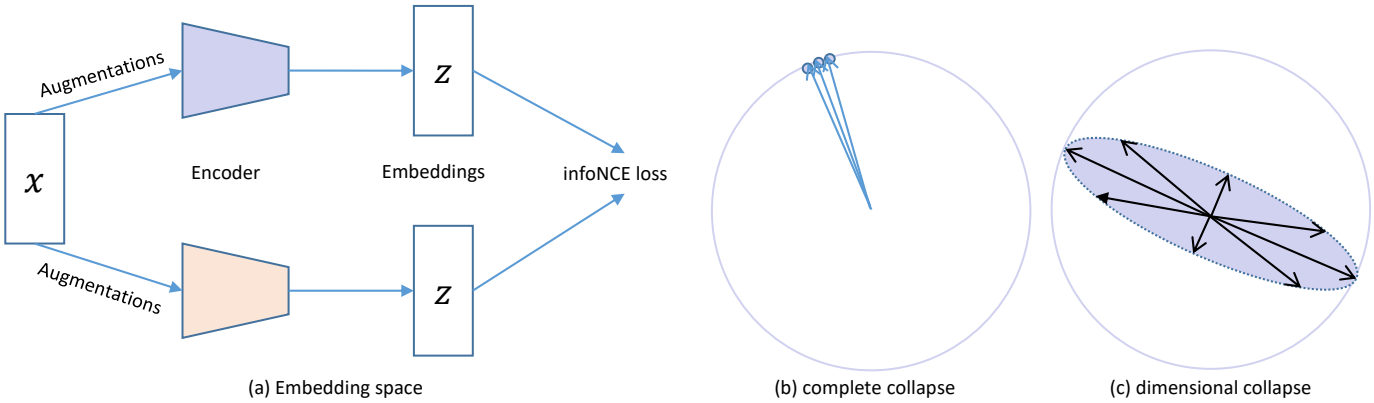

Fig. 10: Demonstration of the collapsing problem. For complete collapse, the embedding feature representations collapse to the same data. For dimensional collapse, the embedding vectors only span a lower-dimensional subspace rather than the whole available embedding space.

supervised signal for learning visual representations. Extensions of [308] include [309], [310]. Lai et al. [311] presented a memory-augmented self-supervised method that enables generalizable and accurate pixel-level tracking. Zhang et al. [312] used the spatial-temporal consistency of depth maps to mitigate forgetting during the learning process. Zhao et al. [313] proposed a novel self-supervised algorithm, named the video cloze procedure (VCP), to learn rich spatial-temporal representations for videos. Other related works include [314]–[319].

4.1.2 Universal sequential SSL models for image processing and CV

The key idea of contrastive predictive coding (CPC) [59] is to learn useful representations by predicting the future in the latent space by using powerful autoregressive models. CPC has been used for not only sequential data such as speech and text but also images [320].

Inspired by the success of GPT [321]–[323] methods developed for natural languages, iGPT [101] examines whether similar models can learn useful representations for images. iGPT [101] explores two training objectives, autoregressive prediction and a denoising objective, making it similar to BERT [324]–[326]. In high-resolution settings, this approach [101] is competitive with other self-supervised methods on ImageNet. Similar to iGPT, the ViT [99], [327] also uses a transformer for vision purposes. The ViT applies a pure transformer to sequences of image patches; it has shown that it can perform very well on image classification tasks. The transformer architecture has been applied to many vision tasks, which can be found in [55], [66], [69], [96], [328]–[330].

4.2 NLP

In NLP, with regard to performing SSL on word embeddings, the continuous bag-of-words (CBOW) model and the continuous skip-gram (SG) model [331] are pioneering works. SSL methods such as BERT [324]–[326] and GPT have been widely used in NLP [134], [332]–[337]. Refer to Hung-yi Lee's machine learning course for more materials on SSL for NLP and speech [2]. SSL has also been used for other sequential data, such as sound data [338].

4.3 Other fields

In the medical field [339], generally, the number of labeled data is limited, and a large number of unlabeled data are available. Therefore, SSL can be used naturally. SSL has been used in the medical field for tasks such as medical image segmentation [340] and 3D medical image analysis [341].

SSL has also been used in many other areas such as bug detection and repair [342].

5 Three Main Trends

First, the theoretical analysis of SSL still lags behind the development of its applications. For example, many studies have been conducted on why BYOL and SimSiam [65] do not collapse [215]. More theoretical studies should be investigated to find the essential reason for this phenomenon, and perhaps a better solution can be found. Additionally, recent research has shown that MIM-based methods are on par with or better than conventional CL-based methods in terms of performance. However, few works have attempted to theoretically analyze the differences between the model representations of these two popular paradigms.

Second, given a fixed downstream task, how can we automatically design an optimal pretext task to make the downstream task have excellent performance? A number of studies have produced various methods, including the pixel-to-propagation consistency method (PixPro) [343] and dense CL (DenseCL) [344]. However, this issue has still not been effectively addressed. More theoretical studies are also needed in this interesting area.

Third, a unified SSL paradigm for multiple modalities is needed. MIM has made great strides in vision, similar to MLM, which has been successful in NLP, demonstrating the possibility of learning paradigm unification. On the other hand, the ViT smooths the gap between the visual and verbal modalities in terms of architecture. A unified transformer model can be built for both CV and NLP tasks.

2. https://speech.ee.ntu.edu.tw/~hylee/ml/2022-spring.php

Recently, studies [345], [346] have moved toward unifying SSL models. They have achieved amazing performance on downstream tasks and have a wide range of applications. NLP is ahead of CV concerning how to use SSL models. CV can take inspiration from the NLP community to address the issue of making better use of pretrained models. How to better unlock the potential of SSL pretraining in CV is a promising research direction.

6 Open questions

First, can SSL benefit from almost unlimited data? We have a large amount of unlabeled data. Can SSL always benefit from more unlabeled data, and can we theoretically find the inflection point?

Second, what is the relationship between SSL and multi-modality learning? Both SSL and multi-modality learning are similar to the learning process of human babies. How can we efficiently combine them to produce a strong learning model?

Third, which SSL algorithm is the best/should I use? There is no standard answer to this question. Since the number of available labeled instances is limited, SSL methods make strong model assumptions. Ideally, one ought to utilize an algorithm whose assumptions fit the structure of the problem that is being examined. This may be difficult in practice. Can we give the user a checklist to determine which method performs the best in certain conditions? This is also a thought-provoking direction for future study.

Fourth, do unlabeled data always help? We believe that the answer is no, as is true for semi-supervised learning methods. There is no free lunch theorem. Poor matching between the model assumption with the problem structure can cause performance degradation.

For instance, a model may assume that the decision boundary should be far away from regions with high densities. However, if the data are actually derived from two heavily overlapping Cauchy distributions, the decision boundary will go right through the densest area, and these algorithms would perform badly.

However, finding a bad match beforehand is difficult and still an open question. This is an issue worth studying.

7 Performance comparison

After we obtain a pretrained model from SSL, we need to evaluate its performance. The canonic solution utilizes the performance achieved on downstream tasks to determine the corresponding feature quality. However, this evaluation metric does not indicate what the network learned via self-supervised pretraining. More evaluation metrics such as network dissection [347] can be used to analyze the interpretability of self-supervised features.

Recently, many MIM methods have emerged, focusing on different aspects compared to those of previous methods. In this section, to clearly demonstrate the performance of different methods, we summarize the classification and transfer learning performance of typical SSL methods on popular datasets. Note that SSL techniques can theoretically be used on data with arbitrary modalities, but here, to simplify the problem, we focus on SSL in the image domain. For SSL in image domain, we compare the performance achieved on several downstream tasks, which mainly include image classification, object detection, and semantic segmentation.

7.1 Comprehensive comparison

In this section, we present the results produced by all the tested algorithms on the corresponding datasets in Table 5. All experimental results are derived from the corresponding original papers or from other papers with annotations. If the experimental results are obtained from an original paper, we do not mark them; otherwise, we mark the data source. If the accuracy of a method reproduced from another work is higher than that achieved in the original paper, we always report the result with better accuracy. Note that to compare a wide range of algorithms, the experimental setup is not strictly aligned. We align the important hyperparameters as much as possible, while the hyperparameters such as the number of training epochs are not necessarily aligned. All experimental results are obtained using the default backbone of the original paper, such as ResNet-50 or ViT-B. For experimental results that do not have corresponding ResNet-50 or ViT-B implementations, we report the results obtained based on other backbones and mark them with subscripts.

Setup. All pretraining is performed on ImageNet-1k. After that, as shown in Table 5, following a common protocol [53] [56], we first compare the above methods by linearly classifying frozen features. Under this setting, based on the features derived from the frozen model, a linear classifier (a fully connected layer followed by the softmax function) is trained. "Finetuning" stands for fine-tuning the entire network. All results represent the top-1 classification accuracies achieved on the ImageNet validation set.

For the object detection and semantic segmentation tasks, we report the results obtained on the widely adopted datasets including PASCAL VOC, COCO, and ADE20k. The metric used for evaluating the object detection results obtained on the PASCAL VOC dataset is the default mean average precision (mAP), i.e., AP_{50}. The object detection task on PASCAL VOC is trained by default using only VOC2007, and the results are annotated for the models using 07+12 data as their training set. The object detection and instance segmentation tasks on COCO uses the bounding-box AP (AP_{bb}) and mask AP (AP_{mk}) as metrics, following [53].

7.2 Summary

First, the accuracy of the linear probe of the CL-based self-supervised algorithm is generally higher than that of the other algorithms. This is because they produce easily measurable and structurally ordered latent spaces, where different categories are separated, and the same categories are clustered together.

Second, pretrained models based on MIM can be fine-tuned to achieve better performance in most cases, whereas pretrained models based on CL do not have this property. This is usually because pretrained models based on CL are more prone to overfitting [348]–[350]. The same phenomenon occurs when fine-tuning pretrained models

for downstream tasks, where MIM-based methods usually significantly improve the performance achieved in downstream tasks, while CL-based methods tend not to help much with downstream tasks.

Third, MIM-based algorithms generally require less computational resources than CL-based methods. CL methods without memory banks require many computational resources because they rely on large batch sizes. Although algorithms based on MIM also tend to use large batch sizes, they do not have to maintain large computational graphs as in CL without a memory bank because they usually lack interexample information interaction. Algorithms based on MIM can use gradient accumulation to save computational resources and facilitate parallel computing, which is beneficial for large models, especially given the increasing importance of large models [351].

8 Conclusions

This paper provides a survey of various aspects of SSL by elaborating on several perspectives, i.e., its algorithms, theory, applications, three main trends, and open questions. We believe that this review will help readers gain an understanding of the existing research on SSL. To conclude, we would like to note that to maintain an appropriate size for the paper, we had to limit the number of references. We therefore apologize to the authors of papers that were not cited.

Acknowledgments

The authors would like to thank Weidi Xie's invited talk at Valse. The authors would also like to thank the members of the Umich Yelab and all my students for their helpful discussions. This work was supported in part by a grant from the National Science Foundation of China (number 62172090), the CAAI-Huawei MindSpore Open Fund, and the Alibaba Group through the Alibaba Innovative Research Program. We thank the Big Data Computing Center of Southeast University for providing the facility support on the numerical calculations in this paper.

