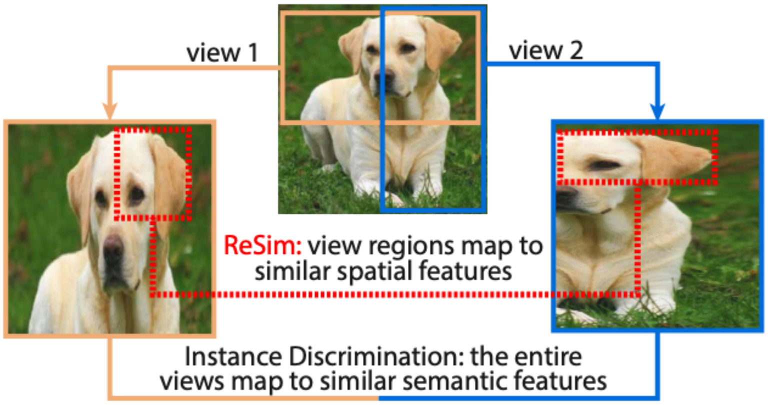
Region Similarity Representation Learning

Background

The progress we see today in the self-supervised pre-training, much of it comes from exploiting the instance discrimination task, where we try to maximize the similarity between two views which are augmented versions of the same original image. Which means, we are trying to map various scales and crops of the given image to a same feature representation.

For example, if there's an image of a dog that was augmented as view 1 and view 2 as shown below:


view 1
view 2
ReSim: view regions map to similar spatial features
Instance Discrimination: the entire views map to similar semantic features

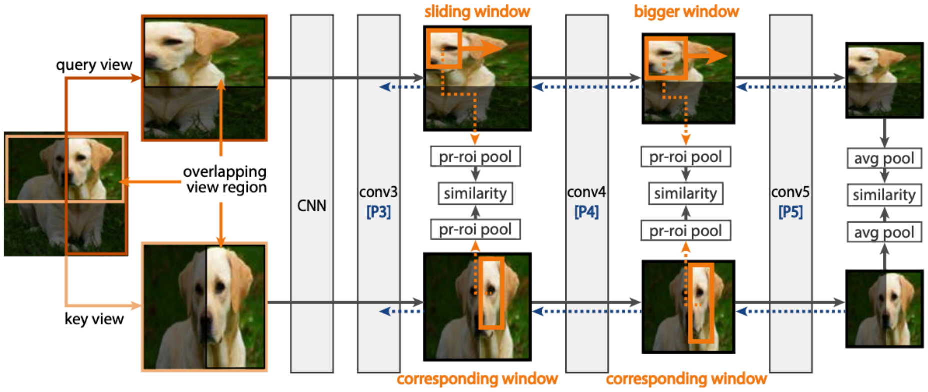
In normal setting, would want to maximize the similarity between these two views, which mean we are mapping both the views to the same feature representation and the instance discrimination model doesn't enforce any spatial consistency in the convolution features.

Ideally, the encoder should be such that, it should map the same image components across the views to the same point in the embedding space. This is necessary for localization sensitive tasks like object detection, segmentation etc.

Key Ideas

Region Similarity Representation Learning (ReSim):

ReSim uses a sliding window across the overlapping regions between the views, then maps these regions to their associated regions in the convolutional layers throughout the network. We then maximize the similarity of these convolution features.



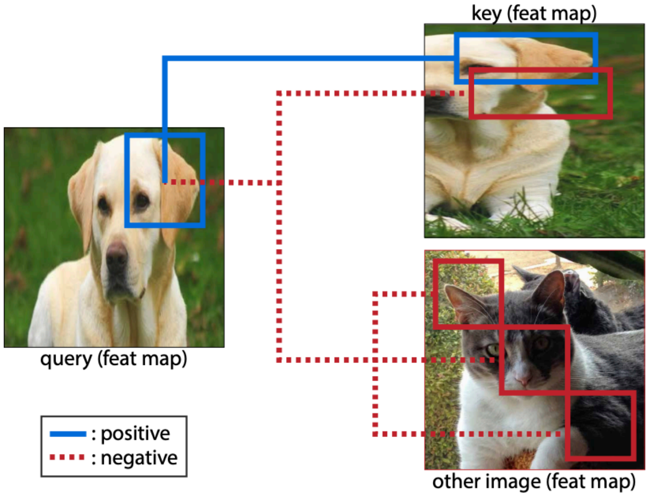
Framework details:

For a given image, augmentations are applied and we get two different views(Query and Key). Now we identify the overlapping regions, and then slide a fixed size window across this overlapping region and generate candidate regions. We use these candidate regions and scale them so to map them to their associated regions in the convolution features throughout the network.

