Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik

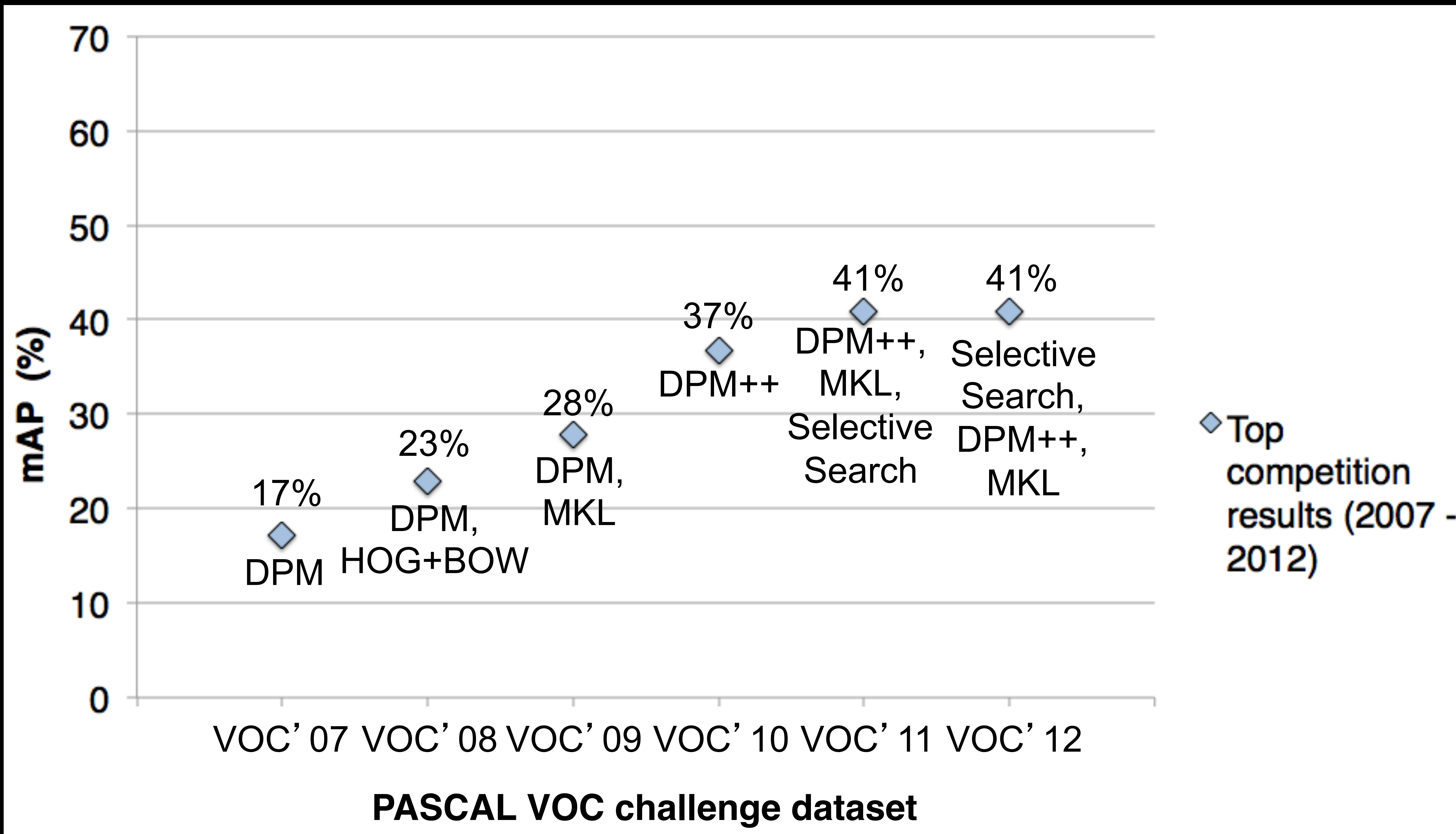
EECS Berkeley UNIVERSITY OF CALIFORNIA

Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik

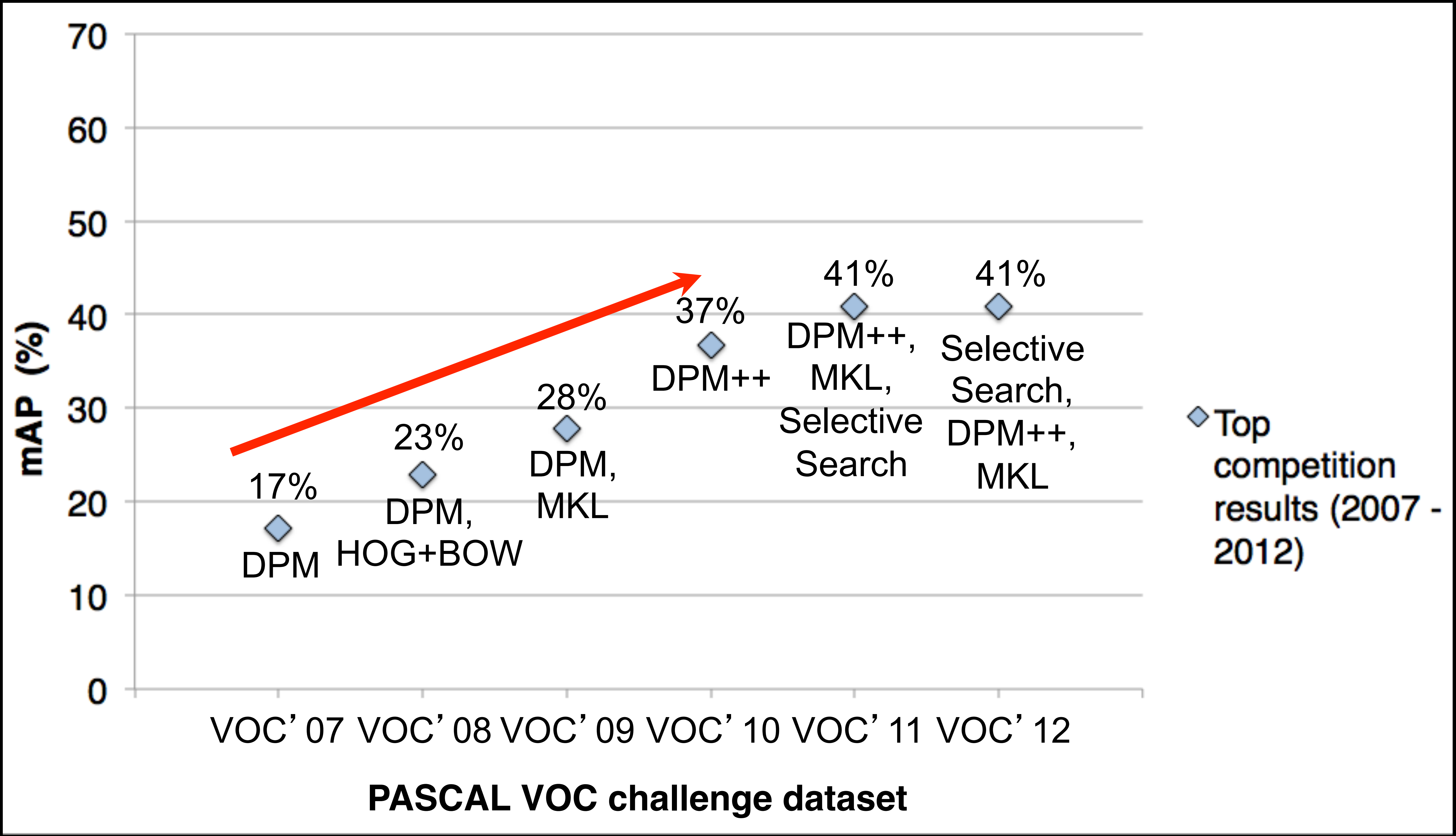
