CYBORGS: Contrastively Bootstrapping Object Representations by Grounding in Segmentation

Renhao Wang[1], Hang Zhao[1,2], and Yang Gao[1,2(✉)]

[1] Tsinghua University, Beijing, China
gaoyangiiis@tsinghua.edu.cn
[2] Shanghai Qi Zhi Institute, Shanghai, China

Abstract. Many recent approaches in contrastive learning have worked to close the gap between pretraining on iconic images like ImageNet and pretraining on complex scenes like COCO. This gap exists largely because commonly used random crop augmentations obtain semantically inconsistent content in crowded scene images of diverse objects. In this work, we propose a framework which tackles this problem via joint learning of representations and segmentation. We leverage segmentation masks to train a model with a mask-dependent contrastive loss, and use the partially trained model to bootstrap better masks. By iterating between these two components, we ground the contrastive updates in segmentation information, and simultaneously improve segmentation throughout pretraining. Experiments show our representations transfer robustly to downstream tasks in classification, detection and segmentation. (Code and pretrained models available at https://github.com/renwang435/CYBORGS).

1 Introduction

Many self-supervised contrastive methods have come to rival and even surpass the performance of fully supervised methods on a number of tasks, including object detection [3,44], semantic segmentation [17,41], video understanding [23,30], and image classification [8,13]. A large portion of these methods rely on random cropping to select positive pairs of image subregions for a self-supervised instance-level discrimination task. Recently, many works have found this random cropping strategy succeeds for iconic image pretraining, but struggles when applied to pretraining on complex scene images. Treating two random crops from the same image as containing semantically similar information works well for images with singular, dominant subjects, like those in ImageNet. But such an

H. Zhao and Y. Gao—Equal advising.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-19821-2_15.

S. Avidan et al. (Eds.): ECCV 2022, LNCS 13691, pp. 260–277, 2022.
https://doi.org/10.1007/978-3-031-19821-2_15

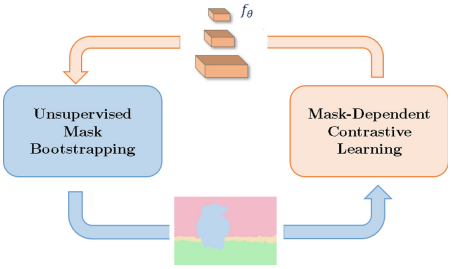


Fig. 1. Mutually improving representation learning and semantic segmentation. In the first stage, we use available segmentation masks to ground contrastive learning. In the second stage, we use representations from the backbone f_θ to bootstrap improved segmentation masks.

assumption inevitably fails due to inconsistent learning signals in scene images full of diverse objects [9,33,40]. To address this issue, prior works have generated random crops in an object-aware manner [3,34,44]. By localizing objects with unsupervised algorithms (e.g. selective search), these works are able to ground random crops around singular objects, validating the assumption that such crops contain similar information about objects.

We argue that utilizing *pixel-level* object information can be even more effective than detection-level boxes. By parsing random crop contents with segmentation masks, we can turn a pair of crops into a diverse wealth of similar and dissimilar object regions, facilitating contrastive self-supervised learning. To fully realize this idea in SSL frameworks, we also need to meet two important criteria. Firstly, these masks should be obtained in an *unsupervised* manner. Secondly, we want to *avoid preprocessing pipelines* to obtain pseudo-segmentations (*e.g.*, graph cut algorithms), which often lack awareness of object-level semantics and require human domain knowledge for good performance [20,50].

To this end, we propose in this work to perform segmentation and concept learning *jointly* (Fig. 1). In the first stage of our framework, we ground self-supervised learning with segmentation information to train a representation backbone. In a periodic second stage, we leverage these representations to bootstrap segmentation masks, which can subsequently be fed back to the first stage to further improve representations. By iterating between these two core stages, we develop representations which strongly generalize to many downstream tasks, and are especially well-aligned with object detection and segmentation. Furthermore, to ameliorate issues of representation collapse, we also optimize a clustering consistency objective during the first stage. We show that the formulation of this loss fits naturally within any contrastive framework, and helps improve masks more reliably between bootstrap cycles. Thus, in **C**ontrastive**Y** **B**ootstrapping **O**bject **R**epresentations by **G**rounding in **S**egmentation (CYBORGS), our contributions are fourfold:

1. We develop the first framework which performs end-to-end, joint self-supervised learning of object-level representations and semantic segmentation, while removing entirely the need for heuristic preprocessing of pseudo-segmentations.
2. We show how to bootstrap segmentation masks robustly by directly clustering on feature maps obtained from a partially pretrained backbone.
3. We demonstrate how to regularize contrastive updates in our framework with an intra-/inter-view cluster consistency loss that is well-aligned with the hyperspherically-distributed contrastive embeddings.
4. With pretraining on complex scene images such as COCO, we demonstrate that grounding in segmentation leads to representations which transfer competitively to a diversity of downstream tasks and real-world, long-tail objects and scene semantics.

2 Related Work

Self-Supervised Representation Learning. SSL methods utilize internal structure as a source of supervision to learn general representations, including auxiliary tasks such as context prediction [13], solving jigsaws [35], inpainting [37], colorization [51], or orientation prediction [27]. Most relevant to our work is contrastive learning, where the goal is to perform instance discrimination, concentrating positive pairs and separating negative pairs of feature embeddings in a latent space [8,15,19,36]. Despite their convincing performance on downstream tasks, the majority of current contrastive-based methods are pretrained on ImageNet, and subject to strong object-centric bias and poor visual grounding [21,34,38,40].

To this end, a number of emerging methods examine self-supervised representation learning on in-the-wild, scene image datasets such as COCO [33,40,45]. CAST improves visual grounding by ensuring crops overlap readily with object regions identified by saliency masks, and guides representation learning using a Grad-CAM loss [39,40]. ORL uses a pretrained self-supervised model to approximate object-level semantic correspondence, thus improving positive-negative identification for contrastive refinement of the pretrained model [45].

Going a step further, CYBORGS and other works obtain object-level semantics through pixel level pseudo-labeling [1,17,20,41,52]. For example, DetCon [20] involves unsupervised preprocessing of images to obtain masks, and uses these masks to aggregate features over object regions for contrastive learning. Crucially, all other previous methods suffer from the disadvantage that mask proposals are generated i) via graph-based algorithms requiring heuristic hyperparameter decisions, and ii) only once before training, with no further learning. In contrast, by integrating object mask proposals and contrastive pretraining into the same loop, CYBORGS iteratively refines and improves both segmentation quality and learned representation quality, jointly.

