Invariant information clustering

Anton Naumov,
Data analyst in ISP RAS
MAY 2021

girafe ai

Based on https://arxiv.org/abs/1807.06653

Foreword

- We use invariant information clustering (IIC)
- Several SOTA results

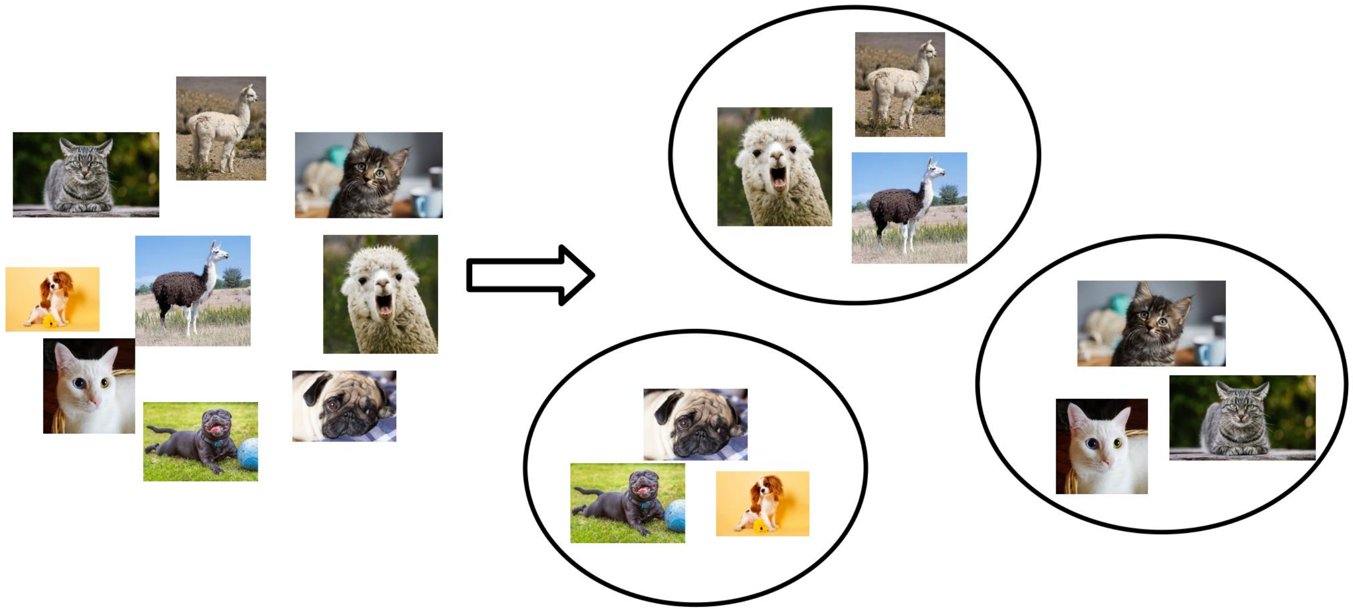
Outline

1. Problem statements
2. Method overview
3. Mutual information loss
4. Couple of tricks
