Unsupervised Object Detection Pretraining with Joint Object Priors Generation and Detector Learning

Anonymous Author(s)
Affiliation
Address
email

1 Related Works

Unsupervised pretraining for backbone. Recent unsupervised pretraining methods, which rely on pretext tasks to learn visual representations [8, 18, 16, 6, 9, 7, 27, 21, 30, 31], have shown considerable performance on transfer learning tasks, outperforming their supervised counterparts. However, compared with considerable performance gains on classification-related tasks, the improvement on dense-prediction tasks [20, 13] are limited. To this end, a growing number of works explore pretext tasks for object detection and instance segmentation. DenseCL [24] and PixPro [29] contrast pixel features on the same physical location under different views to learn pixel-level representations. DetCo [28] exploits supervision on features from different stages of the backbone and from global and local patches to learn consistent representations on image-level and patch-level. [1] proposes point-level region contrast, which enables the model to learn at the point-level to help localization, and at the region-level to help holistic object recognition. Despite the good performance, all these works only focus on pretraining the backbone of object detector, neglecting the detector heads. When these methods are transferred to object detection, the detector heads are initialized from scratch and do not benefit from pretraining, which limits their performance on object detection. In contrast, our JoinDet, which utilizes object priors generated by the model itself as supervision, pretrains the entire model to promote detector learning.

Unsupervised pretraining for object detector. Pretraining the backbone with an pretext task for dense-prediction tasks leaves untrained detection heads which are also a core component when transferring to object detection [19]. Few works attempt to remedy this problem by pretraining the entire detector with various unsupervised pretext tasks. SoCo [26] utilizes selective search to generate object priors and perform contrastive learning on object-level features from the detector head. UP-DETR [10] and DETReg [2] pretrain the detection heads of DETR [3] by forcing them to predict object priors generated by randomly cropping and selective search, respectively. However, randomly cropping hardly provides any effective object prior, and selective search is a heuristics method which is time-consuming, independent from the pretraining process. In contrast to these methods, our proposed JoinDet jointly generates object priors and learns detection, which can gradually update the object priors with learned and improved ones for better supervision during pretraining.

Attention in unsupervised pretraining as supervision. NNCLR [12] and DINO [5] show that the attention maps of the visual transformer can generate semantic segmentation masks even though the model is pretrained without labels. This suggests that self-learned attention can provide effective supervision for dense-prediction tasks. STEGO [17] utilizes off-the-shelf pretrained DINO to extract cross-image feature correspondence (cross-image attention) as supervision to distill segmentation features and train a unsupervised segmentation model. Different from STEGO, our JoinDet exploits the self-attention maps in the transformer encoder to generate multiple object priors as supervision during training. We also show that the self-attention maps can be jointly refined during training to generate progressive object priors for better supervision.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

2 More experimental results

One key factor that contributes to the success of JoinDet is using progressively refined object priors as supervision. We have already shown that the selection of effective object priors have a huge impact on finetuning performance in Sec.4 of the main text. Here, we provide more experimental results to explore the influence of hyperparameters in JoinDet and discuss the possible direction for future works. We implement experiments on single-scale deformable DETR [32]. Unless otherwise specified, we set the momentum coefficient in the Box Smooth Module as 0.45, the clustering IoU threshold in the Box Smooth Module as 0.48. The supervision generated from object priors is updated every 10 epochs by default. JoinDet is pretrained on COCO for 50 epochs and finetune on VOC for 25 epochs. We train 3 different models with different random seeds and report the mean result of AP (COCO format) on VOC.

2.1 Momentum coefficient in the Box Smooth Module

The momentum coefficient m^s in Box Smooth Module controls the shifting speed of supervision, which considers both precedent object priors and current object priors. We ablate the most suitable momentum coefficient for JoinDet in Tab. 1. Firstly, small momentum coefficients, which are smaller than 0.45, represent relative fast shifting speed of supervision, showing significant performance drops. Concretely, when $m^s = 0$, the supervision will be directly replaced with current object priors, neglecting useful precedent object priors and leading to **-2.4** AP drop. Second, when the shifting speed is too slow ($m^s = 0.70$), behindhand object priors are insufficient to guide the current model, which is also harmful (55.4 AP→53.7 AP) for JoinDet.