Unsupervised Segmentation and Clustering. The use of clustering-based approaches in SSL has a long history [5,6,31]. DeepCluster is a seminal work which proposed to train a CNN by alternating between feature clustering to obtain class pseudo-labels, and learning to predict those very labels [5]. PCL and SwAV combine a clustering objective with a contrastive objective, directly encoding semantic structure learned by clustering into a latent representation space [6,31]. Instead of directly improving features by learning to cluster feature prototypes, CYBORGS primarily uses clustering as a mechanism to improve segmentation.

Indeed, clustering-based algorithms have recently found application in a number of unsupervised and self-supervised image segmentation works [11,22,50,52]. Both pixel and region-level contrastive learning methods have been employed to i) improve semantic segmentation for better representation learning [46,52], and ii) vice versa [11,22,26]. To the best of our knowledge, CYBORGS is the first work to consider these two well-studied tasks as complementary, iteratively synergizing them together via a bootstrapping paradigm. Additionally, CYBORGS does not directly optimize for segmentation quality via pixel-level losses, and aims to improve segmentation strictly insofar as it aids in representation learning.

3 CYBORGS

We now describe the details in our proposed framework. In Sect. 3.1, we provide an overview of the abstractions in our work. At its core, we require iteration between two components: a contrastive objective capable of leveraging masks to train an encoder, and an unsupervised method to generate masks from a (partially) trained encoder. In Sect. 3.2 and Sect. 3.3, we describe the particular instantiations of these two components in our demonstration of the framework. Finally, in Sect. 3.4, we show how to construct a self-supervised consistency loss to guide mask generation.

3.1 CYBORGS Framework Abstraction

Following typical contrastive learning frameworks in vision, we begin with a given RGB image $\mathbf{I} \in \mathbb{R}^{3\times H\times W}$ of height H and width W, and two transformations t, t' independently sampled from data augmentation pipelines $\mathcal{T}, \mathcal{T}'$. For the time being, we assume we also have ground truth semantic segmentation masks $\{\mathbf{M}\} \in [0,1]^{C\times H\times W}$. Each $H \times W$ binary mask $\mathbf{M}$ describes pixel-wise class membership for a particular class, for C total classes. Applying the transformations to $\mathbf{I}, \{\mathbf{M}\}$ yields two augmented views $\mathbf{v} = t(\mathbf{I}), \mathbf{v}' = t'(\mathbf{I})$, and two semantic maps $\{\mathbf{m}\} = t(\{\mathbf{M}\}), \{\mathbf{m}'\} = t'(\{\mathbf{M}\})$. Note that every $\mathbf{m}$ contains object-level assignments spatially aligned with view $\mathbf{v}$, and likewise every $\mathbf{m}'$ aligns with $\mathbf{v}'$. After passing view $\mathbf{v}$ to a (fully) convolutional encoder f_θ for featurization, we can extract a (sub)set of intermediate feature maps $\{\mathbf{F}\} = \{\mathbf{y}^{[1]}, \ldots, \mathbf{y}^{[l]}\}$, where $\mathbf{y}^{[l]} = f_\theta^{[l]}(\mathbf{v})$ for layer l. Doing the same for view $\mathbf{v}'$ yields feature maps $\{\mathbf{F}'\}$.

These feature maps inherently contain spatial and latent information about the image, which we can leverage using the segmentation masks. The core idea is conceptually simple and lightweight: we can sample arbitrary regions in the feature maps and apply the binary masks $\{\mathbf{m}\}, \{\mathbf{m}'\}$ to filter out groups of features which correspond to the same underlying object regions. Applying mean pooling, concatenation, or some other general aggregation operator to these groups yields feature vectors containing similar and dissimilar object-level semantics. These positive-negative pairs allow us to use a flexible class of contrastive objectives to train our encoder f_θ. Note that this naturally requires upsampling or downsampling either the masks or the feature maps to the same spatiality, and our framework is entirely agnostic to these details. But a more immediate problem is obtaining reasonable masks $\{\mathbf{M}\}$ to begin with.

A crucial assumption we have thus maintained is that ground truth segmentation masks are available. Indeed, without specification of how object regions correspond to each other across views, the very notion of positives and negatives for a contrastive formulation becomes ill-defined. Previous works which have relied on such masks in a similar fashion have used simple spatial heuristics such as grid-based masks, or more complex unsupervised algorithms such as graph cut segmentations [1,20,52]. Ultimately, we find that these approaches yield unsatisfactory masks which are semantics-unaware, or require significant hand-tuning, especially when employed on scene images. But composing a learning-based procedure is non-trivial; the contrastive objective cannot backpropagate through the non-differentiable augmentations t, t' and modify a mask $\mathbf{m}$ directly (Fig. 2).

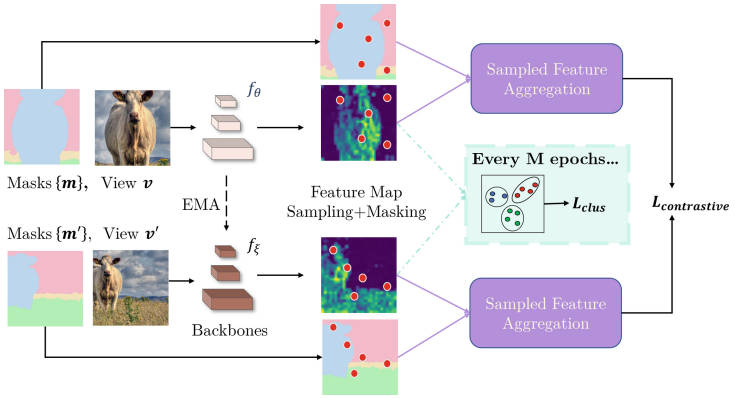


Fig. 2. CYBORGS Training Framework. We sample over the feature maps for different views, using segmentation masks to identify similar and dissimilar object regions. These are aggregated into positive and negative feature vectors, respectively, for the contrastive objective $\mathcal{L}_{mask}$ (Sect. 3.2). Periodically, we also backprop through a clustering consistency loss $\mathcal{L}_{clus}$ (Sect. 3.4).

To this end, our framework bootstraps segmentation masks using representations from the partially trained model f_θ. This idea is motivated by two insights.

