Rich feature hierarchies for accurate object detection and semantic segmentation

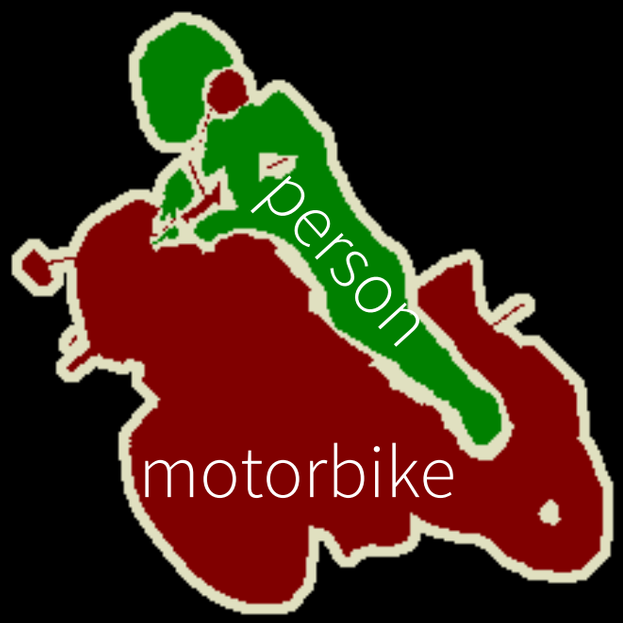
Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik
UC Berkeley

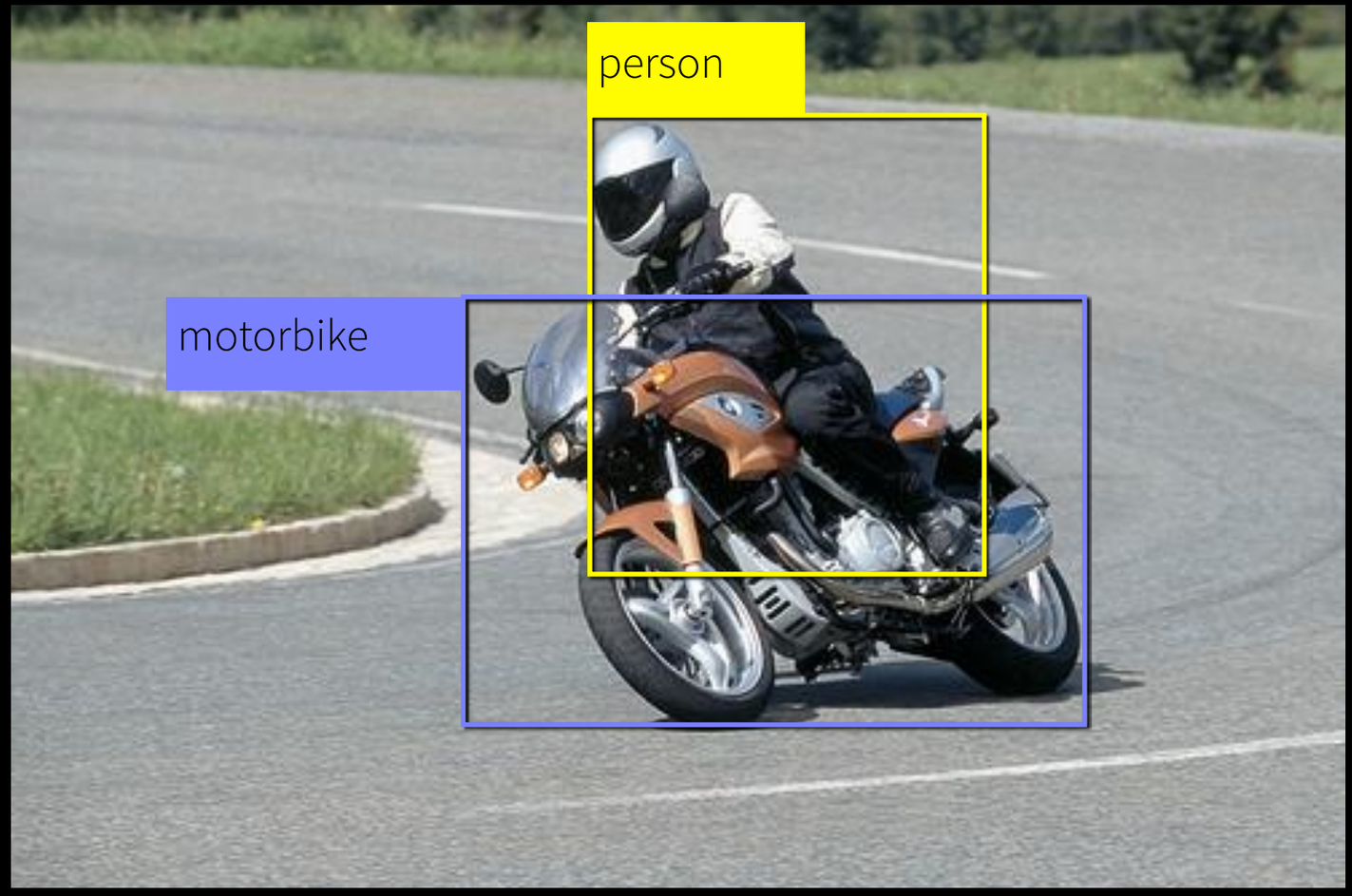
Tech Report @ http://arxiv.org/abs/1311.2524

Detection & Segmentation

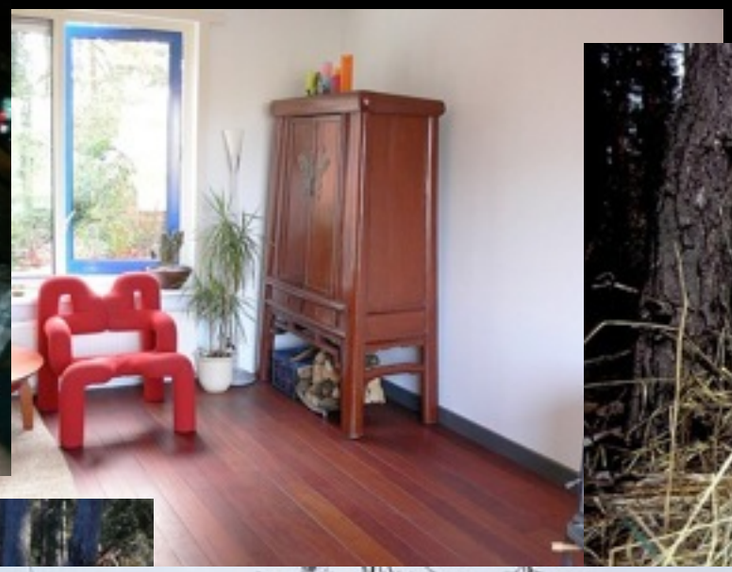
input

background





PASCAL VOC

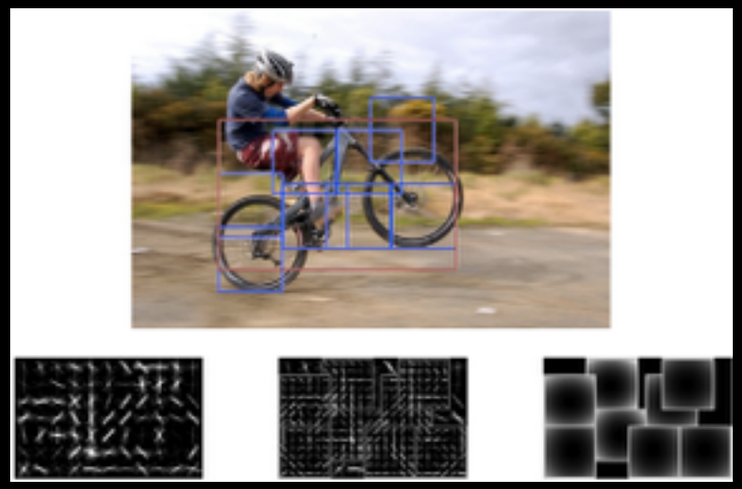




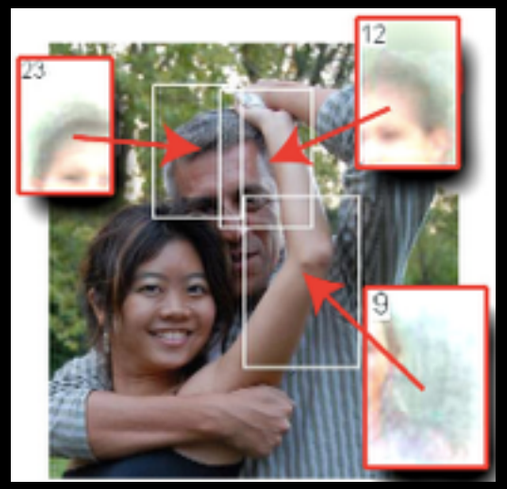
Example PASCAL VOC images

Dominant detection methods

1. Part-based sliding window methods (HOG)

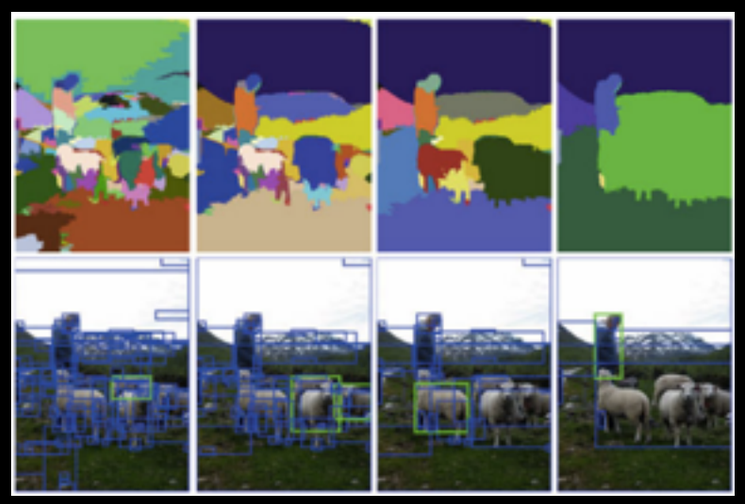


DPM



Poselets

2. Region-proposal classifiers (SIFT++ BoW)



Russell et al. 2006
Gu et al. 2009
van de Sande et al. 2011
> “selective search”

PASCAL VOC epochs (detection)

2007-2010 The Moore's law years

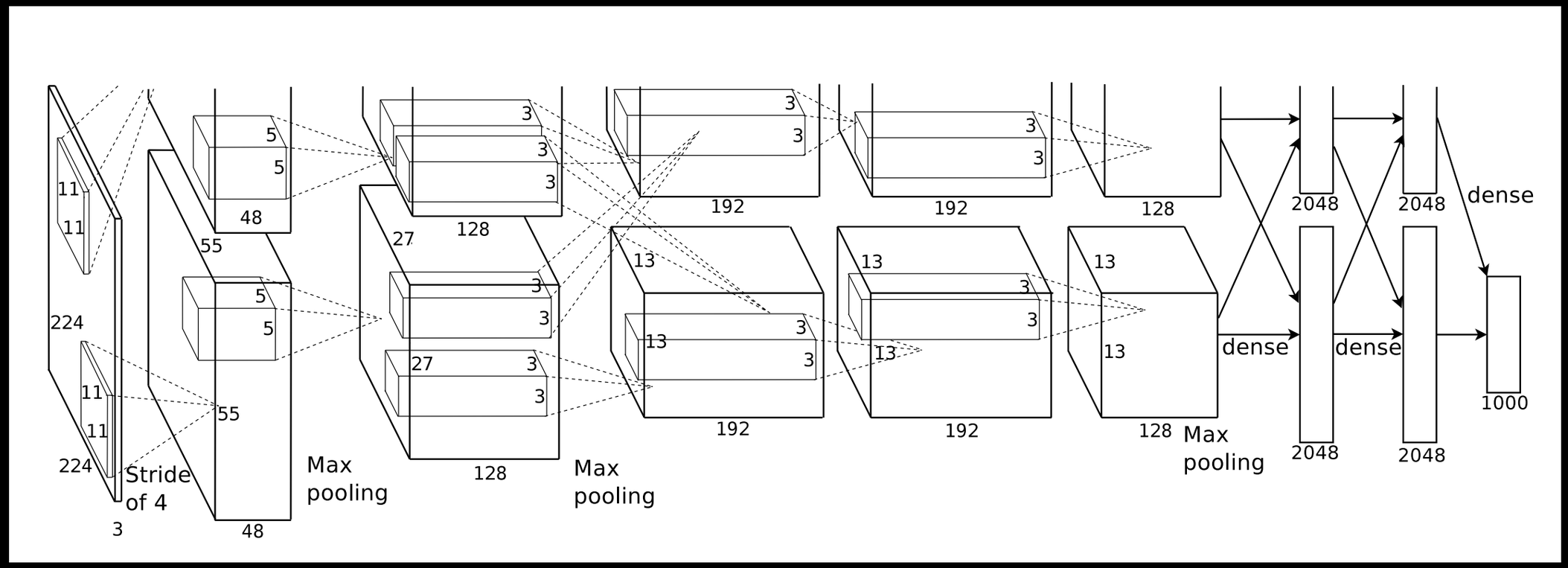
2010-2011 The year of kitchen sinks (or the all-too-soon end of Moore's law)

2011-2012 Stagnation (no new features left, juice all squeezed from context)

2013– *Learning rich features?*

ImageNet LSVRC’12 winner

UToronto “SuperVision” CNN



Krizhevsky, Sutskever, and Hinton.
ImageNet Classification with Deep Convolutional Neural Networks.
NIPS 2012.

cf. LeCun et al. Neural Comp. ’89 & Proc. of the IEEE ‘98

Impressive ImageNet results!

