PiCIE: Unsupervised Semantic Segmentation using Invariance and Equivariance in Clustering

Jang Hyun Cho[1] Utkarsh Mall[2] Kavita Bala[2] Bharath Hariharan[2]
[1]University of Texas at Austin [2]Cornell University

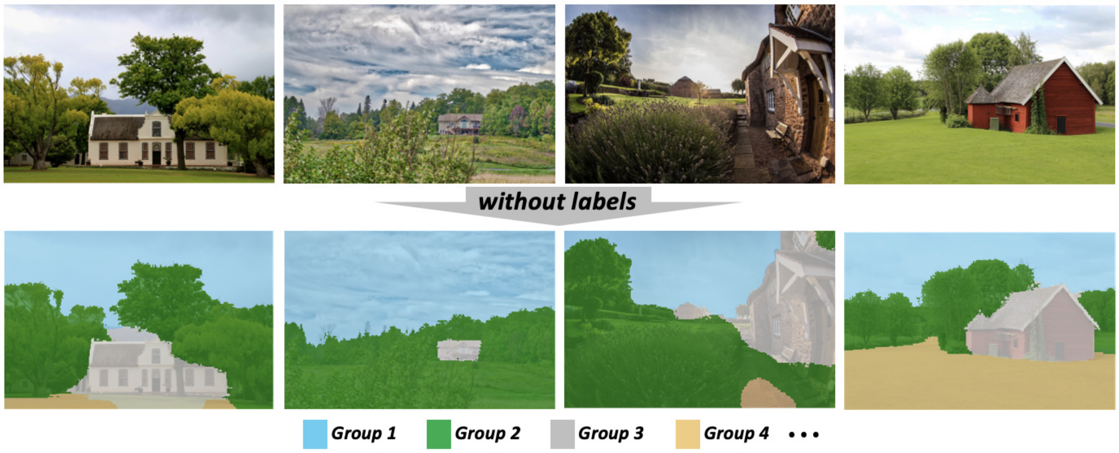

Figure 1: From these unannotated images, we would like a recognition system to discover the concepts of *house, grass, trees* and *sky*, and segment each image accordingly without any supervision.

Abstract

We present a new framework for semantic segmentation without annotations via clustering. Off-the-shelf clustering methods are limited to curated, single-label, and object-centric images yet real-world data are dominantly uncurated, multi-label, and scene-centric. We extend clustering from images to pixels and assign separate cluster membership to different instances within each image. However, solely relying on pixel-wise feature similarity fails to learn high-level semantic concepts and overfits to low-level visual cues. We propose a method to incorporate geometric consistency as an inductive bias to learn invariance and equivariance for photometric and geometric variations. With our novel learning objective, our framework can learn high-level semantic concepts. Our method, ***PiCIE*** *(****Pi****xel-level feature* ***C****lustering using* ***I****nvariance and* ***E****quivariance), is the first method capable of segmenting both* things *and* stuff *categories without any hyperparameter tuning or task-specific pre-processing. Our method largely outperforms existing baselines on COCO [31] and Cityscapes [8] with* ***+17.5*** *Acc. and* ***+4.5*** *mIoU. We show that PiCIE gives a better initialization for standard supervised training. The code is available at* https://github.com/janghyuncho/PiCIE.

1. Introduction

Unsupervised learning from a set of unlabelled images has gained large popularity, but still is mostly limited to single-class, object-centric images. Consider the images shown in Figure 1 (top). Given a collection of these and other unlabeled images, can a machine discover the concepts of "grass", "sky", "house" and "trees" from *each* image? Going further, can it identify *where* in each image each concept appears, and *segment* it out?

A system that is capable of such *unsupervised semantic segmentation* can then automatically discover classes of objects with their precise boundaries, thus removing the substantial cost of collecting and labeling datasets such as COCO. It might even discover objects, materials and textures that an annotator may not know of *a priori*. This can be particularly useful for analyzing novel domains: for example, discovering new kinds of visual structures in satellite imagery. The ability of the system to discover and segment out unknown objects may also prove useful for robots trying to manipulate these objects in the wild.

However, while unsupervised semantic segmentation might be useful, it is also challenging. This is because it combines the problem of class discovery with the challenge of exhaustive pixel labeling. Recent progress in self-

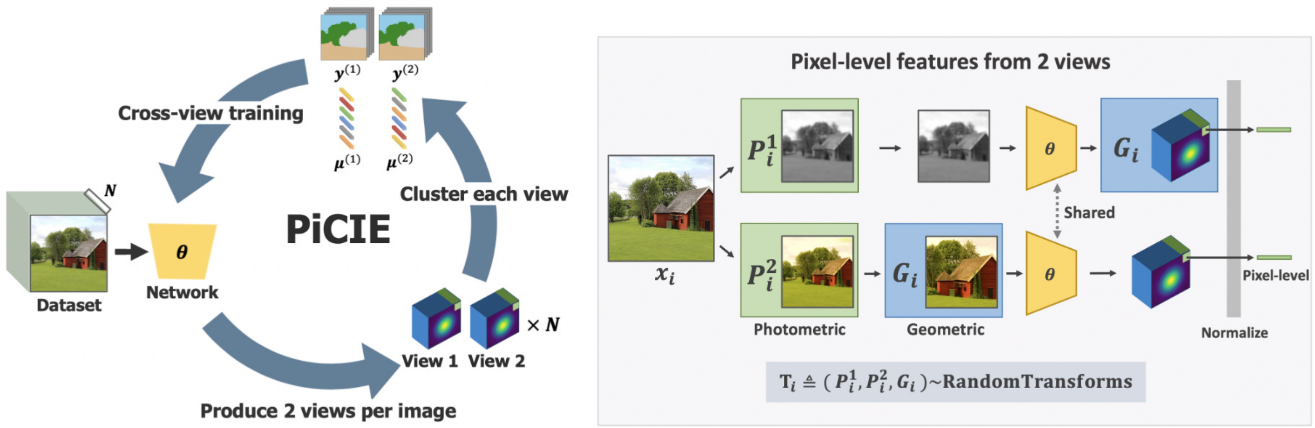

Figure 2: PiCIE overview (**left**) and illustration of multi-view feature computation (**right**). More details in Sec. 3.3.

supervised and unsupervised learning suggests that recognition systems can certainly discover *image-level* classes. However, image-level labeling is easier since the network can simply rely on just a few distinctive, stable features and discard the rest of the image. For example, a recognition system might be able to group all four images of Figure 1 together simply by detecting the presence of roof tiles in each image, and ignoring everything else in the images. In contrast, when *segmenting* the image, no pixel can be ignored; whether it is a distinct object (*thing*) or a background entity (*stuff*), *each and every pixel must be recognized and accurately characterized* in spite of potentially large intra-class variation. As such, very little prior work has tried to tackle this problem of discovering semantic segmentations, with results limited to extremely coarse *stuff* segmentation.