Firstly, the contrastive objective directly improves the encoder f_θ, and thus leveraging the features from f_θ can help us obtain semantic-aware masks which correspondingly improve over the course of training. Secondly, recall that the ultimate goal of our framework is to improve representation learning. Since downstream transfer of representations takes places on f_θ, using the representations from f_θ to construct our segmentation masks ensures that representation quality and bootstrapped mask quality are tied together. Implementation-wise, our framework is agnostic to the actual algorithm employed for mask generation, with the only constraint being that the method cannot rely on ground truth supervision. For concreteness, we illustrate in Sect. 3.3 how to generate robust masks using a simple KMeans clustering-based algorithm on the feature maps $\{\mathbf{F}\}$. By iterating between contrastive updating of f_θ and unsupervised generation of masks, we mutually improve our representations and segmentations.

3.2 Mask-Dependent Contrastive Learning

To demonstrate the utility of our framework, we first choose the loss function from [20] as the particular instantiation of a mask-based contrastive objective for training our encoder in the first stage. We provide a high level review here.

In [20], $\{\mathbf{F}\}$ is a single $2048 \times 7 \times 7$ feature map extracted from the final layer of a standard ResNet-50 encoder processing view $\mathbf{v}$ (before average pooling). The entire feature map is sampled, and the segmentation masks in $\{\mathbf{m}\}$ are spatially downsampled accordingly. Aggregation of $\{\mathbf{F}\}$ is obtained via mask-based pooling for each $\mathbf{m} \in \{\mathbf{m}\}$:

$$\mathbf{h}_\mathbf{m} = \frac{1}{\sum_{i,j} \mathbf{m}[i,j]} \sum_{i,j} \mathbf{m}[i,j]\mathbf{F}[i,j] \tag{1}$$

Feature map $\{\mathbf{F}'\}$ is similarly aggregated after processing view $\mathbf{v}'$ with a target encoder f_ξ, yielding $\mathbf{h}'_{\mathbf{m}'}$. For additional asymmetry, $\mathbf{h}_\mathbf{m}$ is further transformed by an online projector g_θ and predictor q_θ to obtain $\mathbf{v}_\mathbf{m} = q_\theta(g_\theta(\mathbf{h}_\mathbf{m}))$, and $\mathbf{h}_{\mathbf{m}'}$ is transformed by a target projector g_ξ to obtain $\mathbf{v}'_{\mathbf{m}'} = g_\xi(h'_{m'})$. The target parameters ξ are updated as an exponential moving average (EMA) of their online counterparts θ. The final mask-based contrastive objective is given by:

$$\mathcal{L}_{contrastive} = \mathbb{E}_{\mathbf{m},\mathbf{m}' \sim \{\mathbf{m}\},\{\mathbf{m}'\}} \left[-\log \frac{\exp(\mathbf{v}_\mathbf{m} \cdot \mathbf{v}'_{\mathbf{m}'})}{\exp(\mathbf{v}_\mathbf{m} \cdot \mathbf{v}'_{\mathbf{m}'}) + \sum_n \exp(\mathbf{v}_\mathbf{m} \cdot \mathbf{v}_\mathbf{n})} \right] \tag{2}$$

for negative pooled features $\{\mathbf{v}_n\}$ sampled from different masks and images.

In addition, inspired by prior art demonstrating that different layers within a CNN encode information at different semantic resolutions [18,28,47], we also extract and utilize features from throughout ResNet-50, instead of relying solely on features from the final convolutional map as in [20]. By fusing these features together spatially (after upsampling or downsampling), downstream learning is able to leverage information across the semantic spectrum, from low-level local structure, to high-level global style. Further details are available in the appendix.

3.3 Bootstrapping Segmentation Masks

Recall that our framework is agnostic to the particular algorithm used in the second stage bootstrapping of better segmentation masks. For simplicity, we illustrate the details of this stage using a classic KMeans clustering algorithm.

More formally, we begin by considering a batch of B input RGB images $\{\mathbf{I}\} \in \mathbb{R}^{B\times 3\times H\times W}$, and a (fully) convolutional backbone f_θ which has been trained in a self-supervised fashion via the objective in (2). We choose a particular layer $\boldsymbol{\ell}$, and extract the feature map $\mathbf{y}_\theta^{[\boldsymbol{\ell}]} = f_\theta^{[\boldsymbol{\ell}]}(\{\mathbf{I}\}) \in \mathbb{R}^{B\times D_F\times H_F\times W_F}$. We omit the layer index $\boldsymbol{\ell}$ and online encoder parameters θ for brevity, so that $\mathbf{y} \triangleq \mathbf{y}_\theta^{[\boldsymbol{\ell}]}$. We then flatten the feature maps and $\ell 2$-normalize feature-wise, generating a matrix of features $\mathbf{F} \in \mathbb{R}^{(B\cdot H_F\cdot W_F)\times D_F}$. Given a hyperparameter K, representing the number of clusters (or unique object classes) within the segmentation mask, we perform spherical K-means clustering on $\mathbf{F}$, ending up with a matrix of feature prototypes $\mathbf{P} = \{\mu_1, \mu_2, \ldots, \mu_K\} \in \mathbb{R}^{D_F\times K}$. We assign to each cell in the original feature map $\mathbf{y}^{[\boldsymbol{\ell}]}$ a cluster label based on their Euclidean distances to the prototypes in $\mathbf{P}$. Finally, we broadcast the class assignments back to the original dimensions of the image $\mathbf{I}$ via nearest neighbor interpolation, akin to [9] (Fig. 3).

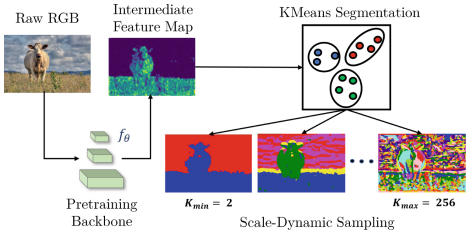


Fig. 3. Bootstrapping Masks. To generate the segmentation masks, we perform simple KMeans clustering on a feature map from the trained backbone, with a dynamic number of clusters.

Periodic Bootstrapping. Performing such a clustering operation on every epoch to regenerate the segmentations can be expensive. Even if computation was not an issue, we empirically find that representations do not improve monotonically with epochs, so bootstrapping masks too frequently can actually lead to worse masks. Moreover, as a result of an undertrained encoder f_θ at the beginning of training, we obtain poorer early clusterings; noisy masks lead to noisy gradients for updating the encoder, and vice versa. Thus, to avoid representation collapse, we periodically bootstrap the segmentation masks every N epochs, where N is a hyperparameter much greater than 1.