Task: 1000-way whole-image classification

Model	Top-1 (val)	Top-5 (val)	Top-5 (test)
SIFT + FVs [7]	—	—	*26.2%*
1 CNN	40.7%	18.2%	—
5 CNNs	38.1%	16.4%	**16.4%**
1 CNN*	39.0%	16.6%	—
7 CNNs*	36.7%	15.4%	**15.3%**

metric: classification error rate (lower is better)

But… does it generalize to other datasets and tasks?

See: Donahue, Jia, et al. DeCAF Tech Report.

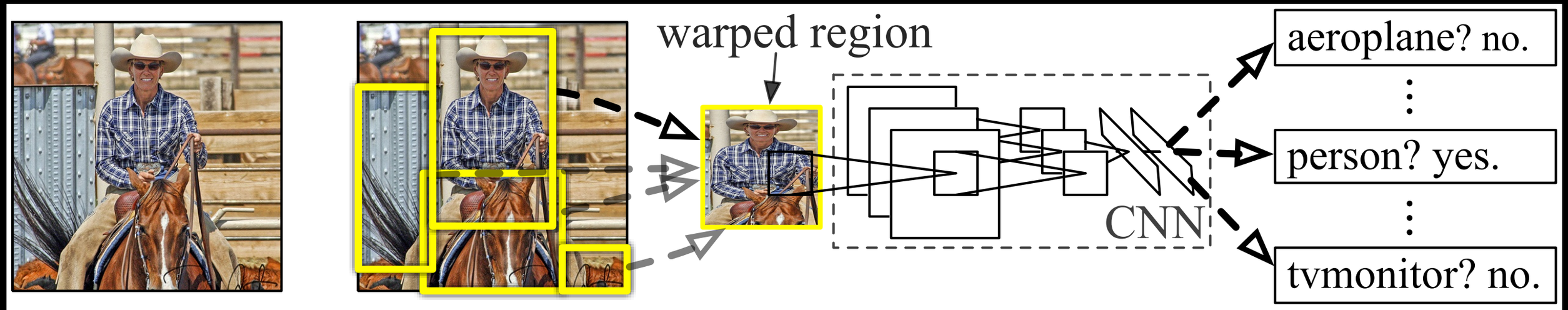
Much debate at ECCV'12

Objective

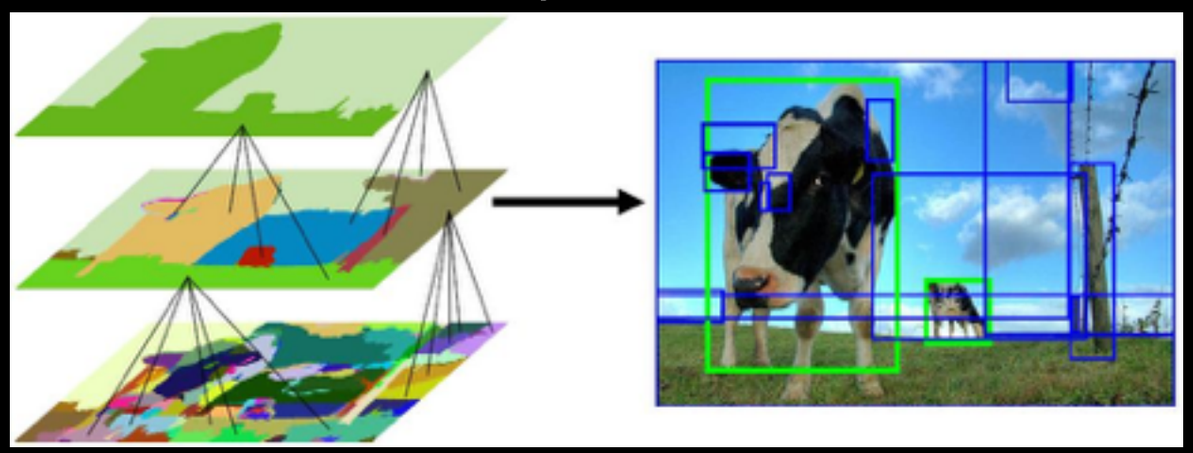
Understand if the SuperVision CNN can be made to work as an object detector.

Object detection system

R-CNN: "Regions with CNN features"



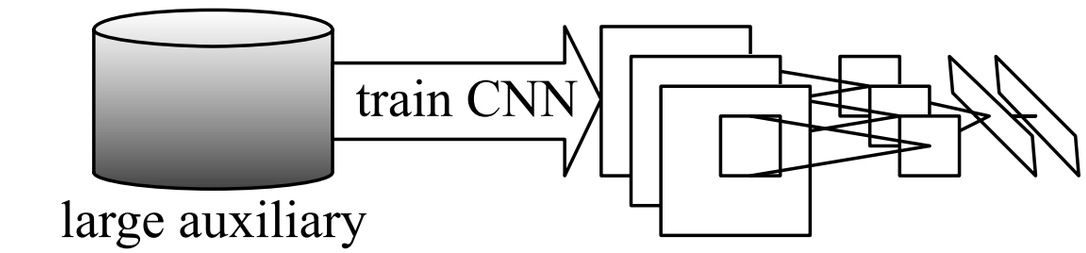
(With a few minor tweaks: semantic segmentation)



(e.g. selective search)

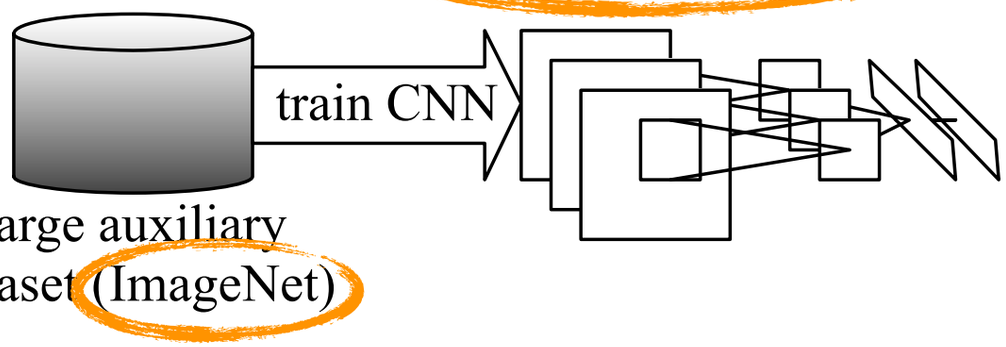
Training

1. Pre-train CNN for **image classification**



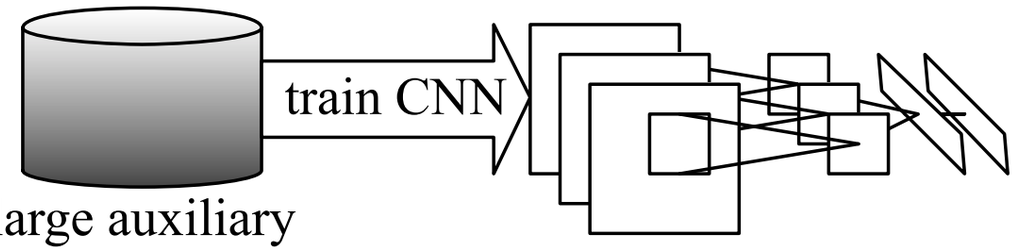
Training

1. Pre-train CNN for **image classification**

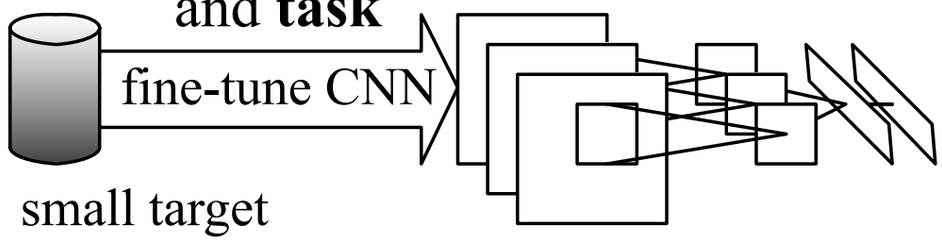


Training

1. Pre-train CNN for **image classification**



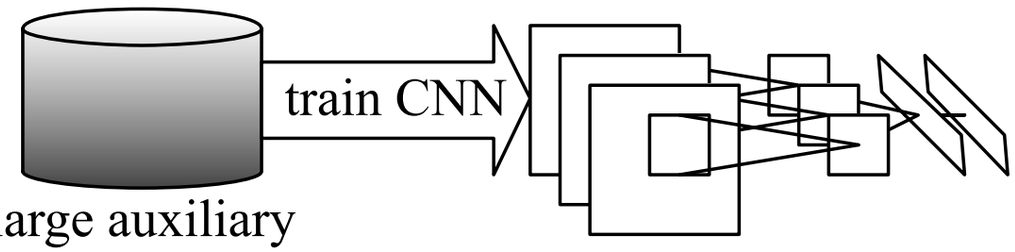
2. Fine-tune CNN on **target dataset** and **task**



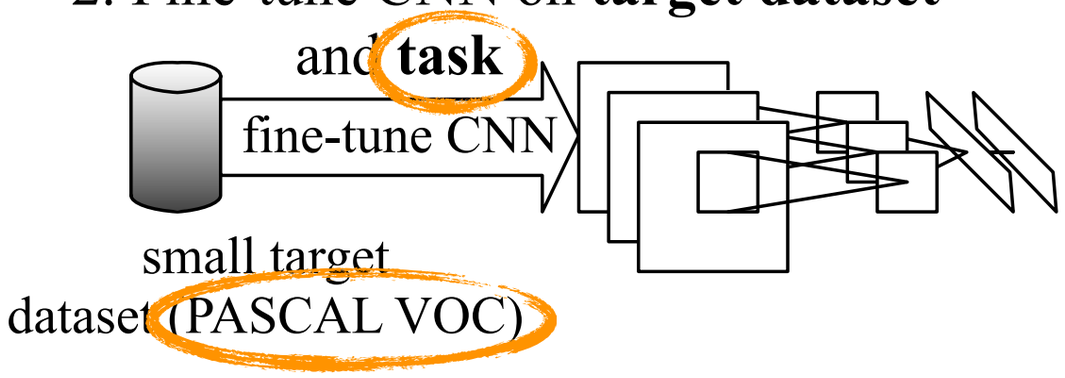
(optional)

