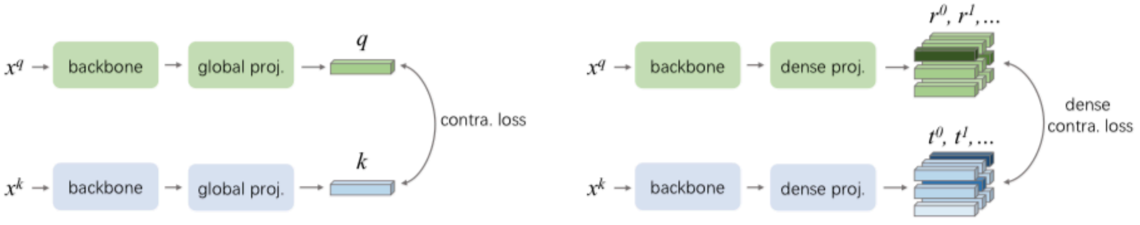
Dense Contrastive Learning for Self-Supervised Visual Pre-Training

This paper talks about the self-supervised pre-training task, and mentions that the progress we see today in the self-supervised pre-training, much of it comes from exploiting the instance discrimination task, where they try to maximize the similarity between two views which are augmented versions of the same original image. The instance discrimination model doesn’t explicitly enforce any spatial consistency in the convolution features. Thus it might not be the best suited for downstream dense prediction tasks such as semantic segmentation or object detection. This paper takes a shot at self-supervised pre-training approach that preserves the spatial features. The approach discussed in this paper has slightly inferior performance compared to Region Similarity Representative Learning I discussed previously.

Key Ideas

The key idea of this paper is, in the projection head, you have two subheads. One which does global average pooling of the features and also introduce a parallel sub head for dense features projection . The dense projection head is different from a normal projection head as it doesn’t have the max pool operation and you preserve the spatial information. It takes features from the backbone and applied 1×1 convolutions on it.



(a) Global Contrastive Learning

(b) Dense Contrastive Learning

DenseCL Pipeline:

- Given a differently augmented views pair of an image, you generate the dense feature maps from the backbone.
- Next, send these to features to the two subheads. The first subhead generated a feature vector after applying the global pooling on the dense feature vectors from the backbone. The second head will just apply some 1×1 convolutions on the dense features and preserve their spatial dimension.
- The feature vector from the first subhead is directly used for instance discrimination task
- The output feature vector of the second head (shape : SxSxD) is used for deep contrastive learning.

Dense contrastive learning:

- For each feature vector r (query) at a spatial position in SxS (where SxS is the spatial shape of the dense features) there are set of encoded keys $t_0, t_1..$ from the other view. The positive key for the query is matched from the set of keys(again the key is one of the feature vector from the SxS feature vectors in the other view). The negative keys are the global average pooled feature vectors from the different images.

$$\mathcal{L}_r = \frac{1}{S^2}\sum_s -\log \frac{\exp(r^s \cdot t^s_+/\tau)}{\exp(r^s \cdot t^s_+) + \sum_{t^s_-} \exp(r^s \cdot t^s_-/\tau)},$$

▼ Total loss

$$\mathcal{L} = (1 - \lambda)\mathcal{L}_q + \lambda\mathcal{L}_r,$$

- Dense correspondence across views (Matching the positive key to the query)
  - For each of the views, the backbone network extracts feature maps $F_1 \in R^{H*W*K}$ and $F_2 \in R^{H*W*K}$. These are sent into the dense projection head that generates dense feature vectors $\theta_1 \in R^{S_h*S_w*E}$ and $\theta_2 \in R^{S_h*S_w*E}$. For convenience, if we assume $S_w = S_h = S$. To match these dense feature vectors from the projection head, they take the feature maps from the backbone (downsampled to match the resolution of the dense feature vectors) and compute cosine similarity between the F_1 and F_2 , and the best matching pairs can be indexed to the dense feature vectors (θ_1, θ_2)