- Aligning the regions between the views :
    - From the widely used augmentations in self-supervised pre-training, only horizontal flipping and Random Resized Crop are the ones that

affect the image co-ordinates.

  - In ReSim pipeline, the horizontal flip is controlled such that either both views are flipped or both views are not flipped.
  - For Random Resized Crop transform denoted by R, to get the corresponding region of query view in the key view, given (t_q, l_q, b_q, r_q) coordinates of the region in the query view, you do $R_K(R_q^-1)$ to convert the (t_q, l_q, b_q, r_q) to the key view region denoted by (t_k, l_k, b_k, r_k).
- After generating these corresponding candidate regions in both of the views, a ROI Pooling (PreciseROI Pooling) is used to extract these regions from the feature maps. Post extraction of these regions, we maximize the similarity of the aligned regions using objective function like contrastive loss.
- **Objective function**:
  - Positive vs Negative samples for region similarity calculation.



  - Similarity function:

    Here, $f(I_q, u_q^i)$ is a PreciseROI Pooling function that takes in a augmented image I_q, u_q is a region defined by (t, l, b, r).

$$E_{i,j}^{\{k_+,k_-\}} = \exp\left(f(\mathcal{I}_q, u_q^i) \cdot f(\mathcal{I}_{\{k_+,k_-\}}, u^j_{\{k_+,k_-\}})/\tau\right), \tag{1}$$

- Region similarity Loss is given by :

$$\mathcal{L}_q^{\text{rs}} = \frac{1}{n}\sum_{i=1}^{n} \frac{E_{i,i}^{k_+}}{E_{i,i}^{k_+} + \sum_{j \neq i} E_{i,j}^{k+} + \sum_{k_-,j} E_{i,j}^{k_-}},$$

- Overall objective :

$$\mathcal{L}_q = \mathcal{L}_q^{\text{rs}} + \lambda \mathcal{L}_q^{\text{is}}.$$

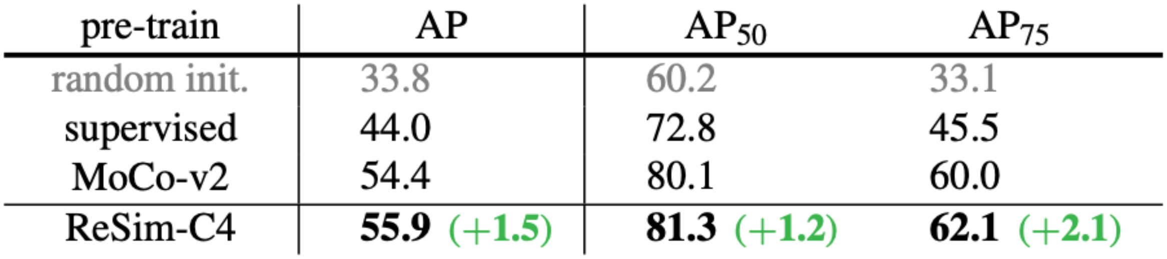
Impressions

During experimentation, IN-1K and its subset IN-100 datasets were used for pre-training. In the ReSim models we use Resnet-50-$C4$ network and Region Similarity is applied on the the convolution features of C_4 block .

Studying the transfer capability of ReSim network in comparison to MoCo-v2 on object detection task. PASCAL VOC dataset is used.