Training

1. Pre-train CNN for **image classification**



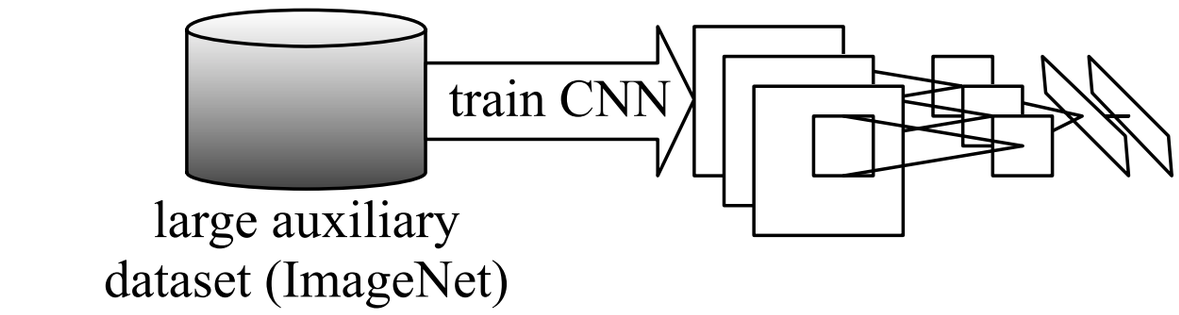
2. Fine-tune CNN on **target dataset**



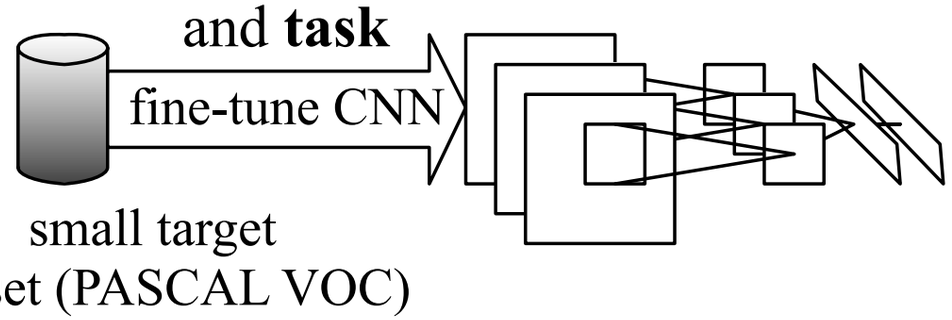
(optional)

Training

1. Pre-train CNN for **image classification**

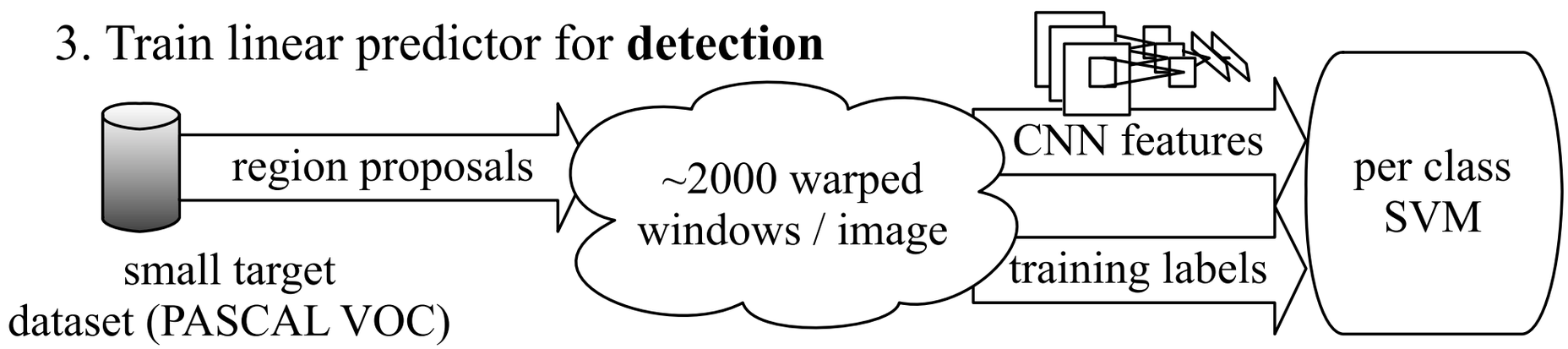


2. Fine-tune CNN on **target dataset** and **task**

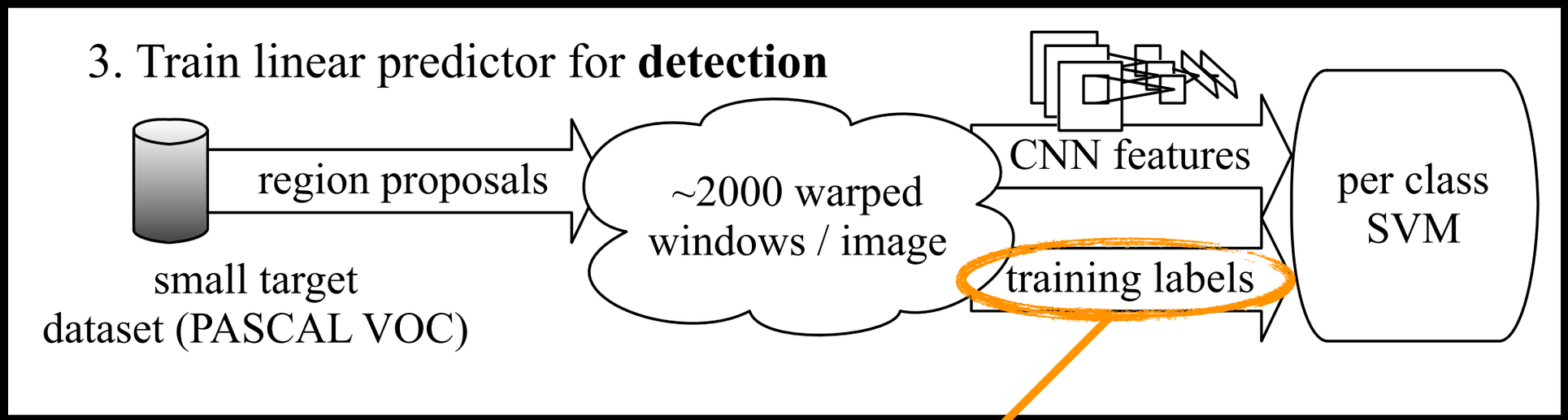


(optional)

3. Train linear predictor for **detection**



Training labels



labeling protocol

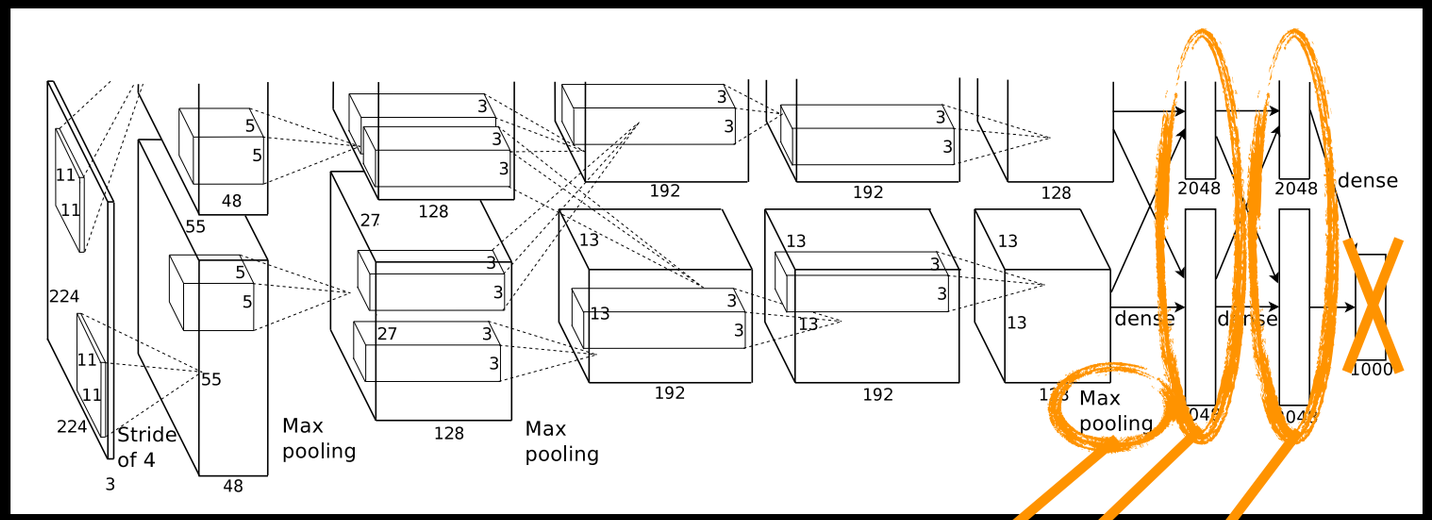
positives = ground truth

negatives = max IoU < 0.3

CNN features for detection

region

warped region



$pool_5$: 6 x 6 x 256 = 9216-dim

6.4% / 15% non-zero

fc_6: 4096-dimensional

71.2% / 20% nz

fc_7: 4096-dimensional

100% / 20% nz

Results

reference	VOC 2007	VOC 2010
DPM v5 (Girshick et al. 2011)	33.7%	29.6%
UVA sel. search (Uijlings et al. 2012)		35.1%
Regionlets (Wang et al. 2013)	**41.7%**	**39.7%**

metric: mean average precision (higher is better)

Results

	VOC 2007	VOC 2010
DPM v5 (Girshick et al. 2011)	33.7%	29.6%
UVA sel. search (Uijlings et al. 2012)		35.1%
Regionlets (Wang et al. 2013)	41.7%	**39.7%**
pre-trained only: R-CNN $pool_5$	40.1%	
pre-trained only: R-CNN fc_6	**43.4%**	
pre-trained only: R-CNN fc_7	42.6%	

metric: mean average precision (higher is better)

Results

	VOC 2007	VOC 2010
DPM v5 (Girshick et al. 2011)	33.7%	29.6%
UVA sel. search (Uijlings et al. 2012)		35.1%
Regionlets (Wang et al. 2013)	41.7%	39.7%
R-CNN $pool_5$	40.1%	
R-CNN fc_6	43.4%	
R-CNN fc_7	42.6%	
R-CNN FT $pool_5$	42.1%	
R-CNN FT fc_6	47.2%	
R-CNN FT fc_7	**48%**	**43.5%**

fine-tuned

metric: mean average precision (higher is better)