Impressions

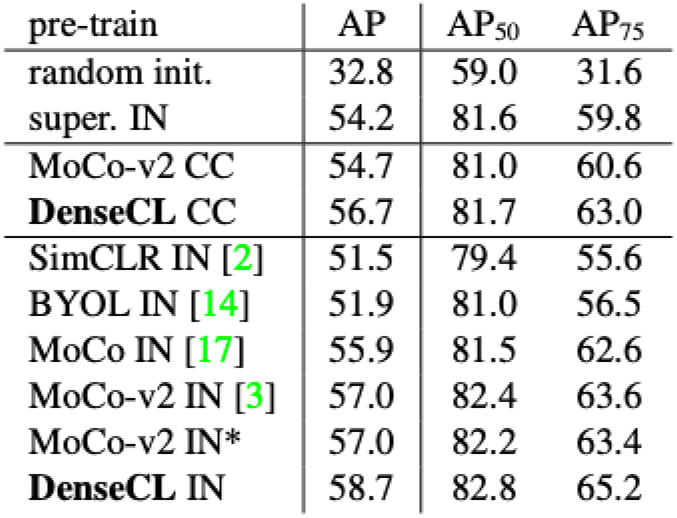
Experiments:

They use MoCo-v2 as their baseline method. The global projection head(instance discrimination task) and the dense projection head have fixed dimensional output (128-D and S^2 128-D respectively)

Datasets : MS-COCO and ImageNet for pre-training.

Evaluations : Fine-tuning on the target task end-to-end

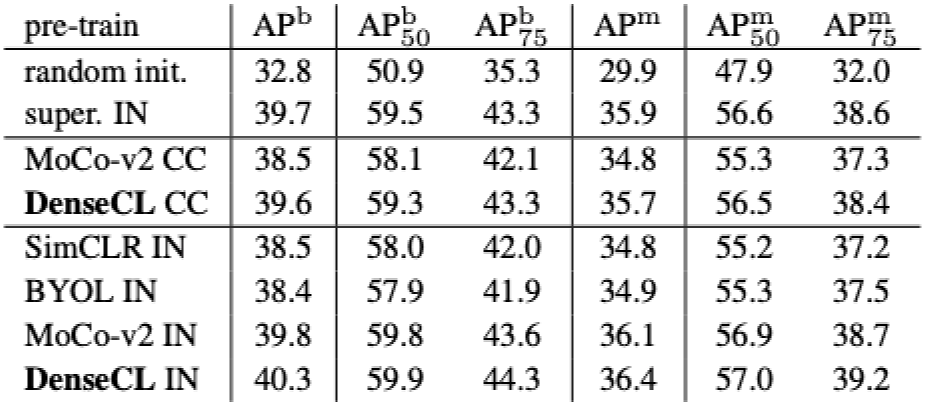
▼ PASCAL VOC object detection:



pre-train	AP	AP_{50}	AP_{75}
random init.	32.8	59.0	31.6
super. IN	54.2	81.6	59.8
MoCo-v2 CC	54.7	81.0	60.6
DenseCL CC	56.7	81.7	63.0
SimCLR IN [2]	51.5	79.4	55.6
BYOL IN [14]	51.9	81.0	56.5
MoCo IN [17]	55.9	81.5	62.6
MoCo-v2 IN [3]	57.0	82.4	63.6
MoCo-v2 IN*	57.0	82.2	63.4
DenseCL IN	58.7	82.8	65.2

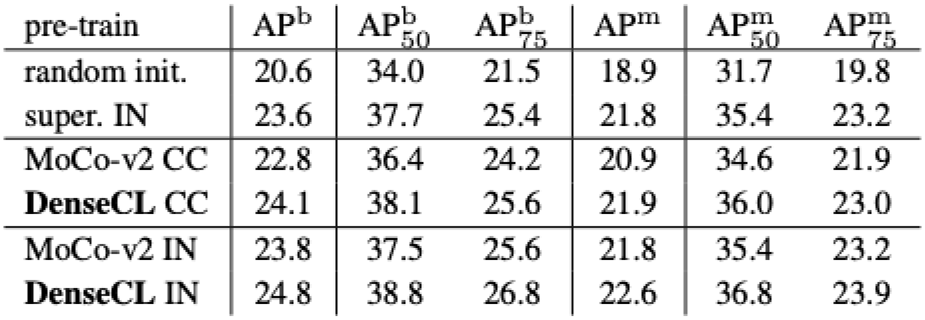
The dense CL when pre-trained on ImageNet (IN) as well as ms-coco (CC) outperform others on the downstream pascal object detection task