References

[1] Y. M. Asano, C. Rupprecht, and A. Vedaldi, "A critical analysis of self-supervision, or what we can learn from a single image," in *International Conference on Learning Representations*, 2020.

[2] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[3] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *International Conference on Machine Learning*, pp. 1096–1103, 2008.

[4] P. Isola, D. Zoran, D. Krishnan, and E. H. Adelson, "Learning visual groups from co-occurrences in space and time," *arXiv preprint arXiv:1511.06811*, 2015.

[5] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *International Conference on Robotics and Automation*, pp. 3406–3413, 2016.

[6] Y. Li, M. Paluri, J. M. Rehg, and P. Dollár, "Unsupervised learning of edges," in *Conference on Computer Vision and Pattern Recognition*, pp. 1619–1627, 2016.

[7] D. Li, W.-C. Hung, J.-B. Huang, S. Wang, N. Ahuja, and M.-H. Yang, "Unsupervised visual representation learning by graph-based consistent constraints," in *European Conference on Computer Vision*, pp. 678–694, 2016.

[8] B. Brattoli, U. Buchler, A.-S. Wahl, M. E. Schwab, and B. Ommer, "Lstm self-supervision for detailed behavior analysis," in *Conference on Computer Vision and Pattern Recognition*, pp. 6466–6475, 2017.

[9] O. Halimi, O. Litany, E. Rodolà, A. Bronstein, and R. Kimmel, "Self-supervised learning of dense shape correspondence," *arXiv preprint arXiv:1812.02415*, 2018.

[10] H. Lee, S. J. Hwang, and J. Shin, "Rethinking data augmentation: Self-supervision and self-distillation," *arXiv preprint arXiv:1910.05872*, 2019.

[11] B. Zoph, G. Ghiasi, T.-Y. Lin, Y. Cui, H. Liu, E. D. Cubuk, and Q. Le, "Rethinking pre-training and self-training," in *Neural Information Processing Systems*, pp. 1–13, 2020.

[12] A. E. Orhan, V. V. Gupta, and B. M. Lake, "Self-supervised learning through the eyes of a child," in *Neural Information Processing Systems*, pp. 9960–9971, 2020.

[13] J. Mitrovic, B. McWilliams, J. Walker, L. Buesing, and C. Blundell, "Representation learning via invariant causal mechanisms," in *International Conference on Learning Representations*, pp. 1–19, 2021.

[14] L. Ericsson, H. Gouk, and T. M. Hospedales, "How well do self-supervised models transfer?," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5414–5423, 2021.

[15] A. Tejankar, S. A. Koohpayegani, V. Pillai, P. Favaro, and H. Pirsiavash, "Isd: Self-supervised learning by iterative similarity distillation," in *International Conference on Computer Vision*, pp. 9609–9618, 2021.

[16] G. Wang, K. Wang, G. Wang, P. H. Torr, and L. Lin, "Solving inefficiency of self-supervised representation learning," in *International Conference on Computer Vision*, pp. 9505–9515, 2021.

[17] S. A. Koohpayegani, A. Tejankar, and H. Pirsiavash, "Mean shift for self-supervised learning," in *International Conference on Computer Vision*, pp. 10326–10325, 2021.

[18] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[19] T. Hua, W. Wang, Z. Xue, S. Ren, Y. Wang, and H. Zhao, "On feature decorrelation in self-supervised learning," in *International Conference on Computer Vision*, pp. 9598–9608, 2021.

[20] VentureBeat, "Yann LeCun, Yoshua Bengio: Self-supervised learning is key to human-level intelligence." https://cacm.acm.org/news/244720-yann-lecun-yoshua-bengio-self-supervised-learning-is-key-to-human-level-intelligence/fulltext.

[21] J. Yu, H. Yin, X. Xia, T. Chen, J. Li, and Z. Huang, "Self-supervised learning for recommender systems: A survey," *arXiv preprint arXiv:2203.15876*, 2022.

[22] Y. Liu, M. Jin, S. Pan, C. Zhou, Y. Zheng, F. Xia, and P. Yu, "Graph self-supervised learning: A survey," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[23] L. Wu, H. Lin, C. Tan, Z. Gao, and S. Z. Li, "Self-supervised learning on graphs: Contrastive, generative, or predictive," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[24] H. H. Mao, "A survey on self-supervised pre-training for sequential transfer learning in neural networks," *arXiv preprint arXiv:2007.00800*, 2020.

[25] M. C. Schiappa, Y. S. Rawat, and M. Shah, "Self-supervised learning for videos: A survey," *arXiv preprint arXiv:2207.00419*, 2022.

[26] G.-J. Qi and M. Shah, "Adversarial pretraining of self-supervised deep networks: Past, present and future," *arXiv preprint arXiv:2210.13463*, 2022.

[27] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 4037–4058, 2021.

[28] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, p. 2, 2020.

[29] C. M. Bishop, *Pattern recognition and machine learning*. Springer, New York, 2006.

[30] V. R. de Sa, "Learning classification with unlabeled data," in *Neural Information Processing Systems*, pp. 112–119, 1994.

[31] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, "A review on generative adversarial networks: Algorithms, theory, and applications," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

Methods	Linear Probe	Fine-Tuning	VOC_det	VOC_seg	COCO_det	COCO_seg	ADE20K_seg	DB
Random:	17.1_A [40]	-	60.2^e_R [65]	19.8_A [40]	36.7_R [53]	33.7_R [53]	-	-
R50 Sup	76.5 [64]	76.5 [64]	81.3^e [65]	74.4 [63]	40.6 [53]	36.8 [53]	-	-
ViT-B Sup	82.3 [66]	82.3 [66]	-	-	47.9 [66]	42.9 [66]	47.4 [66]	-
Context-Based:								
Jigsaw [40]	45.7_R [64]	54.7	61.4_R [39]	37.6	-	-	-	256
Colorization [35]	39.6_R [64]	40.7 [33]	46.9	35.6	-	-	-	-
Rotation [33]	38.7	50.0	54.4	39.1	-	-	-	128
CL Based on Negative Examples:								
Examplar [106]	31.5 [50]	-	-	-	-	-	-	-
Instdisc [50]	54.0	-	65.4	-	-	-	-	256
MoCo v1 [53]	60.6	-	74.9	-	40.8	36.9	-	256
SimCLR [56]	73.9_V [55]	-	81.8^e [65]	-	37.9 [65]	33.3 [65]	-	4096
MoCo v2 [54]	72.2 [65]	-	82.5^e	-	39.8 [68]	36.1 [68]	-	256
MoCo v3 [55]	76.7	83.2	-	-	47.9 [66]	42.7 [66]	47.3 [66]	4096
CL Based on Clustering:								
SwAV [64]	75.3	-	82.6^e [68]	-	41.6	37.8 [68]	-	4096
CL Based on Self-distillation:								
BYOL [63]	74.3	-	81.4^e [65]	76.3	40.4 [68]	37.0 [68]	-	4096
SimSiam [65]	71.3	-	82.4^e [65]	-	39.2	34.4	-	512
DINO [69]	78.2	83.6 [95]	-	-	46.8 [97]	41.5 [97]	44.1 [96]	1024
CL Based on Feature Decorrelation:								
Barlow Twins [67]	73.2	-	82.6^e [68]	-	39.2	34.3	-	2048
VICReg [68]	73.2	-	82.4^e	-	39.4	36.4	-	2048
Masked Image Modeling (ViT-B by default):								
Context Encoder [100]	21.0_A [33]	-	44.5_A [33]	30.0_A	-	-	-	-
BEiT v1 [96]	56.7 [352]	83.4 [95]	-	-	49.8 [66]	44.4 [66]	47.1 [66]	2000
MAE [66]	67.8	83.6	-	-	50.3	44.9	48.1	4096
SimMIM [98]	56.7	83.8	-	-	52.3_{Swin-B} [353]	-	52.8_{Swin-B} [353]	2048
PeCo [354]	-	84.5	-	-	43.9	39.8	46.7	2048
iBOT [95]	79.5	84.0	-	-	51.2	44.2	50.0	1024
MimCo [355]	-	83.9	-	-	44.9	40.7	48.91	2048
CAE [97]	70.4	83.9	-	-	50	44	50.2	2048
data2vec [346]	-	84.2	-	-	-	-	-	2048
SdAE [356]	64.9	84.1	-	-	48.9	43.0	48.6	768
BEiT v2 [352]	80.1	85.5	-	-	-	-	53.1	2048

TABLE 5: Experimental results of all the tested algorithms with respect to linear classification and transfer learning tasks. DB denotes the default batch size. The notation '-' means that the data point does not exist or is not found in the relevant paper. The subscripts A, R and V represent AlexNet, ResNet-50 and ViT-B respectively. The superscript 'e' denotes the usage of extra data, i.e., VOC2012.

[32] T. Nathan Mundhenk, D. Ho, and B. Y. Chen, "Improvements to context based self-supervised learning," in *Conference on Computer Vision and Pattern Recognition*, pp. 9339–9348, 2018.

[33] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *International Conference on Learning Representations*, pp. 1–14, 2018.

[34] P. Agrawal, J. Carreira, and J. Malik, "Learning to see by moving," in *International Conference on Computer Vision*, pp. 37–45, 2015.

[35] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *European Conference on Computer Vision*, pp. 649–666, 2016.

[36] G. Larsson, M. Maire, and G. Shakhnarovich, "Learning representations for automatic colorization," in *European Conference on Computer Vision*, pp. 577–593, 2016.

[37] R. Zhang, J.-Y. Zhu, P. Isola, X. Geng, A. S. Lin, T. Yu, and A. A. Efros, "Real-time user-guided image colorization with learned deep priors," *arXiv preprint arXiv:1705.02999*, 2017.

[38] G. Larsson, M. Maire, and G. Shakhnarovich, "Colorization as a proxy task for visual understanding," in *Conference on Computer Vision and Pattern Recognition*, pp. 6874–6883, 2017.

[39] P. Goyal, D. Mahajan, A. Gupta, and I. Misra, "Scaling and benchmarking self-supervised visual representation learning," in *International Conference on Computer Vision*, pp. 6391–6400, 2019.

[40] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *European Conference on Computer Vision*, pp. 69–84, 2016.

[41] U. Ahsan, R. Madhok, and I. Essa, "Video jigsaw: Unsupervised learning of spatiotemporal context for video action recognition," in *Winter Conference on Applications of Computer Vision*, pp. 179–189, 2019.

[42] C. Wei, L. Xie, X. Ren, Y. Xia, C. Su, J. Liu, Q. Tian, and A. L. Yuille, "Iterative reorganization with weak spatial constraints: Solving arbitrary jigsaw puzzles for unsupervised representation learning," in *Conference on Computer Vision and Pattern Recognition*, pp. 1910–1919, 2019.

[43] D. Kim, D. Cho, D. Yoo, and I. S. Kweon, "Learning image representations by completing damaged jigsaw puzzles," in *Winter Conference on Applications of Computer Vision*, pp. 793–802, 2018.