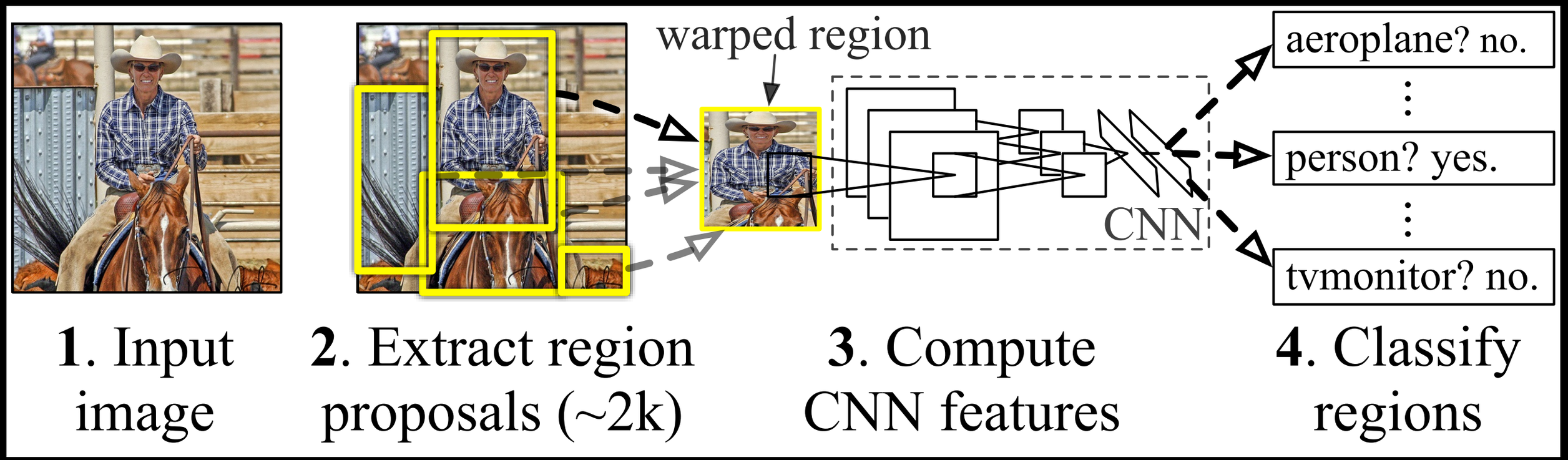
Results — update

	VOC 2007	VOC 2010
DPM v5 (Girshick et al. 2011)	33.7%	29.6%
UVA sel. search (Uijlings et al. 2012)		35.1%
Regionlets (Wang et al. 2013)	41.7%	39.7%
pre-trained only: R-CNN $pool_5$	~~40.1%~~ **44.0%**	
pre-trained only: R-CNN fc_6	~~43.4%~~ **46.2%**	
pre-trained only: R-CNN fc_7	~~42.6%~~ **43.5%**	

metric: mean average precision (higher is better)

CV and DL together

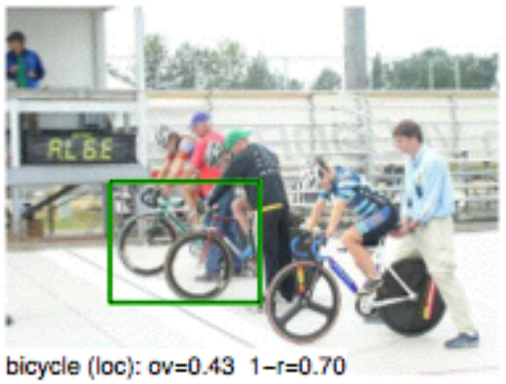
Good features are not enough!



Computer Vision | Deep Learning | Computer Vision

Top bicycle FPs (AP 62.5%)


bicycle (loc): ov=0.36 1−r=0.78

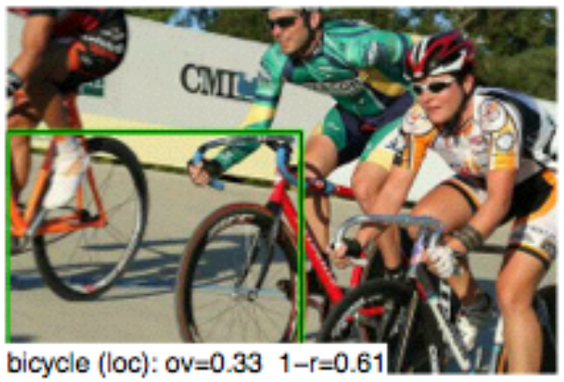

bicycle (loc): ov=0.43 1−r=0.70

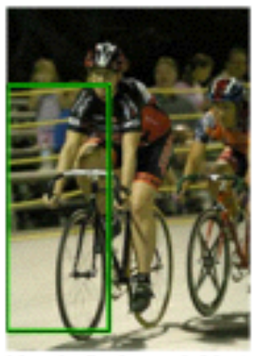

bicycle (loc): ov=0.32 1−r=0.69


bicycle (loc): ov=0.43 1−r=0.67


bicycle (loc): ov=0.34 1−r=0.66

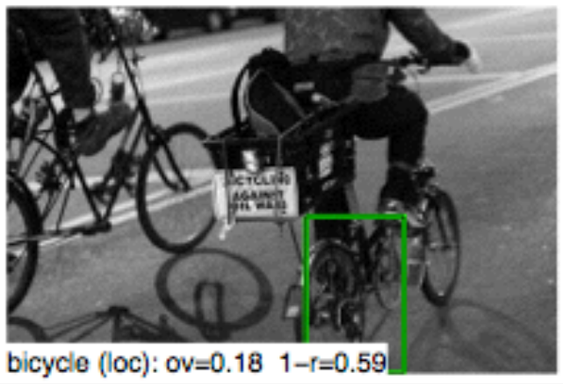

bicycle (loc): ov=0.47 1−r=0.65


bicycle (loc): ov=0.33 1−r=0.61


bicycle (loc): ov=0.28 1−r=0.61

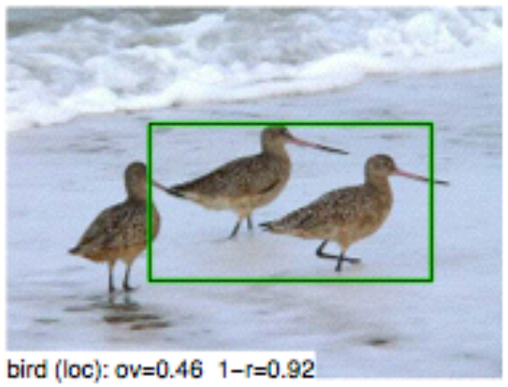

bicycle (sim): ov=0.00 1−r=0.60

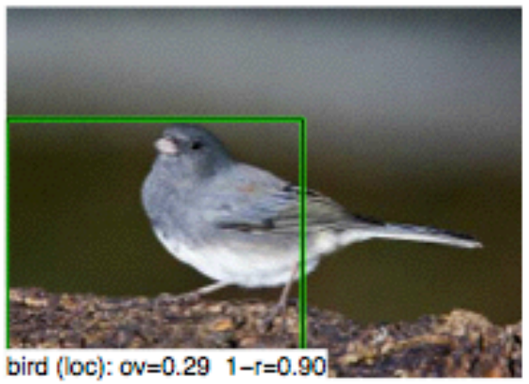

bicycle (sim): ov=0.00 1−r=0.59


bicycle (loc): ov=0.18 1−r=0.59


bicycle (loc): ov=0.46 1−r=0.58

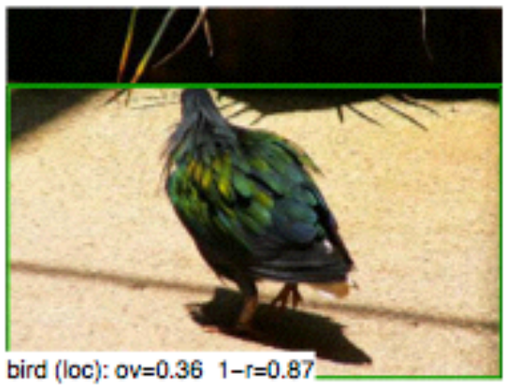
Top bird FPs (AP 41.4%)

