Summary of SCAN: Learning to Classify Images without Labels

Rakibuzzaman Mahmud: 1141444
Master Seminar: Deep Learning in Computer Vision

February 17, 2021

Overview

Convolutional neural networks have made breakthroughs in processing image, video, speech, and text and have recently been substantially improving upon the state of the art in image classification. But CNN often achieves their strong performance through supervised learning, which requires a huge amount of labeled dataset. But when there are no labels to train a network how do we get the network to learn meaningful features from the images?

Combining representation learning with clustering is one of the most promising approaches for unsupervised learning. The method described in the paper called SCAN(Semantic Clustering by Adopting Nearest neighbors) decouples the feature representation part and the clustering part resulting in a state of the art accuracy. The authors try to solve the problem with 3 steps: Self Supervised Learning: a self-supervised task from representation learning tries to extract semantically meaningful features. Clustering: clustering the similar representation using Nearest-neighbor-based Clustering. Self Labelling: using already well-classified images and fixing mistakes through self-labeling. SCAN method is the first to perform well on a large-scale dataset for image classification and shows promising results on ImageNet. Furthermore, it outperforms several semi-supervised learning methods with unlabelled data.

Keywords: Unsupervised Learning, Self-Supervised Learning, Image Classification, Clustering.

1 Introduction

A conventional image classification task involves labels that govern the features it learns through a Loss function. Humans are better at detecting objects where CNN's use supervised learning algorithms that require a huge amount of labeled images. However, acquiring labeled data or manually labeling data is often an expensive process and in the real-world scenario labeled data are hard to come by.

There are many approaches to solve this problem such as semi-supervised, self-supervised [1] semi-supervised learning is the branch of machine learning concerned with using labeled as well as unlabelled data to perform certain learning tasks. But they still require some amount of labeled data to get better performance. However, a major question remains. How can we classify images when there are no ground-truths available during the training or what if we do not know how many classes are there to classify?

To address these questions researchers have been trying to use unsupervised learning methods.[2] The goal of this approach is to cluster similar images together, while images in different clusters are dissimilar. Applying clustering algorithms to images is a difficult task. For instance, consider two images of dogs, one black and one white. The different pixel values cannot be directly clustered. So we need a feature selector. But Hand-engineered features are quite tedious to produce and do not perform that well in practice. Self-supervised learning such as Representation learning [3][4] can be used to generate feature representations from images. When we have a better representation of an image, we can cluster them using a loss function. But this approach will tend to focus on low-level features and we can not be sure if this will be semantically meaningful.

The method described in the paper separates the feature representation and the clustering resulting in inaccuracy almost as similar to the supervised method. Also, to further improve results the authors applied Fine-Tuning through self-labeling. The authors of the paper also observed, depending on the K value, sometimes semantically different images were clustered together. To address this issue, images that had a probability of belonging to a cluster were selected and labeled and these labeled images were used with their neighbors to update the weights and Cross entropy loss that optimizes those data points. Which as a result, makes the predictions more accurate.

2 Related work

This paper relates to many broad areas of research: The major topics that are covered in the paper are briefly discussed below.

2.1 Convolutional Neural Network

The Convolutional Neural Network (CNN) uses deep learning architecture inspired by the visual perception system of the living creatures. [5]

The basic components of a CNN consist of three types of layers, convolutional layer, pooling layer, and fully-connected layer. The convolutional layer learns feature representations of image data. The pooling layer aims to preserve the detected features in a smaller representation by discarding less significant data at the cost of spatial resolution. A CNN may consist of multiple convolutional and pooling layers and they are connected with one or more fully-connected layers that aim to perform high-level reasoning [11]. They take all neurons in the previous layer, connect them to every single neuron of the current layer to generate semantic information.[6]

CNN's have been applied in image classification for a long time.[7, 8] With their better capability of joint feature and classifier learning CNN's can achieve better classification accuracy on large scale datasets Compared to the other methods.[9] A point to be noted that, all the best results are coming from supervised training using labeled data. However, these labeled data are not easy to collect. Furthermore, these systems are difficult to scale up and can not handle many types of fine-grained classes. To solve this problem, some researchers propose to find localized regions in an unsupervised manner.[2]

2.2 Self-supervised representation learning

Self-supervised learning is a subset of unsupervised learning methods, in which CNN is explicitly trained with automatically generated labels.