Table 1: Pretrain JoinDet with different momentum coefficients. When momentum coefficient $m^s = 0$, the supervision will be directly changed to current object priors. AP on VOC is reported.

Method	ms	10 epochs	25 epochs
DETReg	-	46.0	53.9
JoinDet	0.70	47.1	53.7
	0.45	**49.0**	**55.3**
	0.20	48.3	54.3
	0.05	47.7	54.3
	0	48.0	53.0

2.2 Clustering IoU threshold in the Box Smooth Module

When precedent object priors and current object priors have large IoUs, which are bigger than the threshold, corresponding priors (boxes) will be clustered in the same cluster. The box coordinates and scores of all boxes in a specific cluster will be used to generate a new box for supervision. Experimental results of using different clustering IoU thresholds are summarized in Tab. 2. First, we find 0.48 as an optimal hyperparameter, suggesting that duplicate object priors with larger thresholds and scarce object priors with smaller thresholds are both harmful for pretraining. Second, the performance variation with different cluter IoU thresholds are relatively slight (at most -1.3 AP), which indicates that our proposed method is robust to the clustering IoU thresholds.

2.3 Update frequency

As generated object priors are progressively refined during pretraining, we update object priors every 10 epochs as the supervision. As shown in Tab. 3, when the momentum coefficient is fixed (0.45), updating the supervision too frequently (every 1 epoch) leads to a significant performance drop, which indicates that a stable supervision is very important to unsupervised pretraining for object detector. We argue that the performance drop brought by frequent updating can be remedied with a proper momentum coefficient as discussed in Sec.2 of the main text, which we remain for the future work.

Table 2: Pretrain JoinDet with different clustering IoU thresholds. AP on VOC is reported.

Method	IoU threshold	10 epochs	25 epochs
DETReg	-	46.0	53.9
JoinDet	0.35	46.2	54.1
	0.40	47.1	53.9
	0.48	**49.0**	**55.4**
	0.55	48.1	54.8
	0.60	48.4	54.9
	0.65	47.9	54.8

Table 3: Pretrain JoinDet with different update frequencies. AP on VOC is reported.

Method	Update frequency	10 epochs	25 epochs
DETReg	-	46.0	53.9
JoinDet	1 epoch	40.7	51.3
	5 epochs	46.6	54.0
	10 epochs	**49.0**	**55.4**
	20 epochs	46.8	54.6

3 Additional visualization

Fig. 1 visualizes more progressively refined object priors by JoinDet and fixed object priors by selective search. For select search, we only visualize top 15 object priors. JoinDet generates object priors with less background regions than selective search.

4 The eigen attention map computation method $\mathcal{K}$

According to [25], the eigen attention map in the vision transformer can highlight salient foregrounds by partitioning all features $\mathbf{f}_i \in \mathbb{R}^c$ in output patch features $\mathcal{F} \in \mathbb{R}^{h \times w \times c}$ into the background set $\mathcal{F}^b$ and the foreground set $\mathcal{F}^f$, where $i \in [1, hw]$, and h, w, c denote the height, width, and dimension of output patch features $\mathcal{F}$, respectively. Following [25, 23], we fix the feature partition task by solving a group partition problem on a self-similarity graph $\mathcal{S} = (\mathcal{V}, \mathcal{U})$, where the nodes $\mathcal{V}$ represent all features on $\mathcal{F}$ and the edges $\mathcal{U}$ are based on the cosine similarity between corresponding features, which can be computed by