EECS Berkeley UNIVERSITY OF CALIFORNIA

PASCAL VOC detection history



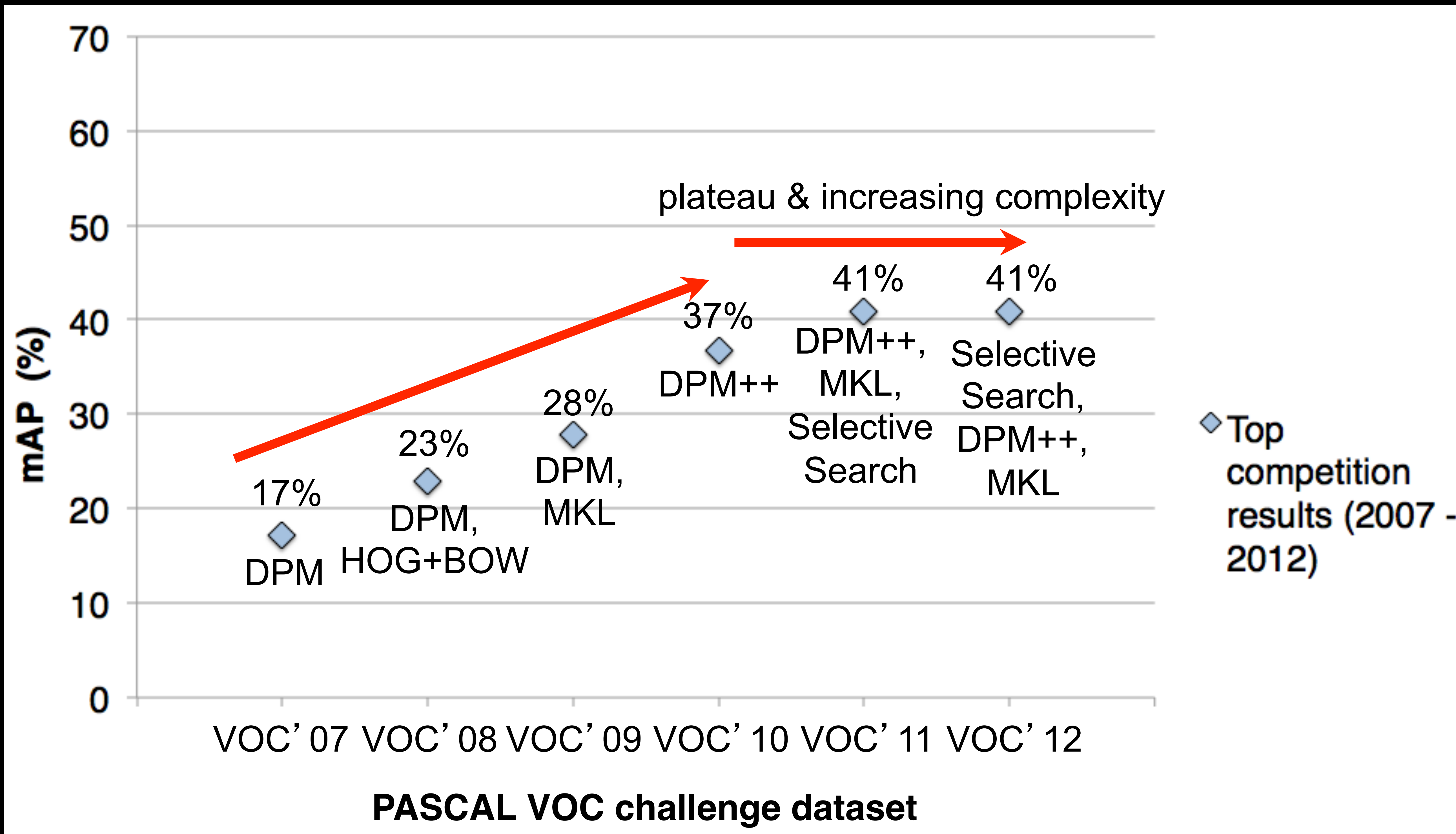
[Source: http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc20{07,08,09,10,11,12}/results/index.html]

