



# Self-supervised Learning in Computer Vision: A Review

Zibei Wang<sup>(✉)</sup>

Tiangong University, Tianjin 300387, China  
15032797455@163.com

**Abstract.** In computer vision, data-driven convolutional neural networks could learn increasingly rich semantic features of images. However, manual annotation of images is an expensive and time-consuming task that hinders development. As a branch of unsupervised learning, self-supervised learning does not rely on labels, avoiding the work of labeling the data. This paper provides a comprehensive discussion of the development of self-supervised learning in computer vision. First, we briefly describes the motivation for proposing self-supervised learning and related concepts, introduces the self-supervised learning paradigm from three aspects, describes the applications of self-supervised learning in computer vision, and finally provides a summary and an outlook on its future development.

**Keywords:** Self-supervised learning · Deep learning · Computer vision

## 1 Introduction

Computer vision tasks require the extraction of image features before proceeding to the next processing step. Initially, traditional feature extraction methods could only extract simple low-level features [1, 2]. The advent of deep learning brought new approaches to feature extraction, and it gradually became mainstream to use labeled data to train convolutional neural networks to extract image features [3]. We refer to such deep learning driven by labeled data as supervised learning. However, with continuous development, supervised learning reveals some drawbacks: poor feature generalization, vulnerability to attacks, etc., and manually labeled data is the main reason limiting its development. Therefore, unlabelled data has become a possible way to break the bottleneck. As the name implies, self-supervised learning is ‘supervised by itself’. As a subclass of unsupervised learning, it uses the image’s information as supervision for training to learn an effective feature representation. The diagram illustrates the training process, using only images for training in the pretext task and then transferring the ConvNets to the downstream task for fine-tuning using labeled data. Our paper summarises and presents the recent developments and applications of self-supervised learning in computer vision and provides an outlook on its future development.

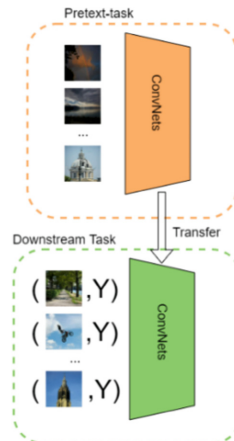
## 2 Background

### 2.1 Visual Feature Learning

Based on the data labels, we can classify visual feature learning into four modes: supervised, semi-supervised, weakly supervised, and unsupervised. The most common one in computer vision is supervised learning. Each image  $X$  has a corresponding label  $Y$ , and the training goal is to narrow the gap between the prediction result and the ground truth. The dataset for semi-supervised learning contains a small amount of labeled data and a large amount of unlabeled data. Each image has a corresponding label in weakly supervised learning, but the label contains noise or is incorrectly labeled. All photos in the dataset are unlabeled for unsupervised learning. The self-supervised learning we present in this paper is a subclass of unsupervised learning and differs from unsupervised learning in that self-supervised learning provides pseudo-labeled supervised training by designing the pretext task.

### 2.2 Self-supervised Learning

Self-supervised learning is divided into two phases: pretext task and downstream task. In the pre-training, the pretext task is designed according to the downstream task. Using pseudo-labeled supervised training, ConvNets learn to extract image features; then, ConvNets transfer to the downstream task and the model is fine-tuned using a small amount of labeled data. Self-supervised learning is achieved because ConvNets learn the feature representation needed for downstream task by solving the pretext task. The usual flow of self-supervised learning is shown in Fig. 1.



**Fig. 1.** Self-supervised learning flow chart.

### 3 Self-supervised Learning Paradigm

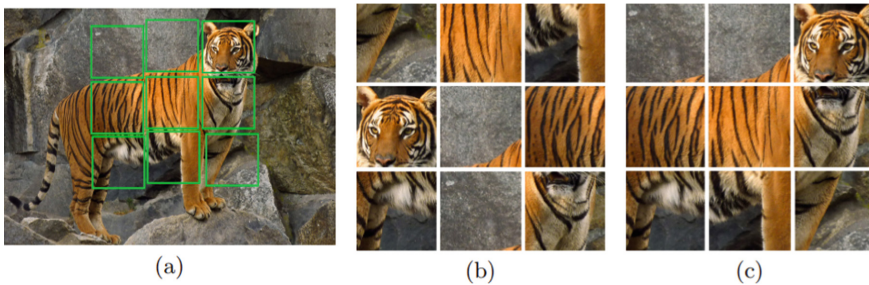
The pretext task teaches ConvNets to extract features. ConvNets are trained by minimizing the error between the pseudo label and the predicted value. We classify the self-supervised learning paradigms into three categories: context-based, generation-based, and contrast learning, depending on the type of pre-task.

