NEURAL INFORMATION PROCESSING SYSTEMS

Self-Supervised Visual Representation Learning with Semantic Grouping

Xin Wen[1], Bingchen Zhao[2, 3], Anlin Zheng[1, 4], Xiangyu Zhang[4], and Xiaojuan Qi[1]

[1]The University of Hong Kong [2]University of Edinburgh [3]LunarAI [4]MEGVII Technology





Why scene-centric pre-training

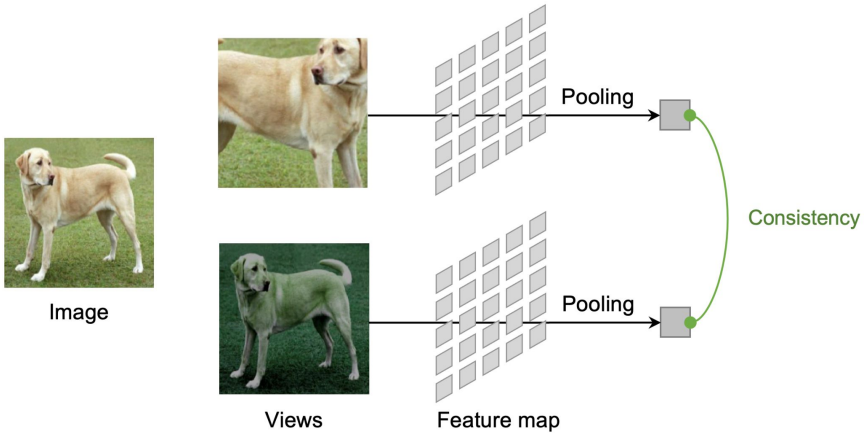
- Uncurated, easier data collection, less human labor
  - Unlabeled ImageNet is not really unlabeled
- Closer to the downstream data distribution (det & seg)
- More information in one image (multiple objs, complex layout)

Table 1: Overview of the training datasets. We sample a uniform and long-tailed (LT) subset of 118K images from ImageNet. On OpenImages, we sample a random subset of 118K images. The complete train splits are used for COCO and BDD100K. The figure shows some examples.

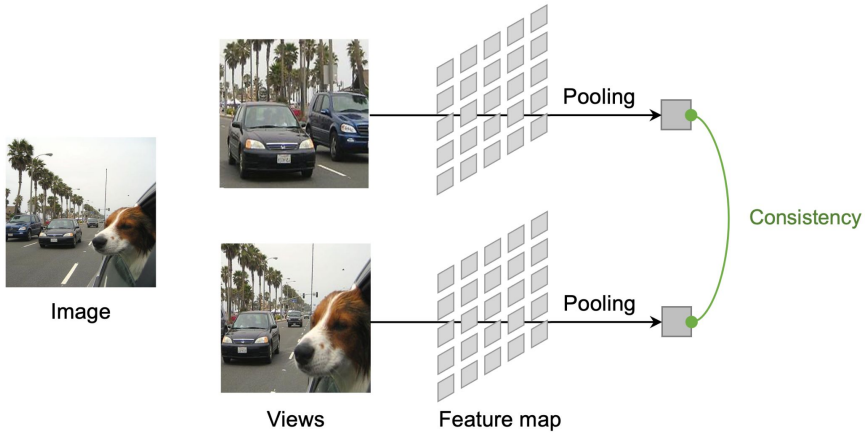
Pretrain Data	#Imgs	#Obj/Img	Uniform	Discriminative
ImageNet-118K [12]	118 K	1.7	✓	✓
ImageNet-118K-LT [12]	118 K	1.7	✗	✓
COCO [31]	118 K	7.3	✗	✓
OpenImages-118K [28]	118 K	8.4	✗	✓
BDD100K [52]	90 K	-	✗	✗

Van Gansbeke, Wouter, et al., Revisiting Contrastive Methods for Unsupervised Learning of Visual Representations, NeurIPS 2021.