$$\begin{aligned} \mathcal{U}_{i,j} &= \begin{cases} 1, & \text{if } \cos(\mathbf{f}_i, \mathbf{f}_j) \geq \tau \\ \epsilon, & \text{otherwise} \end{cases}, \\ \cos(\mathbf{f}_i, \mathbf{f}_j) &= \frac{\langle \mathbf{f}_i, \mathbf{f}_j \rangle}{\|\mathbf{f}_i\|_2 \cdot \|\mathbf{f}_j\|_2}, \end{aligned} \tag{1}$$

where $\mathcal{U}_{i,j}$ denotes the edge between feature $\mathbf{f}_i$ and feature $\mathbf{f}_j$, cos denotes the cosine similarity, τ is a hyper-parameter and ϵ equals a small positive value to ensure that the graph is fully-connected. To partition the graph $\mathcal{S}$ into tow disjoint sets $\mathcal{F}^f$ and $\mathcal{F}^b$, we simply remove edges connecting the two parts. The optimal bi-partitioning of the graph $\mathcal{S}$ can be solved by minimizing the Ncut energy $\mathbb{E}$ [23, 25]:

$$\min_{\mathcal{F}^f, \mathcal{F}^b} \mathbb{E}(\mathcal{F}^f, \mathcal{F}^b) = \min_{\mathcal{F}^f, \mathcal{F}^b} \left[\frac{\mathrm{C}(\mathcal{F}^f, \mathcal{F}^b)}{\mathrm{C}(\mathcal{F}^f, \mathcal{V})} + \frac{\mathrm{C}(\mathcal{F}^f, \mathcal{F}^b)}{\mathrm{C}(\mathcal{F}^b, \mathcal{V})} \right], \tag{2}$$

where $\mathrm{C}(\mathcal{F}^b, \mathcal{F}^f) = \sum_{\mathbf{u} \in \mathcal{F}^b, \mathbf{t} \in \mathcal{F}^f} \mathcal{U}_{\mathbf{u},\mathbf{t}}$ measures the degree of similarity between two sets. By reducing Eq. 2, maximizing the similarity within the sets and minimizing the dissimilarity between two sets can be satisfied simultaneously [23].

Let $\mathbf{1}$ be an vector of all ones, and $\mathbf{x}$ be an dimensional indicator vector, $\mathbf{x}_i = 1$ if node i is is in $\mathcal{F}^f$ and -1, otherwise. Indicating in [23], the optimization problem in Eq. 2, which is NP-complete, can

