Multi-Modal Deep Clustering: Unsupervised Partitioning of Images

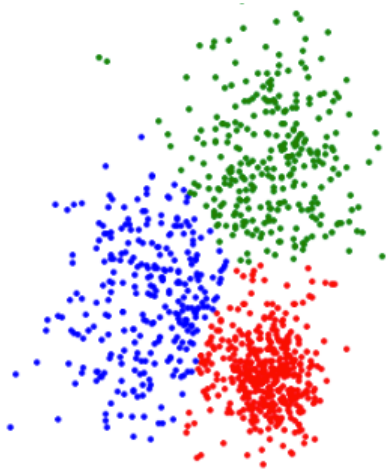
Guy Shiran, Daphna Weinshall

School of Computer Science and Engineering, The Hebrew University of Jerusalem

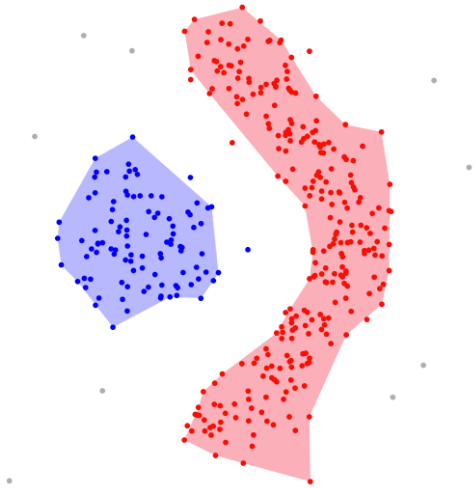
25th International Conference on Pattern Recognition (ICPR2020)

Data Clustering

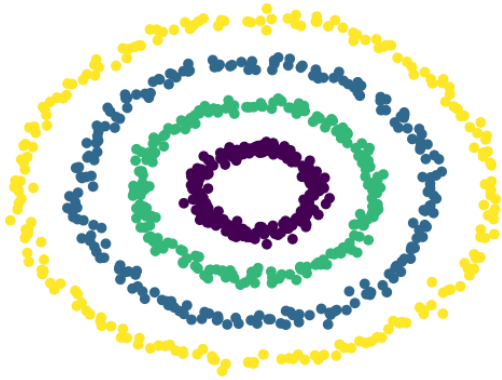
The objective of data clustering is to partition data points into groups such that points in each group are more similar to each other than to data points in the other groups.



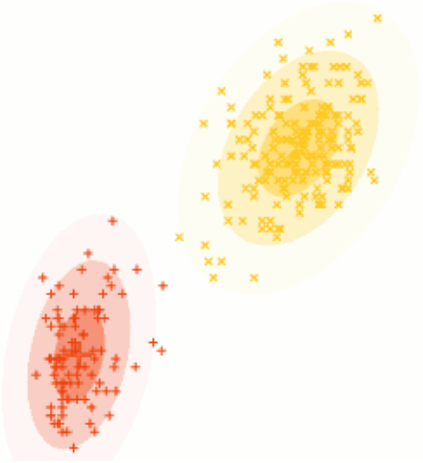
Centroid-based
(e.g k-means)



Density-based
(e.g DBSCAN)



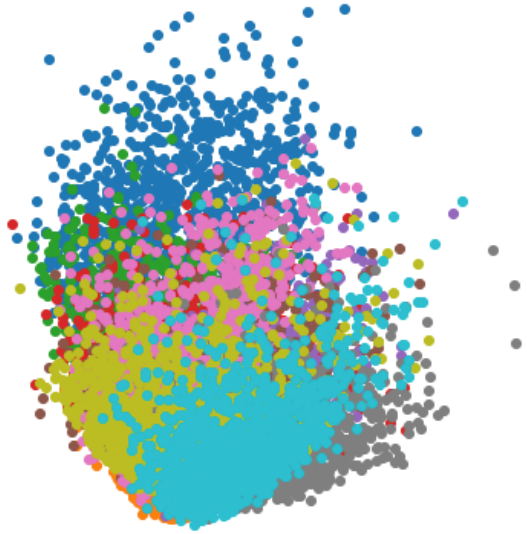
Spectral-based
(e.g Spectral Clustering)