Scene-centric pre-training requires more than instance discrimination

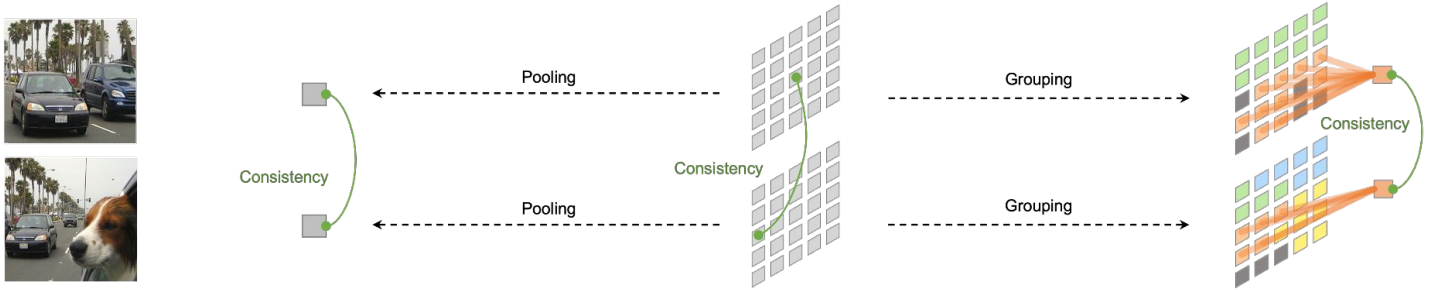


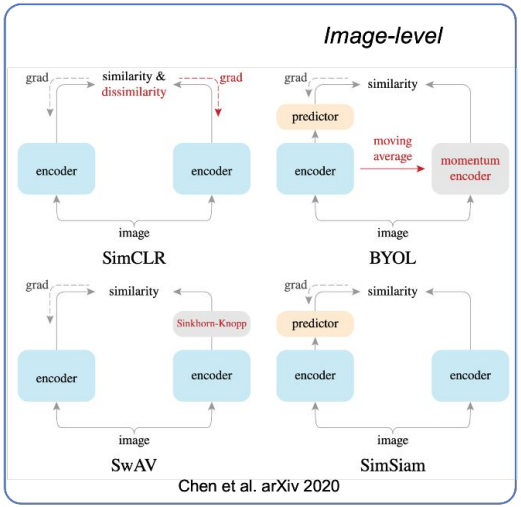
Seems good for object-centric images

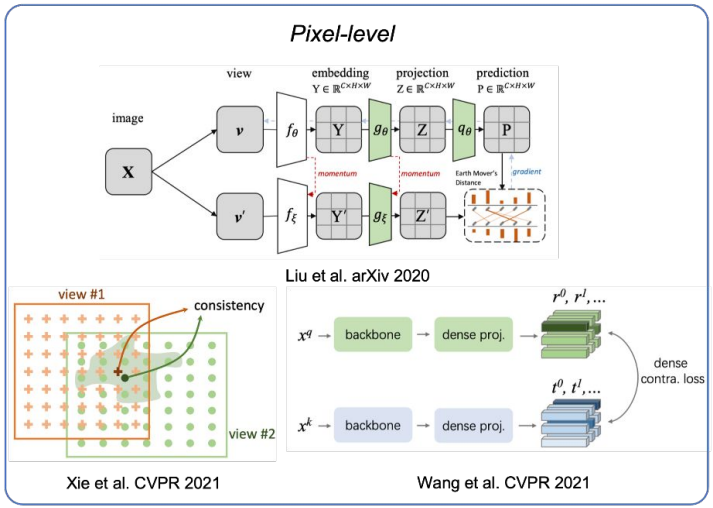


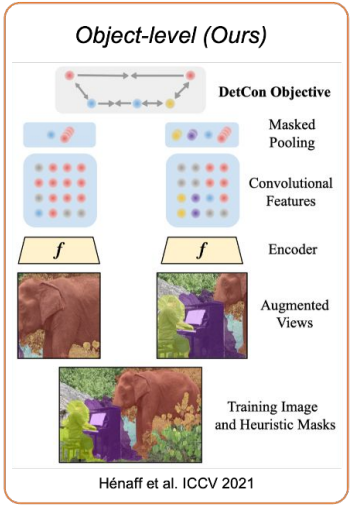
Scene-centric images should not be simply treated as a single feature vector!

*Image-level, pixel-level, and **object-level** contrastive learning*



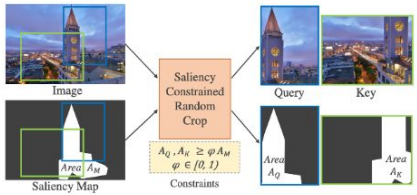




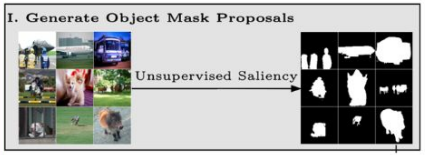


Fantastic objects and where to find them (without supervision)?

Saliency estimators

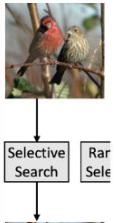


Selvaraju et al. CVPR 2021

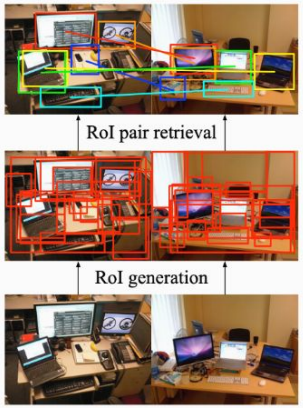


Gansbeke et al. ICCV 2021

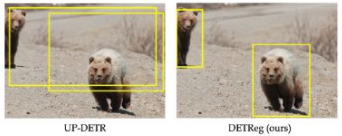
Selective search



Wei et al. NeurIPS 2021

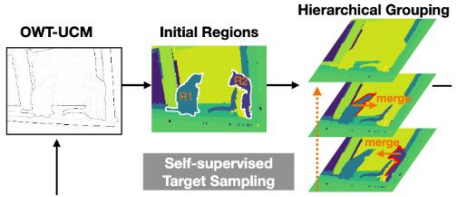


Xie et al. NeurIPS 2021



Bar et al. CVPR 2022

Heuristic segmentation methods

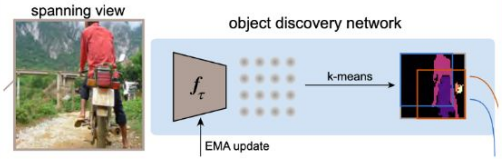


Zhang et al. NeurIPS 2020

Training Image and Heuristic Masks

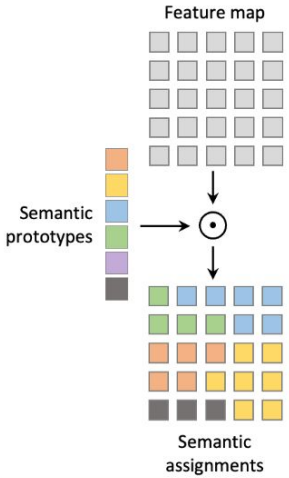
Henaff et al. ICCV 2021

Heuristic clustering methods

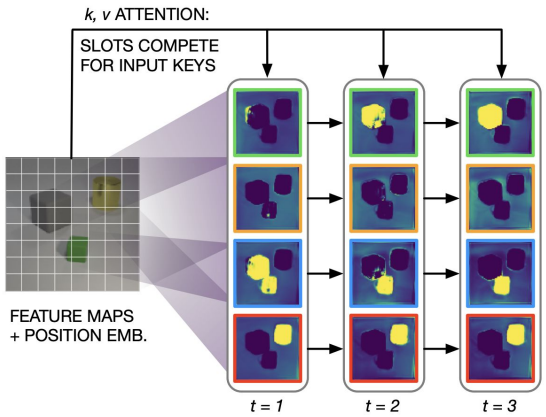


Henaff et al. ECCV 2022

Semantic Grouping (Ours)



On object discovery: the Slot Attention mechanism



(a) Slot Attention module.

