南京航空航天大學
Nanjing University of Aeronautics and Astronautics

模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

Invariant Information Clustering for Unsupervised Image Classification and Segmentation

Xu Ji
University of Oxford
xuji@robots.ox.ac.uk

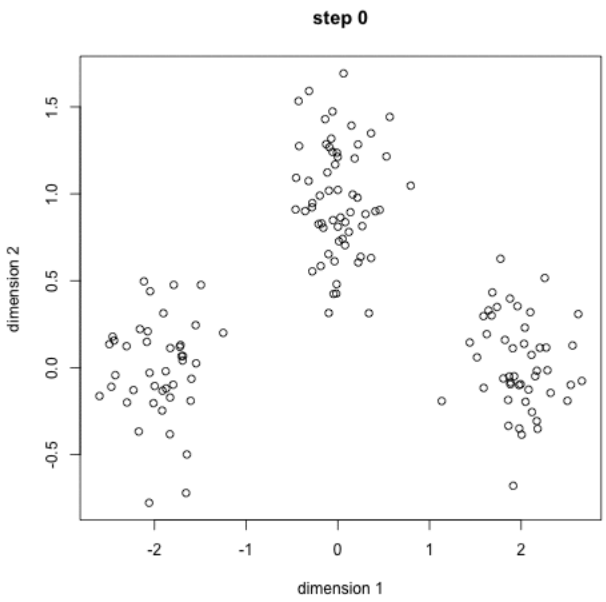
João F. Henriques
University of Oxford
joao@robots.ox.ac.uk

Andrea Vedaldi
University of Oxford
vedaldi@robots.ox.ac.uk

ICCV 2019

Clustering

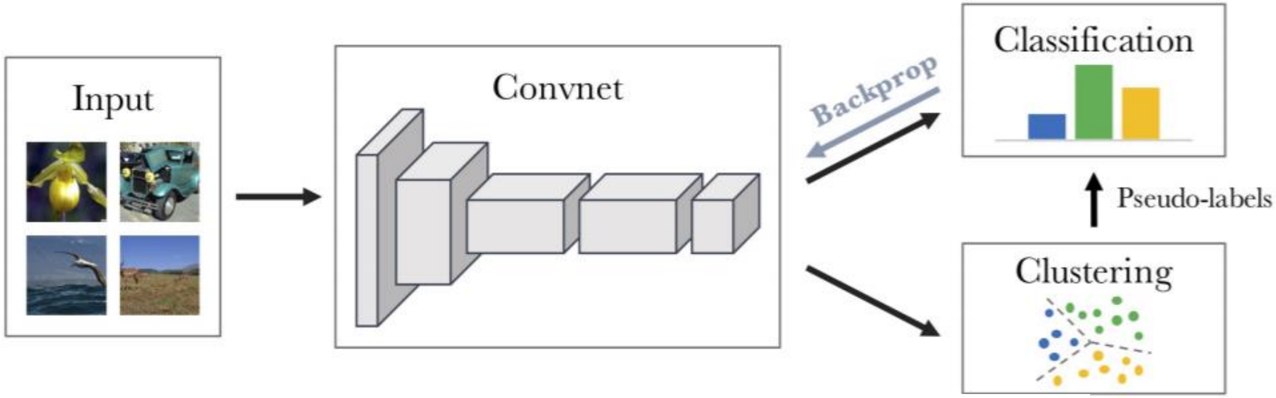
- **K-mean**
  1. Randomly initialize k center points
  2. Each data point is classified by computing the distance between that point and each group center
  3. Recompute the group center
  4. Repeat 2-3



- Potential Limitations
  - How to decide k center points (k-mean++)
  - The curse of dimensionality

Deep Clustering

- **Representation + K-means**



  - Representation learning: $\min_{\theta, W} \frac{1}{N} \sum_{n=1}^{N} \ell\left(g_W\left(f_\theta(x_n)\right), y_n\right)$
  - Clustering: $\min_{C \in \mathbb{R}^{d \times k}} \frac{1}{N} \sum_{n=1}^{N} \min_{y_n \in \{0,1\}^k} \|f_\theta(x_n) - C y_n\|_2^2 \quad \text{such that} \quad y_n^\top 1_k = 1.$
- Issue of **trivial solutions**
  - Empty clusters: An optimal decision boundary is to **assign all of the inputs to a single cluster**.
  - Trivial parametrization: If the **vast majority of images is assigned to a few clusters**, the parameters θ will exclusively discriminate between them.