Scale-Dynamic Sampling. The choice of K also merits discussion. Given access to some oracle, a natural choice might be to set K equal to the number of unique object classes within the image. However, as a number of prior works have identified, the semantic context provided by extra "distractor" classes outside of the main object classes can serve as a useful signal for clustering [5,9,24]. But increasing K also requires more images within the bootstrapping batch to perform KMeans reliably on the features, reducing the scalability of our method.

To balance these motivations, for every batch of images where we wish to bootstrap segmentations, we dynamically sample integer K uniformly between $K_{min} = 2$ and $K_{max} = 256$, inclusive. Intuitively, $K_{min} = 2$ represents a mask which imparts the model with simple foreground-background semantics, while the upper bound of $K_{max} = 256$ yields an oversegmentation (COCO offers only 81 labeled object segmentation classes.) By varying K in such a fashion, not only do we maintain efficiency in bootstrapping, but we also reintroduce our model to information of varying semantic scale on every bootstrap cycle. As we show in Sect. 4.4, this technique improves the robustness of our representations.

3.4 Consistency as a Curriculum for Segmentation

Despite the use of periodic bootstrapping and scale-dynamic sampling, we find that the long training schedules employed in contrastive learning can still lead to divergence between our representation learning and semantic segmentation objectives. This is because our framework up to now improves the segmentation only *implicitly*. While we are optimizing on every iteration our contrastive objective in (2), regularly improving our encoded representations, the bootstrapping of masks is optimization-free with respect to the encoder. Without an update signal to explicitly encode the semantics of desirable vs. non-desirable segmentations, the encoder over-prioritizes the goal of representation learning, and can diverge from a feature distribution which yields good segmentations.

Clustering Consistency. To this end, we reuse a universal paradigm in contrastive learning: similar objects across different scenes and different views should have similar labels. We introduce a clustering consistency loss, similar to that employed in [11], which can be applied more regularly every M epochs, where M is more frequent than the every N epochs used per bootstrapping cycle.

Concretely, recall the feature map $\mathbf{y} \in \mathbb{R}^{D_F \times H_F \times W_F}$ and feature prototypes $\mathbf{P} = \{\mu_1, \mu_2, \ldots, \mu_K\} \in \mathbb{R}^{D_F \times K}$ we obtained in Sect. 3.3 after processing view $\mathbf{v}$ using the online encoder f_θ. We obtain a similar map $\mathbf{y}'$ and set of prototypes $\mathbf{P}' = \{\mu'_1, \mu'_2, \ldots, \mu'_K\}$ after featurizing $\mathbf{v}'$ with the target encoder f_ξ. Consider the feature at pixel $[i, j]$ within $\mathbf{y}$, for an arbitrary $1 \leq i \leq H_F$ and $1 \leq j \leq W_F$. With a slight abuse of notation, we let $\mu_{[i,j]}$ represent the prototype this feature is assigned to under $\mathbf{P}$ (and similarly, $\mu'_{[i,j]}$ the assignment of $\mathbf{y}'[i, j]$ under $\mathbf{P}'$). Then we define a clustering consistency loss via:

$$\mathcal{L}_{clus} = \frac{1}{H_F W_F} \sum_{i=1}^{H_F} \sum_{j=1}^{W_F} \overbrace{d\left(\mu_{[i,j]}, \mathbf{y}_{[i,j]}\right) + d\left(\mu'_{[i,j]}, \mathbf{y}'_{[i,j]}\right)}^{intra-loss} + \overbrace{d\left(\mu'_{[i,j]}, \mathbf{y}_{[i,j]}\right) + d\left(\mu_{[i,j]}, \mathbf{y}'_{[i,j]}\right)}^{inter-loss} \tag{3}$$

where $d(\cdot, \cdot)$ is some distance function. Intuitively, intra-cluster consistency enforces that under one scene, object regions with similar features should be clustered into similar prototypes. Similarly, inter-cluster consistency enforces

that under different scenes, we still wish for features from different regions corresponding to similar objects to be assigned to the same prototype. This forces our learned prototypes to be invariant to differences between views and generalize to object-centric semantics, which translates readily to higher fidelity segmentation masks during bootstrapping updates.

To formulate $d(\cdot, \cdot)$, we draw inspiration from recent work which demonstrates that the infoNCE objective in contrastive learning promotes a feature space which is uniformly distributed on the unit hypersphere [42]. The von Mises-Fisher (vMF) distribution defines a probability density over a unit hypersphere, making it a natural candidate to characterize the feature space learned by our mask-based contrastive objective in (2). We refer readers to a comprehensive treatment in [14] for details. In our setting, we can assume a vMF mixture model where each feature $\mathbf{y}$ is drawn uniformly from one of K vMF distributions, each parameterized by a feature clustering prototype $\mu_1, \mu_2, \ldots, \mu_K$, and sharing a common concentration hyperparameter κ. Then our clustering consistency loss objective is formulated as maximizing the posterior likelihood of a particular encoded feature $\mathbf{y}$ being assigned to its corresponding cluster c under this mixture, with $1 \leq c \leq K$. That is, we seek to minimize the negative log-likelihood given by:

$$d(\mu_{[i,j]}, \mathbf{y}) = -\log p(\mu_{[i,j]=c} \mid \mathbf{y}, \mu_1, \mu_2, \ldots, \mu_K) = -\log \frac{\exp\left(\kappa \mu_{[i,j]}^T \mathbf{y}\right)}{\sum_{c'=1}^{K} \exp\left(\kappa \mu_{c'}^T \mathbf{y}\right)} \quad (4)$$

The vMF clustering loss objective described in (3) also serves an additional purpose towards the beginning of our pretraining pipeline. In the total absence of reliable masks before the first bootstrapping cycle, we train our encoder f_θ strictly with the loss in (3), setting K to a fixed parameter depending on the median number of objects per scene in our dataset (*e.g.*, for COCO, we use $K = 8$). This *vMF warmup period* of W epochs ($W = 5$ in our work) serves to burn in our encoder. After more reasonable representations have been learned, we immediately bootstrap the masks, and subsequent epochs using a combination of the mask-based contrastive loss in (2) and the vMF clustering loss in (3), as their respective periods N and M dictate. For a comprehensive outlining of our algorithm flow, we refer readers to the pseudocode presented in the appendix.