*Slot attention introduces **competing** between queries*

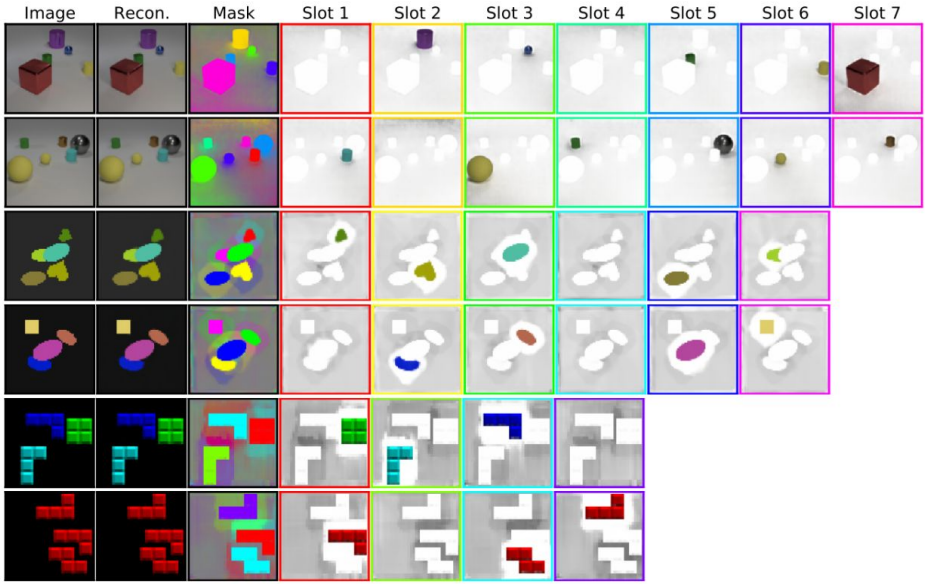
```
# simplified slot attention module
slots = slots_mu + exp(slots_log_sigma)*normal([N, num_slots, slot_size])
inputs = LayerNorm(inputs)
for t in range(T) # loop == multi-layer with weight-sharing
    attn = Softmax(dot(k(inputs), q(slots).T), axis='slots')
    slots = WeightedMean(weights=attn + 1e-6, values=v(inputs))
    slots += MLP(LayerNorm(slots))
```

```
# simplified transformer decoder layer
slots += LayerNorm(self_attn(slots, slots, slots)) # self-attention

attn = Softmax(dot(k(inputs), q(slots).T), axis='inputs') # cross-attention
slots = WeightedSum(weights=attn + 1e-6, values=v(inputs))
slots += LayerNorm(slots)

slots += LayerNorm(MLP(slots))
return slots
```

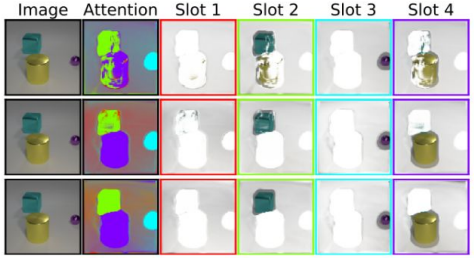
*Transformer decoders' queries are **independent** to each other*

Locatello, Francesco, et al., Object-Centric Learning with Slot Attention, NeurIPS 2020.

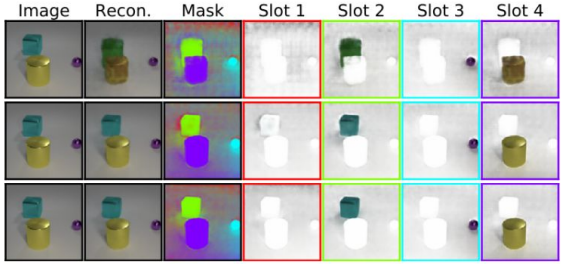
Slot Attention works well on toy data with reconstruction…



(a) Decomposition across datasets.



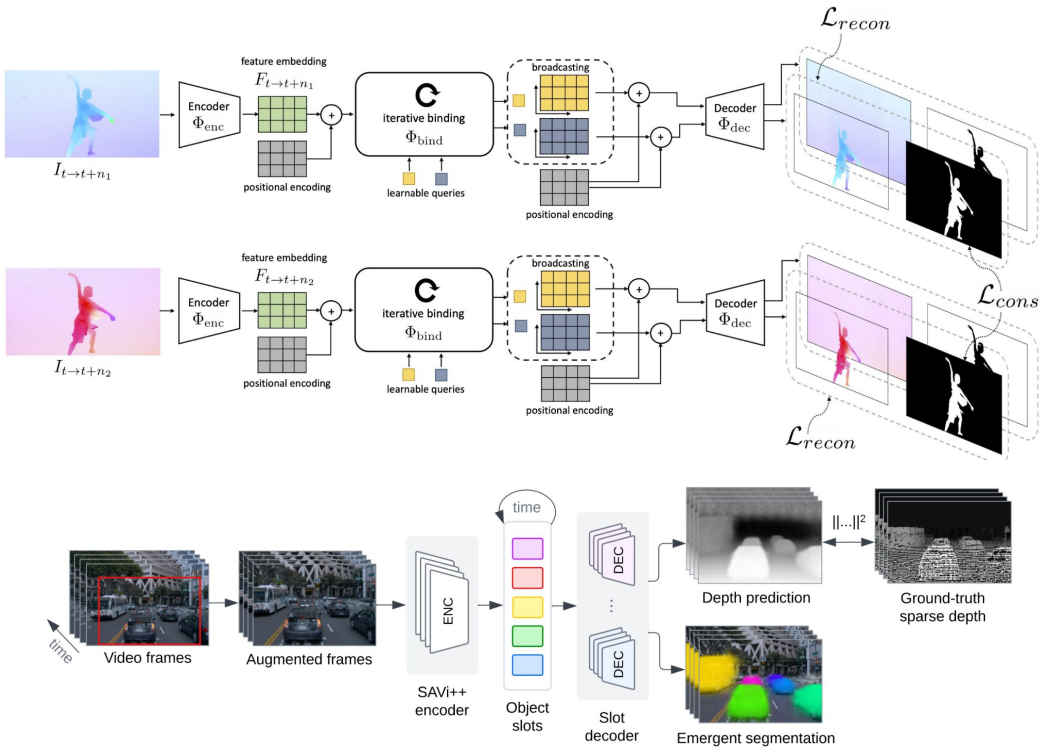
(b) Attention iterations.