Revisiting Deep Clustering

- **How to learn the representation Φ ?**


$\boldsymbol{x}$ $\quad\quad$ $\boldsymbol{x}'$

For a paired data $(\boldsymbol{x}, \boldsymbol{x}')$, we would like to learn a representation $\boldsymbol{\phi}$ that

- emphasizes their **common information** (the class)
- neglects the **instance-specific information** (pose, …)

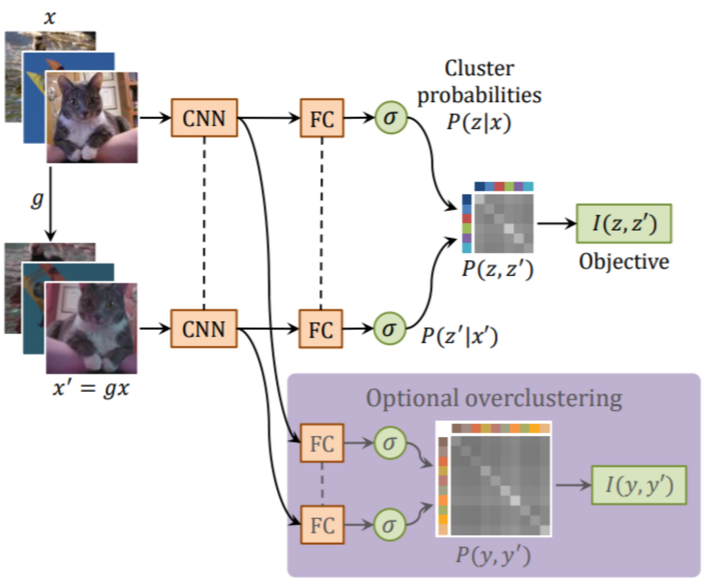
- **Solution**: $\max_{\Phi} I(\Phi(\mathbf{x}), \Phi(\mathbf{x}'))$ (mutual information)

$$I(z, z') = H(z) - \boxed{H(z|z')} \qquad z = \Phi(x)$$

Instance-specific information

Invariant Information Clustering

- Compute the mutual information $I(\Phi(\mathbf{x}), \Phi(\mathbf{x}'))$
  - The neural network Φ is terminated by a softmax layer.



$$P(z = c|\mathbf{x}) = \Phi_c(\mathbf{x})$$

$$\mathbf{P} = \frac{1}{n}\sum_{i=1}^{n} \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}'_i)^\top$$ (a batch)

$$\mathbf{P}_{cc'} = P(z = c, z' = c')$$

$$I(z, z') = I(\mathbf{P}) = \sum_{c=1}^{C}\sum_{c'=1}^{C} \mathbf{P}_{cc'} \cdot \ln \frac{\mathbf{P}_{cc'}}{\mathbf{P}_c \cdot \mathbf{P}_{c'}}.$$

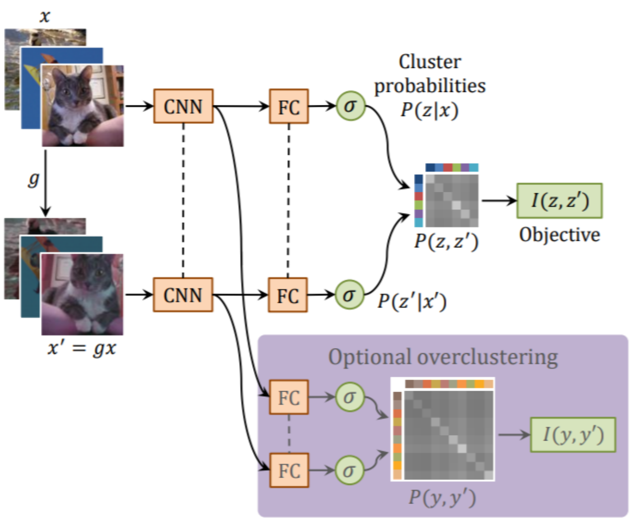
$$\mathrm{I}(X;Y) = \sum_{y \in \mathcal{Y}}\sum_{x \in \mathcal{X}} p(x,y) \log\left(\frac{p(x,y)}{p(x)\,p(y)}\right),$$

- Why degenerate solutions are avoided.
  - If all samples are assigned to a single cluster, *H(z)* is not maximized.