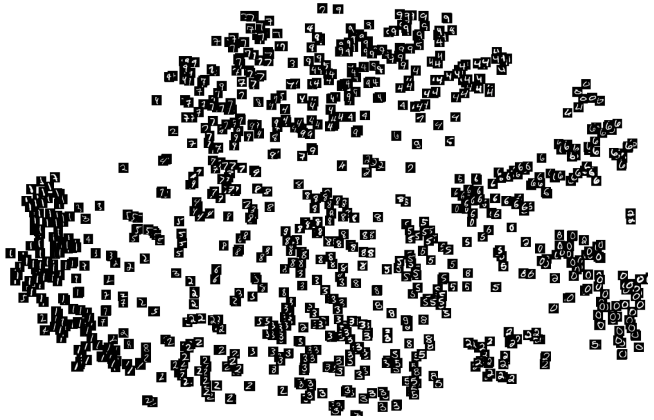
Distribution-based
(e.g GMMs)

Image Clustering

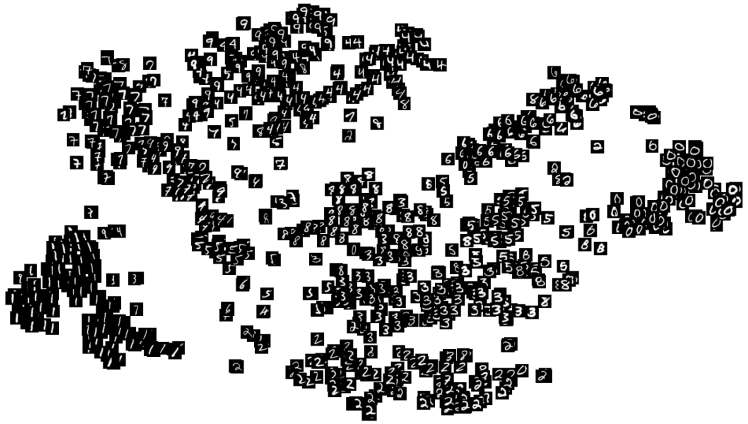
Images in pixel space are highly entangled - semantically similar images are not necessarily similar in the high dimensional space the images reside in. We would like to produce more meaningful representations to work on.



PCA of MNIST



t-SNE of MNIST
(in pixel space)

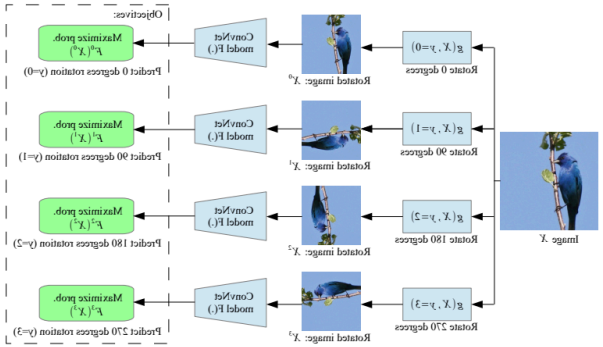


t-SNE of MNIST
(in AE latent space)

Approaches

Two main paradigms:

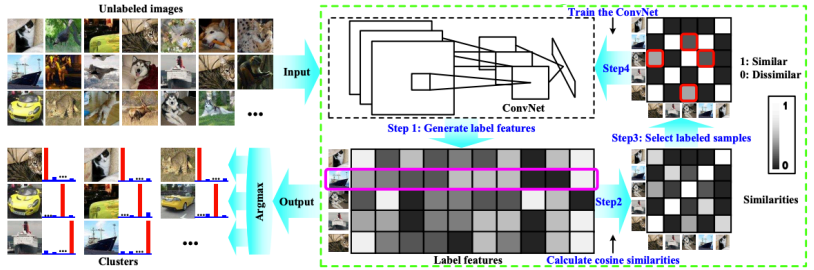
Step 1: Unsupervised representation learning
Step 2: Apply clustering algorithm