(c) Reconstructions per iteration.

Locatello, Francesco, et al., Object-Centric Learning with Slot Attention, NeurIPS 2020.

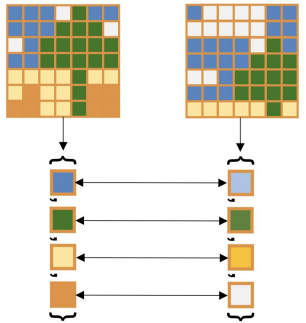
*And also on real-world data if **motion** or **depth** is available…*

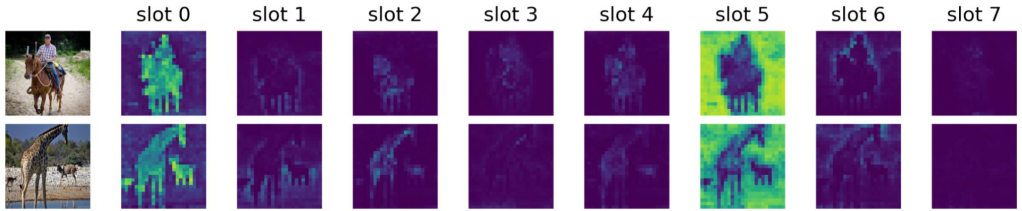


[1] Yang, Charig, et al., Self-Supervised Video Object Segmentation by Motion Grouping, ICCV 2021.
[2] Elsayed, Gamaleldin F., et al., SAVi++: Towards End-to-End Object-Centric Learning from Real-World Videos, NeurIPS 2022

And…, what about real-world images?

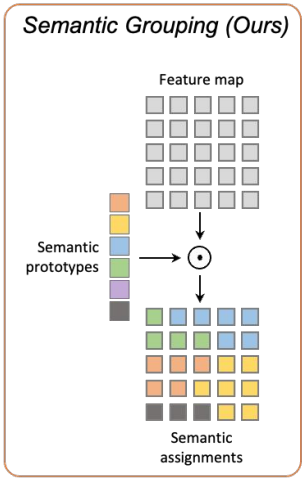
- No “cheating” cues like motion or depth
- Reconstruction-based paradigm no longer work
  - Though it worked well on toy data
  - The training cost is also not acceptable
- Contrastive learning faces ambiguity in the learning objective
  - The key is to define an objective that forces the emergence of objectness
  - Matching the slots procured from two views is hard
    - Since the representations are premature
  - The best we reached in this path is a good foreground-background discriminator (by the end of my undergrad thesis)





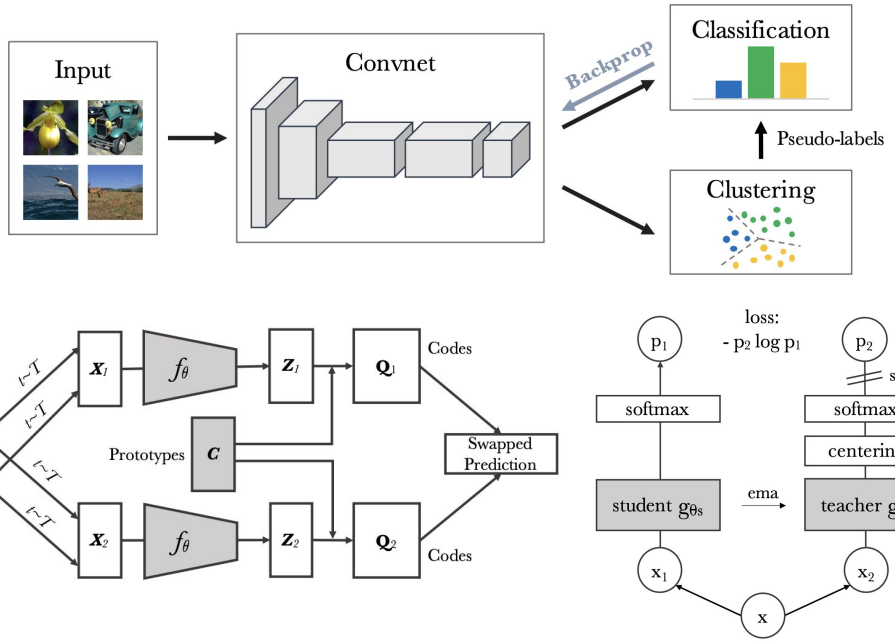
Philosophy shift: from bottom-up to top-down

- Bottom-up object discovery
  - Objects are directly induced from a ***single image***
  - The cues for objectness are mostly ***low-level***
    - E.g., motion, depth, reconstruction
  - Do not generalize to the real-world