#### 3.1 Context-Based Image Feature Learning

Context-based pretext tasks are usually designed based on the semantic feature associations between each image part. After pre-training, the model can learn semantic information about the different objects and between objects in the image.

Clustering is a commonly used method in unsupervised learning to classify images by extracting semantic features [4]. The distance between image features in the same cluster in the feature space is as small as possible. The feature space distance between images in different clusters is as large as possible.

Learning feature representations by identifying the rotation angle of an image was proposed by et al. [5]. To recognize the rotation angle of the image, the model needs to understand the feature information in the image that describes the subject, such as position, category, etc. In the pretext task, the authors applied four rotation angles - 0,90,180,270 - to the original input image as pseudo-labels. After the transformation, the image was passed through ConvNets to predict the rotation angle. However, the method has limitations in learning images with rotational invariance. Feng et al. [6] proposed an improved method that considers rotationally invariant images to address this limitation. In the pretext task, the authors add a branch. For images with rotation invariance, their features are mapped onto the feature space, and the image feature representation is learned by computing the distance between feature vectors.



**Fig. 2.** The visualization of the Jigsaw Image Puzzle.

In 2015, Doersch et al. [7] divided images into nine patches of equal size, randomly selected two of them, and predicted the relative positions. The method provides ideas for designing pretext tasks using image spatial contextual relationships, and researchers proposed a series of tasks related to the spatial location of image patches [8–10]. Noroozi et al. [8] proposed a nine-patch sorting task between image patches, as shown in Fig. 2.

Figure 2(a) represents the selection of nine patches from an image, Fig. 2(b) shows the disordered order of the to-be-sorted patches, and Fig. 2(c) shows the image after reordering. In addition, the creation of subsets of permutations needs to be carefully considered—the number of permutations increases, especially when only two patches differ between the two ordering methods. Therefore, the authors used the Hamming distance to filter the ranking subsets to obtain an appropriate ranking subset.

### 3.2 Generation-Based Image Feature Learning

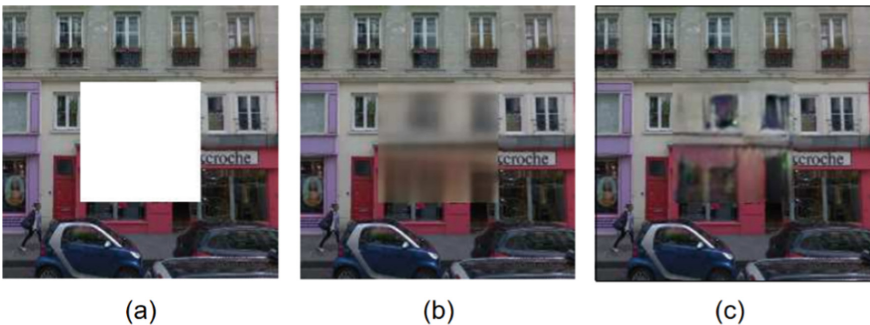
For learning the feature representation, generative-based pretext tasks are often built on Generative Adversarial Networks (GAN) [11] and Autoencoders. In such pretext tasks, the original image is usually pre-trained as a pseudo-labeled supervised model.

GAN has two main components: the generator(G) and the discriminator(D). The G tries to “trick” the D by generating images based on latent vectors similar to the true values. The D training goal is to distinguish the true images from latent vectors generated by true vectors. Mathematically, the game between the generator and the discriminator is defined as:

$$\min_G \max_D \mathbb{E}_{X \sim p_{data}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))] \quad (1)$$

The Autoencoder is an unsupervised learning model. The training process consists of encoding and decoding, with the input image  $X$  being encoded to obtain latent vectors and the reconstructed image  $X'$  being obtained through the decoding process. The model is trained on the original image  $X$  to learn the mapping relationships. MSE Loss is usually used as the objective function.