[44] R. Santa Cruz, B. Fernando, A. Cherian, and S. Gould, "Deeppermnet: Visual permutation learning," in *Conference on Computer Vision and Pattern Recognition*, pp. 3949–3957, 2017.

[45] R. Santa Cruz, B. Fernando, A. Cherian, and S. Gould, "Visual permutation learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 12, pp. 3100–3114, 2018.

[46] X. Zhan, X. Pan, Z. Liu, D. Lin, and C. C. Loy, "Self-supervised learning via conditional motion propagation," in *Conference on Computer Vision and Pattern Recognition*, pp. 1881–1889, 2019.

[47] K. Wang, L. Lin, C. Jiang, C. Qian, and P. Wei, "3d human pose machines with self-supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 5, pp. 1069–1082, 2019.

[48] M. Noroozi, H. Pirsiavash, and P. Favaro, "Representation learning by learning to count," in *International Conference on Computer Vision*, pp. 5898–5906, 2017.

[49] I. Misra and L. v. d. Maaten, "Self-supervised learning of pretext-invariant representations," in *Conference on Computer Vision and Pattern Recognition*, pp. 6707–6717, 2020.

[50] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Conference on Computer Vision and Pattern Recognition*, pp. 3733–3742, 2018.

[51] N. Zhao, Z. Wu, R. W. Lau, and S. Lin, "What makes instance discrimination good for transfer learning?," in *International Conference on Learning Representations*, pp. 1–11, 2021.

[52] Y. Cao, Z. Xie, B. Liu, Y. Lin, Z. Zhang, and H. Hu, "Parametric

instance classification for unsupervised visual feature learning," *Neural Information Processing Systems*, pp. 1–11, 2020.
[53] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.
[54] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," *arXiv preprint arXiv:2003.04297*, 2020.
[55] X. Chen, S. Xie, and K. He, "An empirical study of training self-supervised visual transformers," in *International Conference on Computer Vision*, pp. 9640–9649, 2021.
[56] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International Conference on Machine Learning*, pp. 1597–1607, 2020.
[57] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. Hinton, "Big self-supervised models are strong semi-supervised learners," in *Neural Information Processing Systems*, pp. 1–13, 2020.
[58] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Conference on Computer Vision and Pattern Recognition*, pp. 1735–1742, 2006.
[59] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2019.
[60] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *International Conference on Artificial Intelligence and Statistics*, pp. 297–304, 2010.
[61] M. Zheng, S. You, F. Wang, C. Qian, C. Zhang, X. Wang, and C. Xu, "Ressl: Relational self-supervised learning with weak augmentation," *arXiv preprint arXiv:2107.09282*, 2021.
[62] N. Zhao, Z. Wu, R. W. Lau, and S. Lin, "Distilling localization for self-supervised representation learning," in *AAAI Conference on Artificial Intelligence*, pp. 10990–10998, 2021.
[63] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, *et al.*, "Bootstrap your own latent: A new approach to self-supervised learning," in *Neural Information Processing Systems*, pp. 1–14, 2020.
[64] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," in *Neural Information Processing Systems*, 2020.
[65] X. Chen and K. He, "Exploring simple siamese representation learning," in *Conference on Computer Vision and Pattern Recognition*, pp. 15750–15758, 2021.
[66] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Conference on Computer Vision and Pattern Recognition*, pp. 16000–16009, 2022.
[67] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow twins: Self-supervised learning via redundancy reduction," in *International Conference on Machine Learning*, 2021.
[68] A. Bardes, J. Ponce, and Y. LeCun, "Vicreg: Variance-invariance-covariance regularization for self-supervised learning," in *International Conference on Learning Representations*, pp. 1–12, 2022.
[69] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *International Conference on Computer Vision*, pp. 9650–9660, 2021.
[70] Y. Wang, X. Shen, S. X. Hu, Y. Yuan, J. L. Crowley, and D. Vaufreydaz, "Self-supervised transformers for unsupervised object discovery using normalized cut," in *Conference on Computer Vision and Pattern Recognition*, pp. 14543–14553, 2022.
[71] E. Hoffer, I. Hubara, and N. Ailon, "Deep unsupervised learning through spatial contrasting," *arXiv preprint arXiv:1610.00243*, 2016.
[72] M. Yang, M. Liao, P. Lu, J. Wang, S. Zhu, H. Luo, Q. Tian, and X. Bai, "Reading and writing: Discriminative and generative modeling for self-supervised text recognition," *arXiv preprint arXiv:2207.00193*, 2022.
[73] H. Wu, Y. Qu, S. Lin, J. Zhou, R. Qiao, Z. Zhang, Y. Xie, and L. Ma, "Contrastive learning for compact single image dehazing," in *Conference on Computer Vision and Pattern Recognition*, pp. 10551–10560, 2021.
[74] R. Zhu, B. Zhao, J. Liu, Z. Sun, and C. W. Chen, "Improving contrastive learning by visualizing feature transformation," in *International Conference on Computer Vision*, pp. 10306–10315, 2021.
[75] M. Yang, Y. Li, Z. Huang, Z. Liu, P. Hu, and X. Peng, "Partially view-aligned representation learning with noise-robust contrastive loss," in *Conference on Computer Vision and Pattern Recognition*, pp. 1134–1143, 2021.
[76] L. Xiong, C. Xiong, Y. Li, K.-F. Tang, J. Liu, P. Bennett, J. Ahmed, and A. Overwijk, "Approximate nearest neighbor negative contrastive learning for dense text retrieval," in *International Conference on Learning Representations*, pp. 1–12, 2021.
[77] J. Li, P. Zhou, C. Xiong, and S. C. Hoi, "Prototypical contrastive learning of unsupervised representations," in *International Conference on Learning Representations*, pp. 1–12, 2021.
[78] K. Kotar, G. Ilharco, L. Schmidt, K. Ehsani, and R. Mottaghi, "Contrasting contrastive self-supervised representation learning pipelines," in *International Conference on Computer Vision*, pp. 9949–9959, 2021.
[79] S. Liu, H. Fan, S. Qian, Y. Chen, W. Ding, and Z. Wang, "Hit: Hierarchical transformer with momentum contrast for video-text retrieval," in *International Conference on Computer Vision*, pp. 11915–11925, 2021.
[80] A. Islam, C.-F. Chen, R. Panda, L. Karlinsky, R. Radke, and R. Feris, "A broad study on the transferability of visual representations with contrastive learning," in *International Conference on Computer Vision*, pp. 8845–8855, 2021.
[81] J. Li, C. Xiong, and S. C. Hoi, "Learning from noisy data with robust representation learning," in *International Conference on Computer Vision*, pp. 9485–9494, 2021.
[82] H. Cha, J. Lee, and J. Shin, "Co2l: Contrastive continual learning," in *International Conference on Computer Vision*, pp. 9516–9525, 2021.
[83] O. J. Hénaff, S. Koppula, J.-B. Alayrac, A. v. d. Oord, O. Vinyals, and J. Carreira, "Efficient visual pretraining with contrastive detection," in *International Conference on Computer Vision*, pp. 10086–10096, 2021.
[84] D. Dwibedi, Y. Aytar, J. Tompson, P. Sermanet, and A. Zisserman, "With a little help from my friends: Nearest-neighbor contrastive learning of visual representations," in *International Conference on Computer Vision*, pp. 9588–9597, 2021.
[85] J. Cui, Z. Zhong, S. Liu, B. Yu, and J. Jia, "Parametric contrastive learning," in *International Conference on Computer Vision*, pp. 715–724, 2021.
[86] A. Shah, S. Sra, R. Chellappa, and A. Cherian, "Max-margin contrastive learning," in *AAAI Conference on Artificial Intelligence*, 2022.
[87] L. Jing, P. Vincent, Y. LeCun, and Y. Tian, "Understanding dimensional collapse in contrastive self-supervised learning," in *International Conference on Learning Representations*, pp. 1–11, 2022.
[88] J. Zhang, X. Xu, F. Shen, Y. Yao, J. Shao, and X. Zhu, "Video representation learning with graph contrastive augmentation," in *ACM International Conference on Multimedia*, pp. 3043–3051, 2021.
[89] S. Lal, M. Prabhudesai, I. Mediratta, A. W. Harley, and K. Fragkiadaki, "Coconets: Continuous contrastive 3d scene representations," in *Conference on Computer Vision and Pattern Recognition*, 2021.
[90] Q. Hu, X. Wang, W. Hu, and G.-J. Qi, "Adco: Adversarial contrast for efficient learning of unsupervised representations from self-trained negative adversaries," in *Conference on Computer Vision and Pattern Recognition*, 2021.
[91] Y. Kalantidis, M. B. Sariyildiz, N. Pion, P. Weinzaepfel, and D. Larlus, "Hard negative mixing for contrastive learning," in *Neural Information Processing Systems*, pp. 1–12, 2020.
[92] G. Bukchin, E. Schwartz, K. Saenko, O. Shahar, R. Feris, R. Giryes, and L. Karlinsky, "Fine-grained angular contrastive learning with coarse labels," in *Conference on Computer Vision and Pattern Recognition*, 2021.
[93] S. Purushwalkam and A. Gupta, "Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases," in *Neural Information Processing Systems*, pp. 1–12, 2020.
[94] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," in *Neural Information Processing Systems*, pp. 18661–18673, 2020.
[95] J. Zhou, C. Wei, H. Wang, W. Shen, C. Xie, A. Yuille, and T. Kong, "ibot: Image bert pre-training with online tokenizer," in *International Conference on Learning Representations*, pp. 1–12, 2022.
[96] H. Bao, L. Dong, S. Piao, and F. Wei, "Beit: Bert pre-training of image transformers," in *International Conference on Learning Representations*, pp. 1–13, 2022.

[97] X. Chen, M. Ding, X. Wang, Y. Xin, S. Mo, Y. Wang, S. Han, P. Luo, G. Zeng, and J. Wang, "Context autoencoder for self-supervised representation learning," *arXiv preprint arXiv:2202.03026*, 2022.

[98] Z. Xie, Z. Zhang, Y. Cao, Y. Lin, J. Bao, Z. Yao, Q. Dai, and H. Hu, "Simmim: A simple framework for masked image modeling," in *Conference on Computer Vision and Pattern Recognition*, pp. 9653–9663, 2022.

[99] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021.

[100] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Conference on Computer Vision and Pattern Recognition*, pp. 2536–2544, 2016.

[101] M. Chen, A. Radford, R. Child, J. Wu, H. Jun, P. Dhariwal, D. Luan, and I. Sutskever, "Generative pretraining from pixels," in *International Conference on Machine Learning*, pp. 1691–1703, 2020.

[102] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, "Zero-shot text-to-image generation," in *International Conference on Machine Learning*, pp. 8821–8831, 2021.

[103] C. Feichtenhofer, H. Fan, Y. Li, and K. He, "Masked autoencoders as spatiotemporal learners," *arXiv preprint arXiv:2205.09113*, 2022.

[104] Z. Tong, Y. Song, J. Wang, and L. Wang, "Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training," *arXiv preprint arXiv:2203.12602*, 2022.