- Top-down (semantic) object discovery
  - First learn semantic prototypes from the ***whole dataset***
    - Each prototype can represent a semantic class (e.g., cat, dog)
  - Then assign a nearest-neighbor prototype to each pixel (pseudo-labeling)
  - Pixels with the same pseudo-label forms a group (object)
  - The cues for objectness comes from ***high-level*** semantics

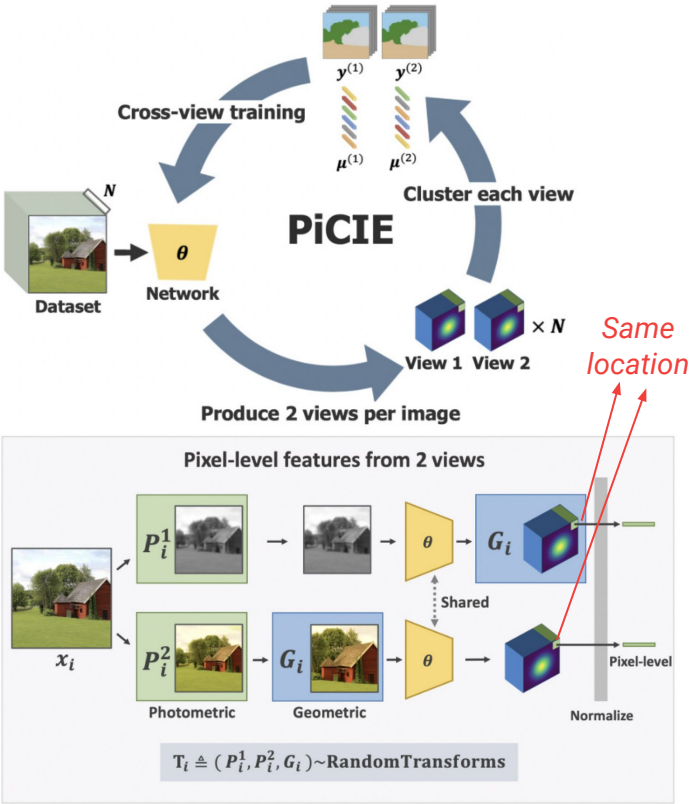
Towards semantic prototypes: deep clustering



- Key insight
  - Two augmented views of the same image should have the same cluster assignments (soft pseudo-labels)
- Key problem
  - Pseudo-labeling
    - K-means
    - Sinkhorn-Knopp
    - Self-distillation
  - Avoiding collapse
    - Stop-gradient
    - Centering & sharpening on the predictions

[1] Caron, Mathilde, et al, Deep Clustering for Unsupervised Learning of Visual Features, ECCV 2018.
[2] Caron, Mathilde, et al, Unsupervised Learning of Visual Features by Contrasting Cluster Assignments, NeurIPS 2020.
[3] Caron, Mathilde, et al. Emerging Properties in Self-Supervised Vision Transformers, ICCV 2021.

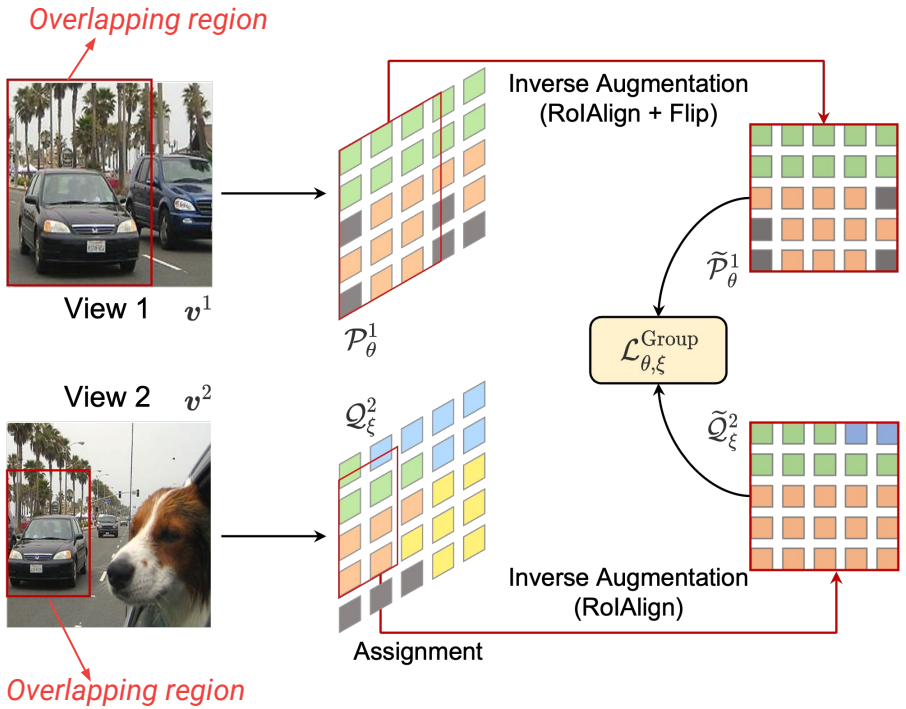
Deep clustering on pixels: unsupervised semantic seg.



Cho, Jang Hyun, et al., PiCIE: Unsupervised Semantic Segmentation Using Invariance and Equivariance in Clustering, CVPR 2021.

