DistinctiveNet: Self-supervised Objectness Losses for Detection

Xu Ji, João Henriques, Andrea Vedaldi (Research group: VGG)

State-of-the-art object detection uses convolutional neural networks (CNNs) trained on large, manually labelled datasets. Labelling data is expensive.

How can we learn an object detector without labels?

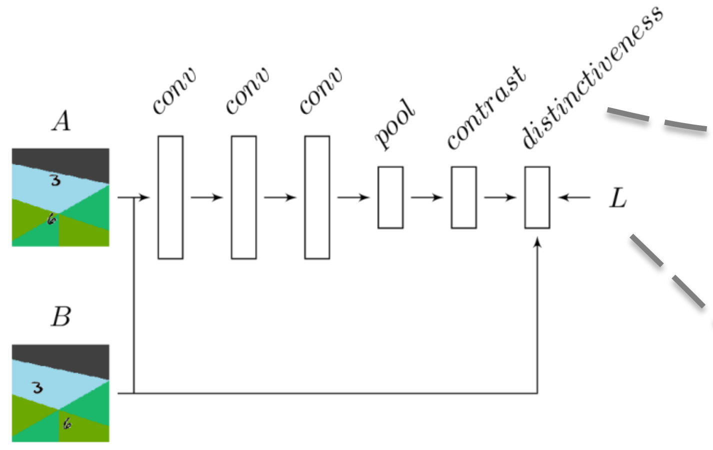

Fig. 1: Architecture. Learning to detect objects in A using self-supervised loss L and auxiliary image B (containing objects of A in different positions). Batchnorm, ReLUs, softmax not shown.

1. Use peakiness (non-entropy) as a learning signal

Call the CNN output $x \in \mathcal{R}^{h' \times w' \times c}$, a heatmap with c channels. The loss encourages spatial and channel peakiness (equivalent notions: minimizing entropy, maximizing sparsity, maximizing one-hotness). Let *i* and *k* be spatial and channel coordinates. We try 12 different variants, such as cross-entropy:

$$L = -\sum_{i,k} \log(x_{i,k}) \cdot x_{i,k}$$

and one-hot cross-entropy, which assumes one heatmap pixel per object:

$$L = \sum_{i} \left[-\max_{k'} x_{i,k'} + \log\left(\sum_{k'} e^{x_{i,k'}} \right) \right] + \sum_{k} \left[-\max_{i'} x_{i',k} + \log\left(\sum_{i'} e^{x_{i',k}} \right) \right]$$

2. Restrict detections to *distinctive* areas of the image

Objects are distinctive (different from surroundings). For any two images containing the same object, sliding one image over the other and taking a similarity metric should result in a peak at the translation where one object is directly over the other.