4 Experiments

In our experiments, we aim to demonstrate that joint learning of general representations and semantic segmentation can be successfully accomplished via our bootstrapping method. We show strong performance on multiple downstream tasks (Sect. 4.2), surprisingly robust segmentation performance over a long-tailed distribution of objects (Sect. 4.3), and a convincing array of ablations which validate our design choices and methodological contributions (Sect. 4.4).

4.1 Experimental Settings

Datasets. Given our primary goal of learning on images in the wild, we follow previous works [33,40,43] and pretrain on the train2017 split of the MS COCO dataset [32]. With ∼118k images of natural settings, MS COCO is widely adopted as a benchmark more reflective of real-world scenarios across a breadth of downstream tasks of interest, such as object detection or instance segmentation. For a relevant quantitative comparison, note that the heavily object-dominant ImageNet dataset contains on average 1.1 objects per image, whereas the average scene image in COCO contains 7.3 objects [43]. Crucially, we use no scene-level, object-level, or pixel-level label information in our pretraining pipeline.

Implementation Details. To enable easy comparison to other SSL works in similar settings [33,40,43,45], we use a ResNet-50 backbone in all of our models. Other architectural details such as the dimensionality of projection and prediction MLPs described in Sect. 3.2 follow directly from BYOL [15].

For our mask-based contrastive objective in (2), we aggregate features from `res2`, `res3`, `res4`, downsampling all layers to a spatial resolution of 7×7. This allows us to leverage a lightweight but comprehensive semantic hierarchy. We bootstrap the segmentation masks every $N = 100$ epochs, performing clustering on the feature map from `res2.b2` in batches of 16 images (where `b2` refers to block 2). We use a vMF warmup period of 5 epochs; outside the warmup period, the vMF clustering loss is employed every 5 epochs with weight $\lambda = 0.1$ and $\kappa = 10$. In pretraining, we use the LARS optimizer [49] with a batch size of 64 across 8 NVIDIA RTX 3090 s s for 800 epochs. The initial learning rate is set to 0.1, and the weight decay is $1.5e^{-6}$. Clustering is implemented via GPU-accelerated mini-batch approximation using the FAISS library [25].

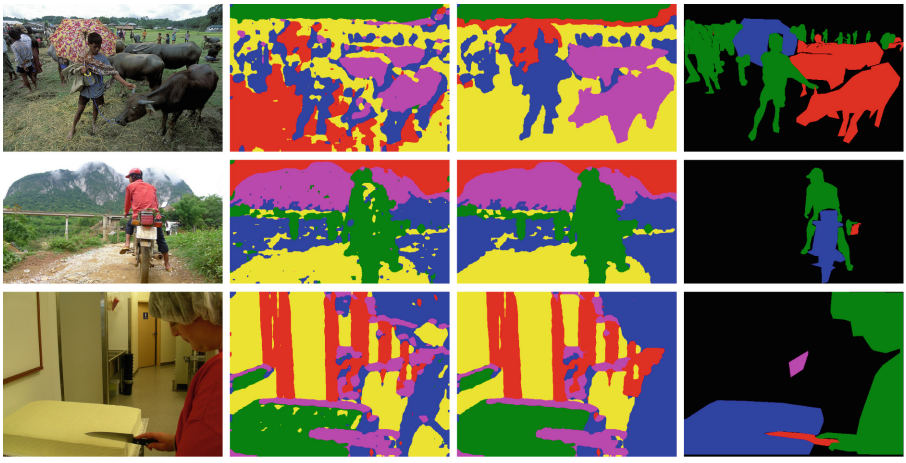
Table 1. Transfer Learning on Downstream Tasks. We report strong, state-of-the-art performance across linear classification on VOC07, semi-supervised finetuning on ImageNet-1k, and transfer on VOC object detection and COCO instance segmentation. All methods are pretrained on COCO with a ResNet-50 backbone, and finetuned on the reported datasets.

	Method	VOC07 clf.	IN-1k, 1% Labels		IN-1k, 10% Labels		VOC Detection			COCO instance segmentation					
		mAP	Top-1 acc.	Top-5 acc.	Top-1 acc.	Top-5 acc.	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
1)	SIMCLR [8]	78.1	23.4	46.4	52.2	77.4	–	–	–	37.0	56.8	40.3	33.7	53.8	36.1
2)	MOCO-V2 [10]	82.2	28.2	54.7	57.1	81.7	54.7	81.0	60.6	38.5	58.1	42.1	34.8	55.3	37.3
3)	BYOL [15]	84.5	28.4	55.9	58.4	82.7	55.5	81.7	61.7	39.5	59.3	43.2	35.6	56.5	38.2
4)	BAI ET AL. [1]	–	–	–	–	–	57.1	82.1	63.8	39.8	59.6	43.7	35.9	56.9	38.6
5)	DENSECL [43]	83.8	–	–	–	–	56.7	81.7	63.0	39.6	59.3	43.3	35.7	56.5	38.4
6)	CAST [40]	73.1	–	–	–	–	54.2	80.1	59.9	36.7	56.7	39.9	33.6	53.6	35.8
7)	ORL [45]	86.7	31.0	58.9	60.5	**84.2**	55.8	82.1	62.3	40.3	60.2	44.4	36.3	57.3	38.9
8)	CYBORGS(ours)	**86.9**	**31.3**	**59.4**	**61.7**	**84.2**	**58.0**	**83.0**	**64.3**	**42.0**	**62.6**	**46.2**	**38.0**	**59.7**	**40.8**

4.2 Main Results: Representation Learning

We follow standard downstream transfer-based protocols to evaluate the strength of representations learned by CYBORGS. In particular, we begin with *frozen* linear evaluation on VOC07 and semi-supervised transfer on ImageNet-1k. In comparison to similar state-of-the-art self-supervised methods pretrained on COCO, we achieve improvements of +0.2 mAP for VOC07 and +0.3%, +1.2% in top-1 accuracy for semi-supervised 1% and 10% on IN-1k, respectively. While these gains are only incremental, image classification requires semantic-level knowledge [2,38,48], whereas we design our method around leveraging pixel-level information, and so even marginal gains are a surprising windfall.