A rapid rise in performance



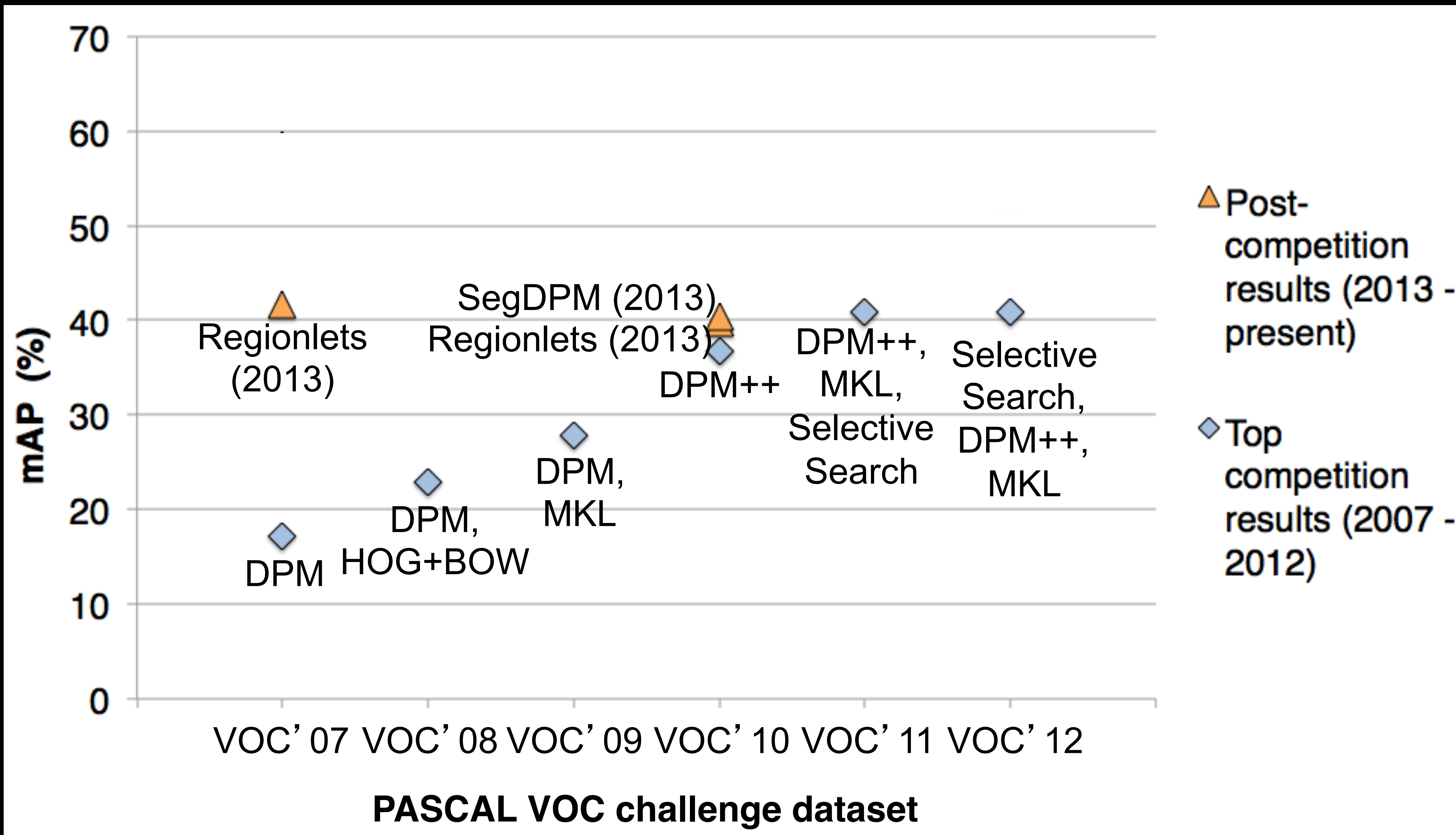
[Source: http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc20{07,08,09,10,11,12}/results/index.html]

Complexity and the plateau



[Source: http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc20{07,08,09,10,11,12}/results/index.html]