In this paper, we take a step towards a practically useful unsupervised semantic segmentation system: we present an approach that is able to segment out all pixels, be they *things* or *stuff*, at a much finer granularity than prior art. Our approach is based on a straightforward objective that codifies only two common-sense constraints. First, pixels that have a similar appearance (i.e., they cluster together in a learned feature space) should be labeled similarly and vice versa. Second, pixel labels should be *invariant* to color space transformations and *equivariant* to geometric transformations. Our results show that using these two objectives alone, we can train a ConvNet based semantic segmentation system *end-to-end* without any labels.

We find that in spite of its simplicity, our approach far outperforms prior work on this task, *more than doubling the accuracy* of prior art (Figure 1, bottom). Our clustering-based loss function (the first objective above) leads to a much simpler and easier learning problem compared to prior work, which instead tries to learn parametric pixel classifiers. But the invariance and equivariance objectives are key. They allow the convolutional network to connect together pixels across scale, pose and color variation, something that prior systems are unable to do. This increased robustness to invariance also allows our approach to effectively segment *objects*. We vindicate these intuitions through an ablation study, where we find that each of these contributes significant improvements in performance.

In sum, our results show that convolutional networks can learn to not only discover image-level concepts, but also semantically parse images without any supervision. This opens the door to true large-scale discovery, where such a trained network can automatically *surface* new classes of objects, materials or textures from only an unlabeled, uncurated dataset.

2. Related Work

Learning for clustering. Using deep neural networks to learn cluster-friendly embedding space has been widely studied [4, 5, 58, 57, 51, 14, 54]. DEC [51] and IDEC [14] train embedding function by training autoencoder (AE) [49] with reconstruction loss. DeepCluster and its variants [4, 5, 57] explicitly cluster the feature vectors of the entire dataset using k-means [38] in order to assign *pseudo-labels* to each data point, and then train an encoder network. All these methods share a common philosophy that iterative optimization of clustering loss improves the feature space to account for high-level visual similarity.

Apart from a representation learning perspective, there have been a number of recent works that tackle classification without labels by clustering data points [51, 18, 17, 23, 48, 55]. IIC [23], SeLa [55] and other works [48, 34, 60, 16] maximize mutual information between two versions of soft cluster assignments from a single image. Maximizing mutual information prevents the network from falling into a degenerate solution, but effectively enforces uniform distribution over clusters. Hence, unsupervised clustering is expected to work only with well-balanced datasets such as MNIST [28] and CIFAR [26]. Recent works [48, 55] tested on larger-scale datasets such as ImageNet [9], still assume a balanced set of single-class, object-centric images. Since these methods do not explicitly perform clustering on data, they are called *implicit clustering* methods, contrary to *explicit clustering* [51, 14, 4, 5, 57, 52, 56, 42, 29].

Segmentation without labels. In clustering, each data point is assumed to be semantically homogeneous. This condition is invalid when images contain more than one semantic class, such as scene-centric datasets [11, 31, 8, 15]. In fact, the majority of *common* images are not object-centric, and therefore one cannot simply use off-the-shelf clustering methods to obtain semantic understanding of an arbitrary dataset. The problem reduces to *semantic segmentation* by clustering pixel-level features.

There has been a number of recent attempts to semantic segmentation without labels. IIC [23] simply extends mutual information-based clustering to pixel-level representation by outputting a probability map over image pixels. AC [37] uses an autoregressive model [47] to obtain probabilities of pixels over categories, which then maximizes mutual information across two different "orderings" of autoregression. Both works are limited to *stuff* categories due to the following two reasons. First, a mixture of *stuff* and *things* categories introduces severe data imbalance since there are far more *stuff* pixels than *things* pixels in real-world images. Such imbalance leads the mutual information maximization to forcibly balance the size of clusters and hence leads to noisy representation as major classes (*stuff* categories) subsume minor classes (*things* categories). Second, each method exploits the *local spatial consistency* condition; a pixel needs to be semantically (and visually) consistent with its neighboring pixels. This condition is only valid with *stuff* categories (e.g., *sky*) and not often true with *things* categories. Other methods [6, 2] based on GANs [12, 25] learn to generate foreground masks of a given image, but are limited to a single-category setting. Our method is free from such assumptions and the results show that our method is capable of segmenting both *stuff* and *things* categories together well with uncurated images.

Equivariance learning. Equivariance learning has been studied in object and keypoints tracking [36, 35, 1, 27], facial landmark detection [46, 50, 22], and keypoint detection [45, 59, 44, 43, 32]. The central idea in these works is to train a model that predicts consistent key points between two images, with the underlying assumption that two images share a common instance. This enables unsupervised learning of semantically consistent and geometrically structured representation learning. The general objective is to directly minimize the L2 distance between two feature vectors that correspond to the semantically equivalent locations on images. However, using MSE loss with clustering is often sensitive to the choice of hyper-parameters, which is often infeasible or prone to overfit in unsupervised setting. Furthermore, individual feature vector may contain noisy low-level visual cues which can overwhelm the gradient flow during back-propagation. Our method instead learns equivariance by enforcing consistent clustering assignments between two views and hence only cluster-centered visual cues affect the loss (detail in Sec. 3.3).

3. PiCIE

We are given an *uncurated, unlabeled* dataset of images taken from some domain $\mathcal{D}$. On this dataset, we want to discover a set of visual classes $\mathcal{C}$ and learn a semantic segmentation function f_θ. When provided an unseen image from $\mathcal{D}$, f_θ should be able to assign every pixel a label from the set of classes $\mathcal{C}$.