Since The labels are automatically generated during self-supervised training large-scale datasets can be used for self-supervised training. Trained with these generated labels, self-supervised methods achieve promising results, and the gap between supervised methods and self-supervised learning methods are getting closer with new researches.[10]

Representation learning is a set of techniques that allows a system to automatically discover the representations needed for feature detection or classification from image data. A wide variety of representation learning methods have been proposed for the self-training of deep convolutional neural networks.

These methods use various pre-designed tasks called pretext tasks, which do not require annotated data to learn the representation and they are applied in-painting [11], patch context and jigsaw puzzles [3]; [12]; [13], clustering [14]; [15] colorization [16], generation [17]; [18], predicting transformations [4]; [19] and predicting rotation[4].

But still, they have not managed to match the performance of supervised-learned representations, but they have proved to be good alternatives for transferring on other vision tasks, such as object recognition, object detection, and semantic segmentation.

2.3 Data Clustering

Much research has been devoted to data clustering methods. We can primarily divide the existing methods into three categories: distance-based, density-based, and connectivity-based methods. [20]

Distance-based methods, such as the K-means [21], seek to find the relationship between data points based on various distance metrics. Density-based methods attempt to cluster data points via a proper density function, including the density-based spatial clustering of applications with noise.[22] Compared with the previous methods, connectivity-based methods cluster data points into a cluster if they are highly connected. [23]

2.4 Combination of Feature learning and Clustering

Recently, several methods have been proposed to combine feature learning with clustering into a single model. Such as deep embedded clustering (DEC)[2] which is a method that simultaneously learns feature representations and cluster assignments using deep neural networks. DEC learns a mapping from the data space to a lower-dimensional feature space in which it iteratively optimizes a clustering objective.

There are also other methods i.e: DAC [24] which leverages the architecture of CNNs for image clustering. Additionally, Another clustering method DeepCluster [25] that jointly learns the parameters of a neural network and the cluster assignments of the resulting features. DeepCluster iteratively groups the features with a standard clustering algorithm, k-means, and uses the subsequent assignments as supervision to update the weights of the network.

3 Method

The method of the paper can be described in a three step process:

First, Producing semantically meaningful feature representations using a pretext task, which is done in a Self supervised manner.

Second, Classify each image and its nearest neighbors together using a novel loss function.

Third, Using a self-labeling approach to reduce noise and make the clustering as close as possible to the class distribution of the k nearest neighbors in the neighbor selection.

3.1 Representation learning for semantic clustering and Pretext Task

The paper uses representation learning as the pretext task τ, that learns an embedding function Φ_θ - parameterized by a neural network with weights θ - that maps images into feature representations.

$$\min_{\theta} d(\Phi_\theta(X_i), \Phi_\theta(T[X_i])). \tag{1}$$

The main idea behind this is that if we transform the images, whether it be cropping or changing contrast, the high level features should remain the same.

After transforming the images and getting a new dataset of transformed images we use a convolution neural network to produce a feature representation of both the images and the transformed images and optimize the distance between the outputs for the image and the transformed image . When the original image and transformed image are in the same cluster with the objective of minimising the distance between $[X_i]$ and their augmentations $(T[X_i])$, the learned representations are much more meaningful.

3.2 A semantic clustering loss

After getting a meaningful representation, we can apply K-means or any clustering algorithm to it . But naively applying K-means to get K clusters can lead to 'cluster degeneracy'.

To solve this issue the paper introduces Semantic clustering loss, which aims to learn a clustering function Φ_η - parameterized by a neural network with weights η which is able to classify a sample X_i and its mined neighbors $\mathcal{N}_{X_i}$ together.