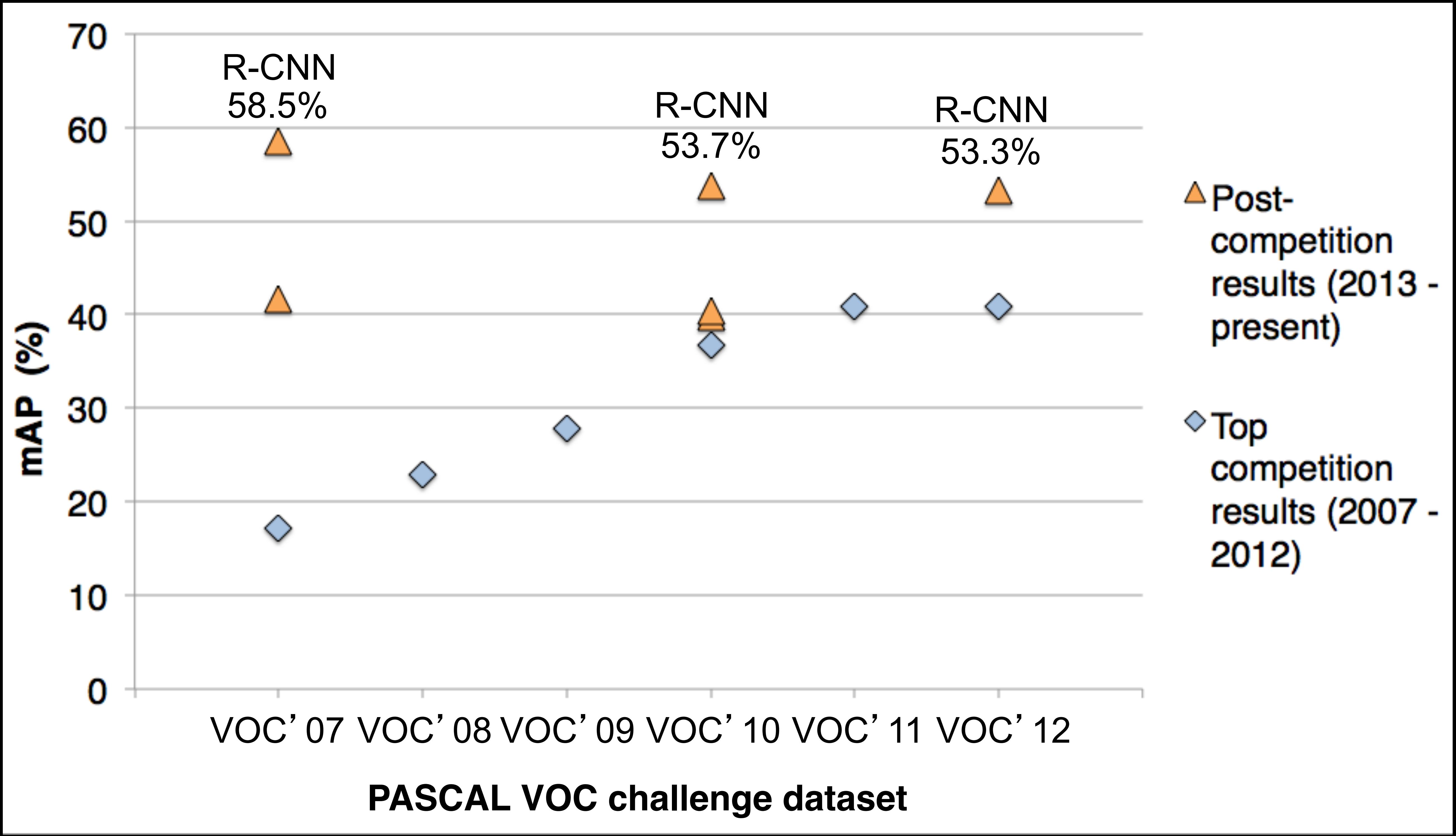
SIFT, HOG, LBP, ...



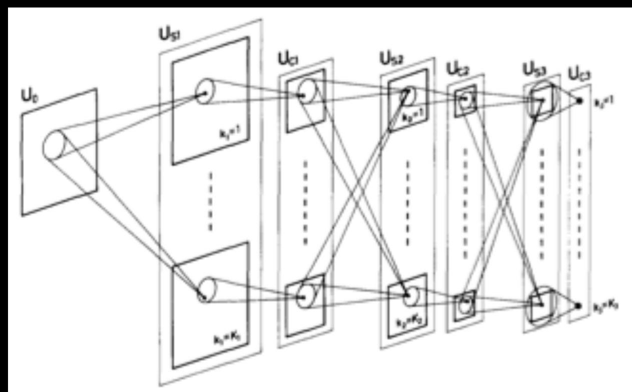
[Regionlets. Wang et al. ICCV’ 13]

[SegDPM. Fidler et al. CVPR’ 13]