We formulate this task of unsupervised image segmentation as pixel-level clustering, where every pixel is assigned to a cluster. Clustering typically requires a good feature space, but no such feature representation exists *a priori*. We therefore propose an approach that learns the feature representation jointly with the clustering. The overall pipeline of **PiCIE**, which stands for **Pi**xel-level feature **C**lustering using **I**nvariance and **E**quivariance, is depicted in Figure 2. We describe our approach below.

3.1. A baseline clustering approach

We begin with prior work that learns a neural network end-to-end for clustering unlabeled images into image-level classes [4, 5, 51, 14, 53]. The key issue tackled in these papers is that clustering images into classes requires strong feature representations, but for training strong feature representations one needs class labels. To solve this chicken-and-egg problem, the simplest solution is the one identified by DeepCluster [4]: alternate between clustering using the current feature representation, and use the cluster labels as pseudo-labels to train the feature representation. One can follow a similar strategy for the unsupervised semantic segmentation task. The only difference is that we need to use an embedding function f_θ that produces a feature map, producing a feature vector for every pixel. The classifier must also operate on individual pixels. One can then alternate between clustering the pixel feature vectors to get pixel pseudo-labels, and using these pseudo-labels to train the pixel feature representation.

Concretely, suppose we have a set of unlabeled images $x_i, i = 1, \dots, n$. Suppose our embedding, denoted by f_θ produces a feature tensor $f_\theta(x)$. This yields a feature representation for every pixel p in the image x. Denote by $f_\theta(x)[p]$ this pixel-level feature representation. Denote by $g_{\mathbf{w}}(\cdot)$ a classifier operating on these pixel feature vectors. Then our baseline approach alternates between two steps:

1. Use the current embedding and k-means to cluster the pixels in the dataset.

$$\min_{\mathbf{y},\boldsymbol{\mu}} \sum_{i,p} \|f_\theta(x_i)[p] - \mu_{y_{ip}}\|^2 \tag{1}$$

where y_{ip} denotes the cluster labels of the p-th pixel in the i-th image, and μ_k is the k-th cluster centroid. (We use mini-batch k-means [39]).

2. Use the cluster labels to train a pixel classifier using standard cross entropy loss.

$$\min_{\theta,\mathbf{w}} \sum_{i,p} \mathcal{L}_{CE}(g_{\mathbf{w}}(f_\theta(x_i)[p]), y_{ip}) \tag{2}$$

$$\mathcal{L}_{CE}(g_{\mathbf{w}}(f_\theta(x_i)[p]), y_{ip}) = -\log \frac{e^{s_{y_{ip}}}}{\sum_k e^{s_k}} \tag{3}$$

where s_k is the k-th class score output by the classifier $g_{\mathbf{w}}(f_\theta(x_i, p))$.

Given this baseline, we now propose the following modifications.

3.2. Non-parametric prototype-based classifiers

The DeepCluster inspired framework above uses a separate, learned classifier. However, in the unsupervised setting with constantly changing pseudo-labels, training a classifier jointly with the feature representation can be challenging. An insufficiently trained classifier can feed noisy gradients into the feature extractor, resulting in noisy clusters for the next training round.

We therefore propose to jettison the parametric pixel classifier $g_{\mathbf{w}}$ entirely. Instead, we label pixels based on their distance to the centroids ("prototypes" [41]) estimated by k-means. This results in the following changed objective.

$$\min_{\theta} \sum_{i,p} \mathcal{L}_{clust}(f_\theta(x_i)[p], y_{ip}, \boldsymbol{\mu}) \tag{4}$$

$$\mathcal{L}_{clust}(f_\theta(x_i)[p], y_{ip}, \boldsymbol{\mu}) = -\log \left(\frac{e^{-d(f_\theta(x_i)[p], \mu_{y_{ip}})}}{\sum_l e^{-d(f_\theta(x_i)[p], \mu_l)}} \right) \tag{5}$$

where $d(\cdot, \cdot)$ is cosine distance.

3.3. Invariance and Equivariance

Jointly learning the feature representation along with the clustering as above will certainly produce clusters that are compact in feature space, but there is no reason why these clusters must be semantic. To get a semantic grouping of pixels, we need to introduce an additional inductive bias. What must this inductive bias be if we have no labels?

The inductive bias we introduce is invariance to photometric transformations and equivariance to geometric transformations: the labeling should not change if the pixel colors are jittered slightly, and when the image is warped geometrically, the labeling should be warped similarly. Concretely, if Y is the output semantic labeling for an image x, and if P and G are photometric and geometric transformations respectively, then the output semantic labeling of a transformed image $G(P(x))$ should be $G(Y)$.

Algorithm 1 PiCIE pseudocode

```
for $x_i \sim \mathcal{D}$ do
    $P_i^{(1)}, P_i^{(2)} \sim$ RandomPhotometricTransforms
    $G_i \sim$ RandomGeometricTransforms
    $z_{i,:}^{(1)} \leftarrow G_i(f_\theta(P_i^{(1)}(x_i)))[:]$
    $z_{i,:}^{(2)} \leftarrow f_\theta(G_i(P_i^{(2)}(x_i)))[:]$
end for
$\mu^{(1)}, \mathbf{y}^{(1)} \leftarrow$ KMeans$(\{z_{ip}^{(1)} : i \in [N], p \in [HW]\})$
$\mu^{(2)}, \mathbf{y}^{(2)} \leftarrow$ KMeans$(\{z_{ip}^{(1)} : i \in [N], p \in [HW]\})$
for $x_i \sim \mathcal{D}$ do
    $z_{i,:}^{(1)} \leftarrow G_i(f_\theta(P_i^{(1)}(x_i)))[:]$
    $z_{i,:}^{(2)} \leftarrow f_\theta(G_i(P_i^{(2)}(x_i)))[:]$
    $\mathcal{L}_{\text{within}} \leftarrow \sum_p \mathcal{L}_{\text{clust}}(z_{ip}^{(1)}, y_{ip}^{(1)}, \mu^{(1)}) + \mathcal{L}_{\text{clust}}(z_{ip}^{(2)}, y_{ip}^{(2)}, \mu^{(2)})$
    $\mathcal{L}_{\text{cross}} \leftarrow \sum_p \mathcal{L}_{\text{clust}}(z_{i}^{(1)}, y_{ip}^{(2)}, \mu^{(2)}) + \mathcal{L}_{\text{clust}}(z_{ip}^{(2)}, y_{ip}^{(1)}, \mu^{(1)})$
    $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{within}} + \mathcal{L}_{\text{cross}}$
    $f_\theta \leftarrow$ backward$(\mathcal{L}_{\text{total}})$
end for
```

Above: PiCIE pseudo-code. Notations consistent with Sec. 3.3.

