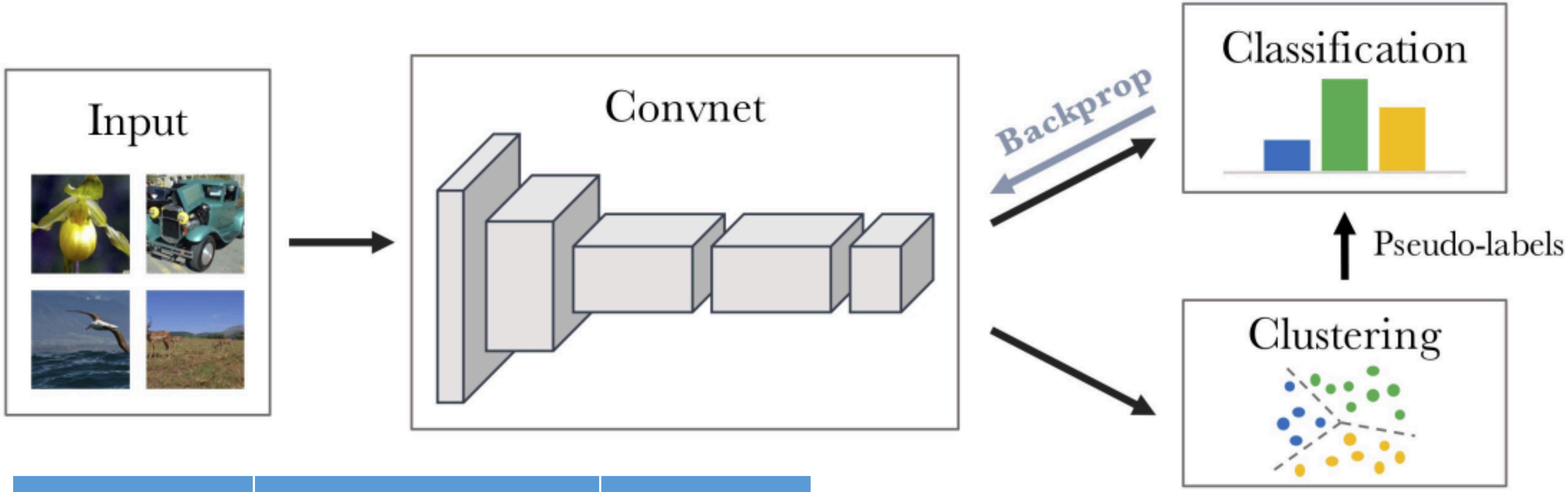
Collaborative Online Deep Clustering for Unsupervised Representation Learning

Xiaohang Zhan[1], Jiahao Xie[2], Ziwei Liu[1], Yew Soon Ong[2], Chen Change Loy[2]

[1]Multimedia Laboratory, The Chinese University of Hong Kong

[2]Nanyang Technological University

Deep Clustering



Backbone	Devices	Time
AlexNet	P100 (x1)	12 days
ResNet-50	GTX 1080TI (x8)	10 days

Low efficiency when trained on ImageNet

[1] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In ECCV, pages 132–149, 2018.