Implementation

- Use generated image pairs, consisting of image x and its randomly perturbed version.

$$\max_{\Phi} I(\Phi(\mathbf{x}), \Phi(g\mathbf{x}))$$



- Auxiliary overclustering
  - Add an auxiliary overclustering head to deal with distractor classes
  - STL-10
    - ➢ 500 training + 800 testing images per class
    - ➢ 100 *k* unlabeled images (with distractor classes)

Experiments

- Unsupervised learning analysis

	STL10	CIFAR10	CFR100-20	MNIST
Random network	13.5	13.1	5.93	26.1
K-means [53]†	19.2	22.9	13.0	57.2
Spectral clustering [49]	15.9	24.7	13.6	69.6
Triplets [46]‡	24.4	20.5	9.94	52.5
AE [5]‡	30.3	31.4	16.5	81.2
Sparse AE [40]‡	32.0	29.7	15.7	82.7
Denoising AE [48]‡	30.2	29.7	15.1	83.2
Variational Bayes AE [34]‡	28.2	29.1	15.2	83.2
SWWAE 2015 [54]‡	27.0	28.4	14.7	82.5
GAN 2015 [45]‡	29.8	31.5	15.1	82.8
JULE 2016 [52]	27.7	27.2	13.7	96.4
DEC 2016 [51]†	35.9	30.1	18.5	84.3
DAC 2017 [8]	47.0	52.2	23.8	97.8
DeepCluster 2018 [7]† ‡	33.4⋆	37.4⋆	18.9⋆	65.6 ⋆
ADC 2018 [24]	53.0	32.5	16.0⋆	99.2
IIC (lowest loss sub-head)	**59.6**	**61.7**	**25.7**	**99.2**
IIC (avg sub-head ± STD)	59.8 ± 0.844	57.6 ± 5.01	25.5 ± 0.462	98.4 ± 0.652

Table 1: **Unsupervised image clustering.** Legend: †Method based on k-means. ‡Method that does not directly learn a clustering function and requires further application of k-means to be used for image clustering. ⋆Results obtained using our experiments with authors' original code.

	STL10
No auxiliary overclustering	43.8
Single sub-head ($h = 1$)	57.6
No sample repeats ($r = 1$)	47.0
Unlabelled data segment ignored	49.9
Full setting	**59.6**

Experiments

- Unsupervised learning analysis

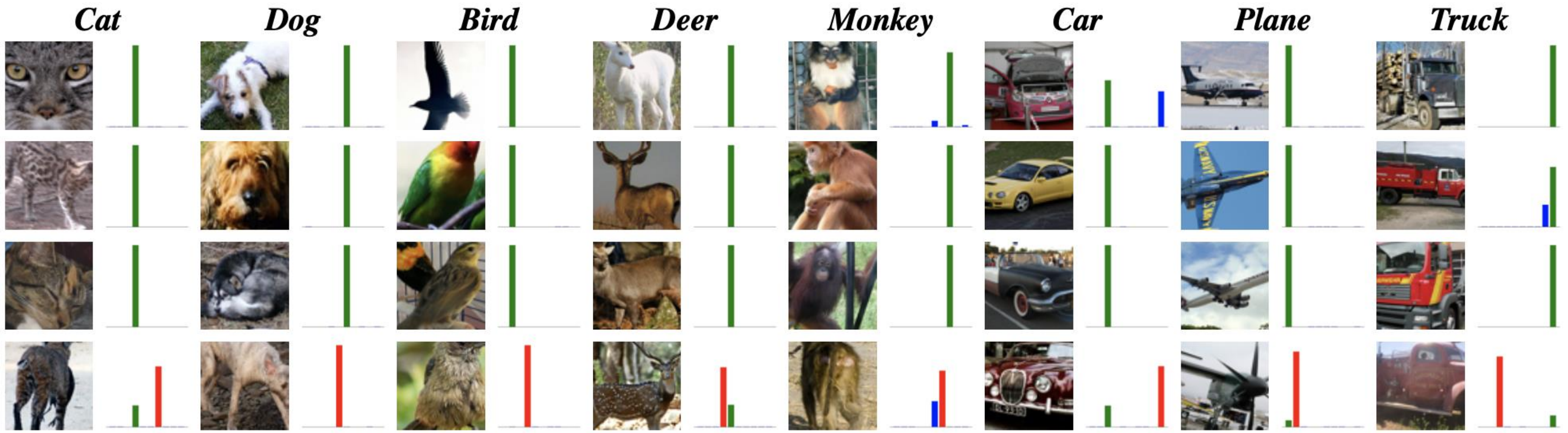


Figure 5: **Unsupervised image clustering (IIC) results on STL10.** Predicted cluster probabilities from the best performing head are shown as bars. Prediction corresponds to tallest, ground truth is green, incorrectly predicted classes are red, and all others are blue. The bottom row shows failure cases.