Implementing this constraint in a joint clustering and learning framework is tricky, since there isn't a ground truth label for each image. The pseudo-ground truth labeling is itself derived from clustering, which is itself produced from the feature maps, and as such itself sensitive to input transformations. Invariance/equivariance in this case therefore means two things: one, we should produce the same *clusters* irrespective of the transformations, and two, the predicted *pixel labels* should exhibit the desired in/equivariance.

3.3.1 Invariance to photometric transformations

We first address the question of invariance. For each image x_i in the dataset, we randomly sample two photometric transformations, $P_i^{(1)}$ and $P_i^{(2)}$. This yields two feature vectors for each pixel p in each image x_i:

$$z_{ip}^{(1)} = f_\theta(P_i^{(1)}(x_i))[p] \tag{6}$$

$$z_{ip}^{(2)} = f_\theta(P_i^{(2)}(x_i))[p] \tag{7}$$

We then perform clustering separately in the two "views" to get two sets of pseudo-labels and centroids:

$$\mathbf{y}^{(1)}, \boldsymbol{\mu}^{(1)} = \arg\min_{\mathbf{y},\boldsymbol{\mu}} \sum_{i,p} \|z_{ip}^{(1)} - \mu_{y_{ip}}\|^2 \tag{8}$$

$$\mathbf{y}^{(2)}, \boldsymbol{\mu}^{(2)} = \arg\min_{\mathbf{y},\boldsymbol{\mu}} \sum_{i,p} \|z_{ip}^{(2)} - \mu_{y_{ip}}\|^2 \tag{9}$$

Given these two sets of centroids and these two sets of pseudo-labels, we use two sets of loss functions:

1. As before, we want the feature vectors to adhere to the clustering labels. Now that we have two views, we

want this to be true in each view:

$$\mathcal{L}_{within} = \sum_{i,p} \mathcal{L}_{clust}(z_{ip}^{(1)}, y_{ip}^{(1)}, \boldsymbol{\mu}^{(1)}) + \mathcal{L}_{clust}(z_{ip}^{(2)}, y_{ip}^{(2)}, \boldsymbol{\mu}^{(2)}) \quad (10)$$

2. Because we posit that the clustering should be invariant to photometric transformations, we also want feature vectors from one view to match the cluster labels and centroids of the other:

$$\mathcal{L}_{cross} = \sum_{i,p} \mathcal{L}_{clust}(z_{ip}^{(1)}, y_{ip}^{(2)}, \boldsymbol{\mu}^{(2)}) + \mathcal{L}_{clust}(z_{ip}^{(2)}, y_{ip}^{(1)}, \boldsymbol{\mu}^{(1)}) \quad (11)$$

This multi-view framework and the cross-view loss achieve two things. First, by forcing feature vectors from one transformation to adhere to labels produced by another, it encourages the network to learn feature representations that will be *labeled* identically irrespective of any photometric transformations. Second, by forcing the same feature representation to be consistent with two different clustering solutions, it encourages the two solutions themselves to match, thus ensuring that the set of concepts discovered by clustering is invariant to photometric transformations.

3.3.2 Equivariance to geometric transformations

A house and a zoomed-in version of the house should be labeled similarly, but may produce vastly different features. More precisely, the segmentation of the zoomed-in house should be a zoomed-in version of the original segmentation. This is the notion of *equivariance* to geometric transformations (such as random crops), which we add in next.

To learn equivariance with respect to geometric transformations, we sample a geometric transformation (concretely, random crop and horizontal flip) G_i for each image. Then, in the above framework, one view uses feature vectors of the transformed image, while the other uses the transformed feature vectors of the original:

$$z_{ip}^{(1)} = f_\theta(G_i(P_i^{(1)}(x_i)))[p] \quad (12)$$

$$z_{ip}^{(2)} = G_i(f_\theta(P_i^{(2)}(x_i)))[p] \quad (13)$$

The other steps are exactly the same. The two views are clustered separately, and the final training objective is the combination of the within-view and cross-view objectives:

$$\mathcal{L}_{total} = \mathcal{L}_{within} + \mathcal{L}_{cross} \quad (14)$$

4. Experiments

4.1. Training details

For all our experiments, we use the Feature Pyramid Network [30] with ResNet-18 [20] backbone pre-trained on ImageNet [9]. The fusion dimension of the feature pyramid is 128 instead of 256. We apply L2 normalization on the feature map of our network. The cluster centroids are computed with mini-batch approximation with GPUs using the FAISS library [39, 24]. For the baselines, we do not use image gradients as an additional input when we use ImageNet-pretrained weight. Except in Table 4, all images are resized and center-cropped to 320 × 320 during training. We used the published codes [4, 23] with minimal modification for the baselines. Other details are in supplementary.

Pre-trained vs random initialization. Prior works [23, 37] train the network from random initialization, but for semantic segmentation it is unnecessary; unlike representation learning literature [19, 7, 13, 61, 4, 5, 57], our goal is to segment a given dataset as accurately as possible, and in a practical scenario one will always choose to initialize from a pre-trained network such as on the ImageNet dataset [9]. Therefore, we train all models with ImageNet-pretrained weights, except that in Table 4 we show PiCIE outperforms all the baselines when trained from scratch as well.

Loss Balancing and Overclustering. As shown in [4, 5, 23], jointly optimizing for a separate set of clusters with higher number improves the stability of clustering as well as the accuracy of the prediction. However, in unsupervised settings hyper-parameter tuning is often infeasible. Thus, we use the generic approach to balance the loss:

$$\mathcal{L} = \lambda_{K_1}\mathcal{L}_{K_1} + \lambda_{K_2}\mathcal{L}_{K_2} \quad (15)$$

$\lambda_{K_1} = \frac{\log K_2}{\log K_1 + \log K_2}$ and $\lambda_{K_2} = \frac{\log K_1}{\log K_1 + \log K_2}$ where K_1 and K_2 are the number of clusters. The intuition is that the magnitude of the cross-entropy loss depends logarithmically on the number of clusters, hence we prevent the overclustering to overwhelm the gradient flow. We fix $K_2 = 100$ and add "+H." in results when applied. Similarly, due to the imbalance of datasets, the computed clusters will have largely different sizes; we apply a balance term for each cluster during the cross-entropy computation.