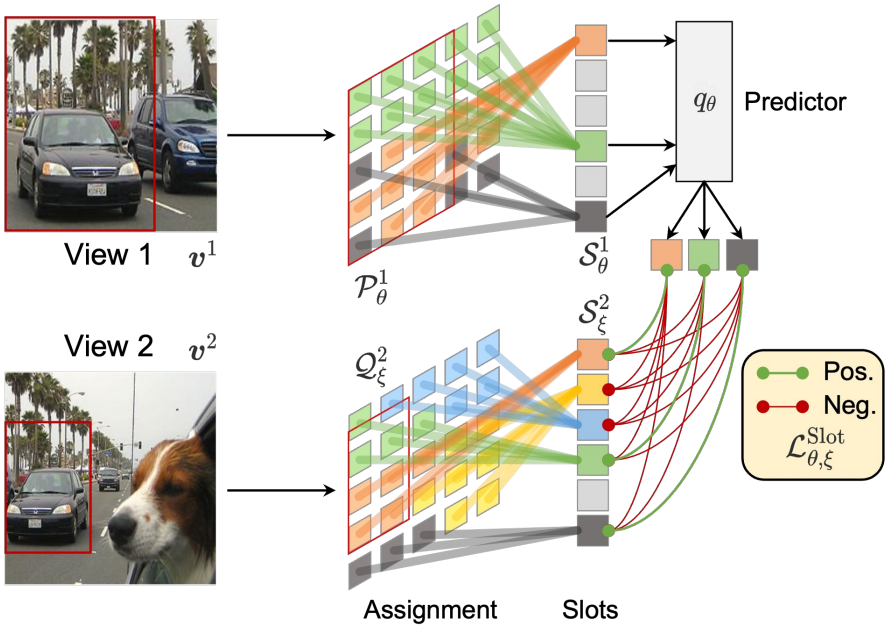
Semantic grouping with pixel-level deep clustering

- **Intuition**: A semantic meaningful grouping should be invariant to data augmentations
- **Consistency**: Pixels that lie in the same location should have similar assignment scores w.r.t. the same set of cluster centers (prototypes)
- **Avoid trivial solution**: The prototypes should be different from each other to ensure that the learned representations are discriminative

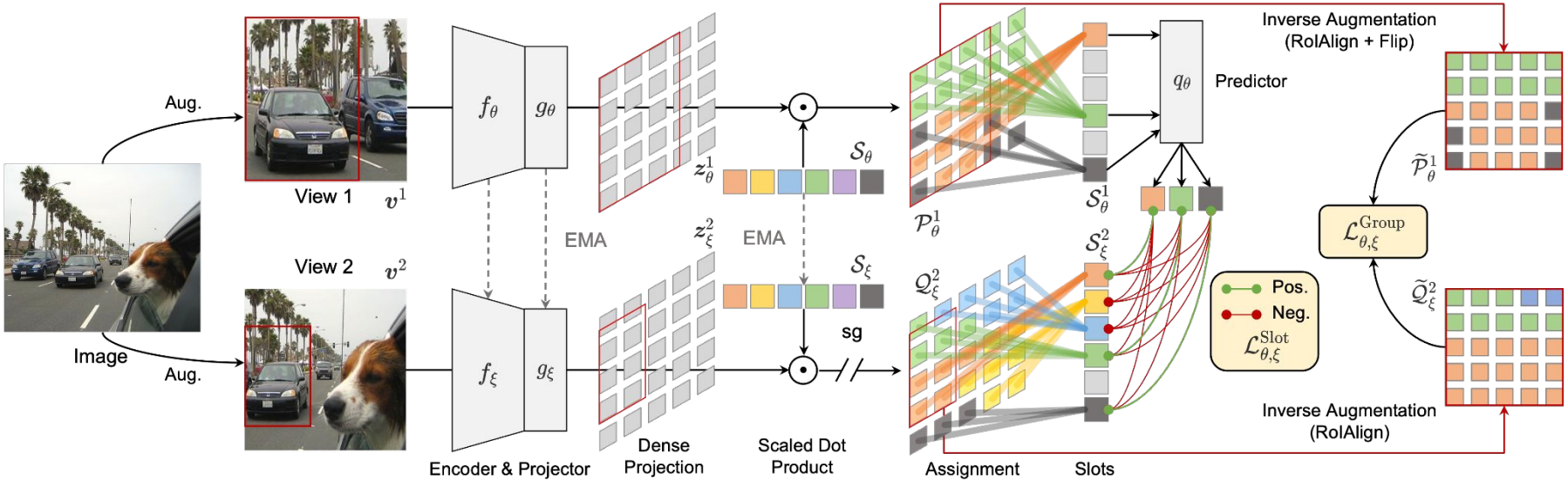


Group-level representation learning by contrasting slots

- MoCo v3-like contrastive learning objective applied to slots/groups
- **Pull force**: Maximize the similarity between different views of the same slot
- **Push force**: Minimize the similarity between slots from another view with different semantics and all slots from other images.



Together: solving segmentation & representation jointly



- Based on a shared pixel embedding function, the model learns to classify pixels into groups according to their feature similarity in a pixel-level deep clustering fashion.
- The model produces group-level feature vectors (slots) through attentive pooling over the feature maps, and further performs group-level contrastive learning.

Results: new SOTA in scene-centric pre-training

Table 2: **Main transfer results with COCO pre-training.** We report the results in COCO [48] object detection, COCO instance segmentation, and semantic segmentation in Cityscapes [13], PASCAL VOC [20] and ADE20K [86]. Compared with other image-, pixel-, and object-level self-supervised learning methods, our method shows consistent improvements over different tasks without leveraging multi-crop [6] and objectness priors. (†: re-impl. w/ official weights; ‡: full re-impl.)