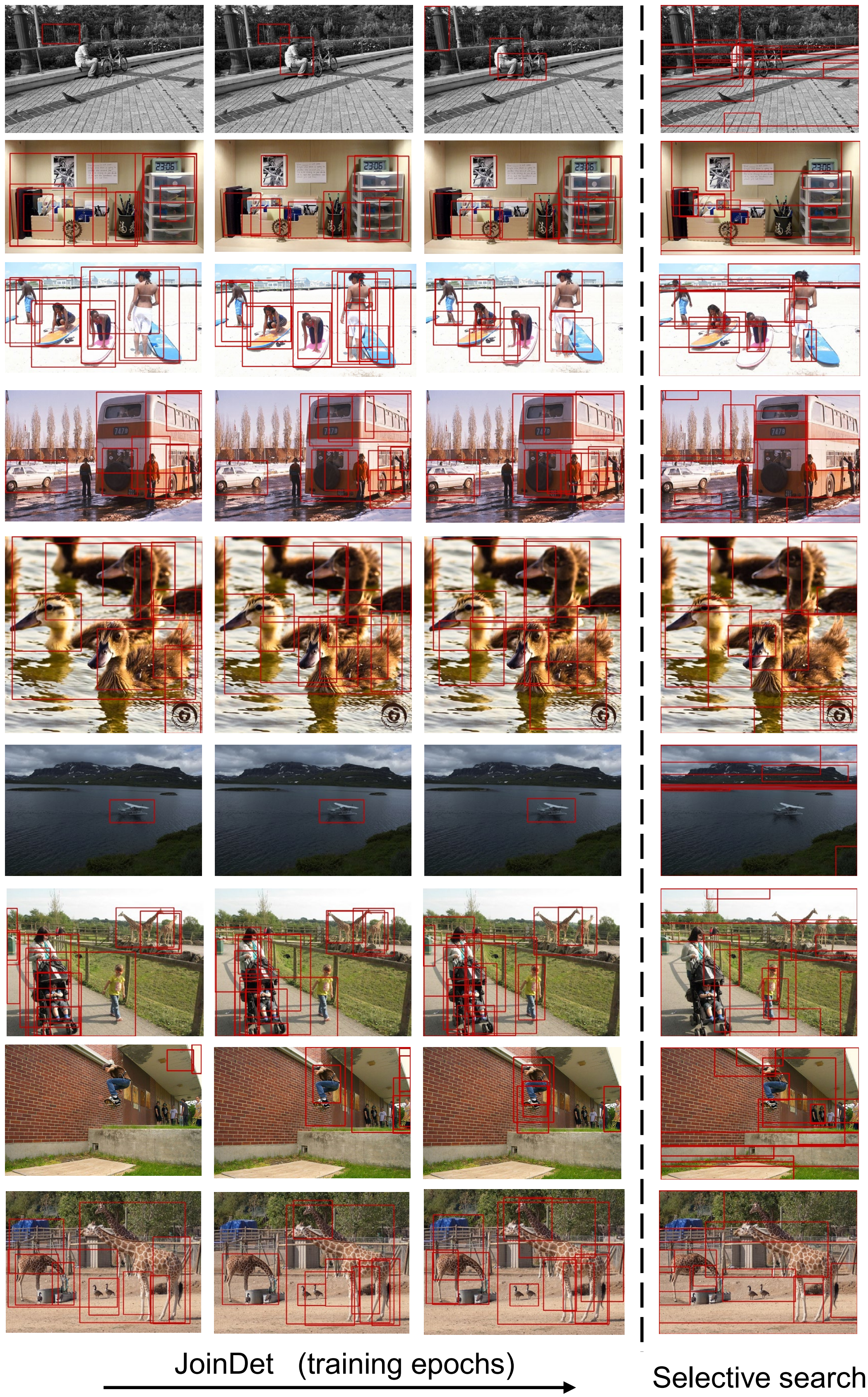

Figure 1: Evolution of object priors generated by JoinDet and object priors generated by selective search. We show that progressively refined object priors in JoinDet contains less background regions.

be equivalently substituted by

$$\min_{\mathbf{x}} \mathbb{E}(\mathbf{x}) = \min_{\mathbf{y}} \frac{\mathbf{y}^T(\mathbf{D}-\mathcal{U})\mathbf{y}}{\mathbf{y}^T\mathbf{D}\mathbf{y}}, \tag{3}$$

where $\mathbf{D}$ is a diagonal matrix with total connection from node i to all other nodes $\mathbf{d}(i) = \sum_j \mathcal{U}_{i,j}$ on its diagonal, $\mathbf{y} \in \{1, -b\}$ and b satisfies $\mathbf{y}^T\mathbf{D}\mathbf{1} = 0$.

Eq. 3 is the Rayleigh quotient [15]. If $\mathbf{y}$ is relaxed to take on real values, Eq. 3 can be minimized by solving

$$(\mathbf{D}-\mathcal{U})\mathbf{y} = \lambda\mathbf{D}\mathbf{y}. \tag{4}$$

Let $\mathbf{z} = \mathbf{D}^{-\frac{1}{2}}\mathbf{y}$, we can rewrite Eq. 4 as

$$\mathbf{D}^{-\frac{1}{2}}(\mathbf{D}-\mathcal{U})\mathbf{D}^{-\frac{1}{2}}\mathbf{z} = \lambda\mathbf{z}. \tag{5}$$