[105] Z. Liu, J. Gui, and H. Luo, "Good helper is around you: Attention-driven masked image modeling," in *AAAI Conference on Artificial Intelligence*, 2023.

[106] A. Dosovitskiy, J. T. Springenberg, M. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with convolutional neural networks," in *Neural Information Processing Systems*, pp. 766–774, 2014.

[107] A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with exemplar convolutional neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 9, pp. 1734–1747, 2015.

[108] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *International Conference on Computer Vision*, pp. 1422–1430, 2015.

[109] P. Bojanowski and A. Joulin, "Unsupervised learning by predicting noise," in *International Conference on Machine Learning*, 2017.

[110] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *International Conference on Machine Learning*, pp. 478–487, 2016.

[111] J. Yang, D. Parikh, and D. Batra, "Joint unsupervised learning of deep representations and image clusters," in *Conference on Computer Vision and Pattern Recognition*, pp. 5147–5156, 2016.

[112] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *European Conference on Computer Vision*, pp. 132–149, 2018.

[113] R. Zhang, P. Isola, and A. A. Efros, "Split-brain autoencoders: Unsupervised learning by cross-channel prediction," in *Conference on Computer Vision and Pattern Recognition*, pp. 1058–1067, 2017.

[114] X. Wang, K. He, and A. Gupta, "Transitive invariance for self-supervised visual representation learning," in *International Conference on Computer Vision*, pp. 1329–1338, 2017.

[115] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*, pp. 8748–8763, 2021.

[116] A. Kolesnikov, X. Zhai, and L. Beyer, "Revisiting self-supervised visual representation learning," in *Conference on Computer Vision and Pattern Recognition*, pp. 1920–1929, 2019.

[117] A. Newell and J. Deng, "How useful is self-supervised pretraining for visual tasks?," in *Conference on Computer Vision and Pattern Recognition*, pp. 7345–7354, 2020.

[118] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *International Conference on Learning Representations*, pp. 1–14, 2014.

[119] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *European Conference on Computer Vision*, pp. 776–794, 2020.

[120] P. Bachman, R. D. Hjelm, and W. Buchwalter, "Learning representations by maximizing mutual information across views," in *Neural Information Processing Systems*, pp. 15535–15545, 2019.

[121] P. Krähenbühl, "Free supervision from video games," in *Conference on Computer Vision and Pattern Recognition*, pp. 2955–2964, 2018.

[122] T. Chen, X. Zhai, M. Ritter, M. Lucic, and N. Houlsby, "Self-supervised gans via auxiliary rotation loss," in *Conference on Computer Vision and Pattern Recognition*, pp. 12154–12163, 2019.

[123] X. Zhai, A. Oliver, A. Kolesnikov, and L. Beyer, "S4l: Self-supervised semi-supervised learning," in *International Conference on Computer Vision*, pp. 1476–1485, 2019.

[124] D. Hendrycks, M. Mazeika, S. Kadavath, and D. Song, "Using self-supervised learning can improve model robustness and uncertainty," in *Neural Information Processing Systems*, pp. 15663–15674, 2019.

[125] Y. Tian, C. Sun, B. Poole, D. Krishnan, C. Schmid, and P. Isola, "What makes for good views for contrastive learning," in *Neural Information Processing Systems*, pp. 1–13, 2020.

[126] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Neural Information Processing Systems*, pp. 2672–2680, 2014.

[127] L. Chen, Y. Zhang, Y. Song, L. Liu, and J. Wang, "Self-supervised learning of adversarial example: Towards good generalizations for deepfake detection," in *Conference on Computer Vision and Pattern Recognition*, 2022.

[128] M. Mathieu, C. Couprie, and Y. LeCun, "Deep multi-scale video prediction beyond mean square error," in *International Conference on Learning Representations*, pp. 1–14, 2016.

[129] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," in *International Conference on Learning Representations*, pp. 1–18, 2017.

[130] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling," in *Neural Information Processing Systems*, pp. 82–90, 2016.

[131] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning representations and generative models for 3d point clouds," in *International Conference on Machine Learning*, pp. 40–49, 2018.

[132] S. Jenni and P. Favaro, "Self-supervised feature learning by learning to spot artifacts," in *Conference on Computer Vision and Pattern Recognition*, pp. 2733–2742, 2018.

[133] J. Donahue and K. Simonyan, "Large scale adversarial representation learning," in *Neural Information Processing Systems*, 2019.

[134] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "Electra: Pre-training text encoders as discriminators rather than generators," in *International Conference on Learning Representations*, 2020.

[135] X. Zeng, Y. Pan, M. Wang, J. Zhang, and Y. Liu, "Realistic face reenactment via self-supervised disentangling of identity and pose," in *AAAI Conference on Artificial Intelligence*, pp. 12154–12163, 2020.

[136] A. Miech, J.-B. Alayrac, L. Smaira, I. Laptev, J. Sivic, and A. Zisserman, "End-to-end learning of visual representations from uncurated instructional videos," in *Conference on Computer Vision and Pattern Recognition*, pp. 9879–9889, 2020.

[137] T. Han, W. Xie, and A. Zisserman, "Self-supervised co-training for video representation learning," in *Neural information processing systems*, pp. 1–12, 2020.

[138] K. Chen, L. Hong, H. Xu, Z. Li, and D.-Y. Yeung, "Multisiam: Self-supervised multi-instance siamese representation learning for autonomous driving," in *International Conference on Computer Vision*, pp. 7546–7554, 2021.

[139] Y. Yu and W. Smith, "Outdoor inverse rendering from a single image using multiview self-supervision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3659–3675, 2022.

[140] Y. Lin, Y. Gou, Z. Liu, B. Li, J. Lv, and X. Peng, "Completer: Incomplete multi-view clustering via contrastive prediction," in *Conference on Computer Vision and Pattern Recognition*, pp. 11174–11183, 2021.

[141] K. Hassani and A. H. Khasahmadi, "Contrastive multi-view representation learning on graphs," in *International Conference on Machine Learning*, 2020.

[142] M. P. Vo, E. Yumer, K. Sunkavalli, S. Hadap, Y. A. Sheikh, and S. G. Narasimhan, "Self-supervised multi-view person associa-

tion and its applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 8, pp. 2794–2808, 2021.
[143] H. Xu, Z. Zhou, Y. Qiao, W. Kang, and Q. Wu, "Self-supervised multi-view stereo via effective co-segmentation and data-augmentation," in *AAAI Conference on Artificial Intelligence*, 2021.
[144] L. Gomez, Y. Patel, M. Rusiñol, D. Karatzas, and C. Jawahar, "Self-supervised learning of visual features through embedding images into text topic spaces," in *Conference on Computer Vision and Pattern Recognition*, pp. 4230–4239, 2017.
[145] L. Jing, Y. Chen, L. Zhang, M. He, and Y. Tian, "Self-supervised feature learning by cross-modality and cross-view correspondences," *arXiv preprint arXiv:2004.05749*, 2020.
[146] L. Jing, Y. Chen, L. Zhang, M. He, and Y. Tian, "Self-supervised modal and view invariant feature learning," *arXiv preprint arXiv:2005.14169*, 2020.
[147] L. Zhang and Z. Zhu, "Unsupervised feature learning for point cloud understanding by contrasting and clustering using graph convolutional neural networks," in *International Conference on 3D Vision*, pp. 395–404, 2019.
[148] Y. Yang, C. Feng, Y. Shen, and D. Tian, "Foldingnet: Point cloud auto-encoder via deep grid deformation," in *Conference on Computer Vision and Pattern Recognition*, pp. 206–215, 2018.
[149] M. Gadelha, R. Wang, and S. Maji, "Multiresolution tree networks for 3d point cloud processing," in *European Conference on Computer Vision*, pp. 103–118, 2018.
[150] Y. Zhao, T. Birdal, H. Deng, and F. Tombari, "3d point capsule networks," in *Conference on Computer Vision and Pattern Recognition*, pp. 1009–1018, 2019.
[151] X. Yu, Y. Rao, Z. Wang, Z. Liu, J. Lu, and J. Zhou, "Pointr: Diverse point cloud completion with geometry-aware transformers," in *International Conference on Computer Vision*, pp. 12498–12507, 2021.
[152] S. A. Baur, D. J. Emmerichs, F. Moosmann, P. Pinggera, B. Ommer, and A. Geiger, "Slim: Self-supervised lidar scene flow and motion segmentation," in *International Conference on Computer Vision*, pp. 13126–13136, 2021.
[153] Y. Chen, J. Liu, B. Ni, H. Wang, J. Yang, N. Liu, T. Li, and Q. Tian, "Shape self-correction for unsupervised point cloud understanding," in *International Conference on Computer Vision*, pp. 8382–8391, 2021.
[154] J. Sun, Y. Cao, C. Choy, Z. Yu, C. Xiao, A. Anandkumar, and Z. M. Mao, "Adversarially robust 3d point cloud recognition using self-supervisions," in *Neural Information Processing Systems*, 2021.
[155] Y. Rao, J. Lu, and J. Zhou, "Global-local bidirectional reasoning for unsupervised representation learning of 3d point clouds," in *Conference on Computer Vision and Pattern Recognition*, pp. 5376–5385, 2020.
[156] Y. Sun, X. Wang, Z. Liu, J. Miller, A. A. Efros, and M. Hardt, "Test-time training with self-supervision for generalization under distribution shifts," in *International Conference on Machine Learning*, 2020.
[157] Y. Gandelsman, Y. Sun, X. Chen, and A. A. Efros, "Test-time training with masked autoencoders," *arXiv preprint arXiv:2209.07522*, 2022.
[158] C. Zhuang, A. L. Zhai, and D. Yamins, "Local aggregation for unsupervised learning of visual embeddings," in *International Conference on Computer Vision*, pp. 6002–6012, 2019.
[159] X. Yan, I. Misra, A. Gupta, D. Ghadiyaram, and D. Mahajan, "Clusterfit: Improving generalization of visual representations," in *Conference on Computer Vision and Pattern Recognition*, pp. 6509–6518, 2020.
[160] X. Zhan, J. Xie, Z. Liu, Y.-S. Ong, and C. C. Loy, "Online deep clustering for unsupervised representation learning," in *Conference on Computer Vision and Pattern Recognition*, pp. 6688–6697, 2020.
[161] Y. Li, X. Wang, S. Qi, C. Huang, Z. L. Jiang, Q. Liao, J. Guan, and J. Zhang, "Self-supervised learning-based weight adaptive hashing for fast cross-modal retrieval," *Signal, Image and Video Processing*, pp. 1–8, 2019.
[162] H. Zhang, M. Wang, R. Hong, and T.-S. Chua, "Play and rewind: Optimizing binary representations of videos by self-supervised temporal hashing," in *ACM International Conference on Multimedia*, pp. 781–790, 2016.
[163] C. Li, C. Deng, N. Li, W. Liu, X. Gao, and D. Tao, "Self-supervised adversarial hashing networks for cross-modal retrieval," in *Conference on Computer Vision and Pattern Recognition*, pp. 4242–4251, 2018.
[164] J. Song, T. He, H. Fan, and L. Gao, "Deep discrete hashing with self-supervised pairwise labels," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 223–238, 2017.
[165] J. Song, H. Zhang, X. Li, L. Gao, M. Wang, and R. Hong, "Self-supervised video hashing with hierarchical binary auto-encoder," *IEEE Transactions on Image Processing*, vol. 27, no. 7, pp. 3210–3221, 2018.
[166] J. J. Sun, A. Kennedy, E. Zhan, D. J. Anderson, Y. Yue, and P. Perona, "Task programming: Learning data efficient behavior representations," in *Conference on Computer Vision and Pattern Recognition*, pp. 2876–2885, 2021.
[167] C. Doersch and A. Zisserman, "Multi-task self-supervised visual learning," in *International Conference on Computer Vision*, pp. 2051–2060, 2017.
[168] Z. Ren and Y. Jae Lee, "Cross-domain self-supervised multi-task feature learning using synthetic imagery," in *Conference on Computer Vision and Pattern Recognition*, pp. 762–771, 2018.
[169] K. Hassani and M. Haley, "Unsupervised multi-task feature learning on point clouds," in *International Conference on Computer Vision*, pp. 8160–8171, 2019.
[170] A. Piergiovanni, A. Angelova, and M. S. Ryoo, "Evolving losses for unlabeled video representation learning," *arXiv preprint arXiv:1906.03248*, 2019.
[171] A. Piergiovanni, A. Angelova, and M. S. Ryoo, "Evolving losses for unsupervised video representation learning," in *Conference on Computer Vision and Pattern Recognition*, pp. 133–142, 2020.
[172] K. Saito, D. Kim, S. Sclaroff, and K. Saenko, "Universal domain adaptation through self supervision," in *Neural Information Processing Systems*, pp. 1–11, 2020.
[173] Y. Sun, E. Tzeng, T. Darrell, and A. A. Efros, "Unsupervised domain adaptation through self-supervision," *arXiv preprint arXiv:1909.11825*, 2019.
[174] M. Noroozi, A. Vinjimoor, P. Favaro, and H. Pirsiavash, "Boosting self-supervised learning via knowledge transfer," in *Conference on Computer Vision and Pattern Recognition*, pp. 9359–9367, 2018.
[175] X. Wang, N. Liu, H. Han, and C. Shi, "Self-supervised heterogeneous graph neural network with co-contrastive learning," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 1726–1736, 2021.
[176] H. Zhang, Q. Wu, J. Yan, D. Wipf, and P. S. Yu, "From canonical correlation analysis to self-supervised graph neural networks," in *Neural Information Processing Systems*, pp. 1–14, 2021.
[177] K. Sun, Z. Zhu, and Z. Lin, "Multi-stage self-supervised learning for graph convolutional networks," *arXiv preprint arXiv:1902.11038*, 2019.
[178] W. Hu, B. Liu, J. Gomes, M. Zitnik, P. Liang, V. Pande, and J. Leskovec, "Strategies for pre-training graph neural networks," *arXiv preprint arXiv:1905.12265*, 2019.
[179] Z. Peng, Y. Dong, M. Luo, X.-M. Wu, and Q. Zheng, "Self-supervised graph representation learning via global context prediction," *arXiv preprint arXiv:2003.01604*, 2020.
[180] Y. You, T. Chen, Z. Wang, and Y. Shen, "When does self-supervision help graph convolutional networks?," *arXiv preprint arXiv:2006.09136*, 2020.
[181] J. Qiu, Q. Chen, Y. Dong, J. Zhang, H. Yang, M. Ding, K. Wang, and J. Tang, "Gcc: Graph contrastive coding for graph neural network pre-training," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1150–1160, 2020.
[182] Z. Hu, Y. Dong, K. Wang, K.-W. Chang, and Y. Sun, "Gpt-gnn: Generative pre-training of graph neural networks," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1857–1867, 2020.
[183] Y. Rong, Y. Bian, T. Xu, W. Xie, Y. Wei, W. Huang, and J. Huang, "Self-supervised graph transformer on large-scale molecular data," in *Neural Information Processing Systems*, 2020.
[184] Y. Zhu, Y. Xu, F. Yu, Q. Liu, S. Wu, and L. Wang, "Deep graph contrastive representation learning," *arXiv preprint arXiv:2006.04131*, 2020.
[185] Y. Zhu, Y. Xu, F. Yu, Q. Liu, S. Wu, and L. Wang, "Graph contrastive learning with adaptive augmentation," in *International World Wide Web Conference*, pp. 2069–2080, 2021.
[186] U. Buchler, B. Brattoli, and B. Ommer, "Improving spatiotemporal self-supervision by deep reinforcement learning," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 770–786, 2018.