R-CNN: Regions with CNN features



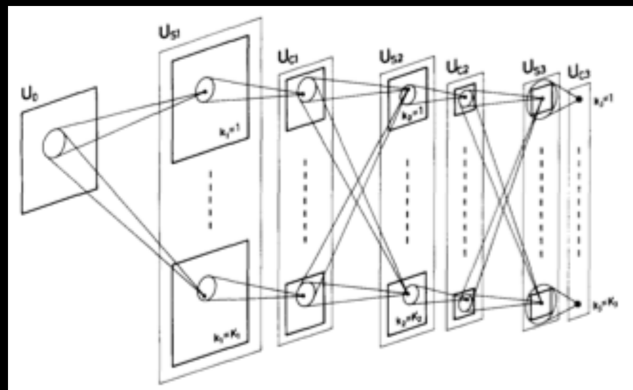
Feature learning with CNNs



Fukushima 1980

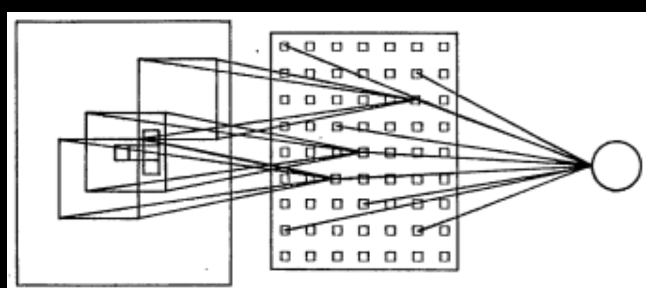
Neocognitron

Feature learning with CNNs



Fukushima 1980

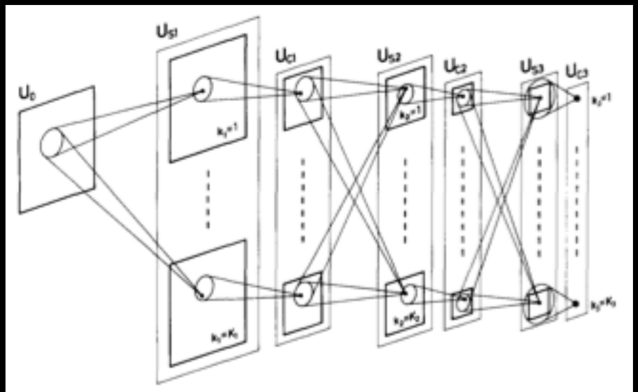
Neocognitron



Rumelhart, Hinton, Williams 1986

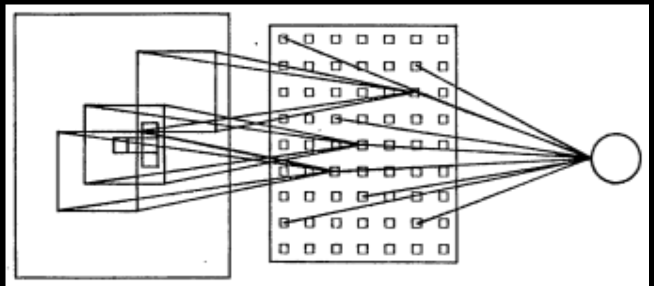
“T” versus “C” problem

Feature learning with CNNs



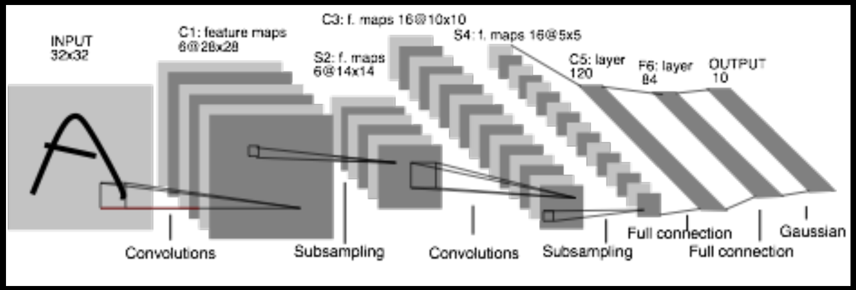
Fukushima 1980

Neocognitron



Rumelhart, Hinton, Williams 1986

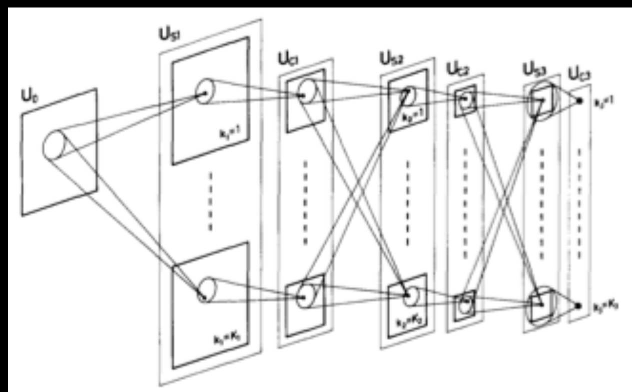
“T” versus “C” problem



LeCun et al. 1989-1998

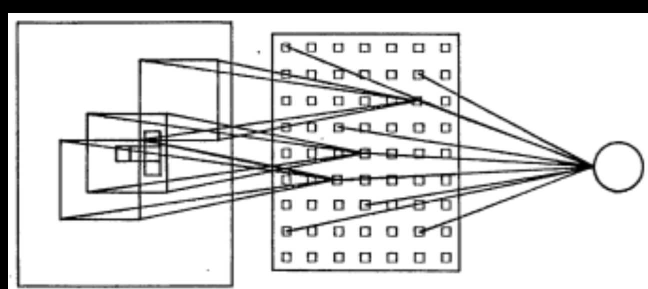
Handwritten digit reading / OCR

Feature learning with CNNs



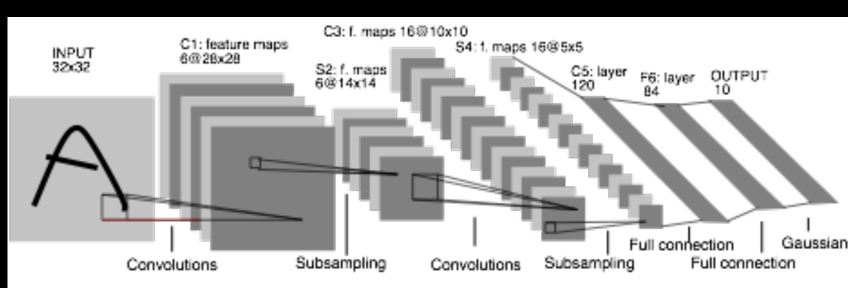