And the energy in 3 can be rewrote as

$$\min_{\mathbf{z}} \frac{\mathbf{z}^T\mathbf{D}^{-\frac{1}{2}}(\mathbf{D}-\mathcal{U})\mathbf{D}^{-\frac{1}{2}}\mathbf{z}}{\mathbf{z}^T\mathbf{z}}. \tag{6}$$

It can be easily proofed that $\mathbf{z}_0 = \mathbf{D}^{-\frac{1}{2}}\mathbf{1}$ is an eigenvector of Eq. 5 with eigenvalue of 0, which satisfied the constraint $\mathbf{y}^T\mathbf{D}\mathbf{1} = 0$. As $(\mathbf{D}-\mathcal{U})$, called the Laplacian matrix, is positive semidefinite, $\mathbf{D}^{-\frac{1}{2}}(\mathbf{D}-\mathcal{U})\mathbf{D}^{-\frac{1}{2}}$ is symmetric positive semidefinite [22]. Therefore $\mathbf{z}_0$ is the smallest eigenvector of Eq. 5, and $\mathbf{z}_1$, the second smallest eigenvector of Eq. 5, is perpendicular to $\mathbf{z}_0$ [23]. According to the Rayleigh quotient [15], $\mathbf{z}_1$, the second smallest eigenvector of Eq. 5, is the real valued solution to minimize the energy in Eq. 6,

$$\mathbf{z}_1 = \arg\min_{\mathbf{z}^T\mathbf{z}_0} \frac{\mathbf{z}^T\mathbf{D}^{-\frac{1}{2}}(\mathbf{D}-\mathcal{U})\mathbf{D}^{-\frac{1}{2}}\mathbf{z}}{\mathbf{z}^T\mathbf{z}}. \tag{7}$$

Consequently, taking $\mathbf{z} = \mathbf{D}^{-\frac{1}{2}}\mathbf{y}$,

$$\mathbf{y}_1 = \arg\min_{\mathbf{y}^T\mathbf{D}\mathbf{1}=0} \frac{\mathbf{y}^T(\mathbf{D}-\mathcal{U})\mathbf{y}}{\mathbf{y}^T\mathbf{D}\mathbf{y}}. \tag{8}$$

Therefore, $\mathbf{y}_1$, the second smallest eigenvectoer of Eq. 4, is the real valued solution that achieves the optimal partition with Ncut energy $\mathbb{E}$ in Eq. 2.

We then reshape the second smallest eigenvectoer $\mathbf{y}_1$ to the eigen attention map $\mathcal{M} \in \mathbb{R}^{h\times w}$, which has the same height and width with output patch features $\mathcal{F}$.

5 Training Details

5.1 Pretraining

Following DETReg [2], we initialize the ResNet50 backbone of JoinDet with SwAV [4], which was pretrained on ImageNet1K [11] for 800 epochs, and fix the backbone during pretraining. Furthermore, a same SwAV encoder is used to extract features of object priors, which are cropped and resized to 128×128. JoinDet follows the default hyperparameter setting and training strategy used in Deformable DETR [32], except that the object embbeding loss with loss weight 1. On COCO [20], models are trained for 50 epochs and the learning rate is decayed by a factor of 0.1 at epoch 40. On ImageNet [11], following DETReg [2], we train models for 5 epochs. Following Deformable DETR [32], we train our models using the Adam optimizer with a base learning rate of 2×10^{-4}, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and set the weight decay as 10^{-4}. We use large scale jittering mentioned in [14] as additional augmentation to alleviate the scale imbalance problem in generated object priors.

5.2 Evaluation

We finetune JoinDet on COCO [20], VOC [13] to evaluate our method. When finetuning, the original classification branch f_{cls} and the object embbeding branch are dropped. We initial a new classification branch using a single fully-connect layer with output dimension c, where c denotes the total categories in the downstream detection datasets.

Full-data finetuning. For COCO, we finetune models for 50 epochs and the learning rate is decayed by a factor of 0.1 at the 40-th epoch. For VOC, following DETReg [2], models are trained for 100 epochs with the learning rate decayed by a factor of 0.1 at the 70-th epoch.