IN-100 Pre-training experiments:

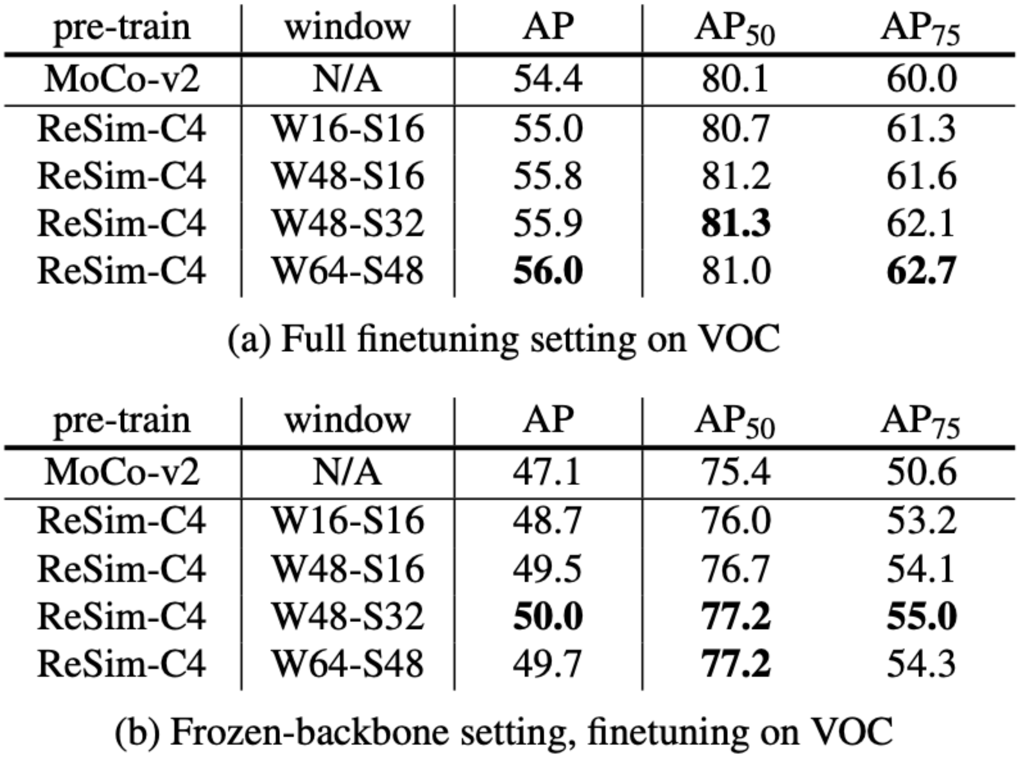
object detection task. **Dataset** : PASCAL VOC



pre-train	AP	AP_{50}	AP_{75}
random init.	33.8	60.2	33.1
supervised	44.0	72.8	45.5
MoCo-v2	54.4	80.1	60.0
ReSim-C4	**55.9** (+1.5)	**81.3** (+1.2)	**62.1** (+2.1)

Frozen backbone for finetuning on VOC:

The backbone is frozen and only the RPN and object classification head is finetuned on the PASCAL VOC



pre-train	window	AP	AP_{50}	AP_{75}
MoCo-v2	N/A	54.4	80.1	60.0
ReSim-C4	W16-S16	55.0	80.7	61.3
ReSim-C4	W48-S16	55.8	81.2	61.6
ReSim-C4	W48-S32	55.9	**81.3**	62.1
ReSim-C4	W64-S48	**56.0**	81.0	**62.7**

(a) Full finetuning setting on VOC

pre-train	window	AP	AP_{50}	AP_{75}
MoCo-v2	N/A	47.1	75.4	50.6
ReSim-C4	W16-S16	48.7	76.0	53.2
ReSim-C4	W48-S16	49.5	76.7	54.1
ReSim-C4	W48-S32	**50.0**	**77.2**	**55.0**
ReSim-C4	W64-S48	49.7	**77.2**	54.3

(b) Frozen-backbone setting, finetuning on VOC

We observe that in the frozen backbone setting, the ReSim model performs better than the MoCo-v2 by a significant margin indicating that
ReSim yields features of higher quality for localization dependent tasks.