[187] D. Guo, B. A. Pires, B. Piot, J.-b. Grill, F. Altché, R. Munos, and M. G. Azar, "Bootstrap latent-predictive representations for multitask reinforcement learning," *arXiv preprint arXiv:2004.14646*, 2020.

[188] N. Hansen, Y. Sun, P. Abbeel, A. A. Efros, L. Pinto, and X. Wang, "Self-supervised policy adaptation during deployment," *arXiv preprint arXiv:2007.04309*, 2020.

[189] S. Gidaris, A. Bursuc, N. Komodakis, P. Pérez, and M. Cord, "Boosting few-shot visual learning with self-supervision," in *International Conference on Computer Vision*, pp. 8059–8068, 2019.

[190] J.-C. Su, S. Maji, and B. Hariharan, "Boosting supervision with self-supervision for few-shot learning," *arXiv preprint arXiv:1906.07079*, 2019.

[191] C. Li, T. Tang, G. Wang, J. Peng, B. Wang, X. Liang, and X. Chang, "Bossnas: Exploring hybrid cnn-transformers with block-wisely self-supervised neural architecture search," in *International Conference on Computer Vision*, 2021.

[192] D. Stutz, M. Hein, and B. Schiele, "Relating adversarially robust generalization to flat minima," in *International Conference on Computer Vision*, pp. 7807–7817, 2021.

[193] L. Fan, S. Liu, P.-Y. Chen, G. Zhang, and C. Gan, "When does contrastive learning preserve adversarial robustness from pretraining to finetuning?," in *Neural Information Processing Systems*, 2021.

[194] Z. Jiang, T. Chen, T. Chen, and Z. Wang, "Robust pre-training by adversarial contrastive learning," in *Neural Information Processing Systems*, pp. 1–12, 2020.

[195] M. Kim, J. Tack, and S. J. Hwang, "Adversarial self-supervised contrastive learning," in *Neural Information Processing Systems*, pp. 1–12, 2020.

[196] S. Gowal, P.-S. Huang, A. van den Oord, T. Mann, and P. Kohli, "Self-supervised adversarial robustness for the low-label, high-data regime," in *International Conference on Learning Representations*, pp. 1–19, 2021.

[197] T. Chen, S. Liu, S. Chang, Y. Cheng, L. Amini, and Z. Wang, "Adversarial robustness: From self-supervised pre-training to fine-tuning," in *Conference on Computer Vision and Pattern Recognition*, pp. 699–708, 2020.

[198] Y. Lin, X. Guo, and Y. Lu, "Self-supervised video representation learning with meta-contrastive network," in *International Conference on Computer Vision*, pp. 8239–8249, 2021.

[199] Y. An, H. Xue, X. Zhao, and L. Zhang, "Conditional self-supervised learning for few-shot classification," in *International Joint Conference on Artificial Intelligence*, pp. 2140–2146, 2021.

[200] Y.-H. H. Tsai, Y. Wu, R. Salakhutdinov, and L.-P. Morency, "Self-supervised learning from a multi-view perspective," *arXiv preprint arXiv:2006.05576*, 2020.

[201] C.-Y. Chuang, J. Robinson, L. Yen-Chen, A. Torralba, and S. Jegelka, "Debiased contrastive learning," in *International Conference on Learning Representations*, 2020.

[202] Y. Yang and Z. Xu, "Rethinking the value of labels for improving class-imbalanced learning," in *Neural Information Processing Systems*, 2020.

[203] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144, 2016.

[204] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," in *Neural Information Processing Systems*, pp. 2172–2180, 2016.

[205] N. Saunshi, O. Plevrakis, S. Arora, M. Khodak, and H. Khandeparkar, "A theoretical analysis of contrastive unsupervised representation learning," in *International Conference on Machine Learning*, pp. 5628–5637, 2019.

[206] J. D. Lee, Q. Lei, N. Saunshi, and J. Zhuo, "Predicting what you already know helps: Provable self-supervised learning," *arXiv preprint arXiv:2008.01064*, 2020.

[207] C. Tosh, A. Krishnamurthy, and D. Hsu, "Contrastive learning, multi-view redundancy, and linear models," in *Algorithmic Learning Theory*, pp. 1179–1206, 2021.

[208] Y. Tian, "Deep contrastive learning is provably (almost) principal component analysis," *arXiv preprint arXiv:2201.12680*, 2022.

[209] J. Z. HaoChen, C. Wei, A. Gaidon, and T. Ma, "Provable guarantees for self-supervised deep learning with spectral contrastive loss," in *Neural Information Processing Systems*, pp. 5000–5011, 2021.

[210] C. Wei, K. Shen, Y. Chen, and T. Ma, "Theoretical analysis of self-training with deep networks on unlabeled data," in *International Conference on Learning Representations*, pp. 1–15, 2021.

[211] M. Tschannen, J. Djolonga, P. K. Rubenstein, S. Gelly, and M. Lucic, "On mutual information maximization for representation learning," in *International Conference on Learning Representations*, pp. 1–12, 2020.

[212] K. Sohn, "Improved deep metric learning with multi-class n-pair loss objective," in *Neural Information Processing Systems*, pp. 1849–1857, 2016.

[213] S. Chen, G. Niu, C. Gong, J. Li, J. Yang, and M. Sugiyama, "Large-margin contrastive learning with distance polarization regularizer," in *International Conference on Machine Learning*, pp. 1673–1683, 2021.

[214] T. Wang and P. Isola, "Understanding contrastive representation learning through alignment and uniformity on the hypersphere," in *International Conference on Machine Learning*, pp. 9929–9939, 2020.

[215] Y. Tian, X. Chen, and S. Ganguli, "Understanding self-supervised learning dynamics without contrastive pairs," in *International Conference on Machine Learning*, pp. 10268–10278, 2021.

[216] S. Pal, A. Datta, and D. D. Majumder, "Computer recognition of vowel sounds using a self-supervised learning algorithm," *Journal of the Anatomical Society of India*, pp. 117–123, 1978.

[217] A. Ghosh, N. R. Pal, and S. K. Pal, "Self-organization for object extraction using a multilayer neural network and fuzziness mearsures," *IEEE Transactions on Fuzzy Systems*, pp. 54–68, 1993.

[218] S.-w. Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin, *et al.*, "Superb: Speech processing universal performance benchmark," in *Interspeech*, pp. 1–6, 2021.

[219] A. Sharma, O. Grau, and M. Fritz, "Vconv-dae: Deep volumetric shape learning without object labels," in *European Conference on Computer Vision*, pp. 236–250, 2016.