Low-Data regimes object detection. Following DETReg [2], we finetune JoinDet with 1%, 10% COCO training set data with 2000 epochs, 400 epochs, respectively. The base learning rate is set as 2×10^{-4} and the learning rate is decayed by a factor of 0.1 at the 1400-th epoch, the 280-th epoch, respectively.

6 Broader impact

We present a more effective general unsupervised object detection pretraining method which can jointly generate object priors and learn to detect. Compared with supervised learning, our method eases the burden of expansive and time-consuming manual labels and benefits from rapidly increasing real-word data. Meanwhile, our method can promote the development on smart healthcare because it can be directly use on medical images without labeling by expertise.

However, several potential issues should be taken into consideration when applied it in real-world scenario. First, similar to other learning methods, there still remains concerns about the interpretability and robustness. Second, pretrained on manually collected datasets, the method might learn biased features when given with biased datasets. Finally, like other unsupervised pretraining methods, our method relies on extra epochs to pretrain the model, which is not efficient during pretraining, leading to more electricity consumption.

References

[1] Yutong Bai, Xinlei Chen, Alexander Kirillov, Alan Yuille, and Alexander C Berg. Point-level region contrast for object detection pre-training. *arXiv preprint arXiv:2202.04639*, 2022.

[2] Amir Bar, Xin Wang, Vadim Kantorov, Colorado J Reed, Roei Herzig, Gal Chechik, Anna Rohrbach, Trevor Darrell, and Amir Globerson. Detreg: Unsupervised pretraining with region priors for object detection. *arXiv preprint arXiv:2106.04550*, 2021.

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.

[4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.

[5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

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

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

[9] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, 2021.

[10] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. In *CVPR*, 2021.

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[12] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *ICCV*, 2021.

[13] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010.

[14] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *CVPR*, 2021.

[15] Gene H Golub and Charles F Van Loan. *Matrix computations*. 2013.

[16] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. In *NeurIPS*, 2020.

[17] Mark Hamilton, Zhoutong Zhang, Bharath Hariharan, Noah Snavely, and William T Freeman. Unsupervised semantic segmentation by distilling feature correspondences. *arXiv preprint arXiv:2203.08414*, 2022.

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

[19] Gabriel Huang, Issam Laradji, David Vazquez, Simon Lacoste-Julien, and Pau Rodriguez. A survey of self-supervised and few-shot object detection. *arXiv preprint arXiv:2110.14711*, 2021.

[20] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

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

[22] Alex Pothen, Horst D Simon, and Kang-Pu Liou. Partitioning sparse matrices with eigenvectors of graphs. *SIMAX*, 1990.

[23] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *TPAMI*, 2000.

[24] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *CVPR*, 2021.

[25] Yangtao Wang, Xi Shen, Shell Hu, Yuan Yuan, James Crowley, and Dominique Vaufreydaz. Self-supervised transformers for unsupervised object discovery using normalized cut. *arXiv preprint arXiv:2202.11539*, 2022.

[26] Fangyun Wei, Yue Gao, Zhirong Wu, Han Hu, and Stephen Lin. Aligning pretraining for detection via object-level contrastive learning. 2021.

[27] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018.

[28] Enze Xie, Jian Ding, Wenhai Wang, Xiaohang Zhan, Hang Xu, Peize Sun, Zhenguo Li, and Ping Luo. Detco: Unsupervised contrastive learning for object detection. In *ICCV*, 2021.

[29] Zhenda Xie, Yutong Lin, Zheng Zhang, Yue Cao, Stephen Lin, and Han Hu. Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. In *CVPR*, 2021.

[30] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*, 2021.

[31] Shaofeng Zhang, Feng Zhu, Junchi Yan, Rui Zhao, and Xiaokang Yang. Zero-cl: Instance and feature decorrelation for negative-free symmetric contrastive learning. In *ICLR*, 2021.

[32] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2020.