The function Φ_η terminates in a softmax function to perform a soft assignment over the clusters $\mathcal{C} = \{1, \ldots, C\}$, with $\Phi_\eta(X_i) \in [0,1]^C$. The probability of sample X_i being assigned to cluster c is denoted as $\Phi_\eta^c(X_i)$. the weights of Φ_η are learned by minimizing the following objective:

$$\begin{gathered}\Lambda = -\frac{1}{|\mathcal{D}|}\sum_{X\in\mathcal{D}}\sum_{k\in\mathcal{N}_X}\log\left\langle\Phi_\eta(X),\Phi_\eta(k)\right\rangle + \lambda\sum_{c\in\mathcal{C}}\Phi_\eta'^c\log\Phi_\eta'^c,\\ \text{with } \Phi_\eta'^c = \frac{1}{|\mathcal{D}|}\sum_{X\in\mathcal{D}}\Phi_\eta^c(X).\end{gathered}\tag{2}$$

The idea is to pass these images and its mined neighbors from the previous stage to a Neural Network to output probabilities for all the classes. The classes are chosen using some knowledge initially, the paper uses the knowledge of ground truth for the evaluation purposes.

The goal of the loss function is to make the clustering of an image as close as possible to the class distribution of the k nearest neighbors. This is done by the dot product of the image vector of probabilities and its neighbor's vector.

3.3 Fine-tuning through self-labeling

After clustering, some of the data points do not belong to the same semantic cluster. These false-positive examples make uncertain predictions. Therefore, the authors used a self-labeling approach to correct for mistakes due to noisy nearest neighbors using the already well-classified examples [26]. To be more specific, each sample was combined with K $\geq$ 1 neighbors, some of which unavoidably do not relate to the corresponding semantic cluster. These false-positive samples point to predictions for which the network is less confident. At the same time, the authors inspected that examples with highly confident predictions tend to be classified into a proper cluster, and consequently, a self-labeling strategy can be used to utilize the already well-classified examples, and correct the errors due to noisy nearest neighbors.

Algorithm 1 Semantic Clustering by Adopting Nearest neighbors (SCAN)

```
Input: Dataset $\mathcal{D}$, Clusters $\mathcal{C}$, Task $\tau$, Neural Nets $\Phi_\theta$ and $\Phi_\eta$, Neighbors $\mathcal{N}_\mathcal{D} = \{\}$.
Optimize $\Phi_\theta$ with task $\tau$ in Eq. 1.                    ▷ Pretext Task Step
for $X_i \in \mathcal{D}$ do
    $\mathcal{N}_\mathcal{D} \leftarrow \mathcal{N}_\mathcal{D} \cup \mathcal{N}_{X_i}$, with $\mathcal{N}_{X_i} = K$ neighboring samples of $\Phi_\theta(X_i)$.
end for
while SCAN-loss decreases do                                          ▷ Clustering Step
    Update $\Phi_\eta$ with SCAN-loss, i.e. $\Lambda(\Phi_\eta(\mathcal{D}), \mathcal{N}_\mathcal{D}, C)$ in Eq. 2
end while
while $Len(Y)$ increases do                                           ▷ Self-Labeling Step
    Y $\leftarrow$ ($\Phi_\eta(\mathcal{D})$ > threshold)
    Update $\Phi_\eta$ with cross-entropy loss, i.e. $H(\Phi_\eta(\mathcal{D}), Y)$
end while
Return: $\Phi_\eta(\mathcal{D})$                                      ▷ $\mathcal{D}$ is divided over $C$ clusters
```

4 Experiments

4.1 Datasets

The dataset that were considered for the experiments are CIFAR10 [27], CIFAR100- 20 [27] STL10 [28] and ImageNet [29].

Table 1: Datasets overview

Dataset	Classes	Train images	Val images	Aspect ratio
CIFAR10	10	50,000	10,000	32 x 32
CIFAR100-20	20	50,000	10,000	32 x 32
STL10	10	5,000	8,000	96 x 96
ImageNet-50	50	64,274	2,500	224 x 224
ImageNet-100	100	128,545	5,000	224 x 224
ImageNet-200	200	256,558	10,000	224 x 224
ImageNet	1000	1,281,167	50,000	224 x 224

4.2 Training setup

The training was done on standard ResNet-18 backbone with batch size 128 and in the fine tuning step with threshold of 0.99 and using Adam optimizer. For every sample, the 20 nearest neighbors are determined through an instance discrimination task based on noise contrastive estimation (NCE) [30] and SimCLR [31] implementation for the instance discrimination task on the dataset and MoCo [32] on ImageNet.

4.3 Validation criterion

The model based on the lowest loss was selected as the best model during the clustering step. For the self-labeling step, Based on the number of confident samples the model's weights were saved.