Fukushima 1980

Neocognitron



Rumelhart, Hinton, Williams 1986

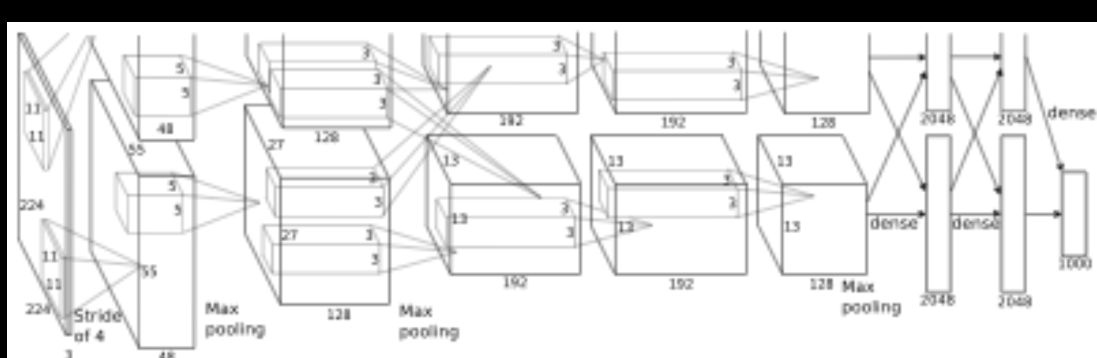
"T" versus "C" problem



LeCun et al. 1989-1998

Handwritten digit reading / OCR

…

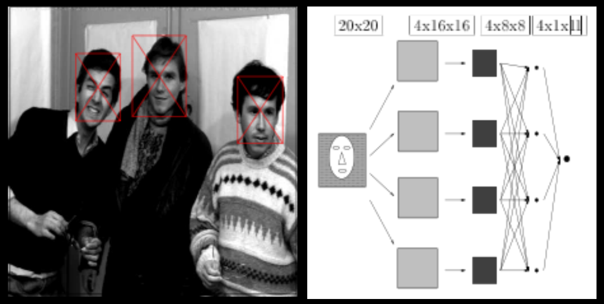


Krizhevksy, Sutskever, Hinton 2012

ImageNet classification breakthrough

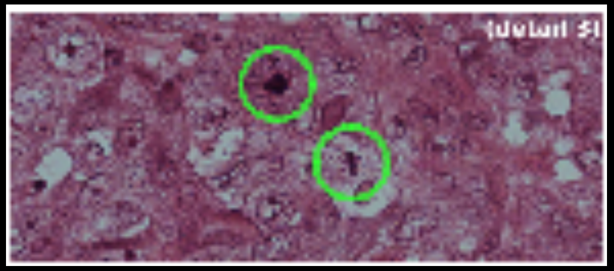
"SuperVision" CNN

CNNs for object detection



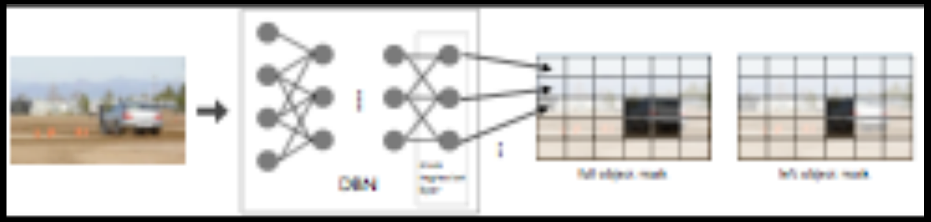
Vaillant, Monrocq, LeCun 1994
Multi-scale face detection

LeCun, Huang, Bottou 2004
NORB dataset



Cireşan et al. 2013
Mitosis detection

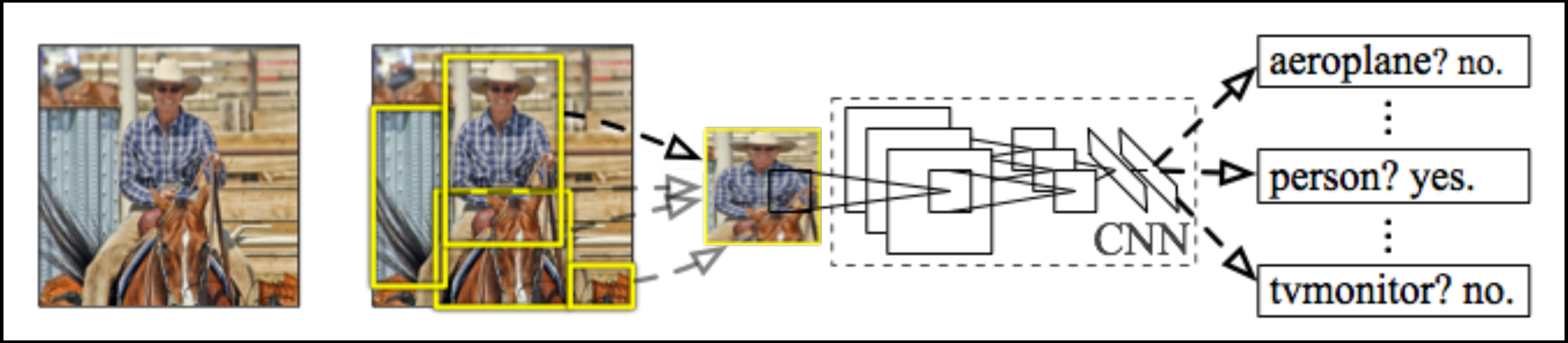
Sermanet et al. 2013
Pedestrian detection



Szegedy, Toshev, Erhan 2013
PASCAL detection (VOC’ 07 mAP 30.