5. Results
6. Summary

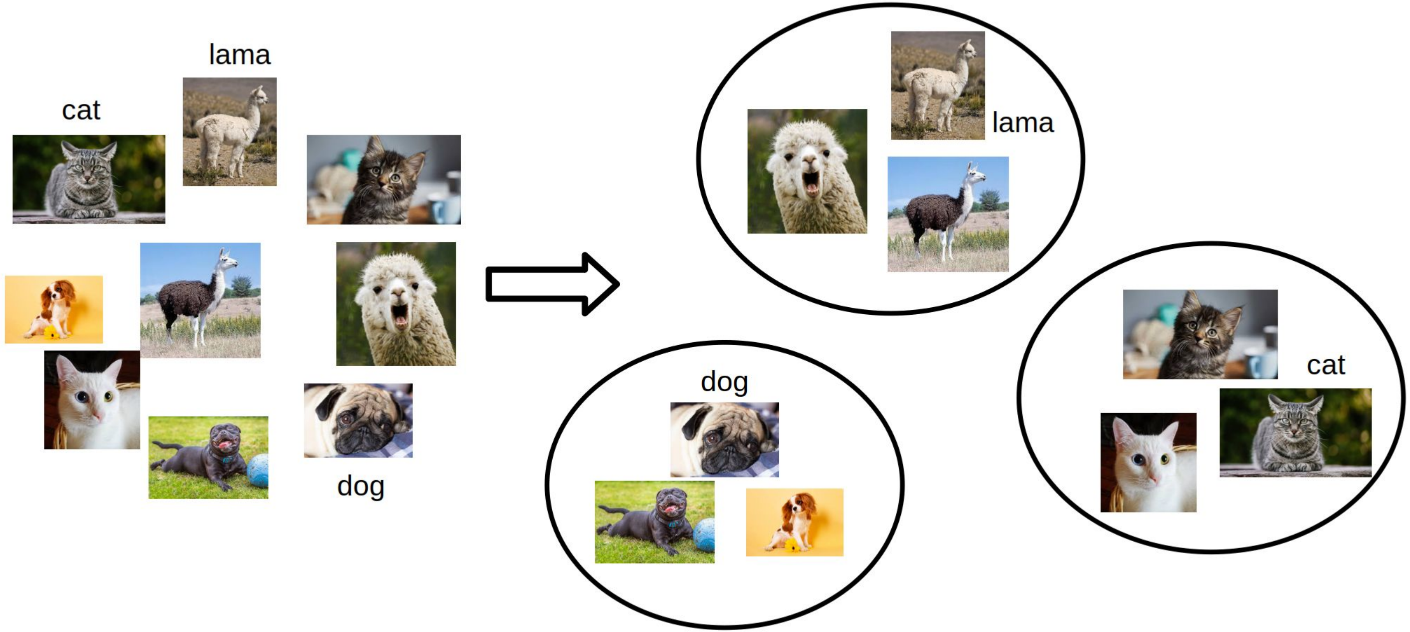
Problem statements

01

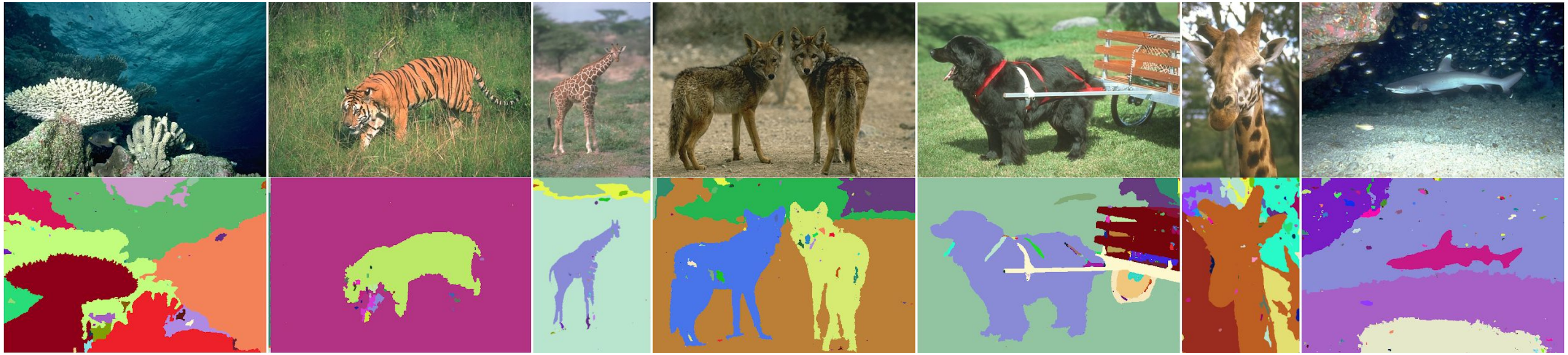
Clustering



Semi-supervised classification

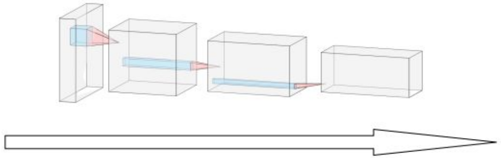


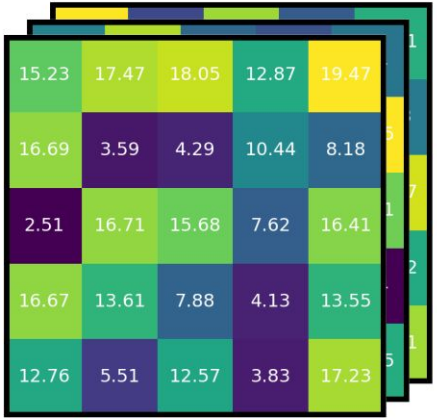
Unsupervised segmentation



Encoder training

Feature
extractor





Input image

Feature maps