4.2. Baselines

We describe the baseline methods that we compare PiCIE to: IIC [23] and modified DeepCluster [4] for segmentation purposes. They are state-of-the-art *implicit* and *explicit* clustering-based learning methods.

IIC. IIC [23] is an implicit clustering method where the network directly predicts the (soft) clustering assignment of each pixel-level feature vector. The main objective is maximizing the mutual information between the predictions of a pixel and neighboring pixel(s). For controlled experiments, we use FPN with ResNet-18 same as PiCIE as well as the first two residual blocks of ResNet-18 (IIC – *res12*) similar to the original shallow VGG-like [40] model (details in

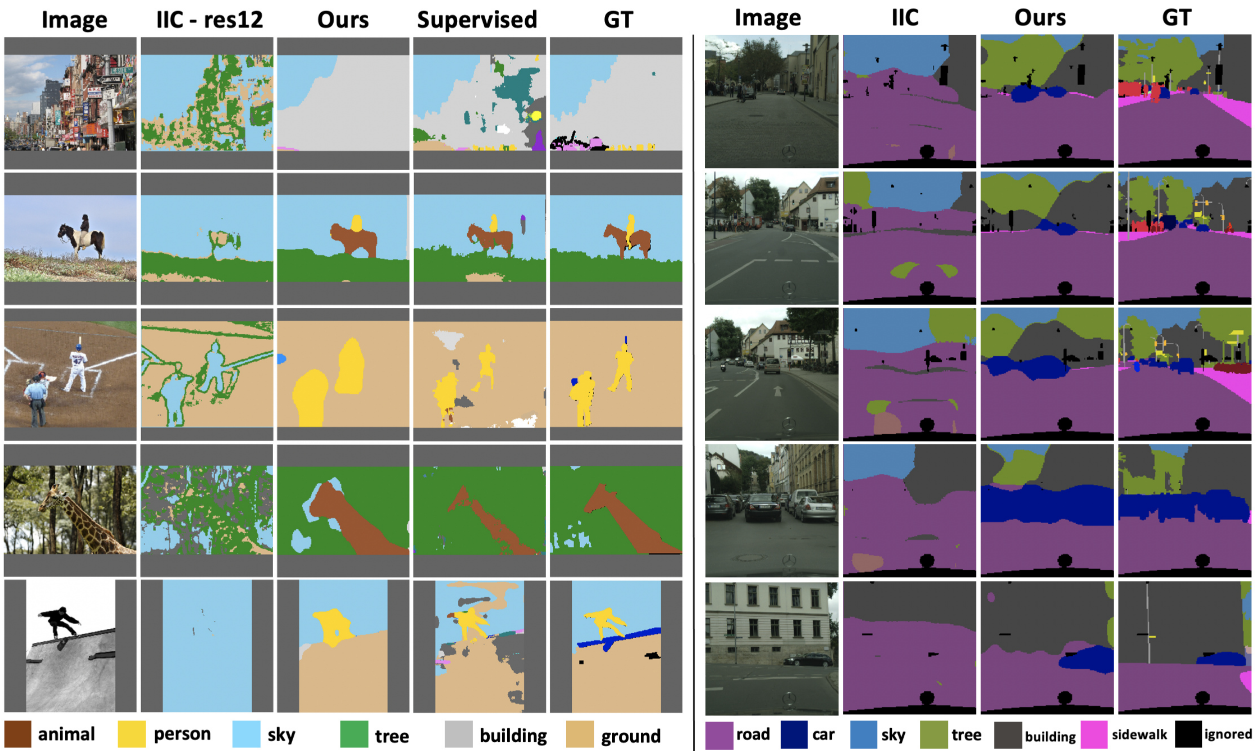

Figure 3: Overall qualitative results on COCO-*All* [31] (**left**) and Cityscapes [8](**right**). Note that we show IIC-res12 for COCO and IIC for Cityscapes to show the best result of the method on each dataset. Each ground truth label is assigned a color and for each cluster, the majority label's color is used. We show some of the color and name matches for better understanding. More in supplementary materials.

supplementary). Following the original paper [23], we used auxiliary over-clustering loss with $K = 45$.

Modified DeepCluster. DeepCluster is an explicit clustering method where the network clusters the feature vectors of given images and uses the cluster assignment as labels to train the network. To adjust to our problem setup, we modify the original DeepCluster to instead cluster pixel-level feature vectors before the final pooling layer. This allows the network to assign a label to each pixel. However, since the size of image explodes the number of feature vectors to cluster, we apply mini-batch k-means [39] to first compute the cluster centroids, assign labels, and train the network.

4.3. Datasets

COCO. Following [23], we evaluate our model on the COCO-Stuff dataset [3]. The COCO-Stuff dataset is a large-scale scene-centric dataset of images with 80 *things* categories and 91 *stuff* categories. We follow the same pre-process as [23] where classes are merged to form 27 (15 *stuff* and 12 *things*) categories. Unless otherwise stated, **we evaluate both *things* and *stuff* categories**, unlike prior works which evaluate only *stuff*.

Cityscapes. We further evaluate our model on the Cityscapes dataset [8]. Cityscapes is a set of images of street scenes from 50 different cities. There are 30 classes of instances that can be further categorized into 8 groups. After filtering out *void* group, we have 27 categories. We train our method as well as IIC and modified DeepCluster with $K = 27$ where K is the number of clusters.

Method	Classifier	Acc.	mIoU
No Train	Linear	17.45	3.70
No Train	Prototype	26.26	8.41
Modified DC	Linear	32.21	9.79
IIC - res12 [23]	Linear	22.45	4.11
IIC [23]	Linear	21.79	6.71
PiCIE	Prototype	48.09	13.84
PiCIE + H.	Prototype	**49.99**	**14.36**

Table 1: COCO-*All* [23] results. Our method is compared to clustering methods adapted to semantic segmentation. "+H." denotes PiCIE trained with auxiliary clustering.