As shown in Fig. 3, Pathak et al. [12] propose to learn the feature representation of an image by repairing the missing part of the image. The middle region of the image is masked to obtain Fig. 3(a), which is subsequently fed into an auto-encoder-like - contextual encoder proposed by the authors to repair the masked region by learning features from the remaining part of the image. The image to be repaired is fed into the encoder to obtain semantic features, and then the decoder generates the repaired image based on the learned features. The context encoder needs to understand the entire image



**Fig. 3.** (a) is the masked image, (b) is the use of reconstruction loss, and (c) is the use of reconstruction loss + adversarial loss.

to succeed in this task. Also, the authors found that adding an adversarial loss to the pixel-level reconstruction loss yields better restoration performance, and the adversarial loss optimizes the model to generate a more realistic image, as shown in Fig. 3(b) and (c). After experiments, the rich semantic features can be learned by restoring the image in the pretext task.

The coloring pretext task is the process of generating a color image from a grey-scale image [13–15]. In the pretext task, Zhang et al. [15] train a model to learn the semantic features of the image to recognize different objects, assigning a color value to each pixel in the image. The authors treat the problem as a classification task, generating color channels based on grey-scale images and using class rebalancing during training to increase the diversity of colors in the results.

For the feasibility of the pretext task, the authors used the “coloring Turing test” to assess it. Participants were asked to choose between the generated image and ground truth. After testing, the images generated by this method successfully deceived 32% of humans, which is significantly higher than previous methods.

Like the coloring task, Zhang et al. [16] propose a pretext task for cross-channel predicting. At the same time, the authors propose that the coloring task suffers from unequal treatment of the different channel features of the image, learning only from the grey-scale image, while the color image is only used to calculate the loss. Therefore, the authors attempt to utilize the full input information during cross-channel coding, allowing different channels to predict each other. The authors split the conventional Autoencoder into two sub-networks connected but not intersecting. The two sub-networks predict other subsets of the input channels between each other. For example, for the coloring task, one sub-network predicts the color channel (channels a and b) based on the L channel, and the other sub-network implements the opposite task (with channels a and b predicting the L channel). The two sub-networks work together to learn the feature representation from the input image.

### 3.3 Contrastive Learning

Contrastive learning is the most common and most important in self-supervised learning. Initially contrastive learning maximises the estimation of mutual information between different views of an image [17–19]. In subsequent studies, e.g., SimCLR [20], MoCo [21, 22], etc., requires the construction of positive and negative two samples, for image  $X$  and a series of data enhancement operations  $T$ . We randomly apply two data enhancement operations  $t_1, t_2 \sim T$ . The resulting images  $t_1(X_1)$  and  $t_2(X_2)$  is positive samples. In contrast, in contrastive learning, the similarity between feature vectors is usually measured using the cosine similarity measure by calculating the cosine angle between the two vectors, as shown below:

$$\cos\_sim(A, B) = \frac{A \cdot B}{\|A\| \|B\|} \quad (2)$$

The objective function of the contrastive learning approach uses a loss representation called InfoNCE:

$$L_{infoNCE} = -\log \frac{\exp(sim(q, k_+)/\tau)}{\exp((sim(q, k_+)/\tau) + \sum_{i=0}^K \exp(sim(q, k_i)/\tau))} \quad (3)$$

where  $q$  is the original sample,  $k_+$  is the positive sample, and  $k_i$  is the negative sample.  $\tau$  is the temperature coefficient. The  $\text{sim}(\cdot)$  function calculates the similarity between two eigenvectors, usually using the cosine similarity introduced in Eq. 2.

SimCLR uses many batches to store negative samples and a training approach similar to a self-distillation model. In contrast, MoCo employs a queue and a moving-averaged encoder to generate a dynamic lookup dictionary.