Method	Epochs	Multi crop	Obj. Prior	COCO detection			COCO segmentation			Semantic seg. (mIoU)		
				AP^b	AP^b_{50}	AP^b_{75}	AP^m	AP^m_{50}	AP^m_{75}	City.	VOC	ADE
random init.	-	✗	✗	32.8	50.9	35.3	29.9	47.9	32.0	65.3	39.5	29.4
Image-level approaches												
MoCo v2‡ [9]	800	✗	✗	38.5	58.1	42.1	34.8	55.3	37.3	73.8	69.2	36.2
Revisit.† [63]	800	✓	✗	40.1	60.2	43.6	36.3	57.3	38.9	75.3	70.6	37.0
Pixel-level approaches												
Self-EMD [49]	800	✗	✗	39.3	60.1	42.8	-	-	-	-	-	-
DenseCL† [68]	800	✗	✗	39.6	59.3	43.3	35.7	56.5	38.4	75.8	71.6	37.1
PixPro‡ [75]	800	✗	✗	40.5	60.5	44.0	36.6	57.8	39.0	75.2	**72.0**	38.3
Object / Group-level approaches												
DetCon† [35]	1000	✗	✓	39.8	59.5	43.5	35.9	56.4	38.7	76.1	70.2	38.1
ORL† [74]	800	✓	✓	40.3	60.2	44.4	36.3	57.3	38.9	75.6	70.9	36.7
Ours (SlotCon)	800	✗	✗	**41.0**	**61.1**	**45.0**	**37.0**	**58.3**	**39.8**	**76.2**	71.6	**39.0**

Eqv. Performance to ImageNet with ⅕ data

Table 3: **Pushing the limit of scene-centric pre-training.** Our method further sees a notable gain in all tasks with extended COCO+ data, showing the great potential of scene-centric pre-training.

Method	Dataset	Epochs	COCO detection			COCO segmentation			Semantic seg. (mIoU)		
			AP^b	AP^b_{50}	AP^b_{75}	AP^m	AP^m_{50}	AP^m_{75}	City.	VOC	ADE
SlotCon	COCO	800	41.0	61.1	45.0	37.0	58.3	39.8	76.2	71.6	39.0
SlotCon	ImageNet	100	41.4	61.6	45.6	37.2	58.5	39.9	75.4	73.1	38.6
SlotCon	ImageNet	200	**41.8**	**62.2**	45.7	**37.8**	59.1	**40.7**	76.3	**75.0**	38.8
ORL [74]	COCO+	800	40.6	60.8	44.5	36.7	57.9	39.3	-	-	-
SlotCon	COCO+	800	**41.8**	**62.2**	**45.8**	**37.8**	**59.4**	40.6	**76.5**	73.9	**39.2**

Table 1: Details of the datasets used for pre-training.

Dateset	#Img.	#Obj./Img.	#Class
ImageNet-1K [15]	1.28M	1.7	1000
COCO [48]	118K	7.3	80
COCO+ [48]	241K	N/A	N/A

Also strong compatibility in object-centric pre-training

Table 4: **Main transfer results with ImageNet-1K pre-training.** Our method is also compatible with object-centric data and shows consistent improvements over different tasks without using FPN [47] and objectness priors. (†: re-impl. w/ official weights; ‡: full re-impl.)

Method	Epochs	w/ FPN	Obj. Prior	COCO detection			COCO segmentation			Semantic seg. (mIoU)		
				AP^b	AP^b_{50}	AP^b_{75}	AP^m	AP^m_{50}	AP^m_{75}	City.	VOC	ADE
random init.	-	✗	✗	32.8	50.9	35.3	29.9	47.9	32.0	65.3	39.5	29.4
supervised	100	✗	✗	39.7	59.5	43.3	35.9	56.6	38.6	74.6	74.4	37.9
Image-level approaches												
MoCo v2† [9]	800	✗	✗	40.4	60.1	44.2	36.5	57.2	39.2	76.2	73.7	36.9
DetCo† [73]	200	✗	✗	40.1	61.0	43.9	36.4	58.0	38.9	76.0	72.6	37.8
InsLoc† [76]	200	✓	✗	40.9	60.9	44.7	36.8	57.8	39.4	75.4	72.9	37.3
Pixel-level approaches												
DenseCL† [68]	200	✗	✗	40.3	59.9	44.3	36.4	57.0	39.2	76.2	72.8	38.1
PixPro† [75]	100	✗	✗	40.7	60.5	44.8	36.8	57.4	39.7	**76.8**	73.9	38.2
Object / Group-level approaches												
DetCon [35]	200	✗	✓	40.6	-	-	36.4	-	-	75.5	72.6	-
SoCo‡ [70]	100	✓	✓	41.6	61.9	45.6	37.4	58.8	40.2	76.5	71.9	37.8
Ours (SlotCon)	100	✗	✗	41.4	61.6	45.6	37.2	58.5	39.9	75.4	73.1	38.6
Ours (SlotCon)	200	✗	✗	**41.8**	**62.2**	**45.7**	**37.8**	**59.1**	**40.7**	76.3	**75.0**	**38.8**

Ablation study

Table 6: **Ablation studies with COCO 800 epochs pre-training.** We show the AP^b on COCO objection detection and mIoU on Cityscapes, PASCAL VOC, and ADE20K semantic segmentation. The default options are marked with a gray background.

(a) **Number of prototypes**

K	COCO	City	VOC	ADE
128	40.7	**76.4**	**71.9**	38.5
256	**41.0**	76.2	71.6	39.0
512	40.9	75.6	71.6	38.9
1024	40.7	75.8	70.9	**39.1**

(b) **Loss balancing**

λ_g	COCO	City	VOC	ADE
0.3	**41.0**	76.1	**72.1**	37.9
0.5	**41.0**	**76.2**	71.6	**39.0**
0.7	40.5	75.2	71.5	38.4
1.0	40.4	74.2	70.1	38.6

(c) **Teacher temperature**

τ_t	COCO	City	VOC	ADE
0.04	40.4	75.5	70.2	37.9
0.07	**41.0**	**76.2**	**71.6**	**39.0**

Understanding the semantic grouping ability: unsupervised semantic segmentation

Table 5: Main results in COCO-Stuff unsupervised semantic segmentation.