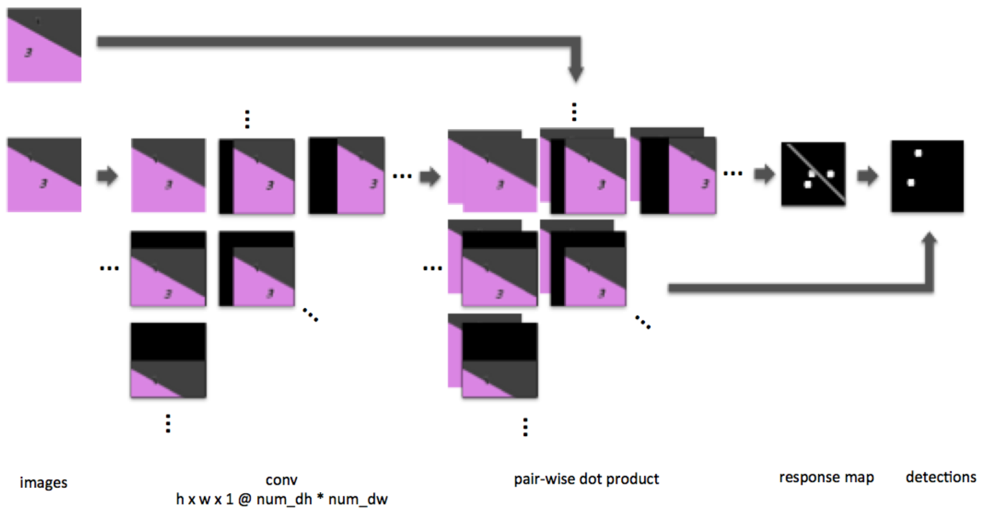

Fig. 2: Distinctiveness filter. We produce a dense set of translations of the auxiliary image using convolution and measure the similarity between each translated auxiliary image with the main image as a score for the corresponding translation (response map). Choosing the top pixels for these translations yields the pixels belonging to objects.

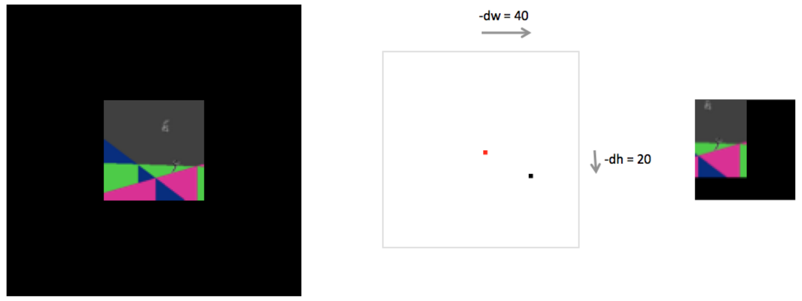

Fig. 3: Translating an image by convolution (i.e. natively in a CNN).

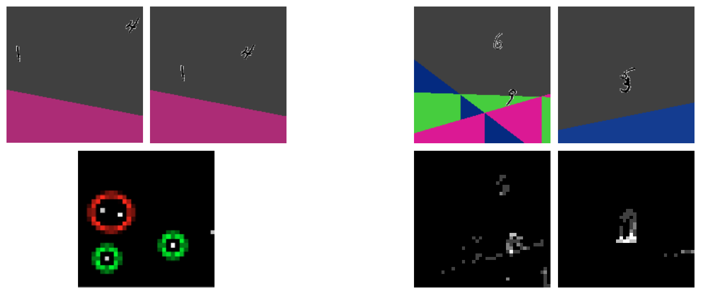

Fig. 4: Example response map (left) and distinctiveness detections (right).

Results

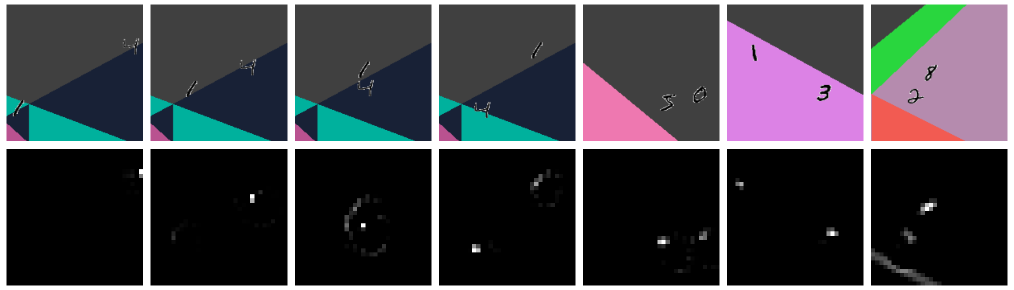

Fig. 5: DistinctiveNet detecting objects (heatmaps are post-contrast layer).