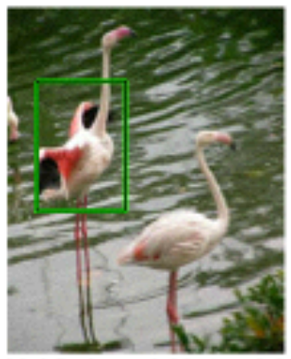

bird (loc): ov=0.46 1−r=0.92


bird (loc): ov=0.29 1−r=0.90


bird (loc): ov=0.49 1−r=0.89

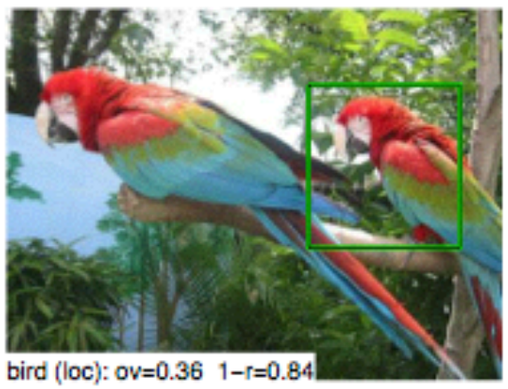

bird (loc): ov=0.35 1−r=0.92

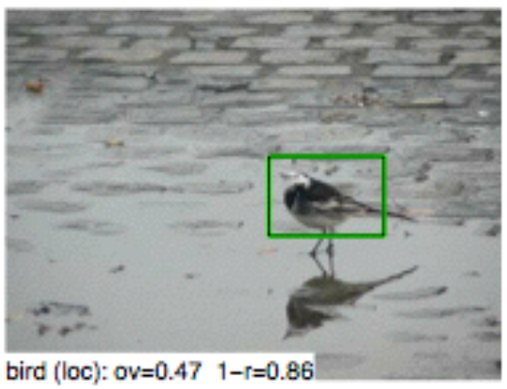

bird (loc): ov=0.36 1−r=0.87


bird (loc): ov=0.39 1−r=0.86

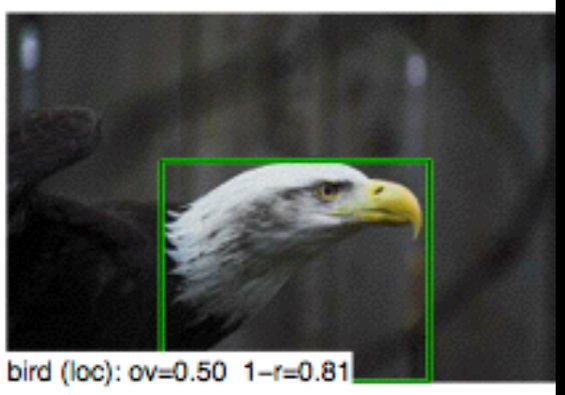

bird (loc): ov=0.32 1−r=0.89


bird (loc): ov=0.38 1−r=0.86


bird (loc): ov=0.36 1−r=0.84


bird (loc): ov=0.47 1−r=0.86

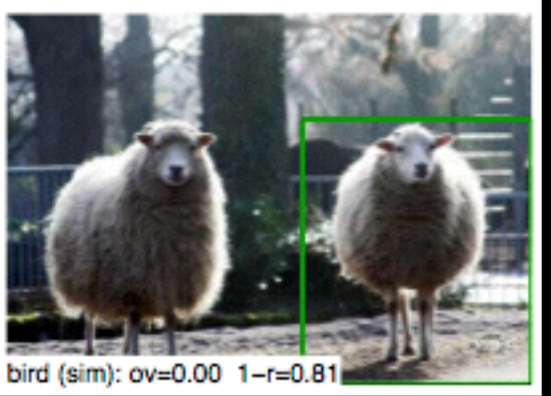

bird (loc): ov=0.32 1−r=0.84


bird (loc): ov=0.50 1−r=0.81

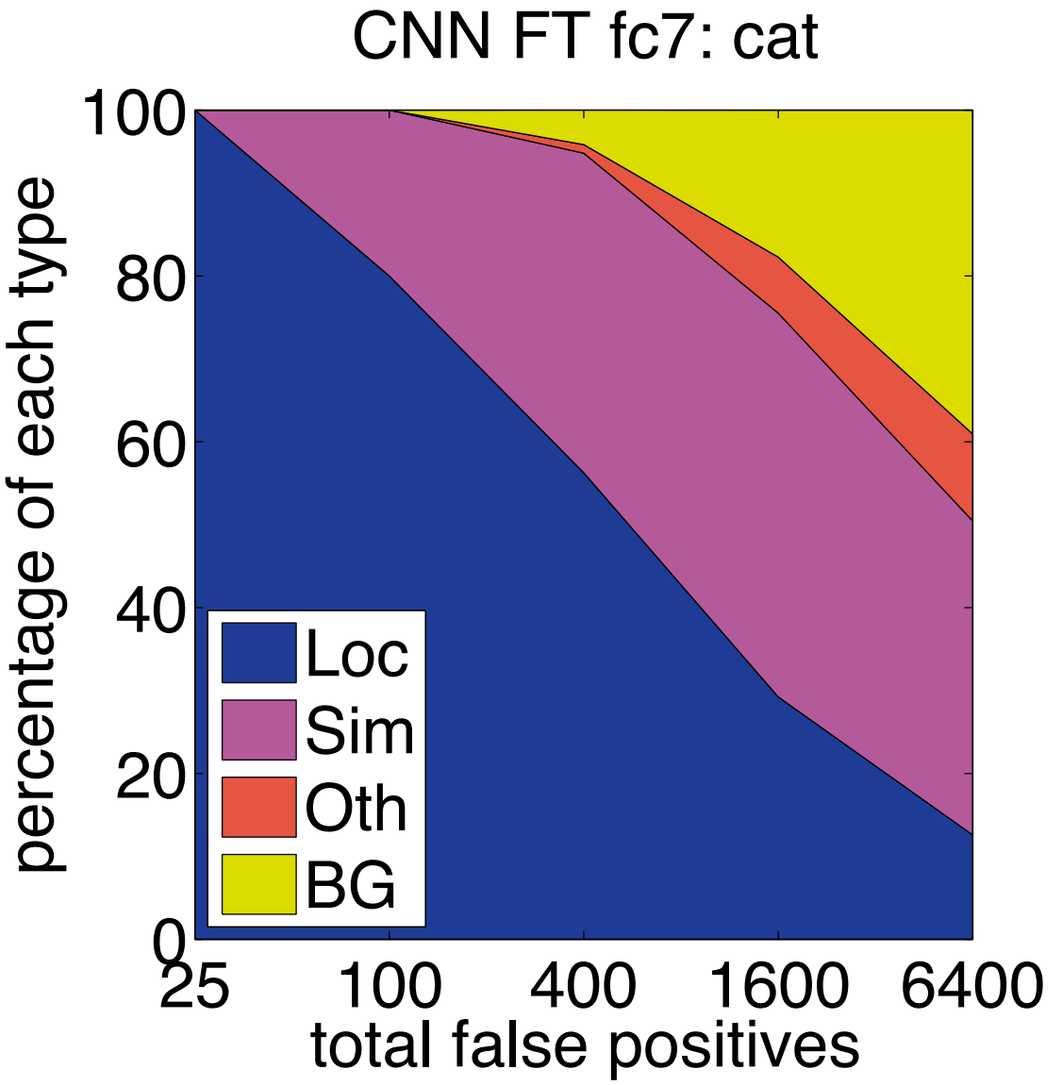

bird (loc): ov=0.34 1−r=0.81


bird (loc): ov=0.21 1−r=0.83

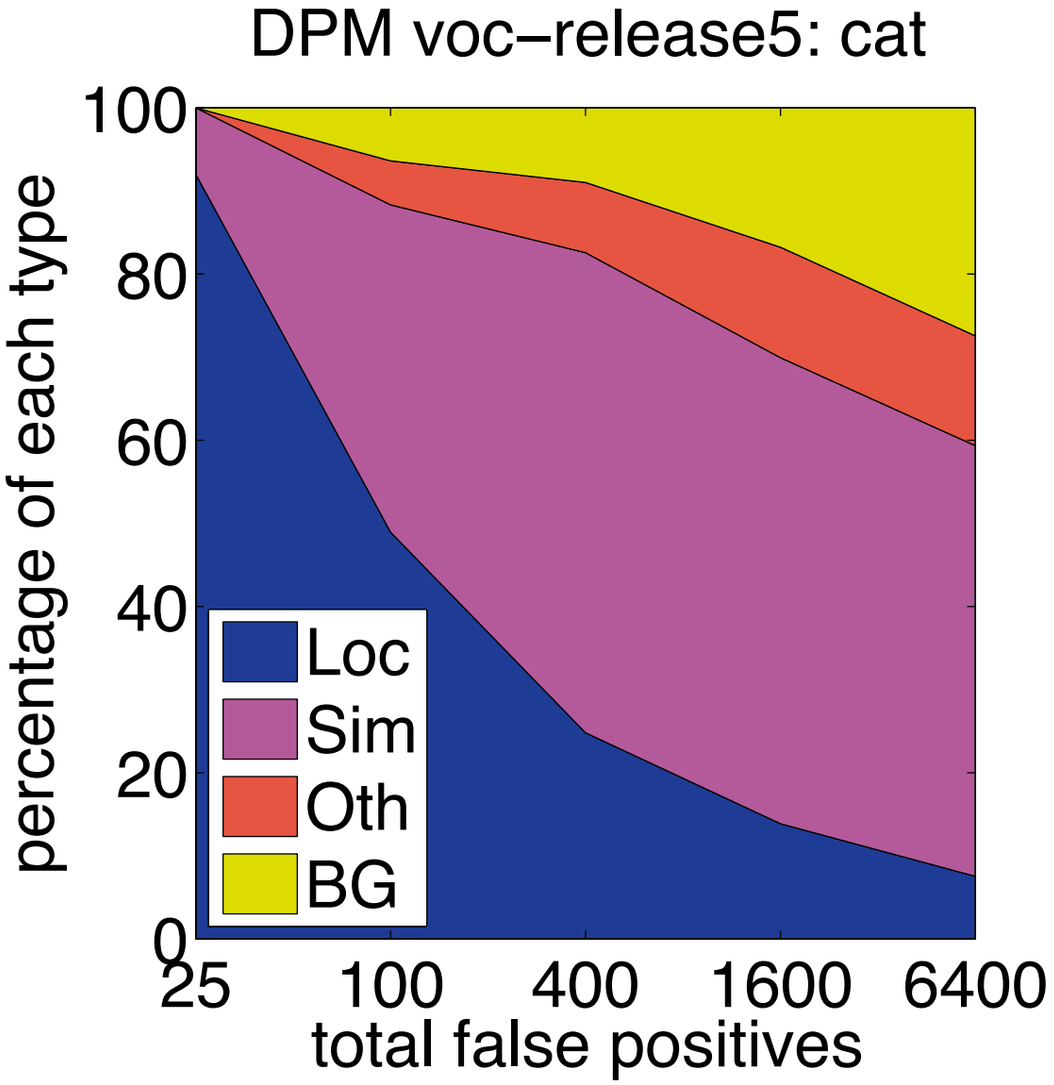

bird (loc): ov=0.18 1−r=0.81


bird (sim): ov=0.00 1−r=0.81

False positive types: cat



AP 56.3%



AP 23.0%

Analysis software from: D. Hoiem, Y. Chodpathumwan, and Q. Dai. “Diagnosing Error in Object Detectors.” *ECCV*, 2012.

Visualizing features

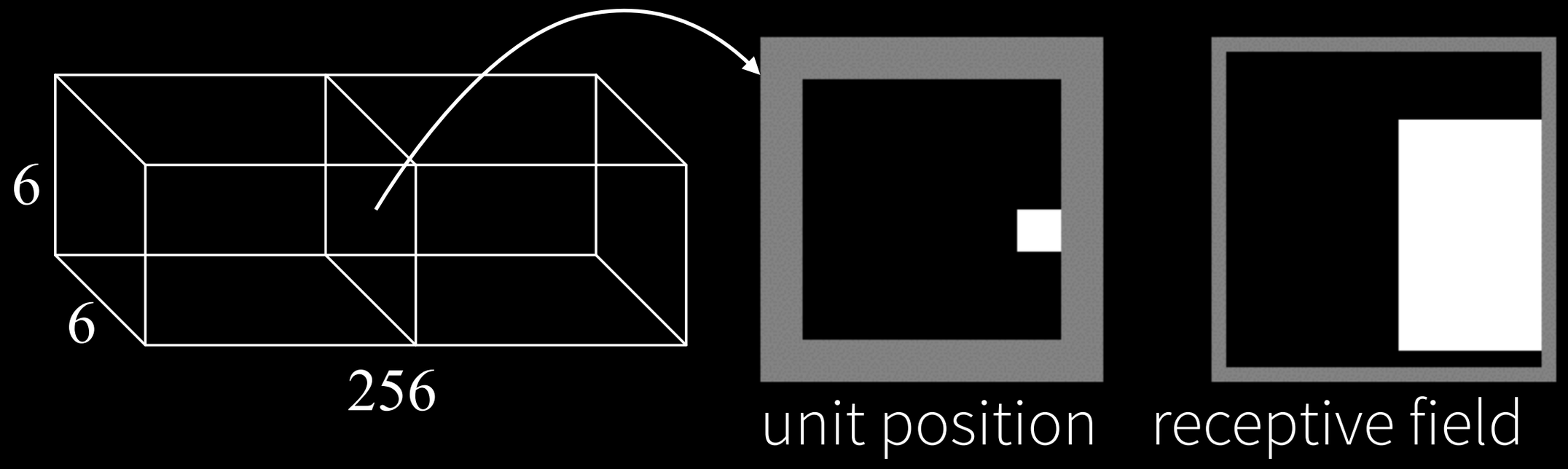
> What does $pool_5$ learn?

> Recap:

> > $pool_5$: max-pooled output of last conv. layer

> > 6 x 6 spatial structure (with 256 channels)

> > receptive field size 163 x 163 (of 224 x 224)