▼ COCO object detection and segmentation.



pre-train	AP^{b}	AP^{b}_{50}	AP^{b}_{75}	AP^{m}	AP^{m}_{50}	AP^{m}_{75}
random init.	32.8	50.9	35.3	29.9	47.9	32.0
super. IN	39.7	59.5	43.3	35.9	56.6	38.6
MoCo-v2 CC	38.5	58.1	42.1	34.8	55.3	37.3
DenseCL CC	39.6	59.3	43.3	35.7	56.5	38.4
SimCLR IN	38.5	58.0	42.0	34.8	55.2	37.2
BYOL IN	38.4	57.9	41.9	34.9	55.3	37.5
MoCo-v2 IN	39.8	59.8	43.6	36.1	56.9	38.7
DenseCL IN	40.3	59.9	44.3	36.4	57.0	39.2

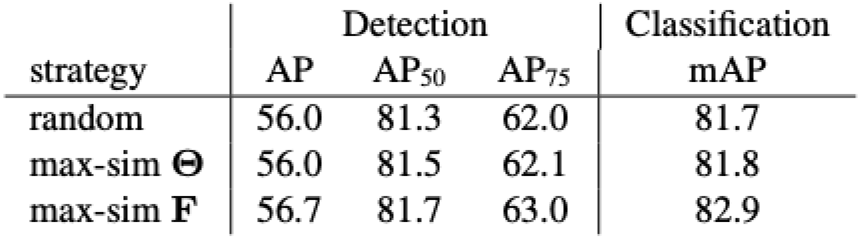
▼ PASCAL VOC semantic segmentation



pre-train	AP^b	AP^b_{50}	AP^b_{75}	AP^m	AP^m_{50}	AP^m_{75}
random init.	20.6	34.0	21.5	18.9	31.7	19.8
super. IN	23.6	37.7	25.4	21.8	35.4	23.2
MoCo-v2 CC	22.8	36.4	24.2	20.9	34.6	21.9
DenseCL CC	24.1	38.1	25.6	21.9	36.0	23.0
MoCo-v2 IN	23.8	37.5	25.6	21.8	35.4	23.2
DenseCL IN	24.8	38.8	26.8	22.6	36.8	23.9

Ablation studies:

Dense Feature Matching strategy:



	Detection			Classification
strategy	AP	AP_{50}	AP_{75}	mAP
random	56.0	81.3	62.0	81.7
max-sim $\boldsymbol{\Theta}$	56.0	81.5	62.1	81.8
max-sim $\mathbf{F}$	56.7	81.7	63.0	82.9

Dense Correspondance visualization:

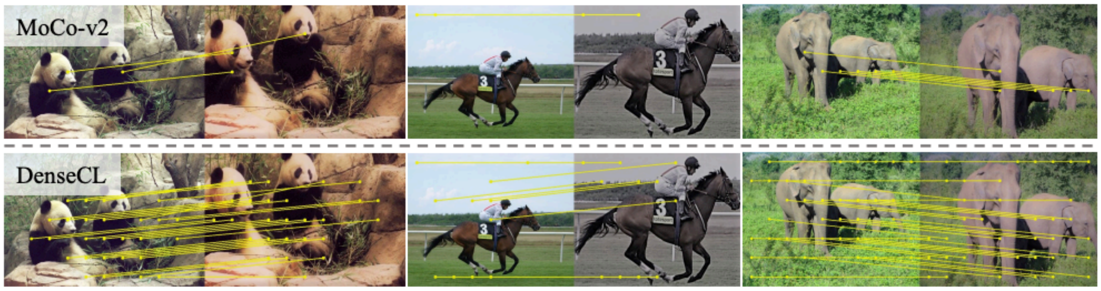


Figure 4 – Visualization of dense correspondence. The correspondence is extracted between two views of the same image, using the 200-epoch ImageNet pre-trained model. DenseCL extracts more high-similarity matches compared with MoCo-v2. Best viewed on screen.