Can we break through the PASCAL plateau with feature learning?

R-CNN: Regions with CNN features



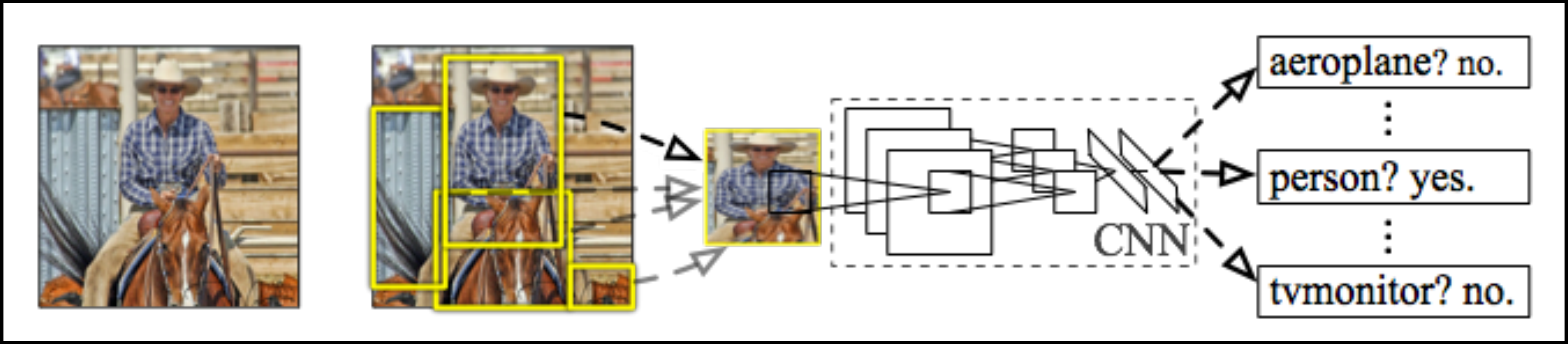
Input image

Extract region proposals (~2k / image)

Compute CNN features

Classify regions (linear SVM)

R-CNN at test time: Step 1

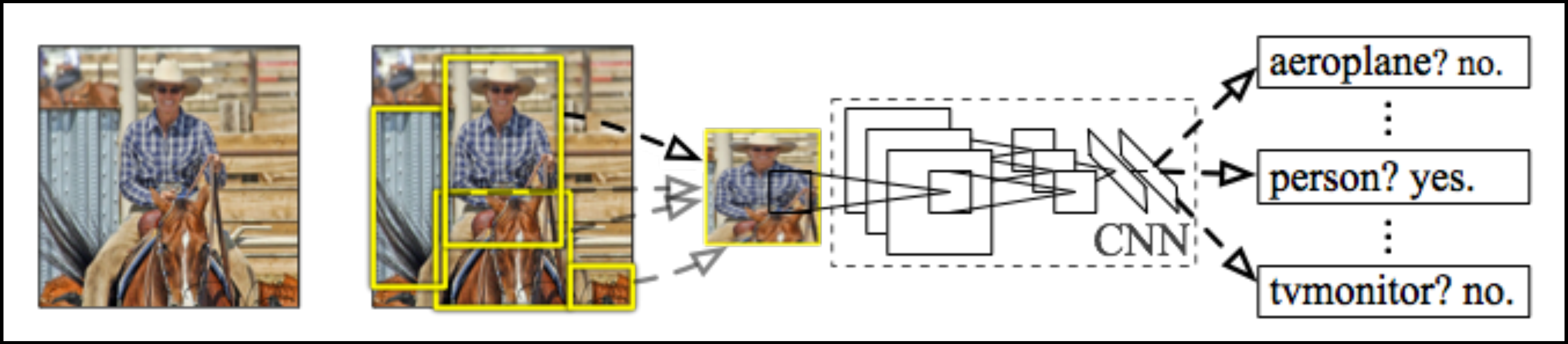


Input image → Extract region proposals (~2k / image)

Proposal-method agnostic, many choices
- Selective Search [van de Sande, Uijlings et al.] (Used in this work)
- Objectness [Alexe et al.]
- Category independent object proposals [Endres & Hoiem]
- CPMC [Carreira & Sminchisescu]

Active area, at this CVPR
- BING [Ming et al.] – fast
- MCG [Arbelaez et al.] – high-quality segmentation