girafe ai

Tasks to consider

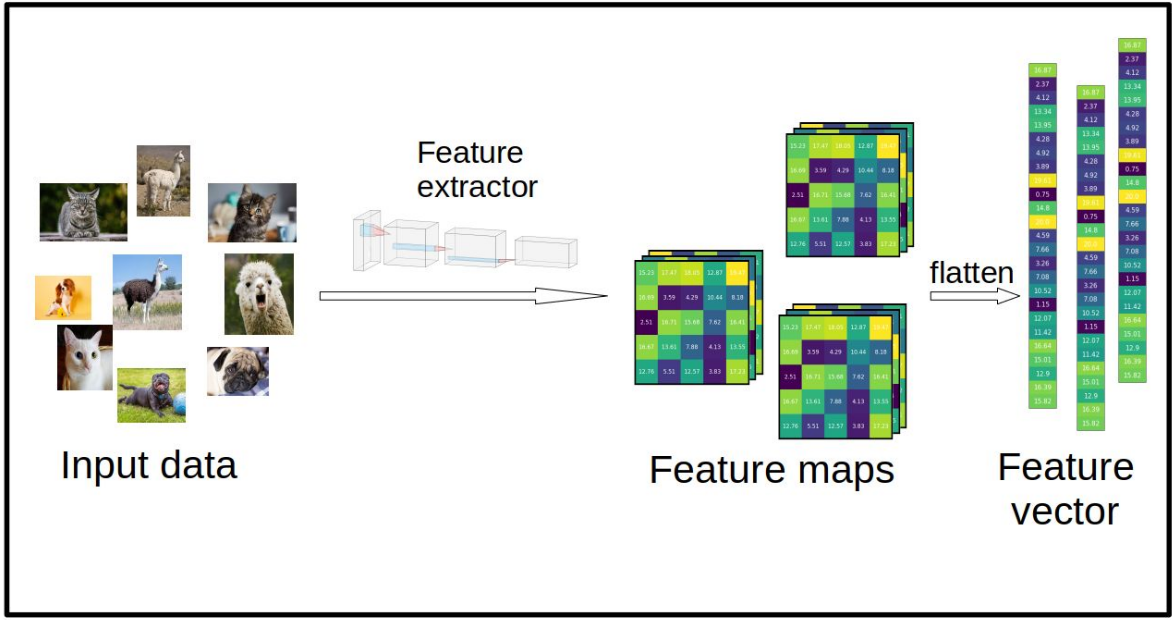
- clustering
- semi-supervised classification
- unsupervised segmentation
- encoder training

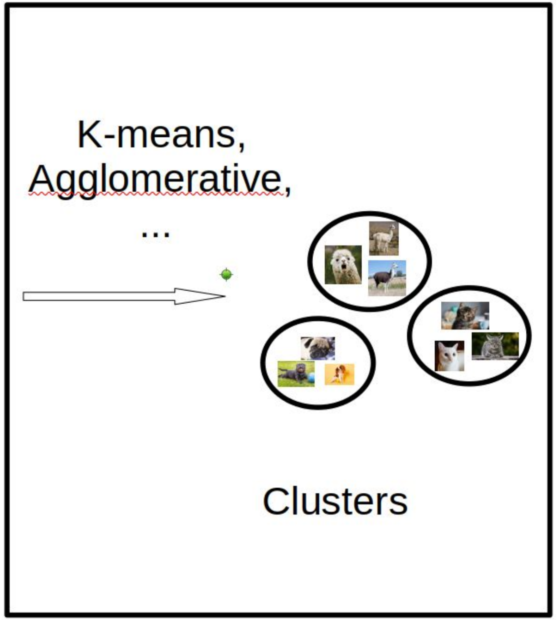
Method overview

girafe
ai

02

2 stage clustering

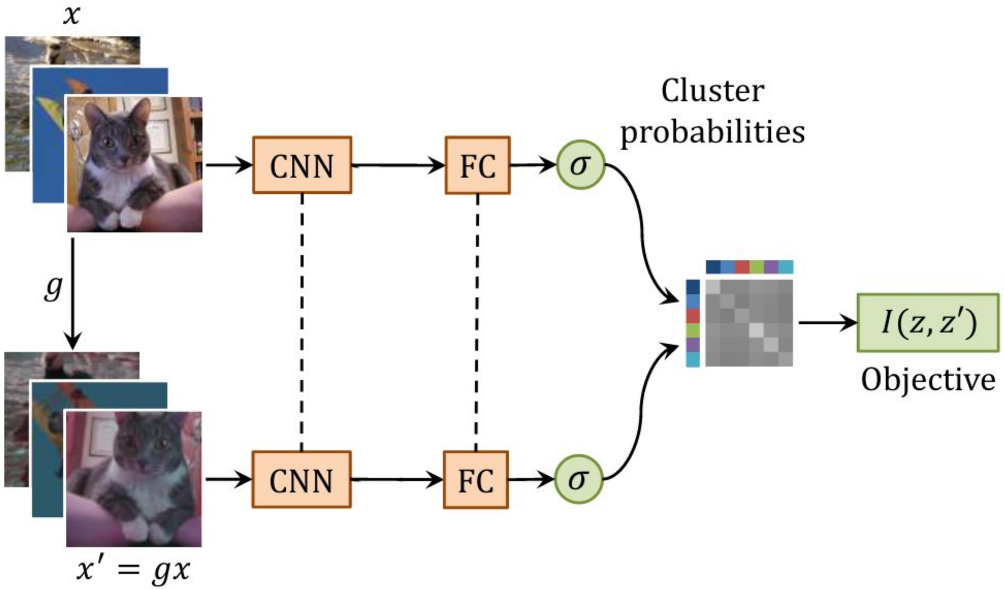




Train feature extractor (i.e. with autoencoder)