4.4. Results

In Table 1, we compare PiCIE with the following baselines: *No Train*, *modified DeepCluster* [4], and *IIC* [23]. Unlike the prior works [23, 37] where only *stuff* categories are considered, we evaluate the models on both *stuff* and *things* categories to test on more realistic setting. Since the majority of scene-centric image dataset consists of *stuff* categories, our evaluation now faces a severe imbalance problem. Also, the learning mechanism of IIC assumes local spatial consistency, which is not often true for *things* cat-

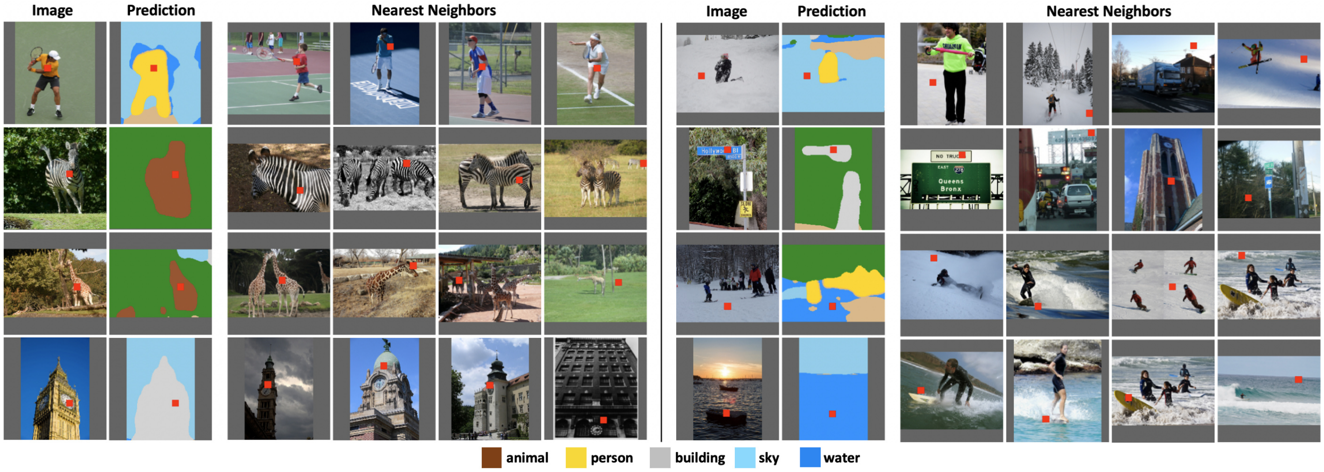

Figure 4: Nearest neighbor results for correctly predicted (**left**) instances and incorrectly predicted (**right**) instances. The red box indicates the position of the particular feature vector (size exaggerated). More details in supplementary materials.

Method	Partition	# Classes	Acc.	mIoU
Modified DC [4]	*Stuff*	15	44.28	**22.24**
IIC [23]			33.91	12.00
PiCIE + H.			**74.56**	17.32
Modified DC [4]	*Things*	12	67.06	11.55
IIC [23]			43.93	13.64
PiCIE + H.			**69.39**	**23.83**
Modified DC [4]	*All*	27	32.21	9.79
IIC [23]			21.79	6.71
PiCIE + H.			**49.99**	**14.36**

Table 2: Results on different partitions of the COCO dataset.

Method	# Classes	Accuracy	mIoU
IIC	27	47.88	6.35
IIC – res12		29.78	4.96
Modified DC		40.67	7.06
PiCIE		**65.50**	**12.31**

Table 3: Cityscapes results.

Method	COCO-*Stuff*
Random CNN	19.4
K-means [38]	14.1
SIFT [33]	20.2
Doersch 2015 [10]	23.1
Isola 2016 [21]	24.3
DeepCluster [4]	19.9
IIC [23]	27.7
AC [37]	30.8
Modified DC	25.26
IIC	27.97
IIC – res12	27.92
PiCIE	**31.48**

Table 4: COCO-*Stuff* results without ImageNet pretrained weight following [23, 37]. First section is from prior works [23, 37] and the last two sections are from our implementation.

egories due to more dynamic shape variations. We found that IIC tends to overfit to low-level visual cues since (implicit) clustering is done within a batch and insufficient supervisory signal is present when an instance has dynamic and complex visual cues. Indeed, in Figure 3 no *things* categories are correctly segmented from IIC results. On the other hand, PiCIE's novel in/equivariance loss enforces geometric consistency as an inductive bias to learn high-level visual concepts, and as shown in Figure 3 PiCIE ("Ours") is capable of segmenting both *stuff* and *things* categories with high accuracy. As a result, Table 1 shows that PiCIE largely outperforms other baselines (+ **17.5** Acc. and **4.5** mIoU). In Table 3, we test the baselines and our method on Cityscapes and show similar level of advantages (+ **18** Acc. and **5.3** mIoU). Finally, Table 4 shows PiCIE outperforms the other models on the benchmark from [23, 37] where the image size is 128×128, models are trained from scratch, and only *stuff* labels are considered for evaluation.

Things vs stuff. In Table 2, we show that PiCIE improves mainly on *things* categories (+**10** mIoU) while maintaining better or compatible performance on *stuff* categories compared to other methods. This indicates that enforcing geometric transformation equivariance was highly effective on *things* categories where the instances objects with distinct shape and boundaries. Furthermore, we show in Table 2 and 4 that PiCIE still outperforms on *stuff* categories with or without ImageNet-pretrained weights.

4.5. Ablation Study

In Table 5, we decompose our method to examine which component affects the performance the most. We gain 5 points by using a non-parametric classifier with cluster centroids. We further gain 3 points with cross-view learning with invariance transformations. Equivariance learning adds another 5.5 points, and with auxiliary over-clustering, we arrive at 49.99 pixel accuracy and 14.36 mIoU.

In Table 6, we test alternatives of different components of PiCIE. First, one could wonder if our cross-view loss can be replaced by MSE loss, directly minimizing the feature

Nonparametric	Photometric	Geometric	Overcluster	Accuracy	mIoU
				34.35	9.88
✓				39.25	9.82
✓	✓			42.55	9.84
✓		✓		46.97	12.04
✓	✓	✓		48.09	13.84
✓	✓	✓	✓	**49.99**	**14.36**

Table 5: Ablation study 1. Our method is decomposed to examine which components affect the performance the most.