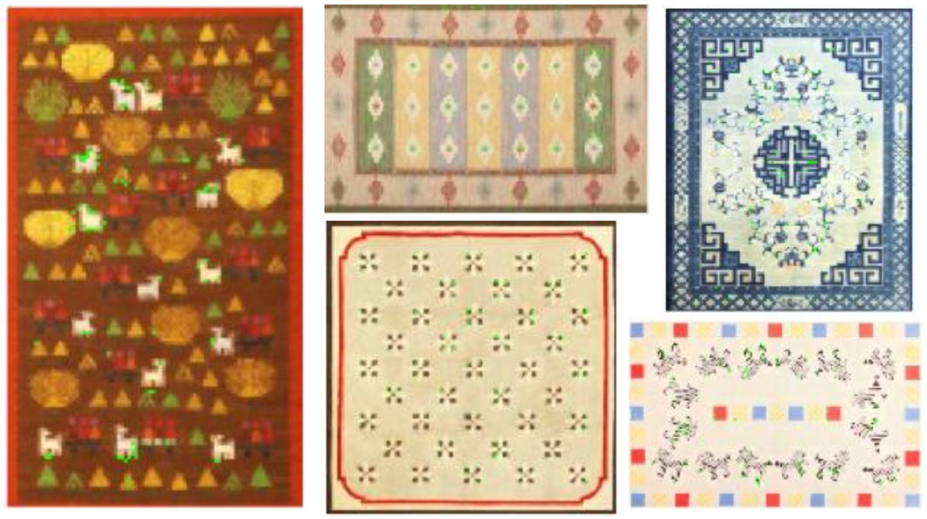

Fig. 6: Looking for repeating patterns in carpets using distinctiveness filter (auxiliary and main images are identical), detections in green.

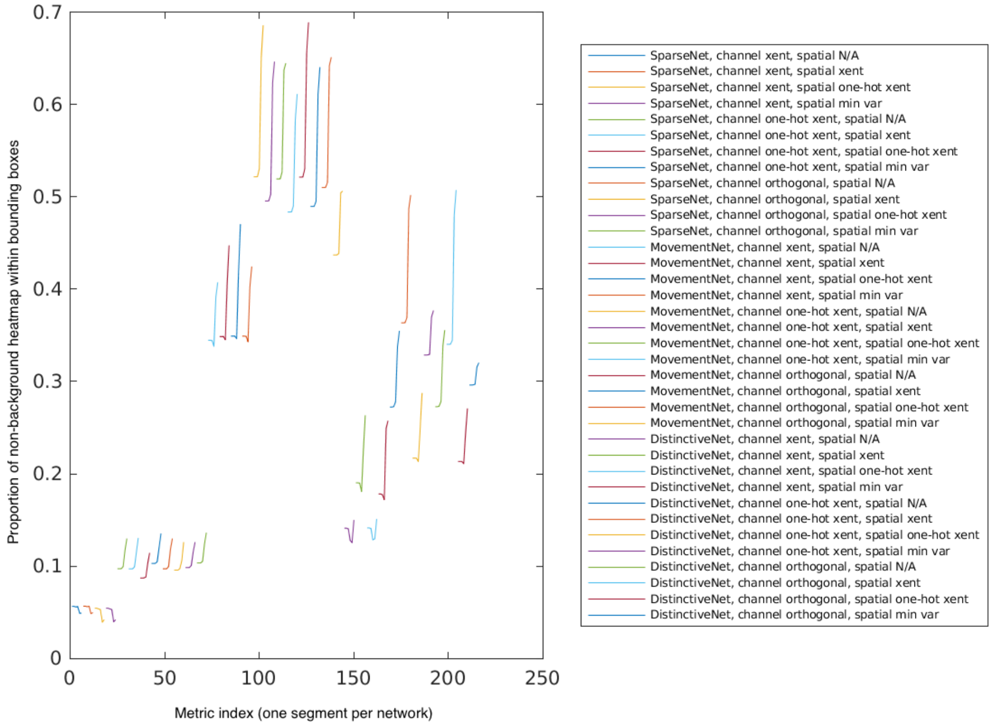

Fig. 7: Precision. DistinctiveNet against SparseNet (peakiness alone) and MovementNet (assumes moving object and stationary background).