Experiments

- Semi-supervised learning analysis

	STL10
Dosovitskiy 2015 [18]†	74.2
SWWAE 2015 [54]†	74.3
Dundar 2015 [19]	74.1
Cutout* 2017 [15]	87.3
Oyallon* 2017 [42]†	76.0
Oyallon* 2017 [42]	87.6
DeepCluster 2018 [7]	73.4⋆
ADC 2018 [24]	56.7⋆
DeepINFOMAX 2018 [27]	77.0
IIC plus finetune†	**79.2**
IIC plus finetune	**88.8**

Table 3: **Fully and semi-supervised classification.** Legend: *Fully supervised method. ⋆Our experiments with authors' code. †Multi-fold evaluation.

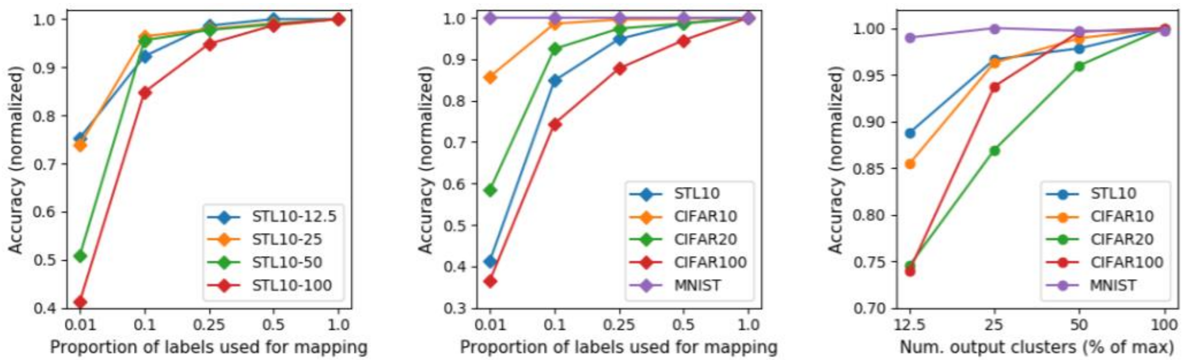


Figure 6: **Semi-supervised overclustering.** Training with IIC loss to overcluster ($k > k_{gt}$) and using labels for evaluation mapping only. Performance is robust even with 90%-75% of labels discarded (left and center). STL10-r denotes networks with output $k = \lceil 1.4r \rceil$. Overall accuracy improves with the number of output clusters k (right). For further details see supplementary material.

$$k = \lceil 1.4r \rceil$$

Experiments

- Semi-supervised learning analysis

	STL10		CIFAR10		CIFAR100-20		CIFAR100		MNIST	
% of max k	k	ACC	k	ACC	k	ACC	k	ACC	k	ACC
100	140	**63.1**	140	**65.0**	280	**34.7**	1000	**20.3**	100	98.6
50	70	61.4	70	62.2	140	33.1	500	20.3	50	98.6
25	35	59.7	35	60.5	70	30.0	250	19.1	25	**98.7**
12.5	18	54.8	18	53.7	35	25.7	125	15.0	13	97.9

Table 3: Absolute accuracy for semi-supervised overclustering experiments in paper fig. 6-right.

STL10	1.0		0.5		0.25		0.1		0.01	
% of max k	n_a	ACC	n_a	ACC	n_a	ACC	n_a	ACC	n_a	ACC
100	5000	**63.1**	2500	**61.0**	1250	58.6	500	52.4	50	25.5
50	5000	61.4	2500	59.8	1250	**59.1**	500	**57.8**	50	30.7
25	5000	59.7	2500	59.2	1250	58.5	500	57.6	50	**44.1**
12.5	5000	54.8	2500	54.8	1250	54.1	500	50.6	50	41.3

	1.0		0.5		0.25		0.1		0.01	
	n_a	ACC	n_a	ACC	n_a	ACC	n_a	ACC	n_a	ACC
STL10	5000	**63.1**	2500	61.0	1250	58.6	500	52.4	50	25.5
CIFAR10	50000	**62.9**	25000	62.7	12500	62.6	5000	62.0	500	53.9
CIFAR100-20	50000	**34.5**	25000	34.0	12500	33.6	5000	31.9	500	20.1
CIFAR100	50000	**20.3**	25000	19.2	12500	17.9	5000	15.1	500	7.43
MNIST-25	60000	**98.9**	30000	98.9	15000	98.9	6000	98.9	600	98.9

Table 4: Absolute accuracy for semi-supervised overclustering experiments in paper fig. 6-left (top) and fig. 6-center (bottom). n_a denotes number of labels used to find mapping from output k to k_{gt} for evaluation.