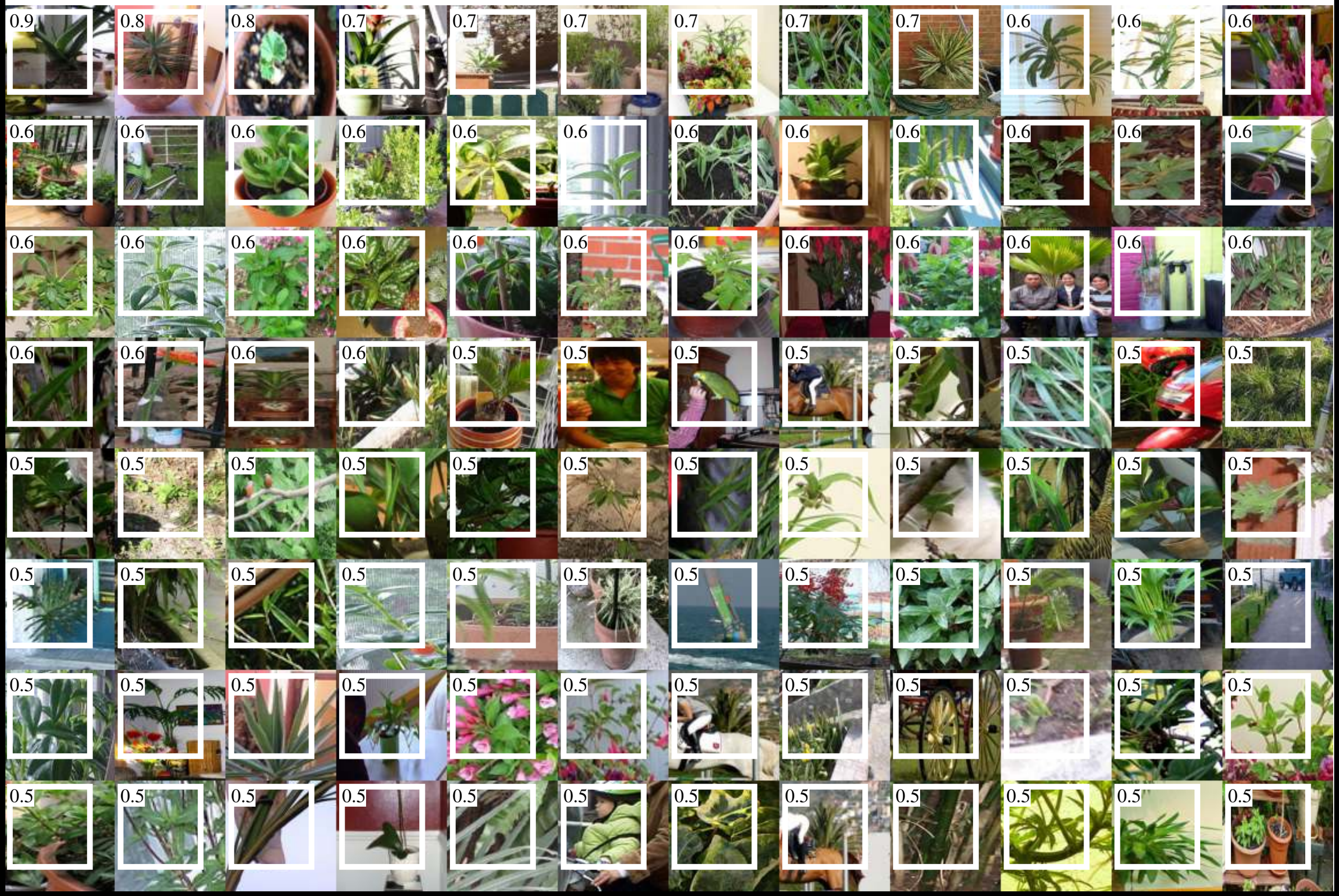


Visualization method

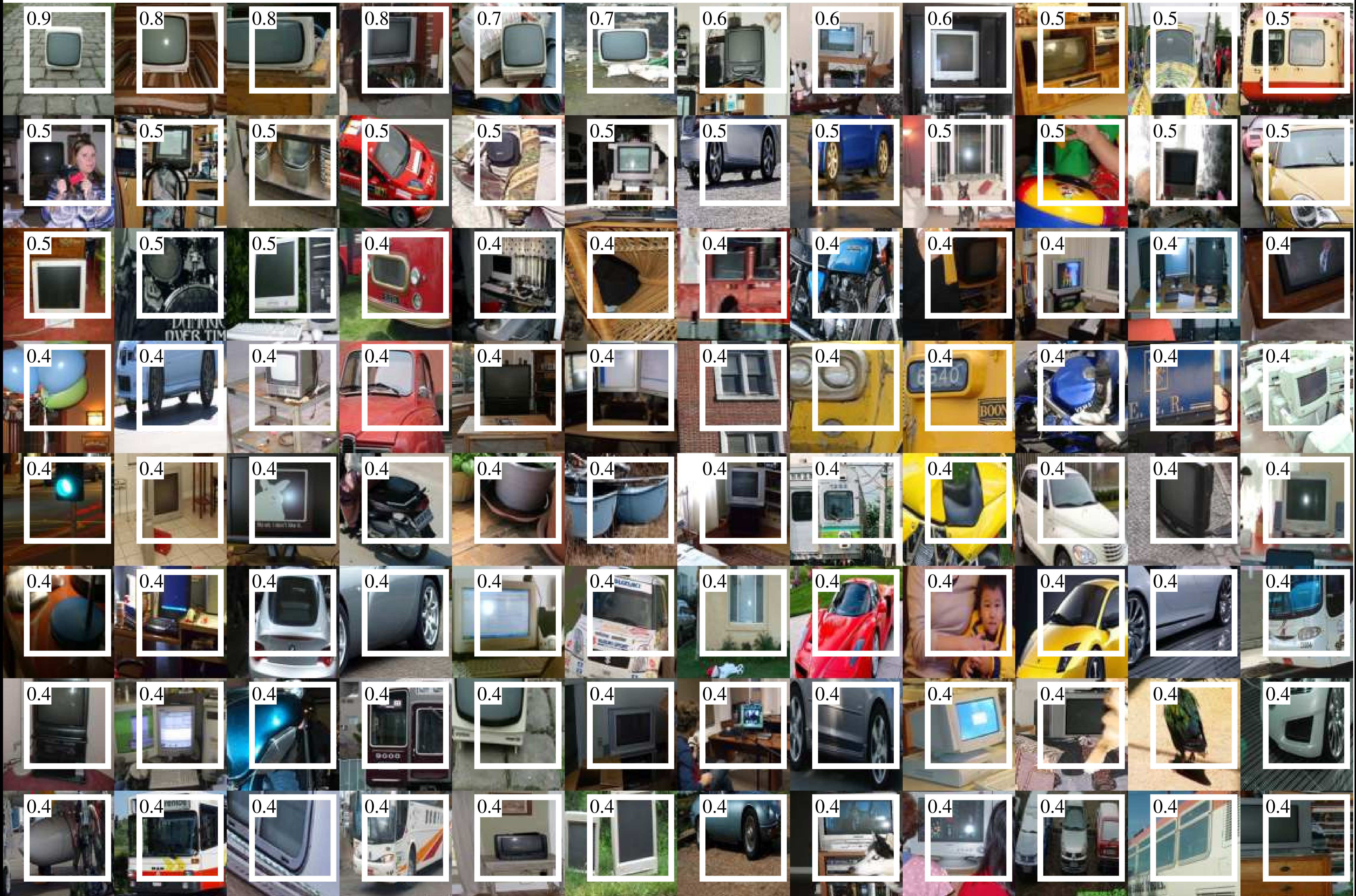
> Select a unit in $pool_5$
> Run it as a detector
> Show top-scoring regions
> Non-parametric, lets unit “speak for itself”

(Used ~10 million held-out regions.)

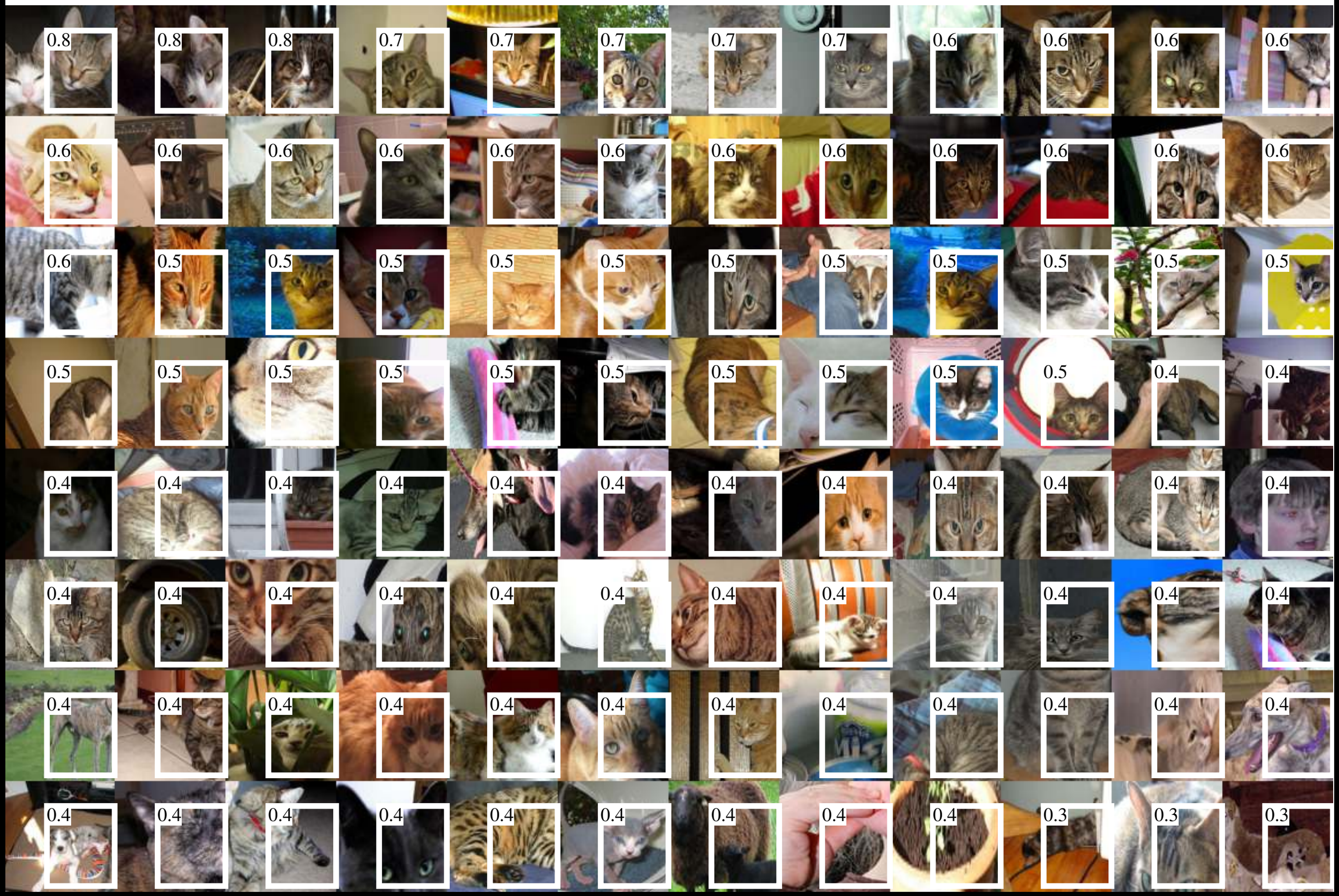
pool5 feature: (3,3,42) (top 1 – 96)



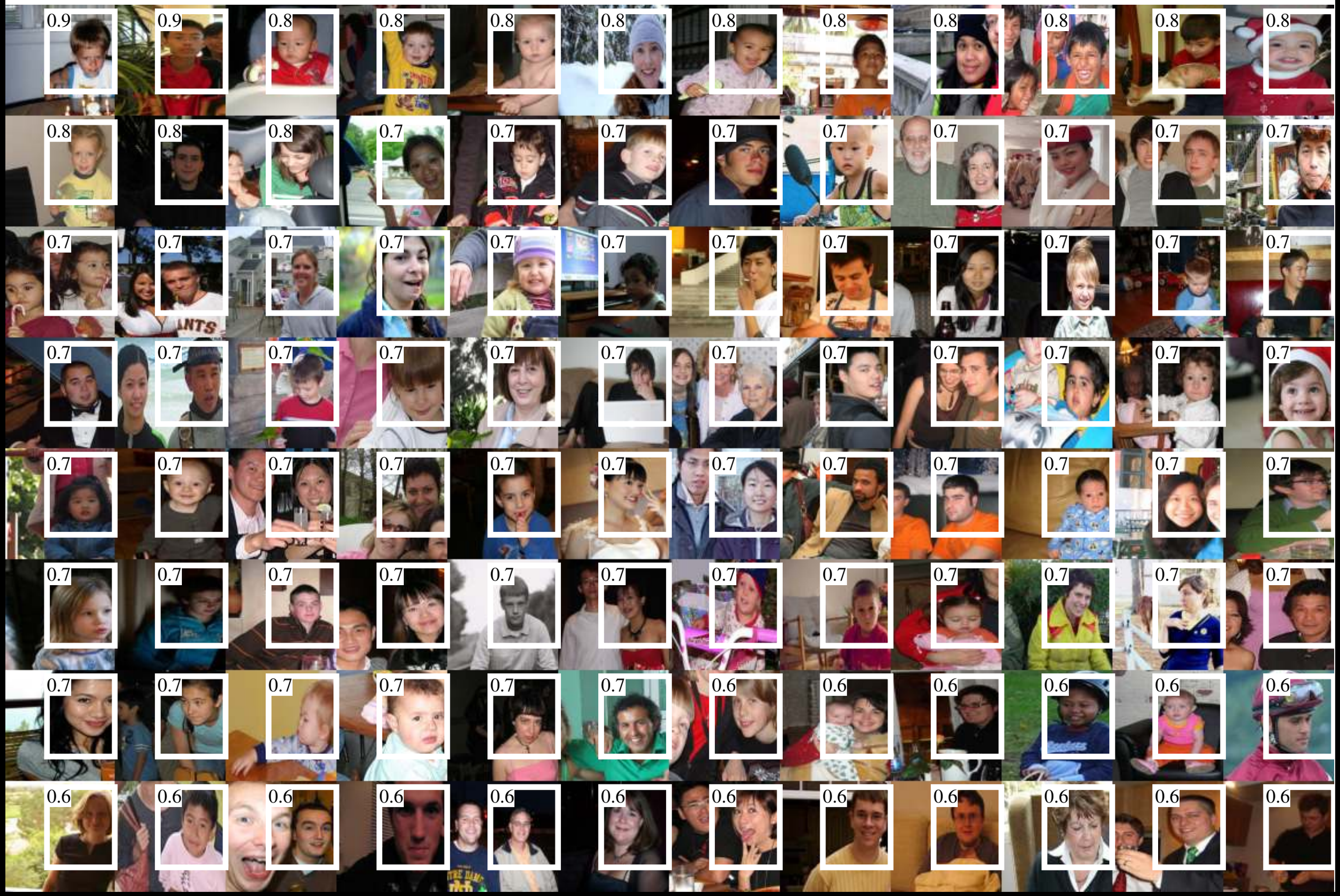
pool5 feature: (3,4,80) (top 1 – 96)