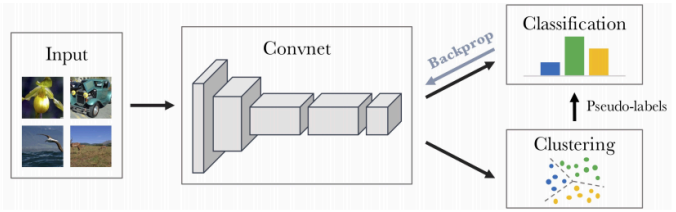
RotNet

K-Means

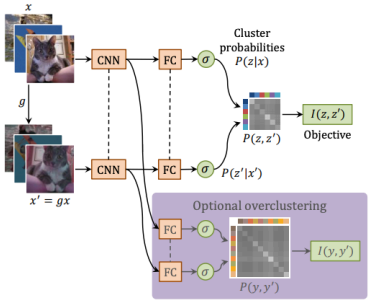
Deep Clustering: Solve representation learning and clustering jointly



Deep Adaptive Clustering



DeepCluster



Invariant Information Clustering

[1] Deep adaptive image clustering. Chang et al. 2017.
[2] Unsupervised representation learning by predicting image rotations. Gidaris et al. 2018.
[3] Deep clustering for unsupervised learning of visual features. Caron et al. 2018.
[4] Invariant information clustering for unsupervised image classification and segmentation. Ji et al. 2019.

Our Method

- In the conventional supervised learning setting, we are given training pairs $\{(x_i, y_i)\}_{i=1}^{n}$ and typically solve:

$$min_\theta \frac{1}{n} \sum_i \ell(f_\theta(x_i), y_i)$$

- In the absence of ground truth labels, we can attempt to learn them alongside model parameters:

$$min_{y_1,\ldots,y_n,\theta} \frac{1}{n} \sum_i \ell(f_\theta(x_i), y_i)$$

- But now we are at risk of cluster collapse, i.e. the trivial solution of:

$$y_1 = y_2 = \cdots = y_n$$

Our Method

- Adopt approach of NAT[1] for feature collapse by using **fixed** targets.
- Given images $\{x_i\}_{i=1}^{n}$, fix n targets randomly sampled from a Gaussian Mixture Model:

$$Y = \{y_i | y_i \in \mathbb{R}^d, \|y_i\|_2 = 1\}, \quad |Y| = n$$

- Learn model $f_\theta : X \to \mathbb{R}^d$ and mapping $P : [n] \to [n]$:

$$min_{P,\theta} \frac{1}{n} \sum_i \ell(f_\theta(x_i), y_{P(i)})$$

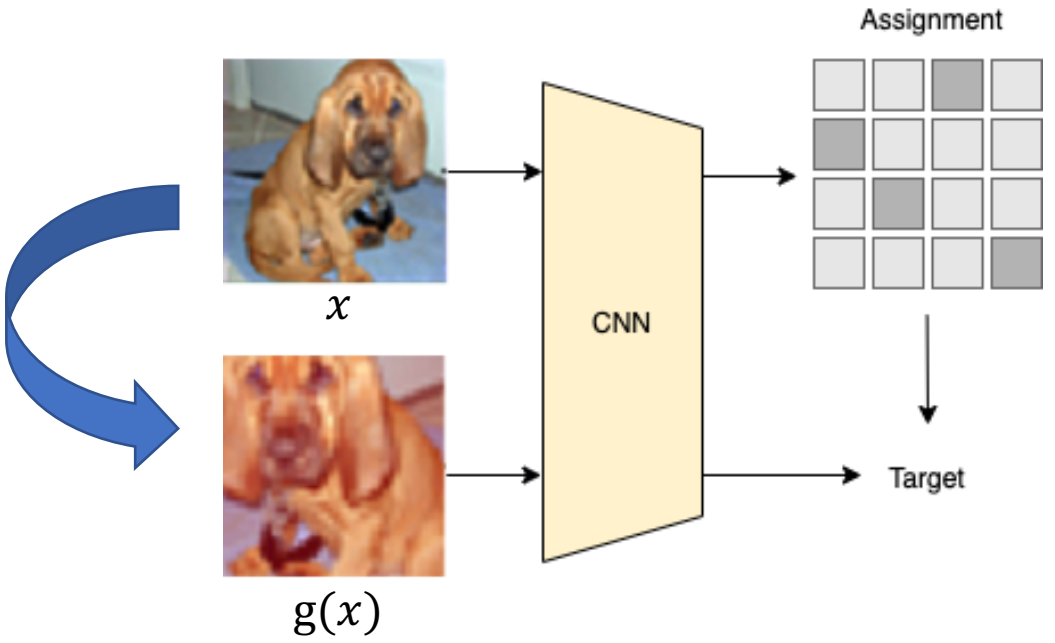
[1] Unsupervised Learning by Predicting Noise, Bojanowski & Joulin, 2017

Our Method

$$min_{P,\theta} \frac{1}{n} \sum_{i} \ell(f_\theta(x_i), y_{P(i)})$$

Two-step mini-batch optimization:

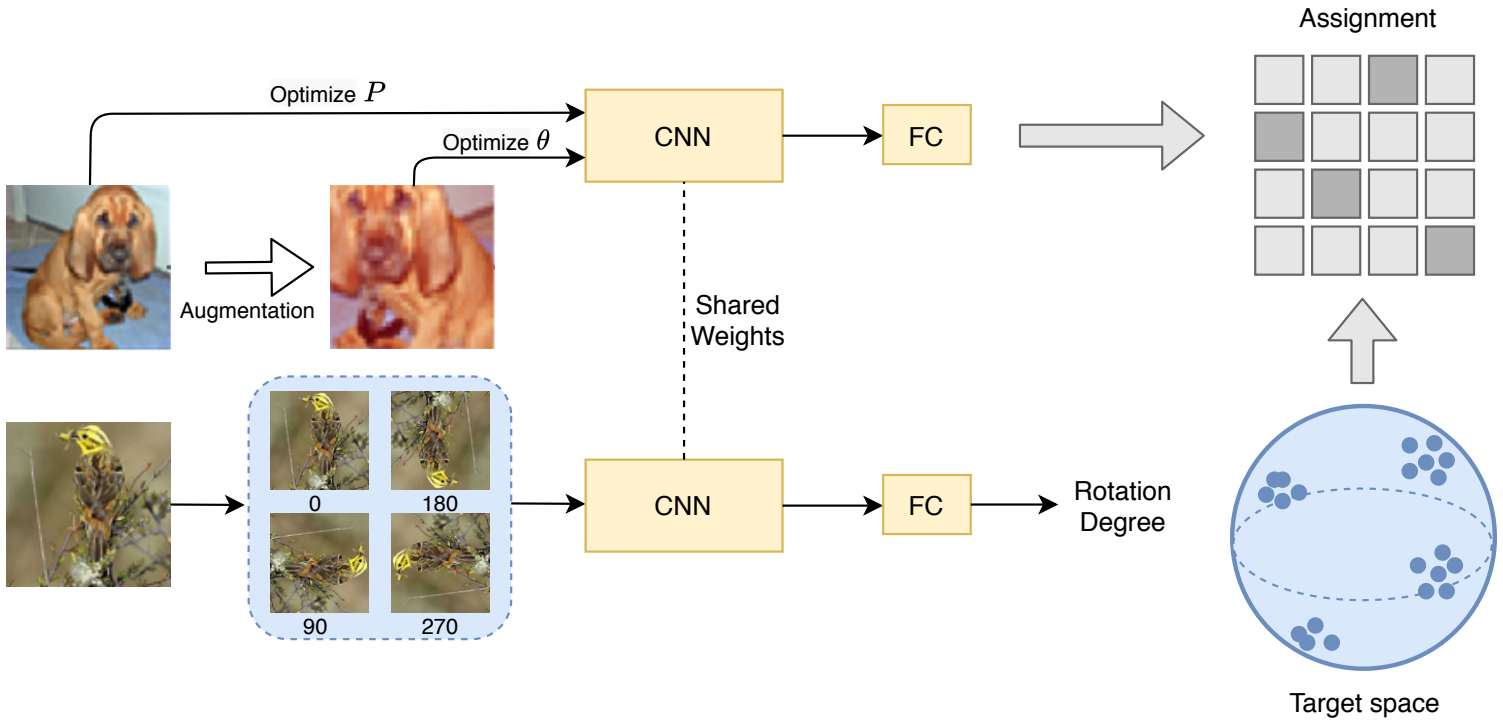
1. Solve for P with original images $\{x_i\}_{i=1}^{b}$ (linear assignment problem) .
2. Solve for θ with transformed images $\{g(x_i)\}_{i=1}^{b}$ and assigned targets (backprop + sgd step).