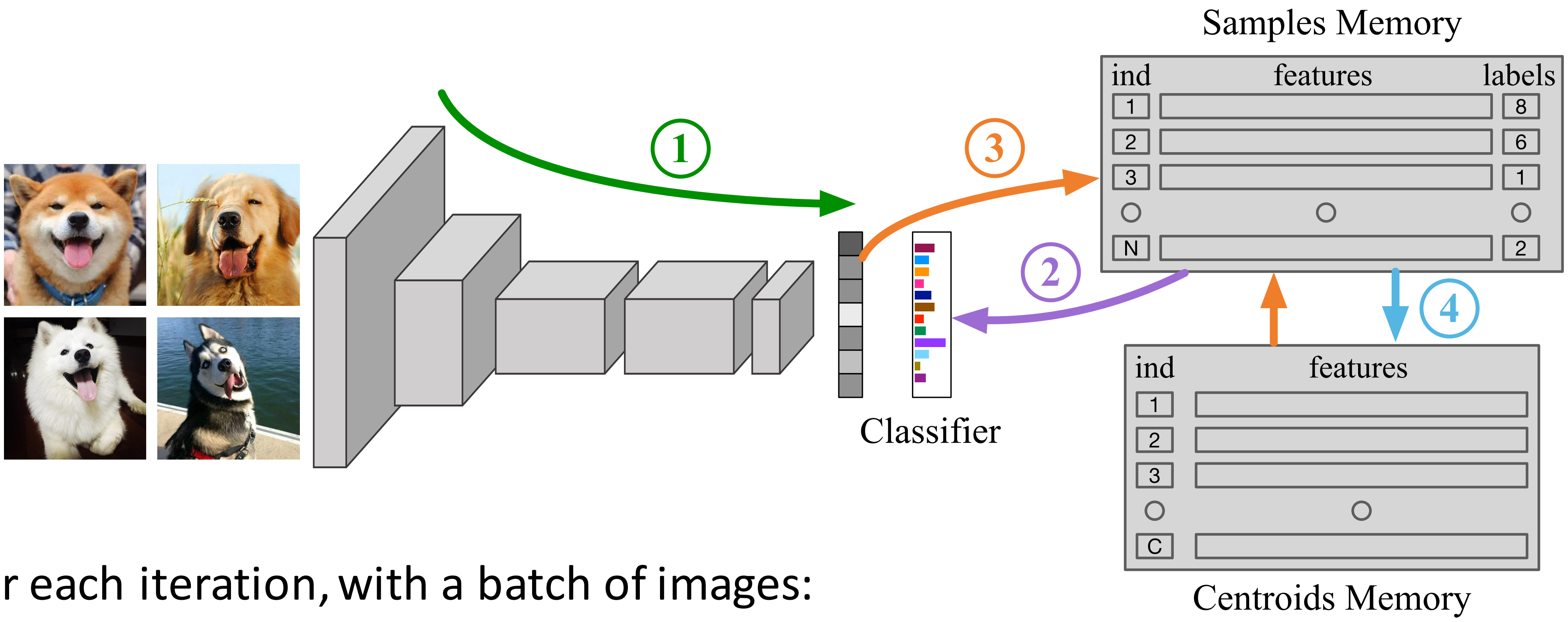
Deep Clustering

For each epoch:

1. extract features of the whole training set ⟸ extra overhead
2. perform clustering and assign new labels
3. randomly initialize the classifier ⟸ long time to converge
4. jointly train CNN + classifier

[1] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In ECCV, pages 132–149, 2018.

Online Deep Clustering (Ours)



For each iteration, with a batch of images:

1. network forward;
2. read labels from samples memory, perform back-propagation to update the CNN;
3. update samples memory: update features, re-assign new labels of this batch;
4. update centroids memory: re-computing involved centroids.

Online Deep Clustering (Ours)

- Loss Re-weighting.
  - Weights are set in each iteration
  - Loss weight: $w_c \propto \frac{1}{\sqrt{N_c}}$
  - To avoid ODC from drifting into a few huge clusters.
- Dealing with Small Clusters.
  - Perform in each iteration.
  - Procedure:

    Repeat until no small clusters exist:

    1. Find a small cluster C;
    2. Disperse samples in C to other normal clusters to make it empty;
    3. Split the largest normal cluster into two parts by K-Means;
    4. Randomly choose one part as the new C.

ODC v.s. DC

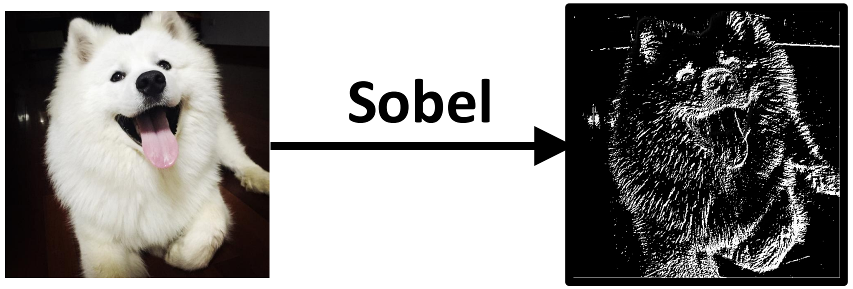
Benefits:

1. The features are stored and continuously updated → No longer need ad-hoc feature extraction. → faster
2. The labels are instantly re-assigned in each iteration rather than in each epoch. → avoids unnecessary back-propagation at the start when labels are noisy → faster
3. The assigned labels are updated smoothly → The classifier evolves steadily → faster and better

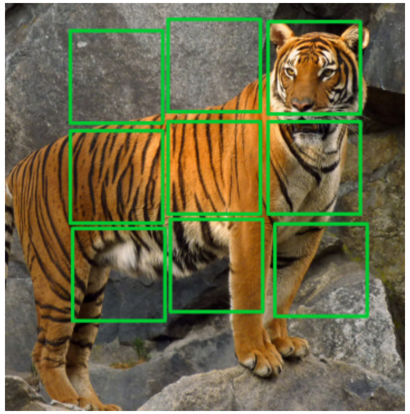
Backbone	Devices	Time	VOC07 (SVM)
DC (AlexNet)	P100 (x1)	12 days	-
DC (ResNet-50)	GTX 1080TI (x8)	10 days	69.12
ODC (ResNet-50)	GTX 1080TI (x8)	2.7 days	69.79

Avoiding "Shortcut" Solutions

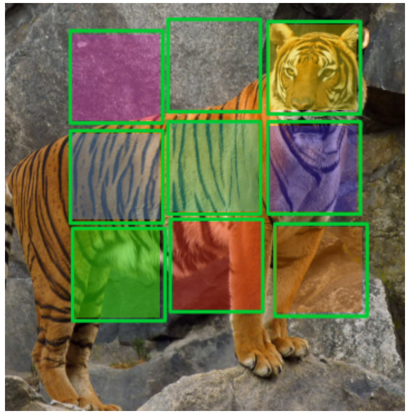
- Color Removal (to avoid clustering according to colors)