4.4 Augmentations

The authors were able to get better results using strong augmentations during the training. RandAugment [33], followed by Cutout [34] for the augmentation part and it was composed of four randomly selected transformations.

4.5 Ablation studies

An ablation study studies the performance of an AI system by removing certain components, to understand the contribution of the component to the overall system. The term is by analogy with biology (removal of components of an organism), and, continuing the analogy, is particularly used in the analysis of artificial neural nets, by analogy with ablative brain surgery.[35]

Table 1: Ablation Method CIFAR10

Setup	ACC (Avg ± Std)
Pretext + K-means	65.9 ± 5.7
SCAN-Loss (SimCLR)	78.7 ± 1.7
(1) Self-Labeling (SimCLR)	10.0 ± 0
(2) Self-Labeling (RA)	87.4 ± 1.6
SCAN-Loss (RA)	81.8 ± 1.7
(1) Self-Labeling (RA)	87.6 ± 0.4

Table 2: Ablation Pretext CIFAR10

Pretext Task	Clustering	ACC (Avg ± Std)
RotNet [16]	K-means	27.1 ± 2.1
	SCAN	74.3 ± 3.9
Inst. discr. [51]	K-means	52.0 ± 4.6
	SCAN	83.5 ± 4.1
Inst. discr. [7]	K-means	65.9 ± 5.7
	SCAN	87.6 ± 0.4

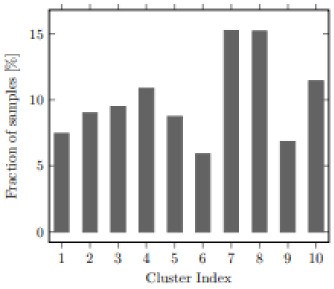

Fig. 3: K-means cluster assignments are imbalanced.

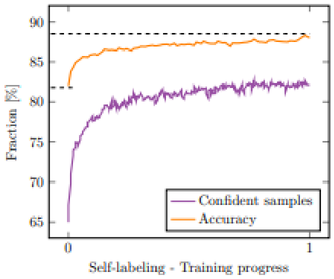

Fig. 4: Acc. and the number of confident samples during self-labeling.

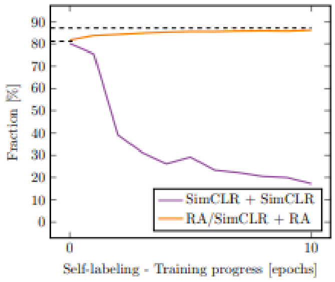

Fig. 5: Self-labeling with SimCLR or RandAugment augmentations.

The paper presented the performance gains through an ablation study on CIFAR10.

Some observation to be noted, K-means clustering with NCE pretext features results in an accuracy of 65.9%. Interestingly, applying K-means to the pretext features outperforms prior state-of-the-art methods for unsupervised classification based on end-to-end learning schemes. This observation supports the paper's primary claim of the benefit of separating clustering and feature learning.

Applying image augmentation to both the samples and their mined neighbors also improves the performance (78.7% vs. 81.8%). During self-labeling, the network slowly becomes more confident and therefore fine-tuning the network enhances the quality of the clustering(81.8% to 87.6%).

5 Comparison

The Table below compares the result to the state-of-the-art on three different benchmarks clustering accuracy (ACC), normalized mutual information (NMI), and adjusted rand index (ARI).