While linear probing has been treated as the gold standard for assessing feature quality, that strong performance in tasks such as detection and segmentation are even more reflective of potent learned representations. We demonstrate convincing state-of-the-art on PASCAL VOC detection and COCO instance segmentation. In comparison to a strong and well-established BYOL baseline, we provide a +2.5 AP improvement on the former, and a +2.5 and +2.4 AP improvement on the latter. Our results on segmentation in particular are noteworthy; while we do not make use of pixel annotations, our bootstrapping scheme clearly aligns with latent information critical to the segmentation task. To further verify this robust segmentation performance, we also perform transfer-based evaluation on CityScapes semantic segmentation [12], as well as LVIS long-tailed instance



(a) Raw RGB **(b)** KMeans Mask **(c)** CRF-Refined Mask **(d)** Ground Truth

Fig. 4. Bootstrapped segmentation masks from a CYBORGS-pretrained encoder on COCO. We show KMeans segmentations on the bilinearly upsampled feature maps for visual quality. During actual bootstrapping, we first segment the feature map, before performing nearest neighbors upsampling, and do not perform CRF refinement on the mask. Colors do not necessarily correspond across images (rows) or between mask types (columns), but are consistent within a single image itself.

segmentation [16], which we detail in the appendix. We achieve state-of-the-art on these datasets amongst all other previous SSL methods pretrained on COCO, with a substantial +3.4 AP and +3.3 AP improvement on LVIS, demonstrating that our framework can also generalize to unseen object structures and semantics.

4.3 Segmentation Quality

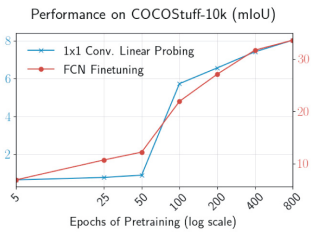
We first confirm qualitatively (Fig. 4) that bootstrapped masks generated by CYBORGS are indeed semantically meaningful. Note that our clustering-based segmentations easily extend beyond the original labeled classes of COCO, despite receiving no ground truth information about pixel labels throughout pretraining.

How does CYBORGS *work with such noisy masks?* In addition to the masks generated by clustering on the feature maps from the backbone encoder, we also show the mask resulting from refinement using a fully connected conditional random field (CRF), using the distances to feature prototypes in latent space as priors, following the protocol described in previous works [7,29]. We argue that although the raw masks at a pixel-level appear to be noisy, their easy refinement into masks closely aligned with ground truth masks indicates that the encoded features are quite well aligned with object-level concepts at the semantic level.

Why Bootstrap Masks? To further demonstrate the robustness of our bootstrapping process for mask generation, we retrain CYBORGS using alternative masks. Instead of bootstrapping masks, we employ random cropping masks (i.e. all pixels in the scene belong to the same class), a 5×5 spatial grid mask and Felzenszwalb-Huttenlocher (FH) masks used in [1,20], detection-level object

Type of Mask	mIoU
Random Crop	15.9
5x5 Grid	18.7
FH Masks	27.7
Obj. Bounding Boxes	29.0
CYBORGS Masks (ours)	33.6
Ground Truth Masks	35.2

(a) Importance of bootstrapping.



(b) Masks (blue) and representations (red) improve jointly.

Fig. 5. Using semantic segmentation performance on COCO-Stuff-10k to evaluate bootstrapping value. (a) Replacing our bootstrapping segmentation core with static boxes from other unsupervised heuristics leads to decreased performance. (b) Note a single epoch of using improved masks can lead to significant gains (epoch 100).

masks acquired via selective search pre-processing, and ground truth masks available in COCO. These masks are generated before pretraining and remain fixed, supplanting our bootstrapping algorithm. Given the unsupervised generalization of masks generated under our framework to a long-tailed distribution of objects (c.f. Fig. 4), we evaluate the representations by transferring the trained backbones to a ResNet-50 FCN and finetuning end-to-end on COCO-Stuff-10k semantic segmentation. COCO-Stuff-10k is a *densely* labeled subset of COCO, comprising of 9k images for training and 1k images for testing, across 171 semantic categories [4]. We verify in Fig. 5a that bootstrapping mask-level information through CYBORGS outperforms detection-level boxes obtained from selective search, and nears performance of pretraining with fixed, stable ground truth masks.

Joint Improvement of Masks and Representations. The harmonious interplay between the representation learning and semantic segmentation components of our framework is one of our major contributions. To ascertain that representations and segmentation quality mutually improve over pretraining, we continue to assess semantic segmentation performance on the COCO-Stuff-10k dataset, for saved checkpoints throughout various stages of pretraining. For a batch of input images, we extract frozen feature maps from the same layer we use to bootstrap segmentation masks (`res2.b2`), and bilinearly interpolate to the original image dimensions. We then add a single layer of 1×1 convolutions to predict the pixel labels, yielding a final setup akin to linear probing in transfer-based evaluation.

Because only this last layer is trainable in the resulting model, segmentation performance is heavily dependent on the quality of the extracted feature maps. Since these are exactly the inputs to our KMeans segmentation algorithm, we obtain transfer results which correlate readily with the quality of our bootstrapped masks. To evaluate representation quality in the same pretrained models, we transfer the ResNet-50 backbones, unfreeze all layers, and add an FCN head, finetuning on COCO-Stuff-10k end-to-end. We perform these evaluations for CYBORGS models pretrained for 5, 25, 50, 100, 200, 500 and 800 epochs. As seen in Fig. 5b, this evaluation scheme demonstrates that mask quality and representation quality improve jointly over the course of pretraining. Note that we bootstrap masks for the first time at the *beginning* of epoch 100 using our partially pretrained backbone; a subsequent iteration over the entire dataset is sufficient to improve both mask and representation semantics significantly.

Table 2. Ablations for design choices in CYBORGS. We report average precision (AP) for object detection on PASCAL VOC `test2007`. Default settings corresp. to Table 1 are highlighted in gray.

N	AP_{all}
0	9.75
10	52.6
50	58.3
100	58.0
200	55.0
400	52.7

(a) Bootstrapping frequency. Mask bootstrapping too often or not enough leads to poor performance.

Case	AP_{all}
CRF	58.6
No CRF	58.0

(b) CRF in bootstrapping. Refining bootstrapped masks with CRFs during pretraining is not necessary.

Layers	AP_{all}
2	55.1
3	55.8
4	52.7
2+4	56.4
2+3+4	58.0

(c) Features in contrastive objective. Leveraging a semantic hierarchy of features is important.