Single	MSE eqv.	No inv.	No balance	Accuracy	mIoU
				48.09	**13.84**
			✓	40.56	11.46
✓				44.31	11.71
	✓			44.15	10.98
✓		✓		41.70	9.92

Table 6: Ablation study 2. One or more components in our method is replaced with alternative options.

vectors of the two views. This leads PiCIE to a suboptimal solution: 1) the direct distance between two feature vectors can be overwhelmed by low-level or irrelevant signals whereas cross-view loss directs the gradient to the nearest centroid, hence only considers relevant signals and 2) MSE loss requires hyperparameter tuning to be jointly used with cross-entropy loss, which is infeasible in purely unsupervised setting. Also, one could doubt if two sets of clustering are necessary; a single clustering with geometric transformation on the predicted labels can be used as an alternative to compute the cross-view loss. However, the two versions of an image contain different information (e.g., *zoomed-in* vs *full house*) that can be mutually beneficial. We test them all (and more) in Table 6 and the results justify our choices.

4.6. Analysis

Nearest neighbor analysis. In Figure 4, we show the nearest neighbors of correctly (left) and incorrectly (right) predicted instances. The nearest neighbors of correctly predicted segments share close high-level semantics (e.g., *person playing tennis, zebra, giraffe*, and *a building with a clock*). This indicates that intra-class semantics are well preserved. The incorrectly predicted segments also have semantically and visually close nearest neighbors. For example, the first row shows that *snow* pixels are confused with *sky* as the two concepts are visually alike. Such visual ambiguity is an inherent limitation of unsupervised methods.

Representation quality. In Table 7, we compare the learned representations by training a linear classifier for each trained method from our main experiments on COCO-*All*. We train with $\eta = 0.001$ for 10 epochs with cross-entropy loss. This allows us to analyze whether the difficulty is from the representation or from clustering. Compared to the unsupervised results from Table 1, baselines have a huge performance gap whereas PiCIE has a minimal gap. This indicates that clustering is where the major difficulty is and PiCIE gives close-to-optimal clustering given learned representation. In Table 8, we show that PiCIE can give better network initialization for supervised training.

Feature Extractor	Normalization	Acc.	mIoU
Modified DC		50.79	13.76
Modified DC	✓	48.61	13.30
IIC		51.49	13.26
IIC	✓	44.50	8.37
No Eqv.		47.73	12.59
No Eqv.	✓	48.58	10.40
Single Cluster		50.34	12.70
Single Cluster	✓	49.24	11.47
MSE		52.01	13.16
MSE	✓	50.61	11.83
PiCIE		54.08	14.11
PiCIE	✓	54.16	13.89
PiCIE + H.		54.65	14.32
PiCIE + H.	✓	**54.75**	**14.77**

Table 7: Transfer learning results. A new linear classifier has been trained on top of the learned embedding network.

Initialization	Normalization	Acc.	mIoU	C-Acc.	C-mIoU
ImageNet		75.48	44.69	55.82	17.36
ImageNet	✓	74.74	43.44	57.24	31.51
Modified DC		75.25	44.37	55.16	18.43
Modified DC	✓	75.27	43.82	57.41	30.27
IIC		75.16	44.26	56.07	20.32
IIC	✓	74.81	44.11	57.30	29.47
PiCIE		75.61	44.40	54.84	17.39
PiCIE	✓	**76.02**	44.97	**59.77**	**32.81**
PiCIE + H.		75.90	**45.60**	58.95	18.38
PiCIE + H.	✓	76.01	45.04	58.94	32.15

Table 8: Re-training results. Trained networks are used as an initialization for standard supervised training. "C-Acc." and "C-mIoU" are clustering results after supervised training. All models are trained from ImageNet-pretrained initialization.

5. Conclusion

In this paper, we introduced a new framework for unsupervised semantic segmentation with clustering. Our main contribution is to incorporate geometric consistency as an inductive bias to learn invariance and equivariance for photometric and geometric variations. Our novel cross-view loss is simple yet highly effective in learning high-level visual concepts necessary to segment *things* categories. Our method is the first unsupervised semantic segmentation that works for both *stuff* and *things* categories without rigorous hyper-parameter tuning or task-specific pre-processing.

Acknowledgements

This work was supported by DARPA Learning with Less Labels program (HR001118S0044), CHS-1617861, CHS-1513967, CHS-1900783, and CHS-1930755.

References

[1] Luca Bertinetto, Jack Valmadre, João F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. *arXiv preprint arXiv:1606.09549*, 2016. 4323

[2] Adam Bielski and Paolo Favaro. Emergence of object segmentation in perturbed generative models. In *Advances in Neural Information Processing Systems*, pages 7256–7266, 2019. 4323

[3] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1209–1218, 2018. 4326

[4] 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. 4322, 4323, 4325, 4326, 4327

[5] Mathilde Caron, Piotr Bojanowski, Julien Mairal, and Armand Joulin. Unsupervised pre-training of image features on non-curated data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2959–2968, 2019. 4322, 4323, 4325

[6] Mickaël Chen, Thierry Artières, and Ludovic Denoyer. Unsupervised object segmentation by redrawing. In *Advances in Neural Information Processing Systems*, pages 12726–12737, 2019. 4323

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020. 4325

[8] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 4321, 4323, 4326

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 4322, 4325

[10] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 1422–1430, 2015. 4327

[11] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 4323

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 4323

[13] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020. 4325

[14] Xifeng Guo, Long Gao, Xinwang Liu, and Jianping Yin. Improved deep embedded clustering with local structure preservation. In *IJCAI*, pages 1753–1759, 2017. 4322, 4323

[15] Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 4323

[16] Divam Gupta, Ramachandran Ramjee, Nipun Kwatra, and Muthian Sivathanu. Unsupervised clustering using pseudo-semi-supervised learning. In *International Conference on Learning Representations*, 2020. 4322

[17] Kai Han, Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Andrea Vedaldi, and Andrew Zisserman. Automatically discovering and learning new visual categories with ranking statistics. In *International Conference on Learning Representations*, 2020. 4322