Dataset	**CIFAR10**			**CIFAR100-20**			**STL10**		
Metric	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
K-means [36]	22.9	8.7	4.9	13.0	8.4	2.8	19.2	12.5	6.1
Triplets [37]	20.5	–	–	9.94	–	–	24.4	–	–
JULE [38]	27.2	19.2	13.8	13.7	10.3	3.3	27.7	18.2	16.4
AEVB [39]	29.1	24.5	16.8	15.2	10.8	4.0	28.2	20.0	14.6
SAE [40]	29.7	24.7	15.6	15.7	10.9	4.4	32.0	25.2	16.1
DAE [41]	29.7	25.1	16.3	15.1	11.1	4.6	30.2	22.4	15.2
SWWAE [42]	28.4	23.3	16.4	14.7	10.3	3.9	27.0	19.6	13.6
AE [43]	31.4	23.4	16.9	16.5	10.0	4.7	30.3	25.0	16.1
GAN [44]	31.5	26.5	17.6	15.1	12.0	4.5	29.8	21.0	13.9
DEC [45]	30.1	25.7	16.1	18.5	13.6	5.0	35.9	27.6	18.6
ADC [46]	32.5	–	–	16.0	–	–	53.0	–	–
DeepCluster [47]	37.4	–	–	18.9	–	–	33.4	–	–
DAC [48]	52.2	40.0	30.1	23.8	18.5	8.8	47.0	36.6	25.6
IIC [49]	61.7	51.1	41.1	25.7	22.5	11.7	59.6	49.6	39.7
Supervised	93.8	86.2	87.0	80.0	68.0	63.2	80.6	65.9	63.1
Pretext [32] + K-means	65.9 ± 5.7	59.8 ± 2.0	50.9 ± 3.7	39.5 ± 1.9	40.2 ± 1.1	23.9 ± 1.1	65.8 ± 5.1	60.4 ± 2.5	50.6 ± 4.1
SCAN* (Avg ± Std)	81.8 ± 0.3	71.2 ± 0.4	66.5 ± 0.4	42.2 ± 3.0	44.1 ± 1.0	26.7 ± 1.3	75.5 ± 2.0	65.4 ± 1.2	59.0 ± 1.6
SCAN† (Avg ± Std)	87.6 ± 0.4	78.7 ± 0.5	75.8 ± 0.7	45.9 ± 2.7	46.8 ± 1.3	30.1 ± 2.1	76.7 ± 1.9	68.0 ± 1.2	61.6 ± 1.8
SCAN† (Best)	**88.3**	**79.7**	**77.2**	**50.7**	**48.6**	**33.3**	**80.9**	**69.8**	**64.6**
SCAN† (Overcluster)	86.2 ± 0.8	77.1 ± 0.1	73.8 ± 1.4	55.1 ± 1.6	50.0 ± 1.1	35.7 ± 1.7	76.8 ± 1.1	65.6 ± 0.8	58.6 ± 1.6

Table 2: State-of-the-art comparison

The proposed method consistently outperforms prior work by large margins on all three metrics, e.g. +26.6% on CIFAR10, +25.0% on CIFAR100-20 and +21.3% on STL10 in terms of accuracy.

5.1 ImageNet Experiments

To check the performance of the proposed SCAN method, the authors used the ImageNet dataset with smaller subsets of 50, 100, and 200 randomly selected classes that outperformed several semi-supervised learning methods without the use of any ground truth annotations.

Finally, the table also includes results that compare with the state-of-the-art in representation learning and when solving the problem in a fully-supervised manner.

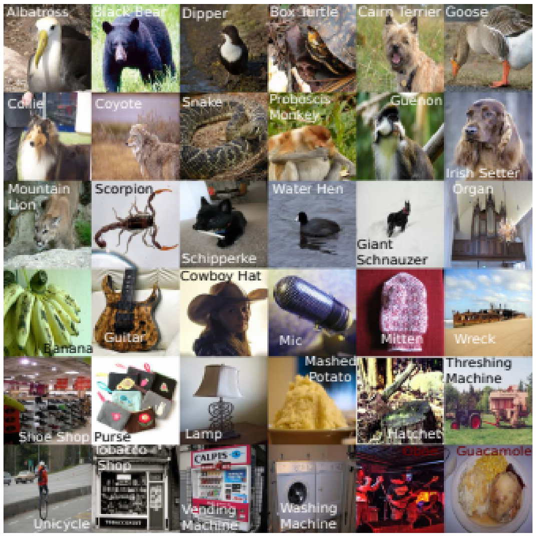

Fig. 9: Prototypes obtained by sampling the confident samples.

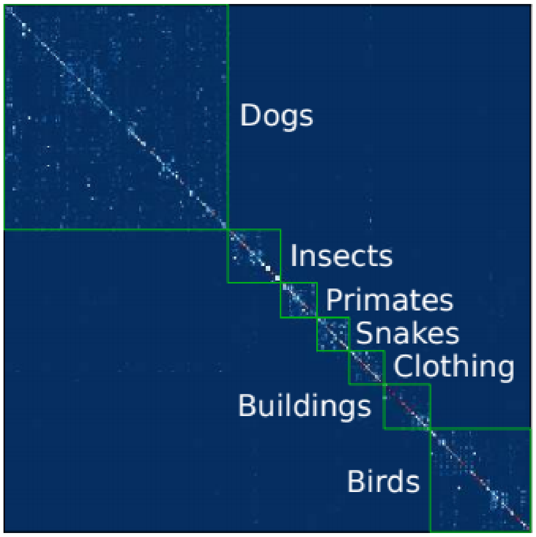

Fig. 10: Zoom on seven superclasses in the confusion matrix on ImageNet.