- Patch gaps (to avoid exploring edge continuity)



- Color jittering (to avoid exploring chromatic aberration)

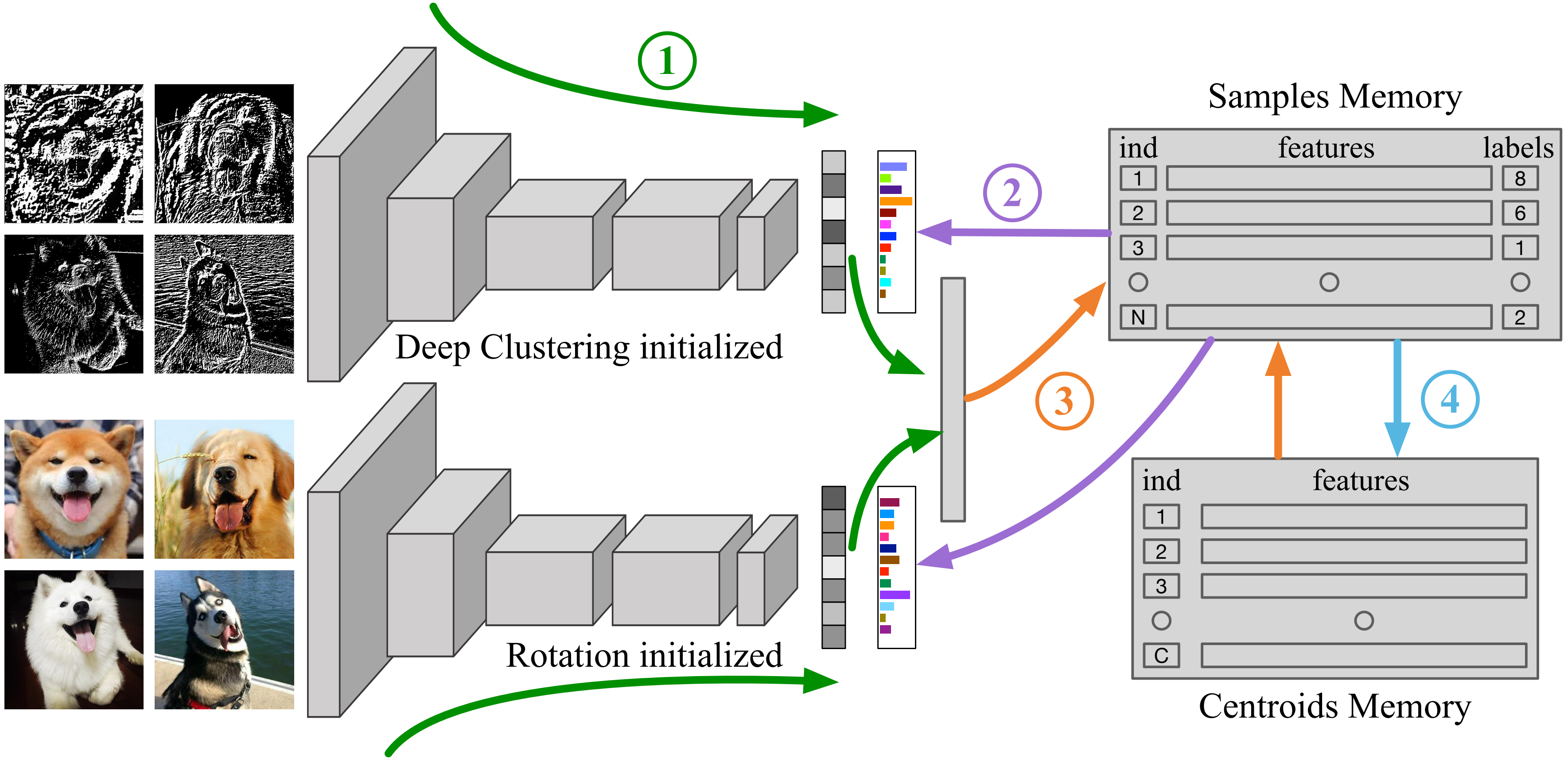


Summary

- Deep clustering
  - Learn: inter-image relationships
  - Shortcut: clustering based on color information
- Colorization
  - Learn: color distribution of different semantic regions
  - Shortcut: simply relying on textures
- Jigsaw Puzzles / Context Prediction
  - Learn: intra-image structures
  - Shortcut: exploring edge continuity, chromatic aberration
- Rotation Prediction
  - Learn: orientation distribution
  - Ambiguity: objects without default orientations

How about allowing these approaches to constrain each other?

Collaborative Online Deep Clustering (Ours)



Experiments

	CODC (2 models)		CODC (3 models)		
components	DC	ROT	DC	ROT	CLS
before CODC	69.12	67.35	69.12	67.35	77.7
after CODC	75.79	74.54	76.48	72.94	78.05
concatenate	76.33		**79.82**		

Table 2. Ablation study on VOC2007 SVM classification.

DC: Deep Clustering
ROT: Rotation Prediction
CLS: Mixup Classification with clustering results.

Thank you for listening