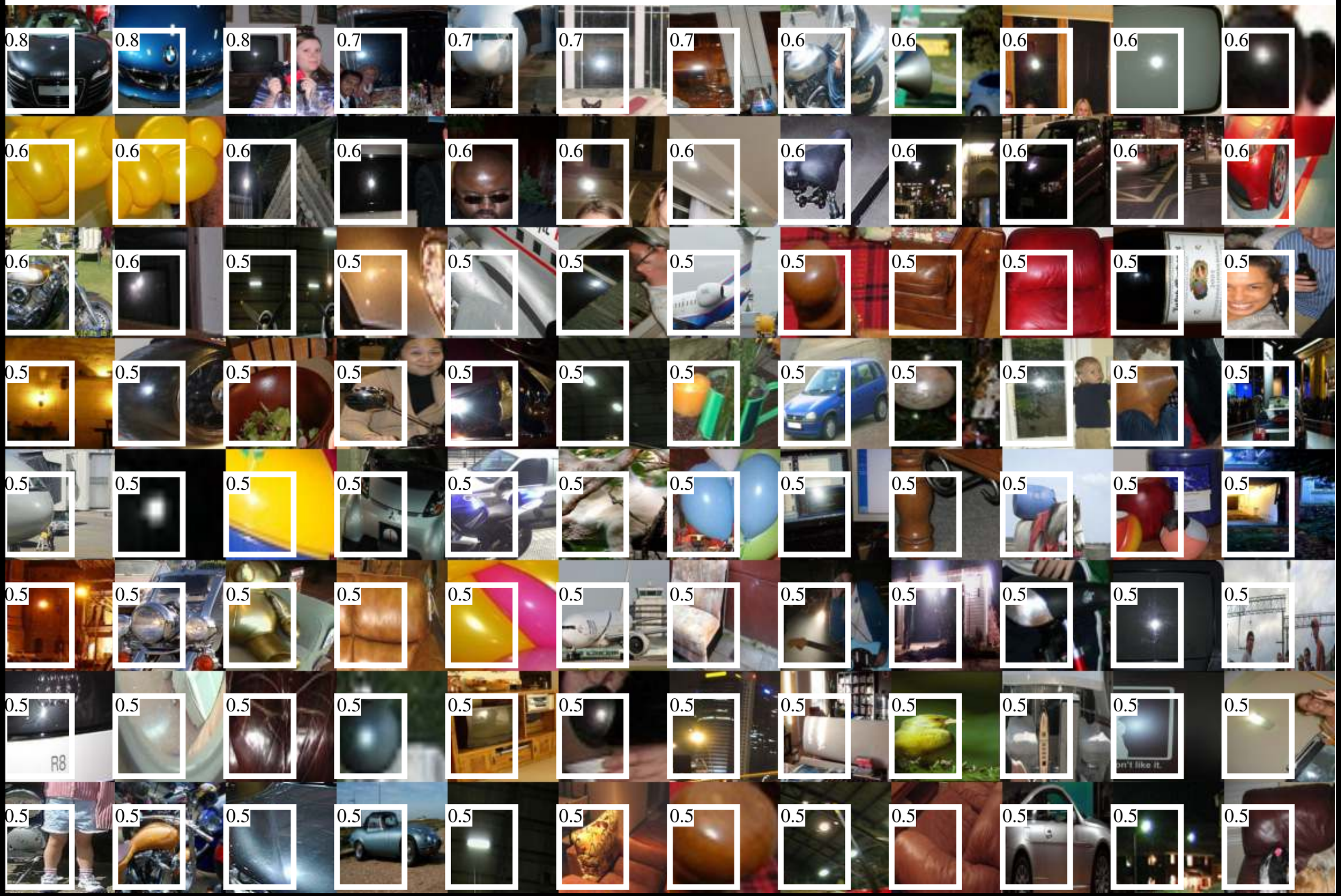
pool5 feature: (4,5,110) (top 1 – 96)



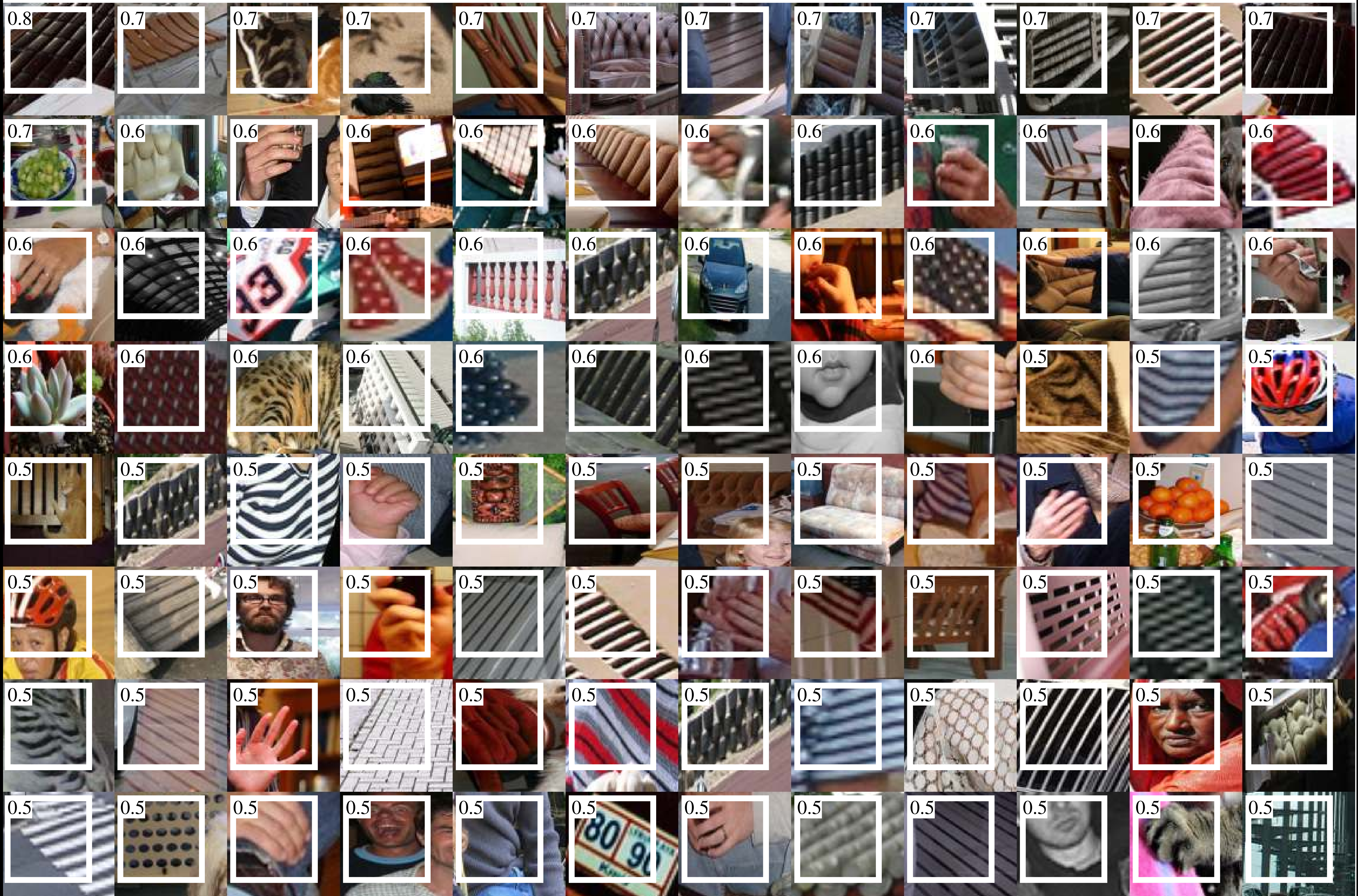
pool5 feature: (3,5,129) (top 1 – 96)



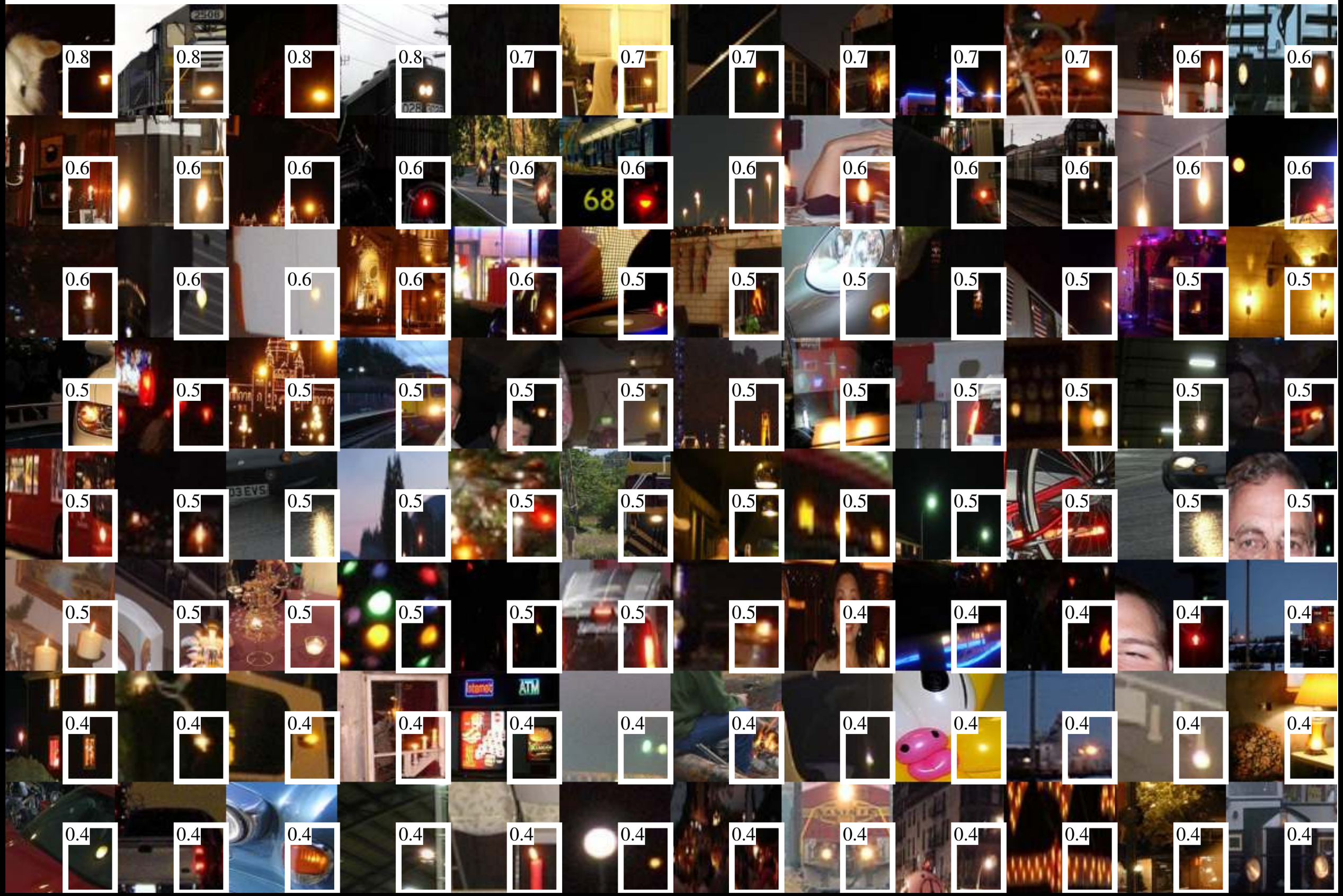
pool5 feature: (4,2,26) (top 1 – 96)