Train clusterisator

IIC approach



Mutual information loss

girafe
ai

03

Mutual information

2 random variables *A* and *B* : values from 0 to N_c

Joint probability distribution: $p_{AB}(a,b)$

The marginals: $p_A(a) = \sum_{b=0}^{N_C} p_{AB}(a,b)$ $\qquad p_B(b) = \sum_{a=0}^{N_C} p_{AB}(a,b)$

Mutual information:

$$\boxed{I(A,B) = \sum_{a=0}^{N_C}\sum_{b=0}^{N_C} p_{AB}(a,b)\log\frac{p_{AB}(a,b)}{p_A(a)p_B(a)}}$$

Mutual information

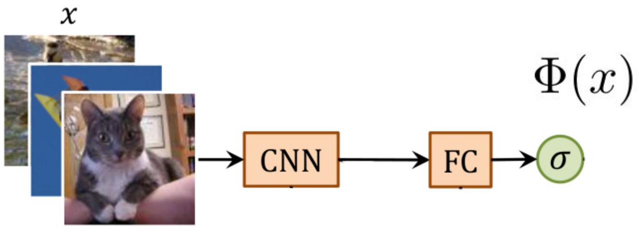
$$I(A,B) = \sum_{a=0}^{N_C} \sum_{b=0}^{N_C} p_{AB}(a,b) \log \frac{p_{AB}(a,b)}{p_A(a) p_B(a)}$$

Properties:

1. Symmetry: $I(A,B) = I(B,A)$
2. $I(A,B) = 0$ if and on if *A* and *B* are independent
3. Non-negative: $I(A,B) > 0$ for any *A* and *B*
4. In some sense maximized when *A* and *B* can be predicted from each other

What are these random variables?

Encoder function: $\Phi(\cdot)$



On original images:

$$\Phi(x_i) = \begin{pmatrix} \Phi_0(x_i) \\ \Phi_1(x_i) \\ \vdots \\ \Phi_{N_C}(x_i) \end{pmatrix}$$

On transformed images:

$$\Phi(gx_i) = \begin{pmatrix} \Phi_0(gx_i) \\ \Phi_1(gx_i) \\ \vdots \\ \Phi_{N_C}(gx_i) \end{pmatrix}$$

Interpretation: $P_{orig}(a|i) = \Phi_a(x_i)$ $P_{trans}(a|i) = \Phi_a(gx_i)$

Joint probabilities estimation

Joints:

$$P_{orig,trans}(a, b) = \frac{1}{N_B} \sum_{i \in batch} \Phi_a(x_i)\Phi_b(gx_i)$$

Symmetrization:

$$P^{sym}_{orig,trans}(a, b) = \frac{1}{2}(P_{orig,trans}(a, b) + P_{orig,trans}(b, a))$$

Mutual information:

$$I(orig, trans) = \sum_{a=0}^{N_C} \sum_{b=0}^{N_C} P_{orig,trans}(a, b) \log \frac{P_{orig,trans}(a, b)}{P_{orig}(a) P_{trans}(b)}$$

Joint probabilities estimation

Joints:

$$P_{orig,trans}(a,b) = \frac{1}{N_B} \sum_{i \in batch} \Phi_a(x_i)\Phi_b(gx_i)$$

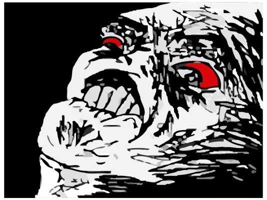
Symmetrization:

$$P^{sym}_{orig,trans}(a,b) = \frac{1}{2}(P_{orig,trans}(a,b) + P_{orig,trans}(b,a))$$

Mutual information:

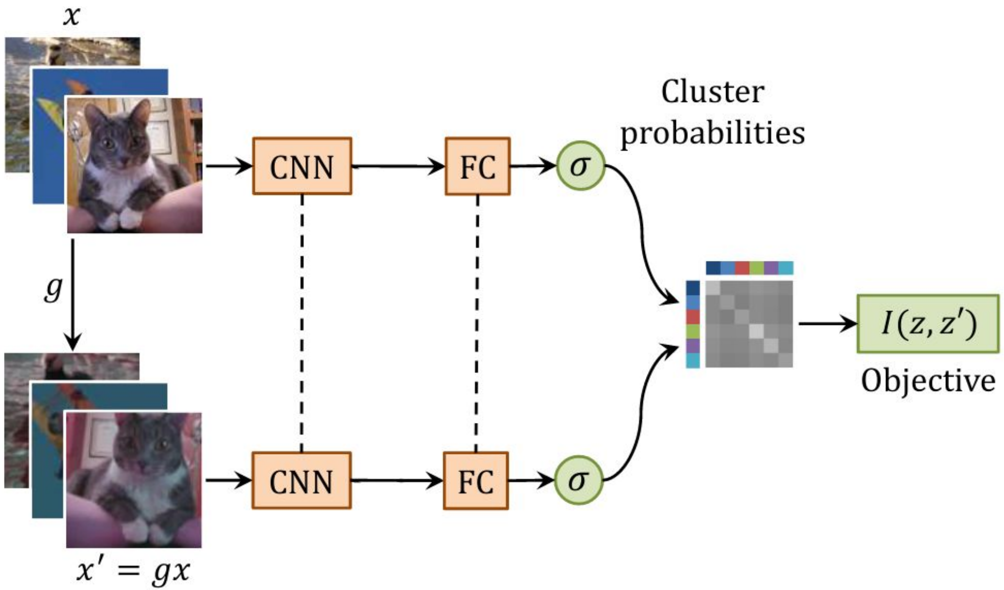
$$I(orig, trans) = \sum_{a=0}^{N_C} \sum_{b=0}^{N_C} P_{orig,trans}(a,b) \log \frac{P_{orig,trans}(a,b)}{P_{orig}(a)P_{trans}(b)}$$

Model assumptions stuff



Joint probabilities estimation

Note: it’s sufficient to estimate joint probabilities



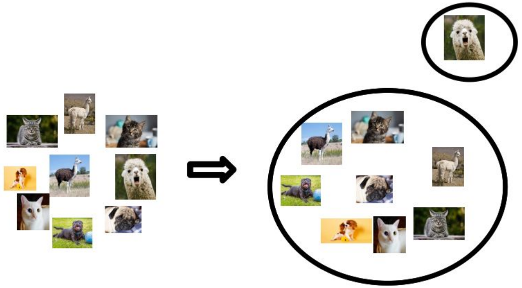
Couple of tricks

girafe
ai

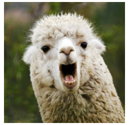
04

Common issues

Clustering degeneracy:



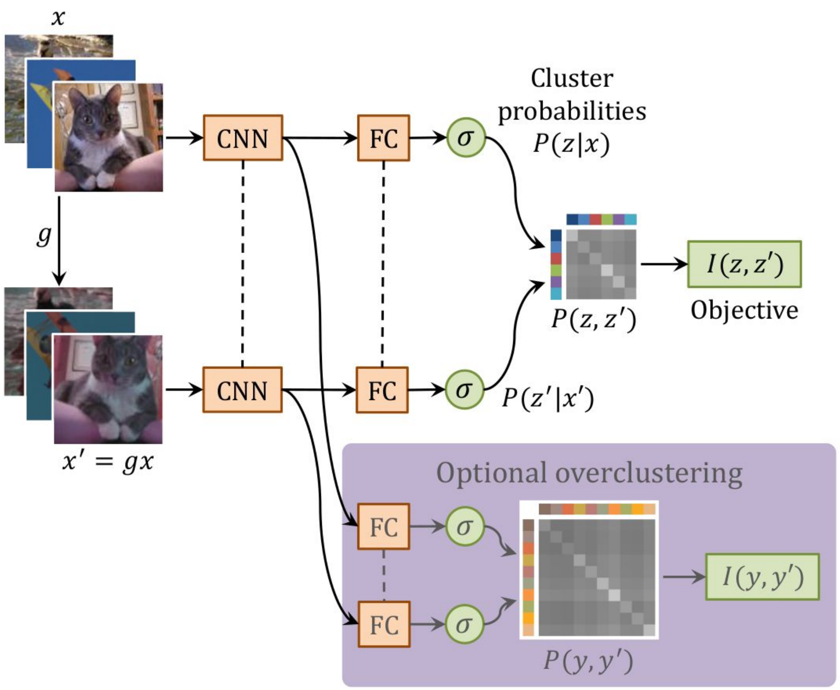
Noisy data:



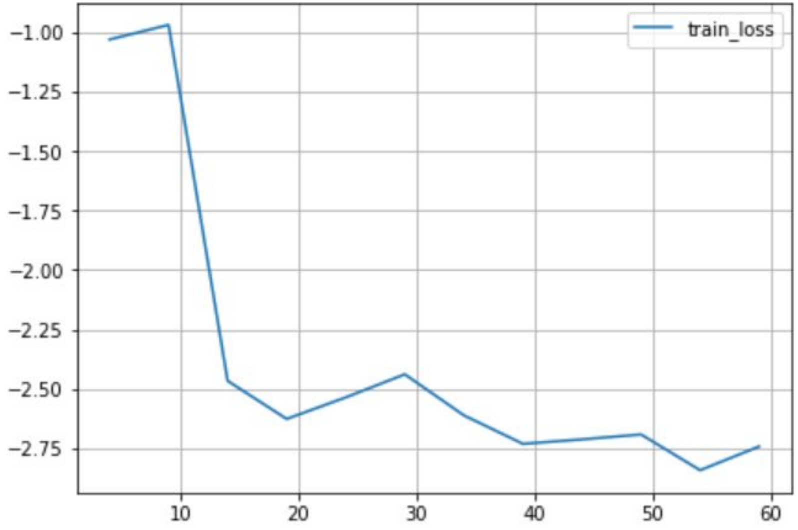
dog

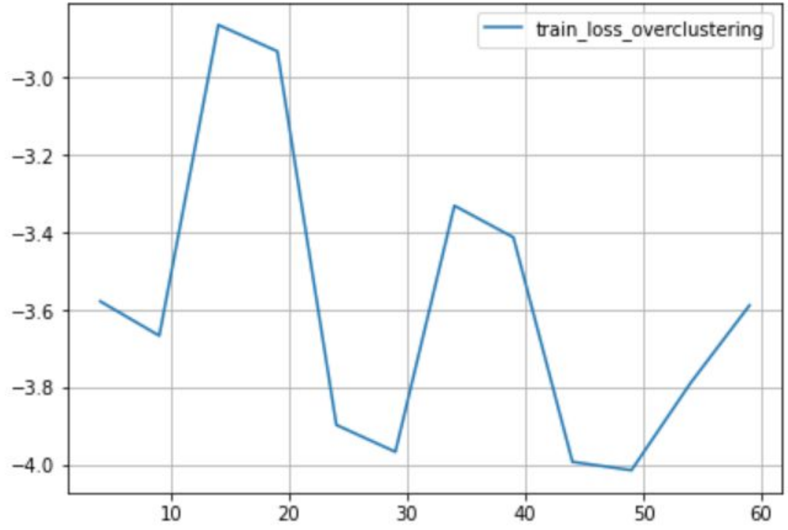
Distractors:

Auxiliary overclustering



Auxiliary overclustering





girafe ai

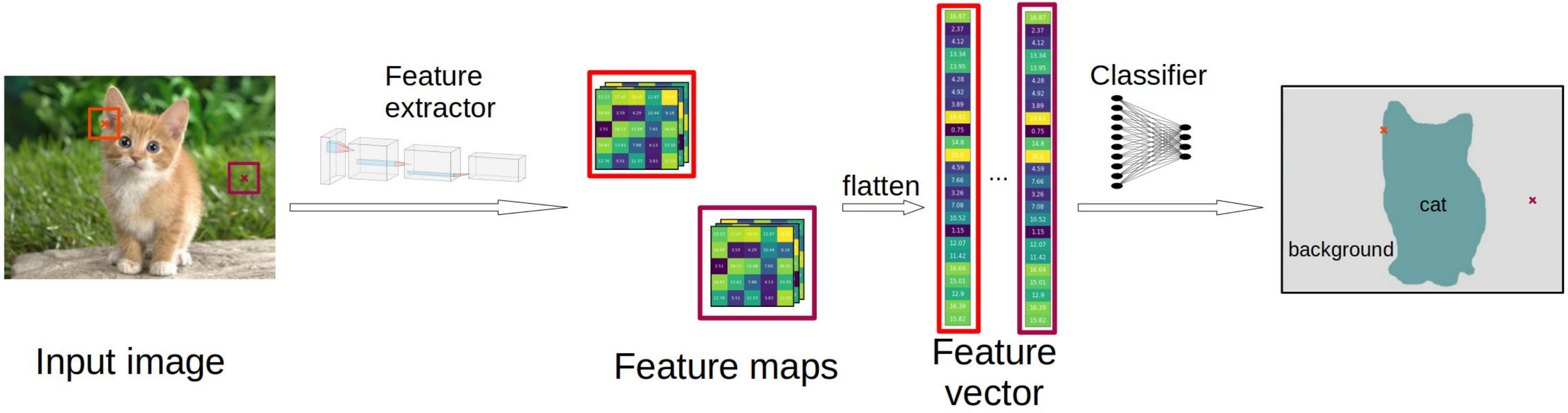
Entropy correction

Another form of mutual information: $$\underbrace{I(A,B) = \frac{1}{2}\big(H(A)+H(B)\big)}_{\text{pushes towards spreading}} - \frac{1}{2}\big(H(A|B)+H(B|A)\big)$$