[220] K. Gong, X. Liang, D. Zhang, X. Shen, and L. Lin, "Look into person: Self-supervised structure-sensitive learning and a new benchmark for human parsing," in *Conference on Computer Vision and Pattern Recognition*, pp. 932–940, 2017.

[221] X. Liang, K. Gong, X. Shen, and L. Lin, "Look into person: Joint body parsing & pose estimation network and a new benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 4, pp. 871–885, 2018.

[222] X. Zhan, X. Pan, B. Dai, Z. Liu, D. Lin, and C. C. Loy, "Self-supervised scene de-occlusion," in *Conference on Computer Vision and Pattern Recognition*, pp. 3784–3792, 2020.

[223] N. Araslanov and S. Roth, "Self-supervised augmentation consistency for adapting semantic segmentation," in *Conference on Computer Vision and Pattern Recognition*, pp. 15384–15394, 2021.

[224] W. Van Gansbeke, S. Vandenhende, S. Georgoulis, and L. Van Gool, "Unsupervised semantic segmentation by contrasting object mask proposals," in *International Conference on Computer Vision*, pp. 10052–10062, 2021.

[225] I. Alonso, A. Sabater, D. Ferstl, L. Montesano, and A. C. Murillo, "Semi-supervised semantic segmentation with pixel-level contrastive learning from a class-wise memory bank," in *International Conference on Computer Vision*, pp. 8219–8228, 2021.

[226] H. Hu, J. Cui, and L. Wang, "Region-aware contrastive learning for semantic segmentation," in *International Conference on Computer Vision*, pp. 16291–16301, 2021.

[227] V. Guizilini, J. Li, R. Ambrus, and A. Gaidon, "Geometric unsupervised domain adaptation for semantic segmentation," in *International Conference on Computer Vision*, pp. 8537–8547, 2021.

[228] D. Pathak, R. Girshick, P. Dollár, T. Darrell, and B. Hariharan, "Learning features by watching objects move," in *Conference on Computer Vision and Pattern Recognition*, pp. 2701–2710, 2017.

[229] X. Zhan, Z. Liu, P. Luo, X. Tang, and C. C. Loy, "Mix-and-match tuning for self-supervised semantic segmentation," in *AAAI Conference on Artificial Intelligence*, pp. 7534–7541, 2018.

[230] Y. Wang, J. Zhang, M. Kan, S. Shan, and X. Chen, "Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation," in *Conference on Computer Vision and Pattern Recognition*, pp. 12275–12284, 2020.

[231] Z. Chen, X. Ye, L. Du, W. Yang, L. Huang, X. Tan, Z. Shi, F. Shen, and E. Ding, "Aggnet for self-supervised monocular depth estimation: Go an aggressive step furthe," in *ACM International Conference on Multimedia*, pp. 1526–1534, 2021.

[232] Z. Chen, X. Ye, W. Yang, Z. Xu, X. Tan, Z. Zou, E. Ding, X. Zhang, and L. Huang, "Revealing the reciprocal relations between self-supervised stereo and monocular depth estimation," in *International Conference on Computer Vision*, pp. 15529–15538, 2021.

[233] G. Wang, F. Manhardt, J. Shao, X. Ji, N. Navab, and F. Tombari, "Self6d: Self-supervised monocular 6d object pose estimation," in *European Conference on Computer Vision*, 2020.

[234] H. Chen, B. Lagadec, and F. Bremond, "Ice: Inter-instance contrastive encoding for unsupervised person re-identification," in *International Conference on Computer Vision*, pp. 14960–14969, 2021.

[235] T. Isobe, D. Li, L. Tian, W. Chen, Y. Shan, and S. Wang, "Towards discriminative representation learning for unsupervised person re-identification," in *International Conference on Computer Vision*, pp. 8526–8536, 2021.

[236] Z. Wang, J. Zhang, L. Zheng, Y. Liu, Y. Sun, Y. Li, and S. Wang, "Cycas: Self-supervised cycle association for learning re-identifiable descriptions," in *European Conference on Computer Vision*, 2020.

[237] G. Iyer, J. Krishna Murthy, G. Gupta, M. Krishna, and L. Paull, "Geometric consistency for self-supervised end-to-end visual odometry," in *CVPR Workshops*, pp. 267–275, 2018.

[238] S. Li, X. Wang, Y. Cao, F. Xue, Z. Yan, and H. Zha, "Self-supervised deep visual odometry with online adaptation," in *Conference on Computer Vision and Pattern Recognition*, pp. 6339–6348, 2020.

[239] W. Wu, Z. Y. Wang, Z. Li, W. Liu, and L. Fuxin, "Pointpwc-net: Cost volume on point clouds for (self-) supervised scene flow estimation," in *European Conference on Computer Vision*, 2020.

[240] G. Xu, Z. Liu, X. Li, and C. C. Loy, "Knowledge distillation meets self-supervision," *arXiv preprint arXiv:2006.07114*, 2020.

[241] J. Walker, A. Gupta, and M. Hebert, "Dense optical flow prediction from a static image," in *International Conference on Computer Vision*, pp. 2443–2451, 2015.

[242] F. Zhu, Y. Zhu, X. Chang, and X. Liang, "Vision-language navigation with self-supervised auxiliary reasoning tasks," in *Conference on Computer Vision and Pattern Recognition*, pp. 10012–10022, 2020.

[243] O. Wiles, A. Koepke, and A. Zisserman, "Self-supervised learning of a facial attribute embedding from video," in *British Machine Vision Conference*, 2018.

[244] Y. Li, J. Zeng, S. Shan, and X. Chen, "Self-supervised representation learning from videos for facial action unit detection," in *Conference on Computer Vision and Pattern Recognition*, pp. 10924–10933, 2019.

[245] Y. Li, J. Zeng, and S. Shan, "Learning representations for facial actions from unlabeled videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 1, pp. 302–317, 2022.

[246] S. Song, S. Jaiswal, E. Sanchez, G. Tzimiropoulos, L. Shen, and M. Valstar, "Self-supervised learning of person-specific facial dynamics for automatic personality recognition," *IEEE Transactions on Affective Computing*, 2021.

[247] X. Niu, S. Shan, H. Han, and X. Chen, "Rhythmnet: End-to-end heart rate estimation from face via spatial-temporal representation," *IEEE Transactions on Image Processing*, vol. 29, pp. 2409–2423, 2020.

[248] X. Niu, Z. Yu, H. Han, X. Li, S. Shan, and G. Zhao, "Video-based remote physiological measurement via cross-verified feature disentangling," in *European Conference on Computer Vision*, 2020.

[249] Y. Xie, Z. Wang, and S. Ji, "Noise2same: Optimizing a self-supervised bound for image denoising," in *Neural Information Processing Systems*, 2020.

[250] T. Huang, S. Li, X. Jia, H. Lu, and J. Liu, "Neighbor2neighbor: Self-supervised denoising from single noisy images," in *Conference on Computer Vision and Pattern Recognition*, 2021.

[251] C. Yang, Z. Wu, B. Zhou, and S. Lin, "Instance localization for self-supervised detection pretraining," in *Conference on Computer Vision and Pattern Recognition*, pp. 3987–3996, 2021.

[252] I. Croitoru, S.-V. Bogolin, and M. Leordeanu, "Unsupervised learning from video to detect foreground objects in single images," in *International Conference on Computer Vision*, pp. 4335–4343, 2017.

[253] E. Xie, J. Ding, W. Wang, X. Zhan, H. Xu, Z. Li, and P. Luo, "Detco: Unsupervised contrastive learning for object detection," *arXiv preprint arXiv:2102.04803*, 2021.

[254] G. Wu, J. Jiang, X. Liu, and J. Ma, "A practical contrastive learning framework for single image super-resolution," *arXiv preprint arXiv:2111.13924*, 2021.

[255] S. Menon, A. Damian, S. Hu, N. Ravi, and C. Rudin, "Pulse: Self-supervised photo upsampling via latent space exploration of generative models," in *Conference on Computer Vision and Pattern Recognition*, pp. 2437–2445, 2020.

[256] R. Girdhar, D. F. Fouhey, M. Rodriguez, and A. Gupta, "Learning a predictable and generative vector representation for objects," in *European Conference on Computer Vision*, pp. 484–499, 2016.

[257] D. Jayaraman and K. Grauman, "Learning image representations tied to ego-motion," in *International Conference on Computer Vision*, pp. 1413–1421, 2015.

[258] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Conference on Computer Vision and Pattern Recognition*, pp. 1851–1858, 2017.

[259] Y. Zou, Z. Luo, and J.-B. Huang, "Df-net: Unsupervised joint learning of depth and flow using cross-task consistency," in *European conference on computer vision*, pp. 36–53, 2018.

[260] Y. Zhang, S. Khamis, C. Rhemann, J. Valentin, A. Kowdle, V. Tankovich, M. Schoenberg, S. Izadi, T. Funkhouser, and S. Fanello, "Activestereonet: End-to-end self-supervised learning for active stereo systems," in *European Conference on Computer Vision*, pp. 784–801, 2018.

[261] Z. Yin and J. Shi, "Geonet: Unsupervised learning of dense depth, optical flow and camera pose," in *Conference on Computer Vision and Pattern Recognition*, pp. 1983–1992, 2018.

[262] Y. Zhao, G. Wang, C. Luo, W. Zeng, and Z.-J. Zha, "Self-supervised visual representations learning by contrastive mask prediction," in *International Conference on Computer Vision*, 2021.

[263] L. Huang, Y. Liu, B. Wang, P. Pan, Y. Xu, and R. Jin, "Self-supervised video representation learning by context and motion decoupling," in *Conference on Computer Vision and Pattern Recognition*, pp. 13886–13895, 2021.

[264] R. Qian, Y. Li, H. Liu, J. See, S. Ding, X. Liu, D. Li, and W. Lin, "Enhancing self-supervised video representation learning via multi-level feature optimization," in *International Conference on Computer Vision*, pp. 7990–8001, 2021.

[265] K. Hu, J. Shao, Y. Liu, B. Raj, M. Savvides, and Z. Shen, "Contrast and order representations for video self-supervised learning," in *International Conference on Computer Vision*, pp. 7939–7949, 2021.

[266] C. Sun, F. Baradel, K. Murphy, and C. Schmid, "Learning video representations using contrastive bidirectional transformer," *arXiv preprint arXiv:1906.05743*, 2019.

[267] M. Tschannen, J. Djolonga, M. Ritter, A. Mahendran, N. Houlsby, S. Gelly, and M. Lucic, "Self-supervised learning of video-induced visual invariances," in *Conference on Computer Vision and Pattern Recognition*, pp. 13806–13815, 2020.

[268] X. He, Y. Pan, M. Tang, Y. Lv, and Y. Peng, "Learn from unlabeled videos for near-duplicate video retrieval," in *International Conference on Research on Development in Information Retrieval*, pp. 1–10, 2022.

[269] A. Miech, D. Zhukov, J.-B. Alayrac, M. Tapaswi, I. Laptev, and J. Sivic, "Howto100m: Learning a text-video embedding by watching hundred million narrated video clips," in *International Conference on Computer Vision*, pp. 2630–2640, 2019.