Layers	AP_{all}
2.b2	58.0
2+3	58.2
2+4	54.4
2+3+4	55.0

(d) Features in KMeans. Earlier maps are more amenable to KMeans segmentation.

K	AP_{all}
$K = 2$	22.6
$K = 81$	42.5
$K = 256$	33.2
$K \sim \mathcal{U}[2, 256]$	58.0

(e) Scale-dynamic sampling. Dynamically sampling cluster resolution for KMeans segmentation works best.

Loss	AP_{all}
Euclidean	53.8
vMF	58.0

(f) Clustering loss. Euclidean distance for (4) leads to significant performance degradation.

M	AP_{all}
0	41.8
1	58.6
5	58.0
10	57.8
50	55.6

(g) vMF Loss Frequency. Applying the vMF curriculum more regularly leads to stronger performance.

λ	AP_{all}
0	41.5
0.001	42.3
0.01	57.8
0.1	58.0
1	57.2

(h) vMF Loss Weight. Performance is sensitive to the presence but not weight of the vMF loss.

4.4 Ablations and Discussion

All ablation models are pretrained using a ResNet-50 backbone, and evaluations are performed on PASCAL VOC detection for faster turnaround time.

Bootstrapping Frequency. We perform a sensitivity analysis on the bootstrapping frequency parameter N, where we regenerate the masks on epoch N, $2N, \ldots,$ using feature maps from the improving encoder (Table 2a). Using only the initial masks obtained under vMF warmup for pretraining (*i.e.*, $N = 0$) leads to collapsed performance. Moreover, bootstrapping the masks too frequently ($N = 10$) also leads to a performance drop, consistent with our hypothesis in Sect. 3.3 that unstable masks which are changing too rapidly can lead to representational collapse. Finally, we also note that bootstrapping too *infrequently* (*i.e.*, $N = 400$) is similarly suboptimal, validating our default chosen schedule.

CRF-Refinement of Masks. Given the qualitative improvements of the CRF-refined masks when performing the final evaluation (c.f. Fig. 4), a natural consideration is to apply CRF post-processing to the masks during every bootstrapping cycle. As we show in Table 2b, this brings only incremental improvements to the resulting representations, which we believe do not justify the increase in computational complexity. This result also further validates our claim in Sect. 4.3 that the representations under CYBORGS are already well-aligned in latent space with respect to the semantic segmentation task.

Usage of Multiple Layers of Features. Throughout our method, there are two points where we potentially use feature maps from multiple layers of our encoder

backbone. The first is in the aggregation of features for our contrastive objective in (2). We show in Table 2c that using `res2`, `res3`, `res4` from our backbone in combination is crucial to performance. This further verifies that leveraging information from across the semantic spectrum learned by the encoder is vital.

The second point is in the bootstrapping of masks, where we use only the feature map from `res2.b2` of our backbone. We show in Table 2d that feature aggregation across multiple layers does not help here. One explanation for this phenomenon is curse of dimensionality; a simple KMeans clustering procedure on extremely high dimensional features aggregated across multiple layers may result in clusters with few or no points.

Scale-Dynamic Sampling. We also perform an ablation on dynamically sampling the semantic resolution of masks during bootstrapping. We compare with a foreground-background masks ($K = 2$), object-level masks ($K = 81$ categories from COCO), and clustering with the same number of unique labels as the default graph-based segmentation algorithm used in [20]. As shown in Table 2e, fixing the KMeans cluster dimension at any level reduces the performance of CYBORGS. This validates our choice to provide the encoder with diverse levels of detail through bootstrapped masks of varying semantic resolution.

vMF Clustering Loss. We verify several properties of the clustering loss in (4). As seen in Table 2f, basing the loss on the vMF distribution, which aligns better with our hyperspherically-distributed embeddings, results in better transfer performance. We also examine the frequency at which the vMF clustering loss is applied, and how sensitive our method is to the weight of this loss. Table 2g validates that applying the loss more frequently increases downstream transfer performance. In combination with Table 2h we note the weight of the loss does not dramatically influence performance, but its presence is important; at $\lambda = 0, 0.001$ or if $M = 0$ (*i.e.*, no application of vMF loss), the performance collapses.

5 Conclusion

We have proposed CYBORGS, a novel self-supervised framework which learns object-level representations and semantic segmentation jointly, in an end-to-end fashion. In pretraining on complex scene images, our representations transfer competitively to a diverse array of downstream tasks, with particularly strong alignment with a long-tailed distribution of object-level segmentation semantics.

Acknowledgements. YG is supported by the Ministry of Science and Technology of the People's Republic of China, the 2030 Innovation Megaprojects "Program on New Generation Artificial Intelligence" (Grant No. 2021AAA0150000). YG is also supported by a grant from the Guoqiang Institute, Tsinghua University. RW would like to thank Yu Sun and Yingdong Hu for valuable edits to the paper, without which this work would not be possible.

References

1. Bai, Y., Chen, X., Kirillov, A., Yuille, A., Berg, A.C.: Point-level region contrast for object detection pre-training. arXiv preprint arXiv:2202.04639 (2022)
2. Ballard, D.H., Zhang, R.: The hierarchical evolution in human vision modeling. Top. Cogn. Sci. **13**(2), 309–328 (2021)
3. Bar, A., et al.: DETReg: unsupervised pretraining with region priors for object detection. arXiv preprint arXiv:2106.04550 (2021)
4. Caesar, H., Uijlings, J., Ferrari, V.: Coco-stuff: Thing and stuff classes in context. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1209–1218 (2018)
5. Caron, M., Bojanowski, P., Joulin, A., Douze, M.: Deep clustering for unsupervised learning of visual features. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision – ECCV 2018. LNCS, vol. 11218, pp. 139–156. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01264-9_9
6. Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A.: Unsupervised learning of visual features by contrasting cluster assignments. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in Neural Information Processing Systems, vol. 33, pp. 9912–9924. Curran Associates, Inc. (2020). https://proceedings.neurips.cc/paper/2020/file/70feb62b69f16e0238f741fab228fec2-Paper.pdf
7. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking Atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587 (2017)
8. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International Conference on Machine Learning, pp. 1597–1607. PMLR (2020)
9. Chen, T., Luo, C., Li, L.: Intriguing properties of contrastive losses. Adv. Neural. Inf. Process. Syst. **34**, 1–9 (2021)
10. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297 (2020)
11. Cho, J.H., Mall, U., Bala, K., Hariharan, B.: PiCIE: unsupervised semantic segmentation using invariance and equivariance in clustering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16794–16804 (2021)
12. Cordts, M., et al.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3213–3223 (2016)
13. Doersch, C., Gupta, A., Efros, A.A.: Unsupervised visual representation learning by context prediction. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1422–1430 (2015)
14. Gopal, S., Yang, Y.: Von mises-fisher clustering models. In: International Conference on Machine Learning, pp. 154–162. PMLR (2014)
15. Grill, J.B., et al.: Bootstrap your own latent - a new approach to self-supervised learning. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in Neural Information Processing Systems, vol. 33, pp. 21271–21284. Curran Associates, Inc. (2020). https://proceedings.neurips.cc/paper/2020/file/f3ada80d5c4ee70142b17b8192b2958e-Paper.pdf
16. Gupta, A., Dollar, P., Girshick, R.: Lvis: a dataset for large vocabulary instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5356–5364 (2019)