Another thing to note is they experiment with different window sizes and strides for generating the candidate region pairs from the overlapping region between the query and key views. From experiments, they reach a optimal window size of W48 and stride of S32 as the best setting.

Please note that these window and stride sizes are at image level, and when applying on the C4 feature maps, they are mapped to the feature map scale (1/16).

IN-1k Pre-training experiments:

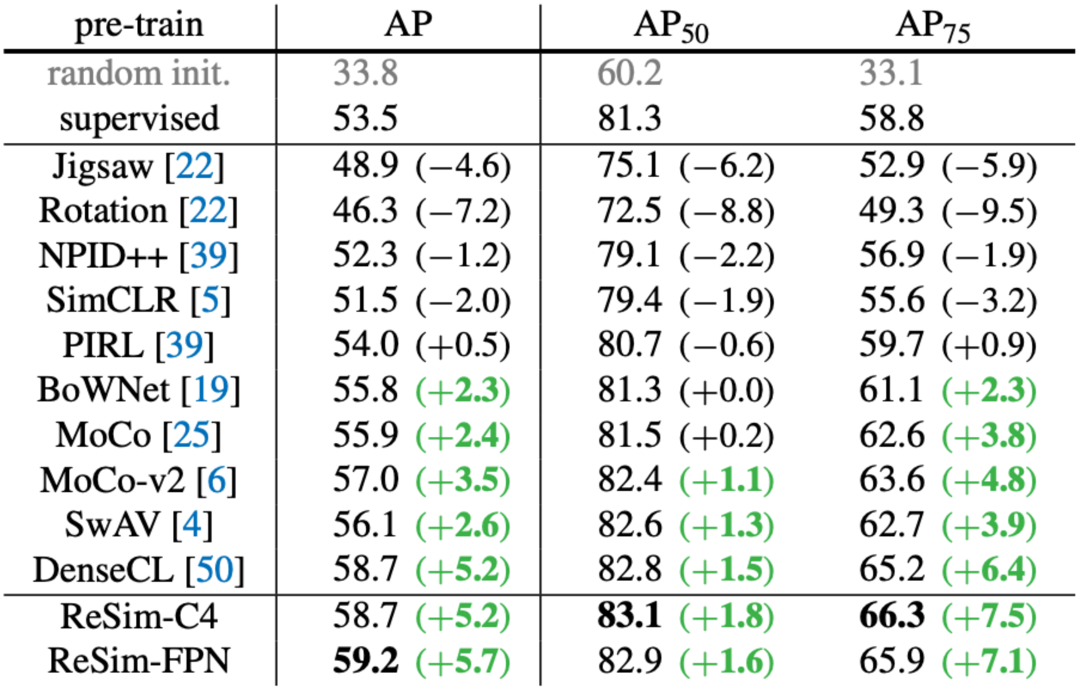
For these experiments, 3 models were used:

ReSim-$C4$: The region level similarity is applied at the C4 feature map of the Resnet

ReSim-FPN : The region level similarity is applied at P3 and P4 levels, but the transferred weights for the downstream tasks are just from the Resnet-50.

ReSim-FPN^T: The region level similarity is applied at P3 and P4 levels, and the downstream task is eligible to use the pre-trained weights from the whole FPN network, so the entire network is transferred.

Object detection task. Dataset : PASCAL VOC



pre-train	AP	AP_{50}	AP_{75}
random init.	33.8	60.2	33.1
supervised	53.5	81.3	58.8
Jigsaw [22]	48.9 (−4.6)	75.1 (−6.2)	52.9 (−5.9)
Rotation [22]	46.3 (−7.2)	72.5 (−8.8)	49.3 (−9.5)
NPID++ [39]	52.3 (−1.2)	79.1 (−2.2)	56.9 (−1.9)
SimCLR [5]	51.5 (−2.0)	79.4 (−1.9)	55.6 (−3.2)
PIRL [39]	54.0 (+0.5)	80.7 (−0.6)	59.7 (+0.9)
BoWNet [19]	55.8 (+2.3)	81.3 (+0.0)	61.1 (+2.3)
MoCo [25]	55.9 (+2.4)	81.5 (+0.2)	62.6 (+3.8)
MoCo-v2 [6]	57.0 (+3.5)	82.4 (+1.1)	63.6 (+4.8)
SwAV [4]	56.1 (+2.6)	82.6 (+1.3)	62.7 (+3.9)
DenseCL [50]	58.7 (+5.2)	82.8 (+1.5)	65.2 (+6.4)
ReSim-C4	58.7 (+5.2)	**83.1** (+1.8)	**66.3** (+7.5)
ReSim-FPN	**59.2** (+5.7)	82.9 (+1.6)	65.9 (+7.1)