Experiments

- Unsupervised and semi-supervised segmentation

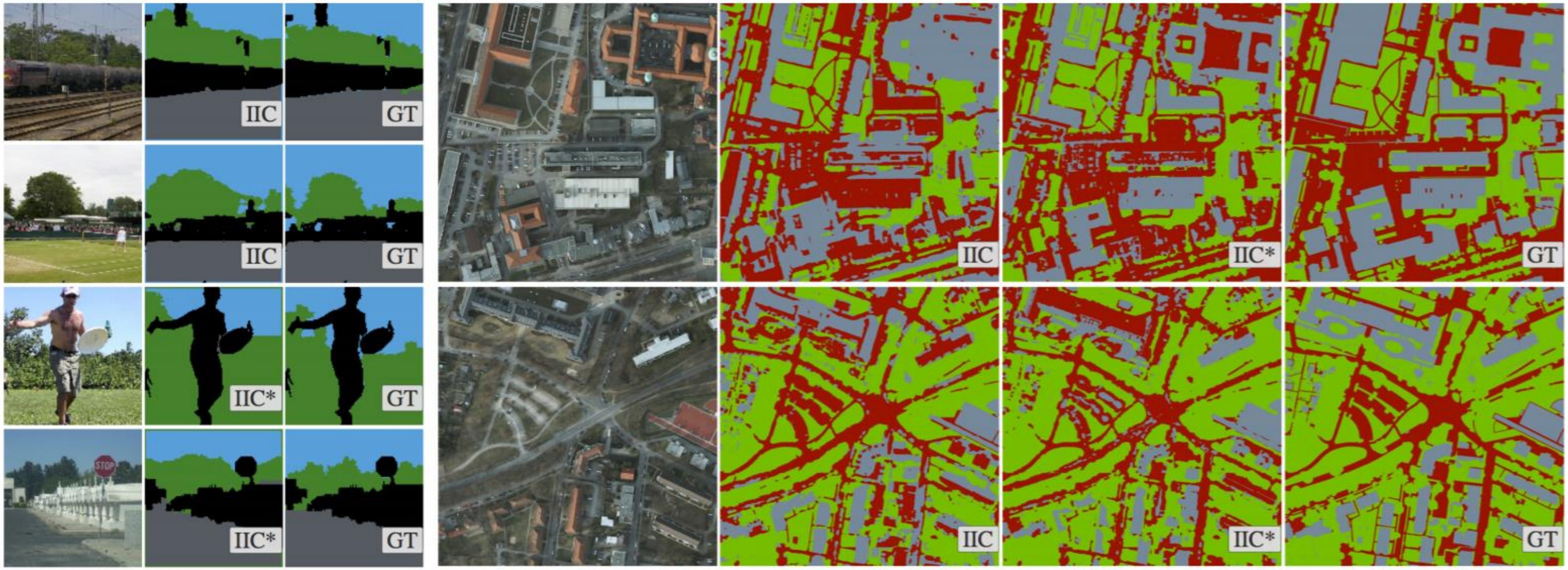


Figure 7: **Example segmentation results (un- and semi-supervised).** Left: COCO-Stuff-3 (non-stuff pixels in black), right: Potsdam-3. Input images, IIC (fully unsupervised segmentation) and IIC* (semi-supervised overclustering) results are shown, together with the ground truth segmentation (GT).

Experiments

- Unsupervised and semi-supervised segmentation

	COCO-Stuff-3	COCO-Stuff	Potsdam-3	Potsdam
Random CNN	37.3	19.4	38.2	28.3
K-means [44]†	52.2	14.1	45.7	35.3
SIFT [39]‡	38.1	20.2	38.2	28.5
Doersch 2015 [17]‡	47.5	23.1	49.6	37.2
Isola 2016 [30]‡	54.0	24.3	63.9	44.9
DeepCluster 2018 [7]† ‡	41.6	19.9	41.7	29.2
IIC	**72.3**	**27.7**	**65.1**	**45.4**

Table 4: **Unsupervised segmentation.** IIC experiments use a single sub-head. Legend: †Method based on k-means. ‡Method that does not directly learn a clustering function and requires further application of k-means to be used for image clustering.

THANKS