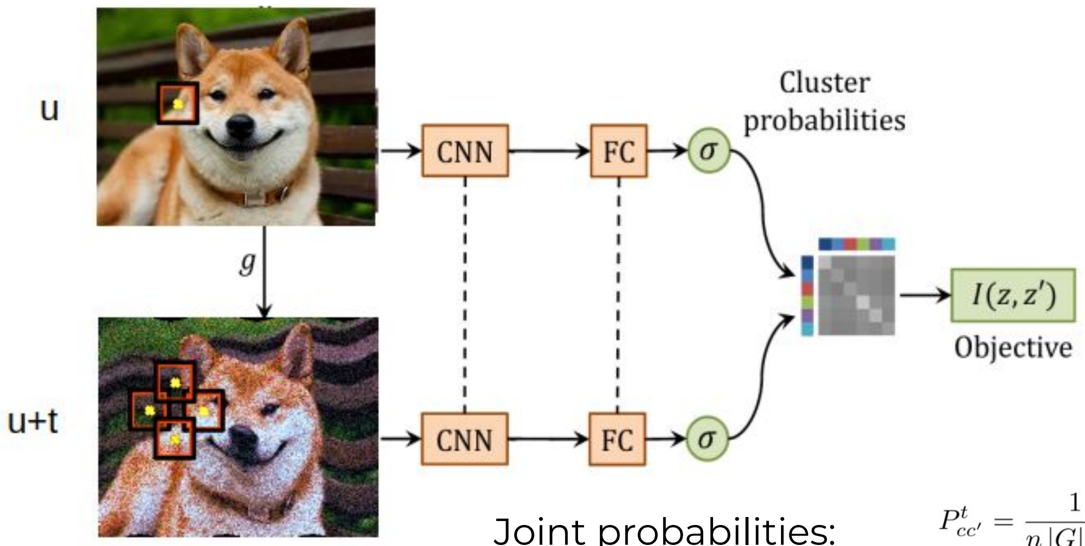
Corrected mutual information: $$I_\lambda(A,B) = I(A,B) + (\lambda - 1)\big(H(A)+H(B)\big)$$

girafe ai

Segmentation



Segmentation



$g \in G$ — number of transforms
$u \in \Omega$ — pixel positions
$t \in T$ — displacements
n — number of images
C — number of classes

Joint probabilities: $$P^t_{cc'} = \frac{1}{n\,|G|\,|\Omega|} \sum_{i=1}^{n} \sum_{g \in G} \sum_{u \in \Omega} P(z = c|x_u) P(z = c'|gx_{u+t})$$

$$I_t = \sum_{c,c'=1}^{C} P^t_{cc'} \ln \frac{P^t_{cc'}}{P^t_c P^t_{c'}}$$

Corrected mutual information: $$I = \frac{1}{|T|} \sum_{t \in |T|} I_t$$

Results

girafe
ai

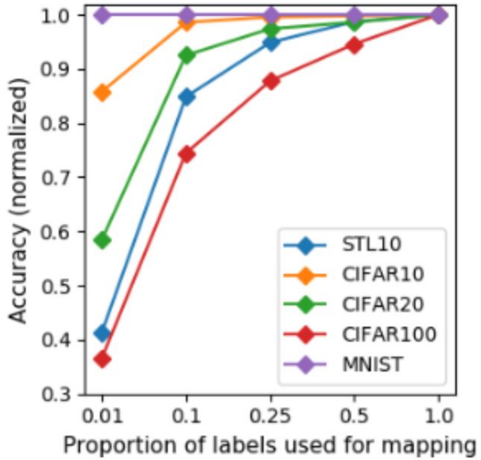
05

Semi-supervised classification

Accuracy

	STL10
Dosovitskiy 2015 [18]†	74.2
SWWAE 2015 [54]†	74.3
Dundar 2015 [19]	74.1
Cutout* 2017 [15]	$\boxed{87.3}$
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.