The presence of negative samples in the original contrastive learning model was critical in preventing the model from collapsing to a trivial solution. Still, the requirement to compare with many negative samples each time increased the difficulty of training while also placing a significant demand on computational resources. As a result, academics have begun to investigate new contrastive learning algorithms to discover new approaches to avoid model collapse and increase model stability without negative samples.

A new contrastive learning algorithm, BYOL [23], was proposed by Grill et al. using only positive samples. The authors constructed asymmetric networks, i.e., an online and a target network. To prevent learning a trivial solution, the authors give a fixed random initialization of the target network, use a stop-gradient for the target network during training, and update the target network with a moving average used by the online network continuous iterations. BYOL is one of the first new contrastive learning algorithms to eschew negative samples and achieves SOTA performance. The authors demonstrate that negative samples are not necessary to prevent model collapse and that the network can be more stable without using negative samples.

From an implementation point of view, BYOL is MoCo without the use of negative pairs. In contrast, the SimSiam proposed by Chen et al. [24] is based on a simpler Siamese network implementation. Instead of using negative samples and momentum encoding in the implementation, the authors use the stop-gradient to prevent the creation of collapsing solutions. Siamese network was also a basis for the success of contrastive learning utilizing only positive samples in an experiment. Because invariant induction bias is introduced during the training process of twin networks, two improved views of the same image should generate the same output. This conclusion offers new directions for future contrastive learning research.

## 4 Application of Self-supervised Learning in Computer Vision

After pre-training with self-supervised learning, the model is transferred to different downstream tasks using a small amount of labeled data for fine-tuning. This section presents dense representation learning and image aesthetic assessment. At the same time, as a recent research hotspot in computer vision, we present the application of Transformer in combination with self-supervised learning.

### 4.1 Dense Representation Learning

Various paradigms of self-supervised learning show good performance on instance-level image processing tasks. However, the feature representations learned, which are sufficient for tasks such as image classification [25–28], are based on the image globally

and ignore the focus on pixel-level features. Still, many other computer vision tasks are related to dense representation, such as image segmentation, target detection, etc. In these tasks, models need to learn more detailed pixel-level features.

Pinheiro et al. [25] propose View-Agnostic Dense Representation (VADeR), an intensive representation learning based on self-supervised learning. To learn pixel-level features of an image, the authors train the model to recognize different views of the same part of the image, for example, the recognition of the eye region of a dog. The authors modified NCE to accommodate pixel-level contrastive learning in the form of a cosine similarity measure of pixel-level similarity. The VADeR-trained model has good results in target detection, keypoint detection, and instance segmentation.

Pixel-level contrastive learning can compensate for the deficiencies in spatial sensitivity of features learned by self-supervised learning and Xie et al. [26] propose new contrastive learning, PixPro. Unlike VADeR, PixPro does not use negative samples in the training process. The authors input two channels with different views of a local feature of the image. The difference between the two channels compared is that one of the channels has an additional Pixel Propagation Module (PPM), which acts as a smoothing-like function. The objective function calculates the pixel differences and consists of two parts: 1) neither is subjected to PPM, and 2) one is subjected to PPM.

## 4.2 Image Aesthetic Assessment

Image aesthetic evaluation is a branch of computational aesthetics. With the proliferation of mobile filming devices, simple reliance on manual screening is no longer sufficient for the exponential growth of images on internet platforms. As a result, computer-assisted human assessment of the aesthetic quality of images has emerged [29]. The aesthetic evaluation of images by humans is somewhat subjective. When undertaking manual annotation, many evaluations from evaluators with various backgrounds must be collected to produce assessments that represent the aesthetics of the general population.