Method	mIoU	pAcc
MaskContrast [64]	8.86	23.03
PiCIE + H. [40]	14.36	49.99
SegDiscover [38]	14.34	**56.53**
Ours (SlotCon)	**18.26**	42.36

Understanding the semantic grouping ability: nearest neighbor visualization

Figure 3: **Examples of visual concepts discovered by SlotCon from the COCO `val2017` split.** Each column shows the top 5 segments retrieved with the same prototype, marked with reddish masks or arrows. Our method can discover visual concepts across various scenarios and semantic granularities regardless of small object size and occlusion. (*best viewed in color*)

On the emergence of objectness…

- Why the prototypes bind to meaningful concepts?
- We adopts three priors:
  - Prior 1: geometric-covariance and photometric-invariance
  - Prior 2: small prototype number
  - Prior 3: meaningful grouping
    - i.e., avoiding collapse
- Given these constraints, optimize the feature space and the prototypes
- Semantic proto. is the only solution

- Concerning granularity
  - Depend on the *prototype number* and *dataset distribution*
- It can bias to occupying categories
  - e.g., the model also discovers human parts and human-related activities
  - While for other animals, one prototype for one general species is enough

Additional ablation studies

Table 8: **Ablation studies with COCO 800 epochs pre-training.** We show the AP^b on COCO objection detection and mIoU on Cityscapes, PASCAL VOC, and ADE20K semantic segmentation. The default options are marked with a gray background.

(a) **Batch size**

B	COCO	City	VOC	ADE
256	40.6	75.9	70.9	38.1
512	**41.0**	**76.2**	71.6	**39.0**
1024	40.7	75.7	**71.8**	38.6

(b) **Type of group-level loss**

Loss	COCO	City	VOC	ADE
Reg.	40.7	75.9	71.0	**39.0**
Ctr.	**41.0**	**76.2**	**71.6**	**39.0**

(c) **Where to apply** invaug**?**

Align	COCO	City	VOC	ADE
Proj.	40.9	75.7	71.4	38.0
Asgn.	**41.0**	**76.2**	**71.6**	**39.0**

(d) **Batch size and image-level objective**

B	$\mathcal{L}_{\text{Image}}$	COCO AP^b	COCO AP^m
512	✗	41.0	**37.0**
512	✓	40.8	36.8
1024	✗	40.7	36.7
1024	✓	**41.1**	**37.0**

(e) **Geometric augmentations**

Method	Geometric aug.	VOC mIoU
Random init.	-	39.5
SlotCon	✓	**71.6**
SlotCon	✗	62.6

Additional visualization results: human-related

Pre-training with autonomous driving data?

Table 7: Transfer learning results with BDD100K pre-training.

Pre-train Data	Method	Cityscapes mIoU
-	Random init.	65.3
COCO	MoCo v2	73.8
COCO	SlotCon	76.2
BDD100K	SlotCon	73.9

Table 1: Overview of the training datasets. We sample a uniform and long-tailed (LT) subset of 118K images from ImageNet. On OpenImages, we sample a random subset of 118K images. The complete train splits are used for COCO and BDD100K. The figure shows some examples.

Pretrain Data	#Imgs	#Obj/Img	Uniform	Discriminative
ImageNet-118K [12]	118 K	1.7	✓	✓
ImageNet-118K-LT [12]	118 K	1.7	✗	✓
COCO [31]	118 K	7.3	✗	✓
OpenImages-118K [28]	118 K	8.4	✗	✓
BDD100K [52]	90 K	-	✗	✗

- Setting
  - Train on BDD100K, then transfer to Cityscapes
- Results
  - Not as good as COCO-pre-train
  - Yet still beats MoCo v2
  - BDD100K is long-tailed, and the images are also less discriminative

Object-centric pre-training on such data is still challenging for us

Statistics about the slots

- How many slots are active on average for each image?
  - Roughly, 7 slots are active on average for one image after convergence.
- How often is one slot active over the whole dataset?
  - Top 5: tree (376), sky (337), streetside car (327), building exterior wall (313), and indoor wall (307)
  - Bottom 5: skateboarder (44), grassland (45), train (56), luggage (57), and airplane (57)

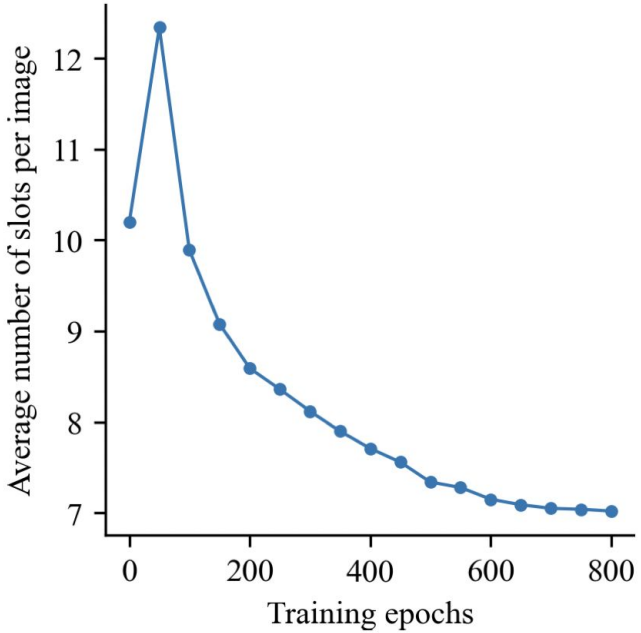


Figure 4: Average number of active slots per image during training on COCO.

Summary

1. We show that the decomposition of natural scenes (semantic grouping) can be done in a learnable fashion and jointly optimized with the representations from scratch.
2. We demonstrate that semantic grouping can bring object-centric representation learning to large-scale real-world scenarios.
3. Combining semantic grouping and representation learning, we unleash the potential of scene-centric pre-training, largely close its gap with object-centric pre-training and achieve state-of-the-art results in various downstream tasks.

NEURAL INFORMATION PROCESSING SYSTEMS

MEGVII 旷视

Thanks!

Code

Twitter Thread

Project Page