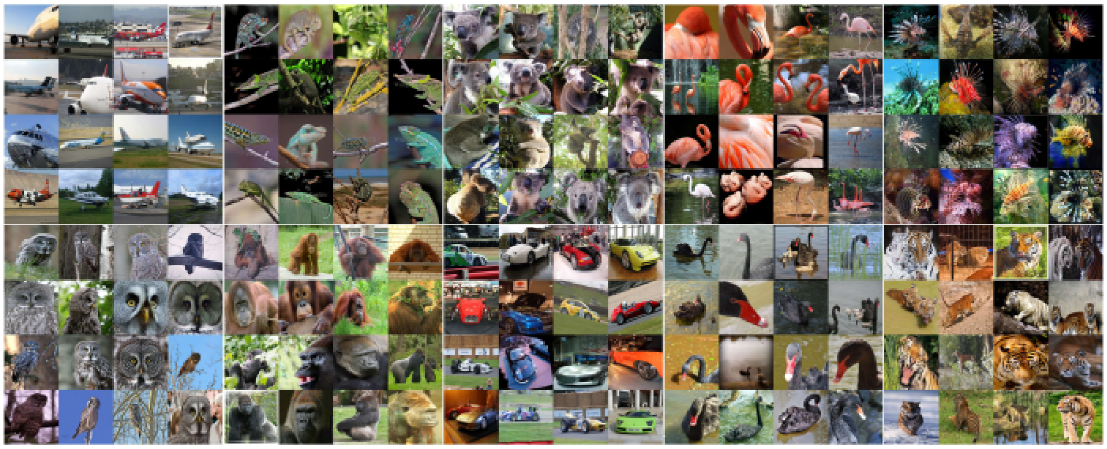

Fig. 11: Clusters extracted by our model on ImageNet (more in supplementary).

The Paper considers the problem of unsupervised image classification on the large-scale ImageNet dataset as shown in Table 3. First smaller subsets of 50, 100, and 200 randomly selected classes are considered and then the sets of 50 and 100 classes are subsets of the 100 and 200 classes respectively.

Table 3: Validation set results for 50, 100 and 200 randomly selected classes from ImageNet.

ImageNet	**50 Classes**				**100 Classes**				**200 Classes**			
Metric	Top-1	Top-5	NMI	ARI	Top-1	Top-5	NMI	ARI	Top-1	Top-5	NMI	ARI
K-means	65.9	-	77.5	57.9	59.7	-	76.1	50.8	52.5	-	75.5	43.2
SCAN*	75.1	91.9	80.5	63.5	66.2	88.1	78.7	54.4	56.3	80.3	75.7	44.1
SCAN†	76.8	91.4	82.2	66.1	68.9	86.1	80.8	57.6	58.1	80.6	77.2	47.0

5.2 Overclustering

The network was trained with the knowledge of the number of ground-truth classes. However, the table 2 also reports the results when the number of clusters does not match the number of ground-truth classes.

It can be observed that the approach does not require knowledge of the exact number of clusters and the increased performance on CIFAR100-20 is related to the higher intra-class variance.

6 Conclusion

The approach described in this paper has produced state of the art results on a variety of datasets. But it comes with its own set of disadvantages. There are many hyperparameters to consider, like the number of clusters, the value of K or the transformation for self-supervised training. Also the algorithm performs less than optimally for large class sizes and these can be observed in the case of ImageNet dataset. Still the paper paves the way for new and better unsupervised image classification models.

References

[1] Mahmut Kaya and H. Bilge. Deep metric learning: A survey. *Symmetry*, 11:1066, 08 2019.

[2] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis, 2016.

[3] Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction, 2016.

[4] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations, 2018.

[5] David H Hubel and Torsten N Wiesel. Receptive fields and functional architecture of monkey striate cortex. *The Journal of physiology*, 195(1):215–243, 1968.