Unsupervised segmentation

COCO-Stuff-3 and Postdam-3

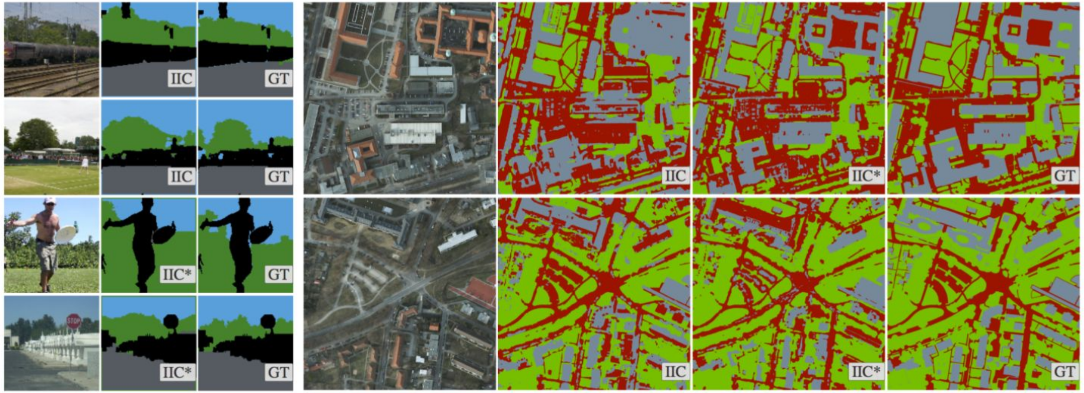


Figure 7: **Example segmentation results (un- and semi-supervised).** Left: COCO-Stuff-3 (non-stuff pixels in black), right: Potsdam-3. Input images, IIC (fully unsupervised segmentation) and IIC* (semi-supervised overclustering) results are shown, together with the ground truth segmentation (GT).

Per-pixel accuracy

	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]‡	$\boxed{54.0}$	$\boxed{24.3}$	$\boxed{63.9}$	$\boxed{44.9}$
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.

girafe ai

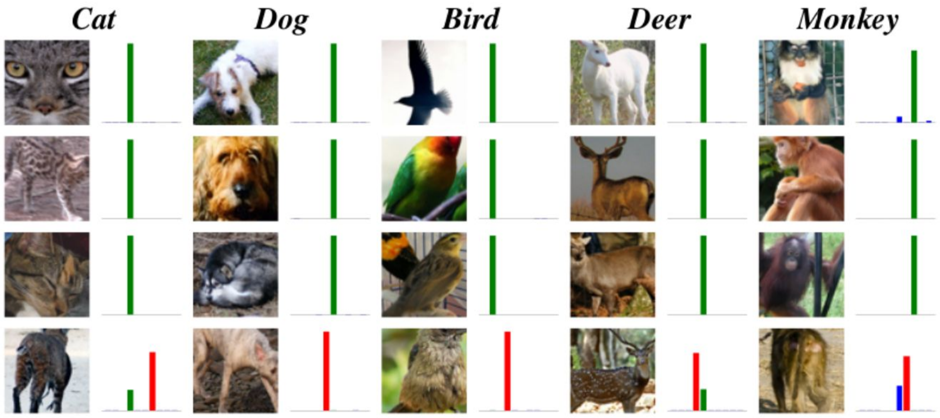
Clustering

Accuracy

	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	$\boxed{52.2}$	$\boxed{23.8}$	97.8
DeepCluster 2018 [7]† ‡	33.4⋆	37.4⋆	18.9⋆	65.6 ⋆
ADC 2018 [24]	$\boxed{53.0}$	32.5	16.0⋆	$\boxed{99.2}$
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 dataset



girafe ai

Overall results

TASK	DATASET	MODEL	METRIC NAME	METRIC VALUE	GLOBAL RANK	RESULT	BENCHMARK
Unsupervised Image Classification	CIFAR-10	IIC	Accuracy	61.7	# 3		Compare
Unsupervised Image Classification	CIFAR-20	IIC	Accuracy	25.7	# 4		Compare
Unsupervised Semantic Segmentation	COCO-Stuff-15	IIC	Accuracy	27.7	# 1		Compare
Unsupervised Semantic Segmentation	COCO-Stuff-3	IIC	Accuracy	72.3	# 1		Compare
Unsupervised MNIST	MNIST	IIC	Accuracy	99.3	# 1		Compare
Unsupervised Image Classification	MNIST	IIC	Accuracy	99.3	# 1		Compare
Unsupervised Semantic Segmentation	Potsdam	IIC	Accuracy	65.1	# 1		Compare
Unsupervised Semantic Segmentation	Potsdam-3	IIC	Accuracy	45.4	# 1		Compare
Unsupervised Image Classification	STL-10	IIC	Accuracy	61.00	# 3		Compare
Image Classification	STL-10	IIC	Percentage correct	88.8	# 37		Compare
Semi-Supervised Image Classification	STL-10	IIC	Accuracy	88.8	# 2		Compare

paperswithcode.com

girafe ai

Summary

1. simple method with several SOTA results
2. multiple problems
3. can be used as encoder trainer
4. data type agnostic (theoretically)

Links

1. Original paper:
   https://arxiv.org/abs/1807.06653
2. Authors git:
   https://github.com/xu-ji/IIC
3. My tutorial (with colab version):
   https://github.com/vandedok/IIC_tutorial

Thanks for attention!

Questions? Additions? Welcome!

girafe
ai