On Pascal VOC, the ReSim-C4 and ReSIM-FPN beats all the pre-existing methods

COCO Object Detection and Instance Segmentation:

pretrain	AP^{bb}	AP$^{bb}_{50}$	AP$^{bb}_{75}$	AP^{mk}	AP$^{mk}_{50}$	AP$^{mk}_{75}$
random init	31.0	49.5	33.2	28.5	46.8	30.4
supervised	38.9	59.6	42.7	35.4	56.5	38.1
MoCo-v2	38.9 (+0.0)	59.2 (−0.4)	42.4 (−0.3)	35.4 (+0.0)	56.2 (−0.3)	37.8 (−0.3)
VADeR [41]	39.2 (+0.3)	59.7 (+0.1)	42.7 (+0.0)	35.6 (+0.2)	56.7 (+0.2)	38.2 (+0.1)
DenseCL [50]†	39.4 (+0.5)	59.9 (+0.3)	42.7 (+0.0)	35.6 (+0.2)	56.7 (+0.2)	38.2 (+0.1)
ReSim-C4	39.3 (+0.4)	59.7 (+0.1)	43.1 (+0.4)	35.7 (+0.3)	56.7 (+0.2)	38.1 (+0.0)
ReSim-FPN	39.5 (+0.6)	59.9 (+0.3)	43.3 (+0.6)	35.8 (+0.4)	57.0 (+0.5)	38.4 (+0.3)
ReSim-FPN^T	**39.8** (+0.9)	**60.2** (+0.6)	**43.5** (+0.8)	**36.0** (+0.6)	**57.1** (+0.6)	**38.6** (+0.5)
ReSim-FPN^T (400 ep)	**40.3** (+1.4)	**60.6** (+1.0)	**44.2** (+1.5)	**36.4** (+1.0)	**57.5** (+1.0)	**38.9** (+0.8)

(a) Mask R-CNN R50-FPN, 1× schedule

pretrain	AP^{bb}	AP$^{bb}_{50}$	AP$^{bb}_{75}$	AP^{mk}	AP$^{mk}_{50}$	AP$^{mk}_{75}$
random init	36.7	56.7	40.0	33.7	53.8	35.9
supervised	40.6	61.3	44.4	36.8	58.1	39.5
MoCo-v2	40.9 (+0.3)	61.5 (+0.2)	44.6 (+0.2)	37.0 (+0.2)	58.4 (+0.3)	39.6 (+0.1)
VADeR [41]	-	-	-	-	-	-
DenseCL [50]†	41.2 (+0.6)	61.9 (+0.6)	45.1 (+0.7)	37.3 (+0.5)	58.9 (+0.8)	40.1 (+0.6)
ReSim-C4	41.1 (+0.5)	61.5 (+0.2)	44.8 (+0.4)	37.1 (+0.3)	58.6 (+0.5)	39.8 (+0.3)
ReSim-FPN	**41.4** (+0.8)	61.8 (+0.5)	45.4 (+1.0)	**37.5** (+0.7)	**59.1** (+1.0)	**40.4** (+0.9)
ReSim-FPN^T	**41.4** (+0.8)	**61.9** (+0.6)	**45.4** (+1.0)	**37.5** (+0.7)	**59.1** (+1.0)	40.3 (+0.8)
ReSim-FPN^T (400 ep)	**41.9** (+1.3)	**62.4** (+1.1)	**45.9** (+1.5)	**37.9** (+1.1)	**59.4** (+1.3)	**40.6** (+1.1)

(b) Mask R-CNN R50-FPN, 2× schedule