[6] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[7] Zhen Zuo, Gang Wang, Bing Shuai, Lifan Zhao, and Qingxiong Yang. Exemplar based deep discriminative and shareable feature learning for scene image classification. *Pattern Recognition*, 48(10):3004–3015, 2015.

[8] Keiller Nogueira, Otávio AB Penatti, and Jefersson A Dos Santos. Towards better exploiting convolutional neural networks for remote sensing scene classification. *Pattern Recognition*, 61:539–556, 2017.

[9] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[10] Longlong Jing and Yingli Tian. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[11] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.

[12] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles, 2017.

[13] Mehdi Noroozi, Ananth Vinjimoor, Paolo Favaro, and Hamed Pirsiavash. Boosting self-supervised learning via knowledge transfer. In *CVPR*, 2018.

[14] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features, 2019.

[15] Xinrui Zhuang, Yuexiang Li, Yifan Hu, Kai Ma, Yujiu Yang, and Yefeng Zheng. Self-supervised feature learning for 3d medical images by playing a rubik's cube, 2019.

[16] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.

[17] Simon Jenni and Paolo Favaro. Self-supervised feature learning by learning to spot artifacts, 2018.

[18] Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. In *NIPS*, 2019.

[19] Liheng Zhang, Guo-Jun Qi, Liqiang Wang, and Jiebo Luo. Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data. In *CVPR*, 2019.

[20] T. Soni Madhulatha. An overview on clustering methods, 2012.

[21] Xin Jin and Jiawei Han. K-means clustering., 2010.

[22] Michal Daszykowski and Beata Walczak. *Density-Based Clustering Methods*, volume 2, pages 635–654. 12 2009.

[23] Jinwon Lee, Sang-Uk Cheon, and Jeongsam Yang. Connectivity-based convolutional neural network for classifying point clouds. *Pattern Recognition*, 112:107708, 2021.

[24] Jianlong Chang, Lingfeng Wang, Gaofeng Meng, Shiming Xiang, and Chunhong Pan. Deep adaptive image clustering. In *Proceedings of the IEEE international conference on computer vision*, pages 5879–5887, 2017.

[25] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149, 2018.

[26] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *arXiv preprint arXiv:2001.07685*, 2020.

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

[28] Yazhou Yao, Jian Zhang, Fumin Shen, Xiansheng Hua, Jingsong Xu, and Zhenmin Tang. Automatic image dataset construction with multiple textual metadata. In *2016 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2016.

[29] 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.

[30] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304, 2010.

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

[32] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[33] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020.

[34] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[35] Richard Meyes, Melanie Lu, Constantin Waubert de Puiseau, and Tobias Meisen. Ablation studies in artificial neural networks. *CoRR*, abs/1901.08644, 2019.

[36] Jianfeng Wang, Jingdong Wang, Jingkuan Song, Xin-Shun Xu, Heng Tao Shen, and Shipeng Li. Optimized cartesian k-means. *IEEE Transactions on Knowledge & Data Engineering*, 2015.

[37] Matthew Schultz and Thorsten Joachims. Learning a distance metric from relative comparisons. In *NIPS*, 2004.

[38] Jianwei Yang, Devi Parikh, and Dhruv Batra. Joint unsupervised learning of deep representations and image clusters. In *CVPR*, 2016.

[39] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[40] Andrew Ng. Sparse autoencoder. *CS294A Lecture notes*, 2011.

[41] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *JMLR*, 2010.

[42] Junbo Zhao, Michael Mathieu, Ross Goroshin, and Yann Lecun. Stacked what-where auto-encoders. *arXiv preprint arXiv:1506.02351*, 2015.

[43] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy layer-wise training of deep networks. In *NIPS*, 2007.

[44] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[45] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *ICML*, 2016.

[46] Philip Haeusser, Johannes Plapp, Vladimir Golkov, Elie Aljalbout, and Daniel Cremers. Associative deep clustering: Training a classification network with no labels. In *German Conference on Pattern Recognition*, 2018.

[47] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.

[48] Jianlong Chang, Lingfeng Wang, Gaofeng Meng, Shiming Xiang, and Chunhong Pan. Deep adaptive image clustering. In *ICCV*, 2017.

[49] Xu Ji, João F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *ICCV*, 2019.