17. Hamilton, M., Zhang, Z., Hariharan, B., Snavely, N., Freeman, W.T.: Unsupervised semantic segmentation by distilling feature correspondences. In: International Conference on Learning Representations (2021)
18. Hariharan, B., Arbeláez, P., Girshick, R., Malik, J.: Hypercolumns for object segmentation and fine-grained localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 447–456 (2015)
19. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9729–9738 (2020)
20. Hénaff, O.J., Koppula, S., Alayrac, J.B., Oord, A., Vinyals, O., Carreira, J.: Efficient Visual Pretraining with Contrastive Detection. In: International Conference on Computer Vision (2021)
21. Herranz, L., Jiang, S., Li, X.: Scene recognition with CNNs: objects, scales and dataset bias. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 571–579 (2016)
22. Hwang, J.J., Yet al.: SegSort: segmentation by discriminative sorting of segments. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 7334–7344 (2019)
23. Jabri, A., Owens, A., Efros, A.: Space-time correspondence as a contrastive random walk. Adv. Neural. Inf. Process. Syst. **33**, 19545–19560 (2020)
24. Ji, X., Henriques, J.F., Vedaldi, A.: Invariant information clustering for unsupervised image classification and segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9865–9874 (2019)
25. Johnson, J., Douze, M., Jégou, H.: Billion-scale similarity search with GPUs. IEEE Trans. Big Data. **7**, 537–547 (2019)
26. Ke, T.W., Hwang, J.J., Yu, S.X.: Universal weakly supervised segmentation by pixel-to-segment contrastive learning. In: International Conference on Learning Representations (2021)
27. Komodakis, N., Gidaris, S.: Unsupervised representation learning by predicting image rotations. In: International Conference on Learning Representations (ICLR) (2018)
28. Kornblith, S., Shlens, J., Le, Q.V.: Do better imagenet models transfer better? In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2661–2671 (2019)
29. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected CRFs with gaussian edge potentials. Adv. Neural. Inf. Process. Syst. **24**, 1–11 (2011)
30. Kuang, H., et al.: Video contrastive learning with global context. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3195–3204 (2021)
31. Li, J., Zhou, P., Xiong, C., Hoi, S.C.: Prototypical contrastive learning of unsupervised representations. In: ICLR (2021)
32. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
33. Liu, S., Li, Z., Sun, J.: Self-EMD: self-supervised object detection without imagenet. arXiv preprint arXiv:2011.13677 (2020)
34. Mo, S., Kang, H., Sohn, K., Li, C.L., Shin, J.: Object-aware contrastive learning for debiased scene representation. Adv. Neural. Inf. Process. Syst. **34**, 1–14 (2021)
35. Noroozi, M., Favaro, P.: Unsupervised learning of visual representations by solving jigsaw puzzles. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9910, pp. 69–84. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46466-4_5

36. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
37. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: feature learning by inpainting. In: Proceedings of the IEEE Conference On Computer Vision and Pattern Recognition, pp. 2536–2544 (2016)
38. Purushwalkam, S., Gupta, A.: Demystifying contrastive self-supervised learning: invariances, augmentations and dataset biases. Adv. Neural. Inf. Process. Syst. **33**, 3407–3418 (2020)
39. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-CAM: visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 618–626 (2017)
40. Selvaraju, R.R., Desai, K., Johnson, J., Naik, N.: Casting your model: learning to localize improves self-supervised representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11058–11067 (2021)
41. Van Gansbeke, W., Vandenhende, S., Georgoulis, S., Van Gool, L.: Unsupervised semantic segmentation by contrasting object mask proposals. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 10052–10062, October 2021
42. Wang, T., Isola, P.: Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In: International Conference on Machine Learning, pp. 9929–9939. PMLR (2020)
43. Wang, X., Zhang, R., Shen, C., Kong, T., Li, L.: Dense contrastive learning for self-supervised visual pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3024–3033 (2021)
44. Xiao, T., Reed, C.J., Wang, X., Keutzer, K., Darrell, T.: Region similarity representation learning. arXiv preprint arXiv:2103.12902 (2021)
45. Xie, J., Zhan, X., Liu, Z., Ong, Y.S., Loy, C.C.: Unsupervised object-level representation learning from scene images. arXiv preprint arXiv:2106.11952 (2021)
46. Xiong, Y., Ren, M., Zeng, W., Urtasun, R.: Self-supervised representation learning from flow equivariance. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 10191–10200, October 2021
47. Xu, J., Wang, X.: Rethinking self-supervised correspondence learning: a video frame-level similarity perspective. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 10075–10085, October 2021
48. Yang, C., Wu, Z., Zhou, B., Lin, S.: Instance localization for self-supervised detection pretraining. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3987–3996 (2021)
49. You, Y.,et al.: Large batch optimization for deep learning: training BERT in 76 minutes. arXiv preprint arXiv:1904.00962 (2019)
50. Zhang, F., Torr, P., Ranftl, R., Richter, S.: Looking beyond single images for contrastive semantic segmentation learning. Adv. Neural. Inf. Process. Syst. **34**, 1–13 (2021)
51. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 649–666. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46487-9_40
52. Zhang, X., Maire, M.: Self-supervised visual representation learning from hierarchical grouping. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in Neural Information Processing Systems, vol. 33, pp. 16579–16590. Curran Associates, Inc. (2020). https://proceedings.neurips.cc/paper/2020/file/c1502ae5a4d514baec129f72948c266e-Paper.pdf