Sheng et al. [30] were the first to propose a self-supervised image aesthetic assessment, providing a new paradigm. The authors propose to design a predicate task using the relationship between some degradation operations and the aesthetic quality of an image. Compared to self-supervised learning methods and supervised image aesthetic evaluation methods, the authors learn the aesthetic features by predicting the class of operations with degradation intensity and eventually achieving good performance on image aesthetic benchmark datasets - AVA, AADB, and CUHK-PQ. Building on the research of using degradation operations, Pirsf et al. propose an innovation [31]. They classified the degradation operations into three categories, enriching the variety of aesthetic quality degradation operations. Also, as the degree of aesthetic quality degradation resulting from images under the same degradation operation with different parameters applied varies, the authors added relative aesthetic quality ranking to the predicate task. The method showed the best performance for aesthetic assessment on both the AVA and TID2013 datasets. Ching et al. [32] designed the pretext task based on saliency and composition features. Inspired by [7], images masked with specific regions are repaired by a generator and then discriminated using a discriminator in the pretext task. When the discriminator is transferred to the downstream task, it automatically focuses on specific

regions of the image. After experiments, the authors found that masking the intersection region of Rule-of-Third achieved better results in the downstream task.

### 4.3 Work with Transformer

Originally widely used in NLP, Transformer was introduced to computer vision by Dosovitskiy et al. [33] in 2020. It was applied to image classification tasks and showed better classification results than CNNs. As two research directions that have recently received much attention in computer vision, researchers have combined self-supervised learning with the Vision Transformer in vision tasks to bring about performance innovations. At the same time, since Vision Transformer can only achieve performance beyond that of CNNs when using large-scale datasets (14M-300M), such as ImageNet-21k, JFT-300M, this imposes a manual labeling burden. Therefore, self-supervised learning with Vision Transformer makes it possible to train using large-scale datasets while avoiding manual labeling.

MoCo v3 [34] was proposed by replacing the backbone with Transformer based on the previous method. Meanwhile, Chen et al. found that instability is an important reason affecting the model's performance and proposed a thought about improving the stability of the model. The authors argue that whether the patch projection layer is involved in training greatly impacts the stability of the model. They find that using a random patch projection layer can effectively retain the information in patches.

To investigate whether self-supervised learning fuels Vision Transformer to extract richer image features, Caron et al. proposed a simple self-distillation model called DINO [35]. The student model was optimized by minimizing the cross-entropy loss between the predictions of the teacher-student model, which was derived from the student model in previous training rounds. The final experimental results show that the Vision Transformer can learn clear semantic segmentation of images under self-supervised learning, which is not available with the supervised Vision Transformer and CNNs.

## 5 Conclusion and Future

In this paper, we provide a general summary of the application of self-supervised learning in computer vision. From the emergence of self-supervised learning to the development of different paradigms of self-supervised learning and their applications in computer vision, Yann Lecun's talk at AAAI 2020 presented the difficulties that deep learning is currently facing and that the widespread use of self-supervised learning is an inevitable choice for the development of deep learning. Training with large-scale datasets brings stunning results and a huge manual labeling task. Therefore, we need to pay attention to and explore the potential of self-supervised learning. It is hoped that self-supervised learning can drive the continuous development of the field of computer vision in future research.

## References

1. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), vol. 1, pp. 886–893. IEEE (2005)



2. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision* **60**(2), 91–110 (2004)
3. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778 (2016)
4. Caron, M., Bojanowski, P., Joulin, A., Douze, M.: Deep clustering for unsupervised learning of visual features. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 132–149 (2018)
5. Gidaris, S., Singh, P., Komodakis, N.: Unsupervised representation learning by predicting image rotations. *arXiv preprint [arXiv:1803.07728](https://arxiv.org/abs/1803.07728)* (2018)
6. Feng, Z., Xu, C., Tao, D.: Self-supervised representation learning by rotation feature decoupling. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10364–10374 (2019)
7. 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)
8. Noroozi, M., Favaro, P.: Unsupervised learning of visual representations by solving jigsaw puzzles. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *Computer Vision – ECCV 2016*. *ECCV 2016*. LNCS, vol. 9910, pp. 69–84. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46466-4\\_5](https://doi.org/10.1007/978-3-319-46466-4_5)
9. Kim, D., Cho, D., Yoo, D., Kweon, I.S.: Learning image representations by completing damaged jigsaw puzzles. In: *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 793–802. IEEE (2018)
10. Chen, P., Liu, S., Jia, J.: Jigsaw clustering for unsupervised visual representation learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11526–11535 (2021)
11. Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., Bharath, A.A.: Generative adversarial networks: an overview. *IEEE Signal Process. Mag.* **35**(1), 53–65 (2018)
12. 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)
13. Larsson, G., Maire, M., Shakhnarovich, G.: Learning representations for automatic colorization. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *Computer Vision – ECCV 2016*. *ECCV 2016*. LNCS, vol. 9908, pp. 577–593. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46493-0\\_35](https://doi.org/10.1007/978-3-319-46493-0_35)
14. Larsson, G., Maire, M., Shakhnarovich, G.: Colorization as a proxy task for visual understanding. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6874–6883 (2017)
15. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *Computer Vision – ECCV 2016*. *ECCV 2016*. LNCS, vol. 9907, pp. 649–666. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46487-9\\_40](https://doi.org/10.1007/978-3-319-46487-9_40)
16. Zhang, R., Isola, P., Efros, A.A.: Split-brain autoencoders: unsupervised learning by cross-channel prediction. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1058–1067 (2017)
17. Tschannen, M., Djolonga, J., Rubenstein, P.K., Gelly, S., Lucic, M.: On mutual information maximization for representation learning. *arXiv preprint [arXiv:1907.13625](https://arxiv.org/abs/1907.13625)* (2019)
18. Bachman, P., Hjelm, R.D., Buchwalter, W.: Learning representations by maximizing mutual information across views. In: *Advances in Neural Information Processing Systems*, vol. 32 (2019)
19. Hjelm, R.D., et al.: Learning deep representations by mutual information estimation and maximization. *arXiv preprint [arXiv:1808.06670](https://arxiv.org/abs/1808.06670)* (2018)

20. 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)
21. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Presented at the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
22. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. arXiv preprint [arXiv:2003.04297](https://arxiv.org/abs/2003.04297) (2020)
23. Grill, J.-B., et al.: Bootstrap your own latent: A new approach to self-supervised learning. arXiv preprint [arXiv:2006.07733](https://arxiv.org/abs/2006.07733) (2020)
24. Chen, X., He, K.: Exploring simple siamese representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15750–15758 (2021)
25. Pinheiro, P.O.O., Almahairi, A., Benmalek, R., Golemo, F., Courville, A.C.: Unsupervised learning of dense visual representations. *Adv. Neural. Inf. Process. Syst.* **33**, 4489–4500 (2020)
26. Xie, Z., Lin, Y., Zhang, Z., Cao, Y., Lin, S., Hu, H.: Propagate yourself: exploring pixel-level consistency for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16684–16693 (2021)
27. Li, X., et al.: Dense semantic contrast for self-supervised visual representation learning. In: Proceedings of the 29th ACM International Conference on Multimedia, pp. 1368–1376 (2021)
28. 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)
29. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Studying aesthetics in photographic images using a computational approach. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) *Computer Vision – ECCV 2006*. *ECCV 2006*. LNCS, vol. 3953, pp. 288–301. Springer, Heidelberg (2006). [https://doi.org/10.1007/11744078\\_23](https://doi.org/10.1007/11744078_23)
30. Sheng, K., et al.: Revisiting image aesthetic assessment via self-supervised feature learning. *Proc. AAAI Conf. Artif. Intell.* **34**(04), 5709–5716 (2020)
31. Pfister, J., Kobs, K., Hotho, A.: Self-supervised multi-task pretraining improves image aesthetic assessment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 816–825 (2021)
32. Ching, J.H., See, J., Wong, L.-K.: Learning image aesthetics by learning inpainting. In: 2020 IEEE International Conference on Image Processing (ICIP), pp. 2246–2250. IEEE (2020)
33. Dosovitskiy, A., et al.: An image is worth  $16 \times 16$  words: Transformers for image recognition at scale. arXiv preprint [arXiv:2010.11929](https://arxiv.org/abs/2010.11929) (2020)
34. Chen, S., Xie, S., He, K.: An empirical study of training self-supervised visual transformers. arXiv e-prints [arXiv:2104.02057](https://arxiv.org/abs/2104.02057) (2021)
35. Caron, M., et al.: Emerging properties in self-supervised vision transformers. arXiv preprint [arXiv:2104.14294](https://arxiv.org/abs/2104.14294) (2021)