[18] Kai Han, Andrea Vedaldi, and Andrew Zisserman. Learning to discover novel visual categories via deep transfer clustering. In *International Conference on Computer Vision (ICCV)*, 2019. 4322

[19] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019. 4325

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4325

[21] Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H Adelson. Learning visual groups from co-occurrences in space and time. *arXiv preprint arXiv:1511.06811*, 2015. 4327

[22] Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object landmarks through conditional image generation. In *Advances in neural information processing systems*, pages 4016–4027, 2018. 4323

[23] Xu Ji, João F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9865–9874, 2019. 4322, 4323, 4325, 4326, 4327

[24] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *arXiv preprint arXiv:1702.08734*, 2017. 4325

[25] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018. 4323

[26] Alex Krizhevsky et al. Learning multiple layers of features from tiny images. 2009. 4322

[27] Tejas D Kulkarni, Ankush Gupta, Catalin Ionescu, Sebastian Borgeaud, Malcolm Reynolds, Andrew Zisserman, and Volodymyr Mnih. Unsupervised learning of object keypoints for perception and control. In *Advances in neural information processing systems*, pages 10724–10734, 2019. 4323

[28] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 4322

[29] Yang Li, Shichao Kan, and Zhihai He. Unsupervised deep metric learning with transformed attention consistency and contrastive clustering loss. *CoRR*, abs/2008.04378, 2020. 4323

[30] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 4325

[31] 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 *European conference on computer vision*, pages 740–755. Springer, 2014. 4321, 4323, 4326

[32] Dominik Lorenz, Leonard Bereska, Timo Milbich, and Bjorn Ommer. Unsupervised part-based disentangling of object shape and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10955–10964, 2019. 4323

[33] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 4327

[34] Willi Menapace, Stéphane Lathuilière, and Elisa Ricci. Learning to Cluster under Domain Shift. In *European Conference on Computer Vision*, Edinburgh, United Kingdom, Aug. 2020. ECCV 2020. 4322

[35] Matthias Minderer, Chen Sun, Ruben Villegas, Forrester Cole, Kevin P Murphy, and Honglak Lee. Unsupervised learning of object structure and dynamics from videos. In *Advances in Neural Information Processing Systems*, pages 92–102, 2019. 4323

[36] David Novotny, Samuel Albanie, Diane Larlus, and Andrea Vedaldi. Self-supervised learning of geometrically stable features through probabilistic introspection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3637–3645, 2018. 4323

[37] Yassine Ouali, Celine Hudelot, and Myriam Tami. Autoregressive unsupervised image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, August 2020. 4323, 4325, 4326, 4327

[38] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011. 4322, 4327

[39] David Sculley. Web-scale k-means clustering. In *Proceedings of the 19th international conference on World wide web*, pages 1177–1178, 2010. 4324, 4325, 4326

[40] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015. 4325

[41] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, pages 4077–4087, 2017. 4324

[42] Kun Su, Xiulong Liu, and Eli Shlizerman. Predict cluster: Unsupervised skeleton based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 4323

[43] Supasorn Suwajanakorn, Noah Snavely, Jonathan J Tompson, and Mohammad Norouzi. Discovery of latent 3d keypoints via end-to-end geometric reasoning. In *Advances in neural information processing systems*, pages 2059–2070, 2018. 4323

[44] James Thewlis, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object frames by dense equivariant image labelling. In *Advances in neural information processing systems*, pages 844–855, 2017. 4323

[45] James Thewlis, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object landmarks by factorized spatial embeddings. In *Proceedings of the IEEE international conference on computer vision*, pages 5916–5925, 2017. 4323

[46] James Thewlis, Hakan Bilen, and Andrea Vedaldi. Modelling and unsupervised learning of symmetric deformable object categories. In *Advances in Neural Information Processing Systems*, pages 8178–8189, 2018. 4323

[47] Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. In *Advances in neural information processing systems*, pages 4790–4798, 2016. 4323

[48] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *European Conference on Computer Vision (ECCV)*, 2020. 4322

[49] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol, and Léon Bottou. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(12), 2010. 4322

[50] O. Wiles, A.S. Koepke, and A. Zisserman. Self-supervised learning of a facial attribute embedding from video. In *British Machine Vision Conference*, 2018. 4323

[51] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pages 478–487, 2016. 4322, 4323

[52] Xueting Yan, Ishan Misra, Abhinav Gupta, Deepti Ghadiyaram, and Dhruv Mahajan. Clusterfit: Improving generalization of visual representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6509–6518, 2020. 4323

[53] Jianwei Yang, Devi Parikh, and Dhruv Batra. Joint unsupervised learning of deep representations and image clusters. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5147–5156, 2016. 4323

[54] Linxiao Yang, Ngai-Man Cheung, Jiaying Li, and Jun Fang. Deep clustering by gaussian mixture variational autoencoders with graph embedding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 4322

[55] Asano YM., Rupprecht C., and Vedaldi A. Self-labelling via simultaneous clustering and representation learning. In *International Conference on Learning Representations*, 2020. 4322

[56] Yunpeng Zhai, Shijian Lu, Qixiang Ye, Xuebo Shan, Jie Chen, Rongrong Ji, and Yonghong Tian. Ad-cluster: Augmented discriminative clustering for domain adaptive person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 4323

[57] Xiaohang Zhan, Jiahao Xie, Ziwei Liu, Yew-Soon Ong, and Chen Change Loy. Online deep clustering for unsupervised representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6688–6697, 2020. 4322, 4323, 4325

[58] Junjian Zhang, Chun-Guang Li, Chong You, Xianbiao Qi, Honggang Zhang, Jun Guo, and Zhouchen Lin. Self-supervised convolutional subspace clustering network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5473–5482, 2019. 4322

[59] Yuting Zhang, Yijie Guo, Yixin Jin, Yijun Luo, Zhiyuan He, and Honglak Lee. Unsupervised discovery of object landmarks as structural representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2694–2703, 2018. 4323

[60] Junjie Zhao, Donghuan Lu, Kai Ma, Yu Zhang, and Yefeng Zheng. Deep image clustering with category-style representation. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XIV*, volume 12359 of *Lecture Notes in Computer Science*, pages 54–70. Springer, 2020. 4322

[61] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 4325