[270] X. Wang and A. Gupta, "Unsupervised learning of visual representations using videos," in *International Conference on Computer Vision*, pp. 2794–2802, 2015.

[271] N. Srivastava, E. Mansimov, and R. Salakhudinov, "Unsupervised learning of video representations using lstms," in *International Conference on Machine Learning*, pp. 843–852, 2015.

[272] T. Han, W. Xie, and A. Zisserman, "Video representation learning by dense predictive coding," in *ICCV Workshops*, 2019.

[273] T. Han, W. Xie, and A. Zisserman, "Memory-augmented dense predictive coding for video representation learning," in *European Conference on Computer Vision*, 2020.

[274] I. Misra, C. L. Zitnick, and M. Hebert, "Shuffle and learn: unsupervised learning using temporal order verification," in *European Conference on Computer Vision*, pp. 527–544, 2016.

[275] B. Fernando, H. Bilen, E. Gavves, and S. Gould, "Self-supervised video representation learning with odd-one-out networks," in *Conference on Computer Vision and Pattern Recognition*, pp. 3636–3645, 2017.

[276] H.-Y. Lee, J.-B. Huang, M. Singh, and M.-H. Yang, "Unsupervised representation learning by sorting sequences," in *International Conference on Computer Vision*, pp. 667–676, 2017.

[277] D. Xu, J. Xiao, Z. Zhao, J. Shao, D. Xie, and Y. Zhuang, "Self-supervised spatiotemporal learning via video clip order predic-

tion," in *Conference on Computer Vision and Pattern Recognition*, pp. 10334–10343, 2019.
[278] D. Wei, J. J. Lim, A. Zisserman, and W. T. Freeman, "Learning and using the arrow of time," in *Conference on Computer Vision and Pattern Recognition*, pp. 8052–8060, 2018.
[279] S. Benaim, A. Ephrat, O. Lang, I. Mosseri, W. T. Freeman, M. Rubinstein, M. Irani, and T. Dekel, "Speednet: Learning the speediness in videos," in *Conference on Computer Vision and Pattern Recognition*, pp. 9922–9931, 2020.
[280] Y. Yao, C. Liu, D. Luo, Y. Zhou, and Q. Ye, "Video playback rate perception for self-supervised spatio-temporal representation learning," in *Conference on Computer Vision and Pattern Recognition*, pp. 6548–6557, 2020.
[281] J. Wang, J. Jiao, and Y.-H. Liu, "Self-supervised video representation learning by pace prediction," in *European Conference on Computer Vision*, 2020.
[282] A. Diba, V. Sharma, L. V. Gool, and R. Stiefelhagen, "Dynamonet: Dynamic action and motion network," in *International Conference on Computer Vision*, pp. 6192–6201, 2019.
[283] S. L. Pintea, J. C. van Gemert, and A. W. Smeulders, "Déja vu: Motion prediction in static images," in *European Conference on Computer Vision*, pp. 172–187, 2014.
[284] S. Purushwalkam and A. Gupta, "Pose from action: Unsupervised learning of pose features based on motion," *arXiv preprint arXiv:1609.05420*, 2016.
[285] C. Gan, B. Gong, K. Liu, H. Su, and L. J. Guibas, "Geometry guided convolutional neural networks for self-supervised video representation learning," in *Conference on Computer Vision and Pattern Recognition*, pp. 5589–5597, 2018.
[286] H. Jiang, G. Larsson, M. Maire Greg Shakhnarovich, and E. Learned-Miller, "Self-supervised relative depth learning for urban scene understanding," in *European Conference on Computer Vision*, pp. 19–35, 2018.
[287] A. Mahendran, J. Thewlis, and A. Vedaldi, "Cross pixel optical-flow similarity for self-supervised learning," in *Asian Conference on Computer Vision*, pp. 99–116, 2018.
[288] N. Sayed, B. Brattoli, and B. Ommer, "Cross and learn: Cross-modal self-supervision," in *German Conference on Pattern Recognition*, pp. 228–243, Springer, 2018.
[289] J. Wang, J. Jiao, L. Bao, S. He, Y. Liu, and W. Liu, "Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics," in *Conference on Computer Vision and Pattern Recognition*, pp. 4006–4015, 2019.
[290] B. Korbar, D. Tran, and L. Torresani, "Cooperative learning of audio and video models from self-supervised synchronization," in *Neural Information Processing Systems*, pp. 7763–7774, 2018.
[291] A. Owens, J. Wu, J. H. McDermott, W. T. Freeman, and A. Torralba, "Ambient sound provides supervision for visual learning," in *European Conference on Computer Vision*, pp. 801–816, 2016.
[292] R. Arandjelovic and A. Zisserman, "Look, listen and learn," in *International Conference on Computer Vision*, pp. 609–617, 2017.
[293] R. Arandjelovic and A. Zisserman, "Objects that sound," in *European Conference on Computer Vision*, pp. 435–451, 2018.
[294] H. Alwassel, D. Mahajan, L. Torresani, B. Ghanem, and D. Tran, "Self-supervised learning by cross-modal audio-video clustering," in *Neural Information Processing Systems*, pp. 1–13, 2020.
[295] S.-W. Chung, J. S. Chung, and H.-G. Kang, "Perfect match: Improved cross-modal embeddings for audio-visual synchronisation," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3965–3969, IEEE, 2019.
[296] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "Videobert: A joint model for video and language representation learning," in *International Conference on Computer Vision*, pp. 7464–7473, 2019.
[297] A. Nagrani, C. Sun, D. Ross, R. Sukthankar, C. Schmid, and A. Zisserman, "Speech2action: Cross-modal supervision for action recognition," in *Conference on Computer Vision and Pattern Recognition*, pp. 10317–10326, 2020.
[298] T. Li and L. Wang, "Learning spatiotemporal features via video and text pair discrimination," *arXiv preprint arXiv:2001.05691*, 2020.
[299] D. Hu, R. Qian, M. Jiang, X. Tan, S. Wen, E. Ding, W. Lin, and D. Dou, "Discriminative sounding objects localization via self-supervised audiovisual matching," *arXiv preprint arXiv:2010.05466*, 2020.
[300] J. C. Stroud, D. A. Ross, C. Sun, J. Deng, R. Sukthankar, and C. Schmid, "Learning video representations from textual web supervision," *arXiv preprint arXiv:2007.14937*, 2020.
[301] J.-B. Alayrac, A. Recasens, R. Schneider, R. Arandjelović, J. Ramapuram, J. De Fauw, L. Smaira, S. Dieleman, and A. Zisserman, "Self-supervised multimodal versatile networks," *arXiv preprint arXiv:2006.16228*, 2020.
[302] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, and S. Levine, "Time-contrastive networks: Self-supervised learning from video," in *IEEE International Conference on Robotics and Automation*, pp. 1134–1141, 2018.
[303] A. Owens and A. A. Efros, "Audio-visual scene analysis with self-supervised multisensory features," in *European Conference on Computer Vision*, pp. 631–648, 2018.
[304] M. Patrick, Y. M. Asano, R. Fong, J. F. Henriques, G. Zweig, and A. Vedaldi, "Multi-modal self-supervision from generalized data transformations," *arXiv preprint arXiv:2003.04298*, 2020.
[305] Y. Cheng, R. Wang, Z. Pan, R. Feng, and Y. Zhang, "Look, listen, and attend: Co-attention network for self-supervised audio-visual representation learning," in *ACM International Conference on Multimedia*, pp. 3884–3892, 2020.
[306] T. Afouras, A. Owens, J. S. Chung, and A. Zisserman, "Self-supervised learning of audio-visual objects from video," in *European Conference on Computer Vision*, 2020.
[307] S.-W. Chung, J. S. Chung, and H.-G. Kang, "Perfect match: Improved cross-modal embeddings for audio-visual synchronisation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 3, pp. 568–576, 2020.
[308] X. Wang, A. Jabri, and A. A. Efros, "Learning correspondence from the cycle-consistency of time," in *Conference on Computer Vision and Pattern Recognition*, pp. 2566–2576, 2019.
[309] X. Li, S. Liu, S. De Mello, X. Wang, J. Kautz, and M.-H. Yang, "Joint-task self-supervised learning for temporal correspondence," in *Neural Information Processing Systems*, pp. 318–328, 2019.
[310] A. Jabri, A. Owens, and A. A. Efros, "Space-time correspondence as a contrastive random walk," in *Neural Information Processing Systems*, pp. 19545–19560, 2020.
[311] Z. Lai, E. Lu, and W. Xie, "Mast: A memory-augmented self-supervised tracker," in *Conference on Computer Vision and Pattern Recognition*, pp. 6479–6488, 2020.
[312] Z. Zhang, S. Lathuiliere, E. Ricci, N. Sebe, Y. Yan, and J. Yang, "Online depth learning against forgetting in monocular videos," in *Conference on Computer Vision and Pattern Recognition*, pp. 4494–4503, 2020.
[313] D. Luo, C. Liu, Y. Zhou, D. Yang, C. Ma, Q. Ye, and W. Wang, "Video cloze procedure for self-supervised spatio-temporal learning," in *AAAI Conference on Artificial Intelligence*, pp. 11701–11708, 2020.
[314] Y. Jafarian and H. S. Park, "Learning high fidelity depths of dressed humans by watching social media dance videos," in *Conference on Computer Vision and Pattern Recognition*, pp. 12753–12762, 2021.
[315] L. Jing and Y. Tian, "Self-supervised spatiotemporal feature learning by video geometric transformations," *arXiv preprint arXiv:1811.11387*, 2018.
[316] C. Vondrick, A. Shrivastava, A. Fathi, S. Guadarrama, and K. Murphy, "Tracking emerges by colorizing videos," in *European Conference on Computer Vision*, pp. 391–408, 2018.
[317] D. Dwibedi, Y. Aytar, J. Tompson, P. Sermanet, and A. Zisserman, "Temporal cycle-consistency learning," in *Conference on Computer Vision and Pattern Recognition*, pp. 1801–1810, 2019.
[318] Z. Lai and W. Xie, "Self-supervised learning for video correspondence flow," in *British Machine Vision Conference*, 2019.
[319] D. Kim, D. Cho, and I. S. Kweon, "Self-supervised video representation learning with space-time cubic puzzles," in *AAAI Conference on Artificial Intelligence*, pp. 8545–8552, 2019.
[320] O. J. Hénaff, A. Srinivas, J. De Fauw, A. Razavi, C. Doersch, S. Eslami, and A. v. d. Oord, "Data-efficient image recognition with contrastive predictive coding," in *International Conference on Machine Learning*, 2020.
[321] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.
[322] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, p. 9, 2019.

[323] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.*, "Language models are few-shot learners," *arXiv preprint arXiv:2005.14165*, 2020.

[324] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[325] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv preprint arXiv:1909.11942*, 2019.

[326] M. Joshi, D. Chen, Y. Liu, D. S. Weld, L. Zettlemoyer, and O. Levy, "Spanbert: Improving pre-training by representing and predicting spans," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 64–77, 2020.

[327] Z. Xie, Y. Lin, Z. Yao, Z. Zhang, Q. Dai, Y. Cao, and H. Hu, "Self-supervised learning with swin transformers," *arXiv preprint arXiv:2105.04553*, 2021.