$g: X \to X$ is a random image transformation

Auxiliary Rotation Loss

- We have a framework that learns image features alongside cluster assignments.
- Can we further enhance the image features to facilitate a better clustering?



Experimental Setup

- ResNet-18 backbone for natural image datasets and 4-layer ConvNet for MNIST.

Name	Classes	Samples	Dimension
MNIST	10	70,000	28×28×1
CIFAR-10	10	60,000	32×32×3
CIFAR-100	20	60,000	32×32×3
STL-10	10	13,000	96×96×3
ImageNet-10	10	13,000	96×96×3
Tiny-ImageNet	200	100,000	64×64×3

Image datasets used in our experiments

Results

	MNIST		CIFAR-10		CIFAR-100		STL-10		ImageNet-10		Tiny-ImageNet	
	NMI	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI	ACC
k-means	0.499	0.572	0.087	0.228	0.083	0.129	0.124	0.192	0.119	0.241	0.065	0.025
SC	0.663	0.696	0.103	0.247	0.090	0.136	0.098	0.159	0.151	0.274	0.063	0.022
AE	0.725	0.812	0.239	0.313	0.100	0.164	0.249	0.303	0.210	0.317	0.131	0.041
DEC	0.772	0.843	0.257	0.301	0.136	0.185	0.276	0.359	0.282	0.381	0.115	0.037
JULE	0.913	0.964	0.192	0.272	0.103	0.137	0.182	0.277	0.175	0.300	0.102	0.033
DAC	0.935	0.978	0.396	0.522	0.185	0.238	0.249	0.303	0.394	0.527	0.190	0.066
IIC	**0.978**	**0.992**	0.512	0.617	0.224	0.257	0.431	0.499	-	-	-	-
DCCM	-	-	0.496	0.623	0.285	0.327	0.376	0.482	0.608	0.710	0.224	0.108
Ours (avg.)	0.971	0.990	**0.703**	**0.820**	**0.418**	**0.446**	**0.593**	**0.694**	**0.719**	**0.811**	**0.274**	**0.119**
Ours (ste)	±.000	±.000	±.011	±.019	±.003	±.006	±0.005	±0.013	±0.008	±0.012	±0.001	±0.001
Ours (best)	0.973	0.991	0.720	0.843	0.423	0.464	0.609	0.741	0.732	0.830	0.277	0.121

Unsupervised clustering results.

Summary

- Clustering framework that trains a ConvNet by learning cluster assignments alongside model parameters by solving a linear assignment problem using the Hungarian algorithm.
- Random image transformations insert prior knowledge of invariance within clusters into model.
- Auxiliary rotation loss is very effective in helping model learn better image features that facilitate a quality clustering.

Our code is available at: https://github.com/guysrn/mmdc