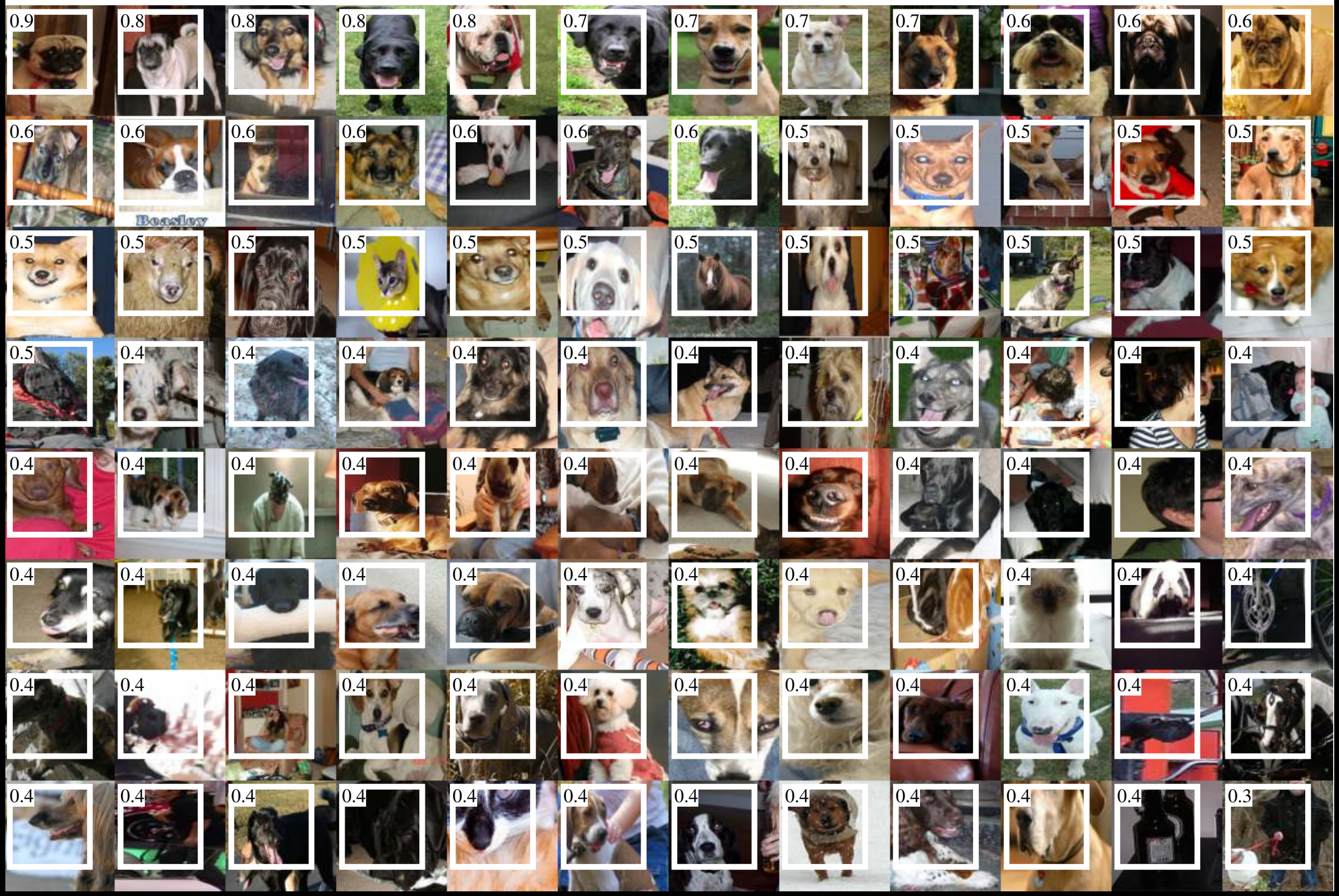
pool5 feature: (3,3,39) (top 1 – 96)



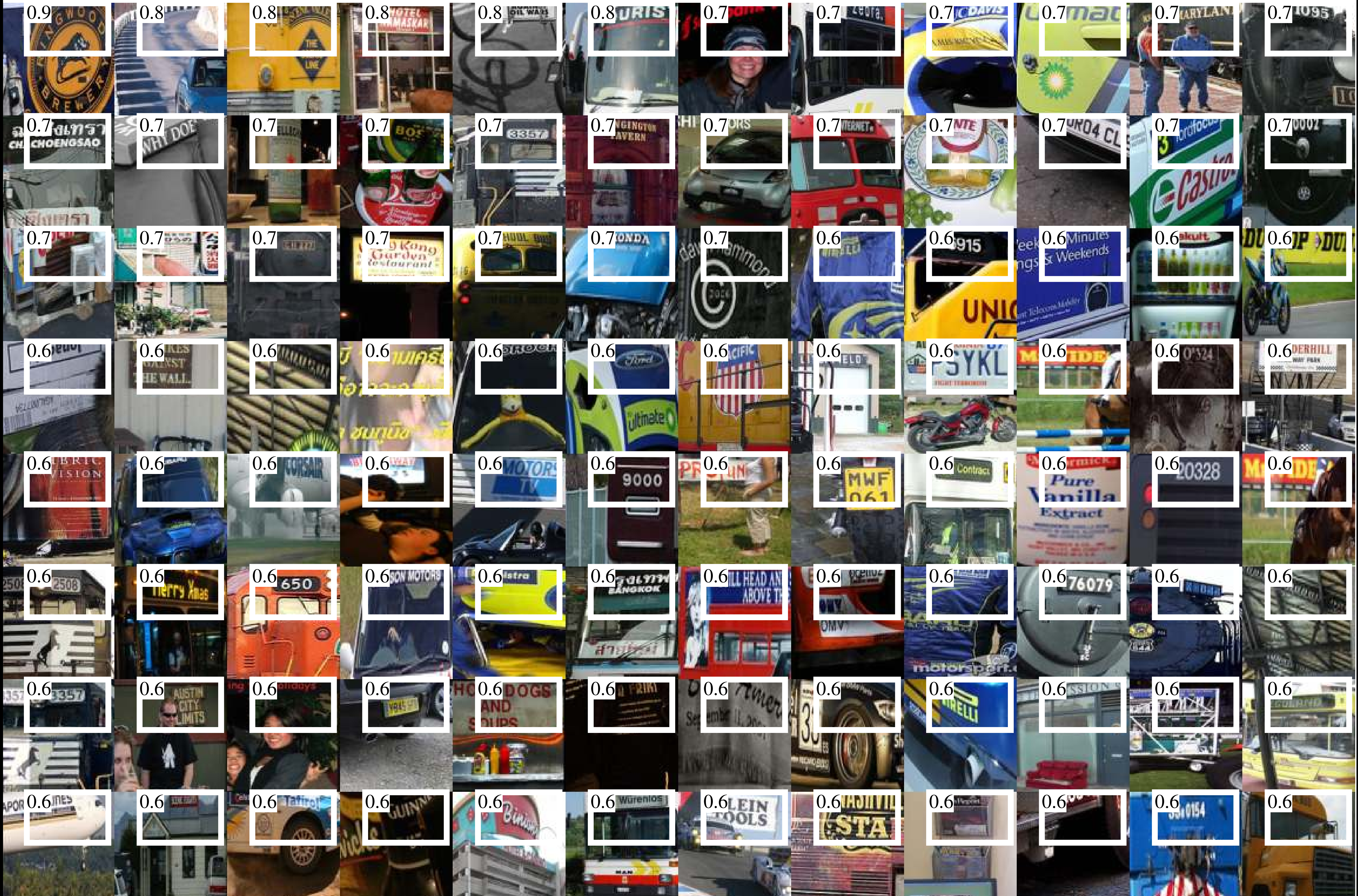
pool5 feature: (5,6,53) (top 1 – 96)



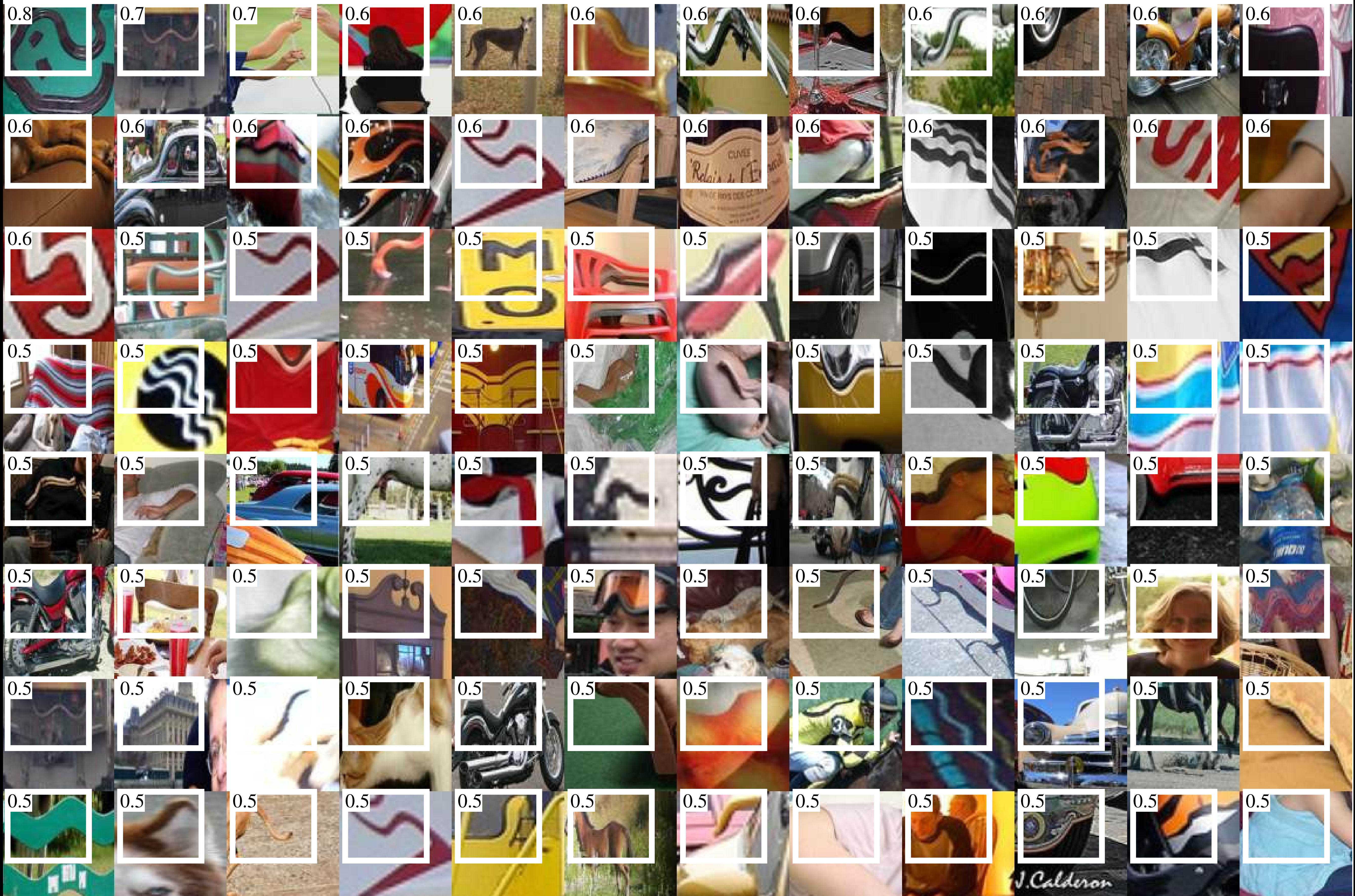
pool5 feature: (3,3,139) (top 1 – 96)



pool5 feature: (1,4,138) (top 1 – 96)



pool5 feature: (2,3,210) (top 1 – 96)



Semantic segmentation

	VOC 2011 test
1. UCB Regions and Parts	40.8%
2. Bonn O_2P	47.6%
3. R-CNN full+fg fc_6	**47.9%**

metric: mean segmentation accuracy (higher is better)

(Carreira & Sminchisescu)