[328] C. Li, J. Yang, P. Zhang, M. Gao, B. Xiao, X. Dai, L. Yuan, and J. Gao, "Efficient self-supervised vision transformers for representation learning," *arXiv preprint arXiv:2106.09785*, 2021.

[329] Z. Li, Z. Chen, F. Yang, W. Li, Y. Zhu, C. Zhao, R. Deng, L. Wu, R. Zhao, M. Tang, and J. Wang, "Mst: Masked self-supervised transformer for visual representation," in *Neural Information Processing Systems*, pp. 1–12, 2021.

[330] C. Ge, Y. Liang, Y. Song, J. Jiao, J. Wang, and P. Luo, "Revitalizing cnn attentions via transformers in self-supervised visual representation learning," in *Neural Information Processing Systems*, pp. 1–14, 2021.

[331] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Neural Information Processing Systems*, pp. 3111–3119, 2013.

[332] L. Kong, C. d. M. d'Autume, W. Ling, L. Yu, Z. Dai, and D. Yogatama, "A mutual information maximization perspective of language representation learning," *arXiv preprint arXiv:1910.08350*, 2019.

[333] J. Wu, X. Wang, and W. Y. Wang, "Self-supervised dialogue learning," *arXiv preprint arXiv:1907.00448*, 2019.

[334] W. Chen, Y. Su, X. Yan, and W. Y. Wang, "Kgpt: Knowledge-grounded pre-training for data-to-text generation," in *Conference on Empirical Methods in Natural Language Processing*, 2020.

[335] J. Guan and M. Huang, "Union: An unreferenced metric for evaluating open-ended story generation," in *Conference on Empirical Methods in Natural Language Processing*, 2020.

[336] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang, "Pre-trained models for natural language processing: A survey," *arXiv preprint arXiv:2003.08271*, 2020.

[337] H. Wang, X. Wang, W. Xiong, M. Yu, X. Guo, S. Chang, and W. Y. Wang, "Self-supervised learning for contextualized extractive summarization," in *Annual Meeting of the Association for Computational Linguistics*, pp. 2221–2227, 2019.

[338] Y. Aytar, C. Vondrick, and A. Torralba, "Soundnet: Learning sound representations from unlabeled video," in *Neural Information Processing Systems*, pp. 892–900, 2016.

[339] H.-Y. Zhou, C. Lu, S. Yang, X. Han, and Y. Yu, "Preservational learning improves self-supervised medical image models by reconstructing diverse contexts," in *International Conference on Computer Vision*, pp. 3499–3509, 2021.

[340] K. Chaitanya, E. Erdil, N. Karani, and E. Konukoglu, "Contrastive learning of global and local features for medical image segmentation with limited annotations," in *Neural Information Processing Systems*, 2020.

[341] J. Zhu, Y. Li, Y. Hu, K. Ma, S. K. Zhou, and Y. Zheng, "Rubik's cube+: A self-supervised feature learning framework for 3d medical image analysis," *Medical Image Analysis*, p. 101746, 2020.

[342] M. Allamanis, H. Jackson-Flux, and M. Brockschmidt, "Self-supervised bug detection and repair," in *Neural Information Processing Systems*, 2021.

[343] Z. Xie, Y. Lin, Z. Zhang, Y. Cao, S. Lin, and H. Hu, "Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning," in *Conference on Computer Vision and Pattern Recognition*, pp. 16684–16693, 2021.

[344] X. Wang, R. Zhang, C. Shen, T. Kong, and L. Li, "Dense contrastive learning for self-supervised visual pre-training," in *Conference on Computer Vision and Pattern Recognition*, pp. 3024–3033, 2021.

[345] W. Wang, H. Bao, L. Dong, J. Bjorck, Z. Peng, Q. Liu, K. Aggarwal, O. K. Mohammed, S. Singhal, S. Som, *et al.*, "Image as a foreign language: Beit pretraining for all vision and vision-language tasks," *arXiv preprint arXiv:2208.10442*, 2022.

[346] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "Data2vec: A general framework for self-supervised learning in speech, vision and language," *arXiv preprint arXiv:2202.03555*, 2022.

[347] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba, "Network dissection: Quantifying interpretability of deep visual representations," in *Conference on Computer Vision and Pattern Recognition*, pp. 6541–6549, 2017.

[348] J. Robinson, L. Sun, K. Yu, K. Batmanghelich, S. Jegelka, and S. Sra, "Can contrastive learning avoid shortcut solutions?," in *Neural Information Processing Systems*, pp. 4974–4986, 2021.

[349] X. Wang and G.-J. Qi, "Contrastive learning with stronger augmentations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–12, 2022.

[350] Y. Wei, H. Hu, Z. Xie, Z. Zhang, Y. Cao, J. Bao, D. Chen, and B. Guo, "Contrastive learning rivals masked image modeling in fine-tuning via feature distillation," *arXiv preprint arXiv:2205.14141*, 2022.

[351] R. Bommasani, D. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, E. Brynjolfsson, S. Buch, D. Card, R. Castellon, N. Chatterji, A. Chen, K. Creel, J. Davis, D. Demszky, C. Donahue, M. Doumbouya, E. Durmus, S. Ermon, J. Etchemendy, K. Ethayarajh, L. Fei-Fei, C. Finn, T. Gale, L. Gillespie, K. Goel, N. Goodman, S. Grossman, N. Guha, T. Hashimoto, P. Henderson, J. Hewitt, D. Ho, J. Hong, K. Hsu, J. Huang, T. Icard, S. Jain, D. Jurafsky, P. Kalluri, S. Karamcheti, G. Keeling, F. Khani, O. Khattab, P. Koh, M. Krass, R. Krishna, R. Kuditipudi, A. Kumar, F. Ladhak, M. Lee, T. Lee, J. Leskovec, I. Levent, X. Li, X. Li, T. Ma, A. Malik, C. Manning, S. Mirchandani, E. Mitchell, Z. Munyikwa, S. Nair, A. Narayan, D. Narayanan, B. Newman, A. Nie, J. Niebles, H. Nilforoshan, J. Nyarko, G. Ogut, L. Orr, I. Papadimitriou, J. Park, C. Piech, E. Portelance, C. Potts, A. Raghunathan, R. Reich, H. Ren, F. Rong, Y. Roohani, C. Ruiz, J. Ryan, C. R'e, D. Sadigh, S. Sagawa, K. Santhanam, A. Shih, K. Srinivasan, A. Tamkin, R. Taori, A. Thomas, F. T. , R. Wang, W. Wang, B. Wu, J. Wu, Y. Wu, S. Xie, M. Yasunaga, J. You, M. Zaharia, M. Zhang, T. Zhang, X. Zhang, Y. Zhang, L. Zheng, K. Zhou, and P. Liang, "On the Opportunities and Risks of Foundation Models," *ArXiv*, 2021.

[352] Z. Peng, L. Dong, H. Bao, Q. Ye, and F. Wei, "Beit v2: Masked image modeling with vector-quantized visual tokenizers," *arXiv preprint arXiv:2208.06366*, 2022.

[353] J. Liu, X. Huang, Y. Liu, and H. Li, "Mixmim: Mixed and masked image modeling for efficient visual representation learning," *arXiv preprint arXiv:2205.13137*, 2022.

[354] X. Dong, J. Bao, T. Zhang, D. Chen, W. Zhang, L. Yuan, D. Chen, F. Wen, and N. Yu, "Peco: Perceptual codebook for bert pre-training of vision transformers," *arXiv preprint arXiv:2111.12710*, 2021.

[355] Q. Zhou, C. Yu, H. Luo, Z. Wang, and H. Li, "Mimco: Masked image modeling pre-training with contrastive teacher," in *ACM International Conference on Multimedia*, pp. 4487–4495, 2022.

[356] Y. Chen, Y. Liu, D. Jiang, X. Zhang, W. Dai, H. Xiong, and Q. Tian, "Sdae: Self-distillated masked autoencoder," *arXiv preprint arXiv:2208.00449*, 2022.

Jie Gui (SM'16) is currently a professor at the School of Cyber Science and Engineering, Southeast University. He received a BS degree in Computer Science from Hohai University, Nanjing, China, in 2004, an MS degree in Computer Applied Technology from the Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei, China, in 2007, and a PhD degree in Pattern Recognition and Intelligent Systems from the University of Science and Technology of China, Hefei, China, in 2010. He has published more than 40 papers in international journals and conferences such as IEEE TPAMI, IEEE TNNLS, IEEE TCYB, IEEE TIP, IEEE TCSVT, IEEE TSMCS, KDD, and ACM MM. He is the Area Chair, Senior PC Member, or PC Member of many conferences such as NeurIPS and ICML. He is an Associate Editor of IEEE Transactions on Circuits and Systems for Video Technology (T-CSVT), Artificial Intelligence Review, Neural Networks, and Neurocomputing. His research interests include machine learning, pattern recognition, and image processing.

Tuo Chen is a PhD student with the Department of Electronic Information, Southeast University. He received his bachelor's degree from the Department of Information Security, Lanzhou University. His main research interests include self-supervised learning and adversarial examples.

Qiong Cao is a Research Scientist at JD Explore Academy. Before that, she was a Senior Researcher at Tencent. Prior to joining Tencent, she was a Postdoctoral Researcher at the Department of Engineering Science, University of Oxford. She obtained her PhD in Computer Science from the University of Exeter.

Zhenan Sun (SM'18) received a B.E. degree in industrial automation from Dalian University of Technology, Dalian, China, in 1999, an M.S. degree in system engineering from Huazhong University of Science and Technology, Wuhan, China, in 2002, and a PhD degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China, in 2006.

Since 2006, he has been a Faculty Member with the National Laboratory of Pattern Recognition, CASIA, and he is currently a professor with the Center for Research on Intelligent Perception and Computing. He has authored/coauthored over 200 technical papers. His current research interests include biometrics, pattern recognition, and CV.

Prof. Sun is an Associate Editor of IEEE Transactions on Biometrics, Behavior, and Identity Science. He is a member of the IEEE Computer Society and IEEE Signal Processing Society and a fellow of IAPR.

Hao Luo received B.S. and PhD degrees from Zhejiang University, China, in 2015 and 2020, respectively. He is currently working at the Alibaba DAMO Academy. His research interests include person re-identification, vision transformer, self-supervised, computer vision, and deep learning.

Dacheng Tao (F'15) is a professor of Computer Science and an ARC Laureate Fellow at the School of Computer Science with the Faculty of Engineering and the Inaugural Director of the UBTECH Sydney Artificial Intelligence Centre at The University of Sydney. His research results in artificial intelligence have been expounded in one monograph and 200+ publications in prestigious journals and at prominent conferences, such as IEEE T-PAMI, IJCV, JMLR, AAAI, IJCAI, NeurIPS, ICML, CVPR, ICCV, ECCV, ICDM, and KDD, where he has received several "best paper" awards. He received the 2018 IEEE ICDM Research Contributions Award and the 2015 Australian Scopus-Eureka prize. He is a fellow of the IEEE, the ACM and the Australian Academy of Science.