R-CNN at test time: Step 2

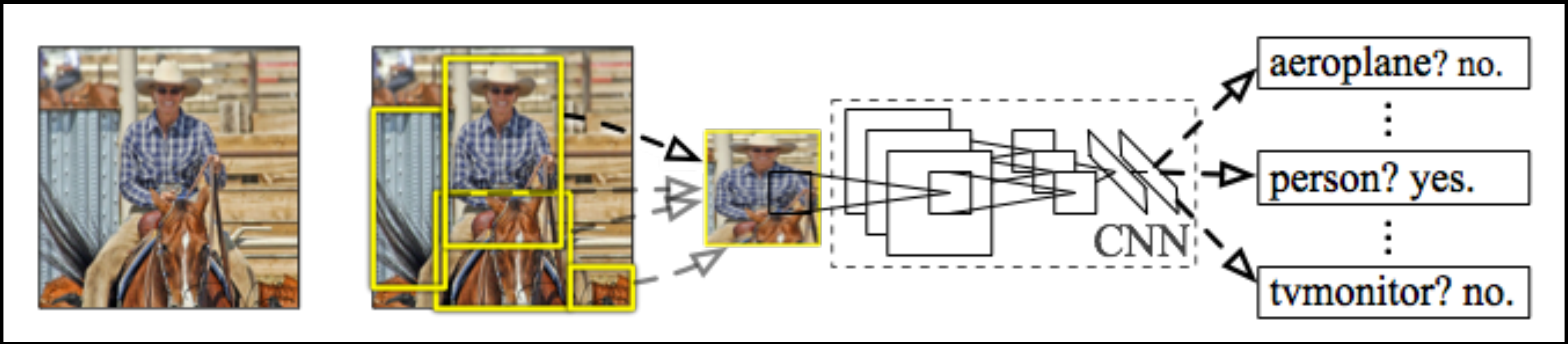


Input image

Extract region proposals (~2k / image)

Compute CNN features

R-CNN at test time: Step 2



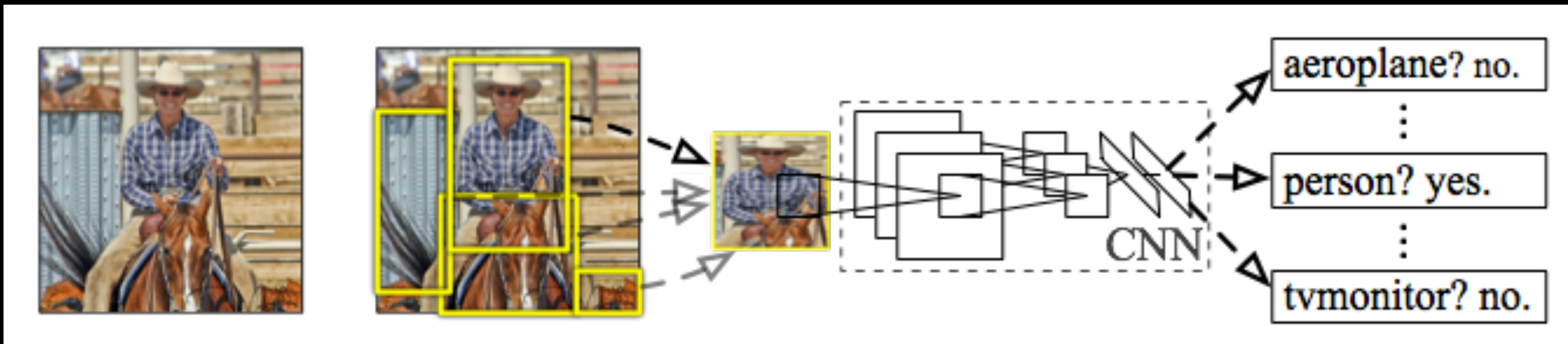
Input image

Extract region proposals (~2k / image)

Compute CNN features

Dilate proposal

R-CNN at test time: Step 2



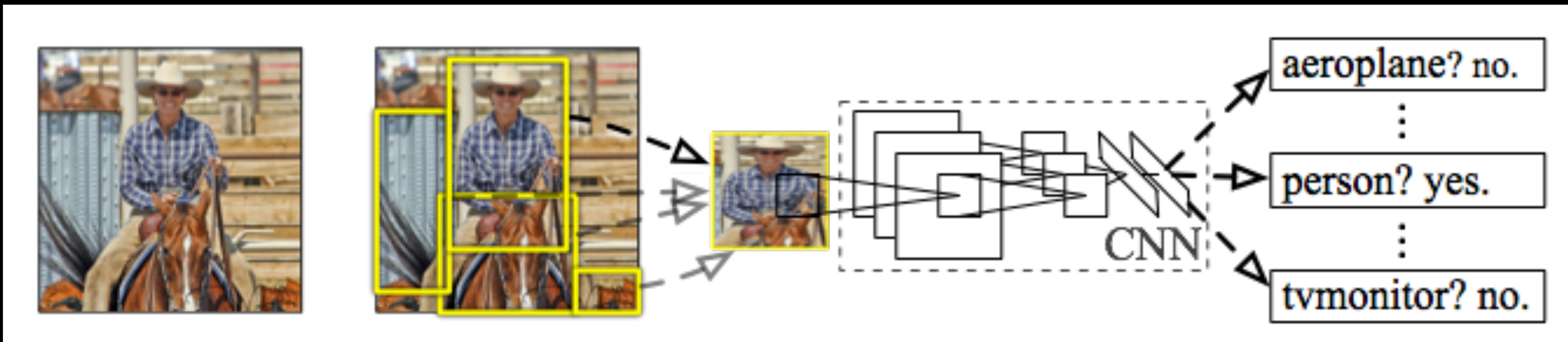
Input image

Extract region proposals (~2k / image)

Compute CNN features

a. Crop

R-CNN at test time: Step 2



Input image

Extract region proposals (~2k / image)

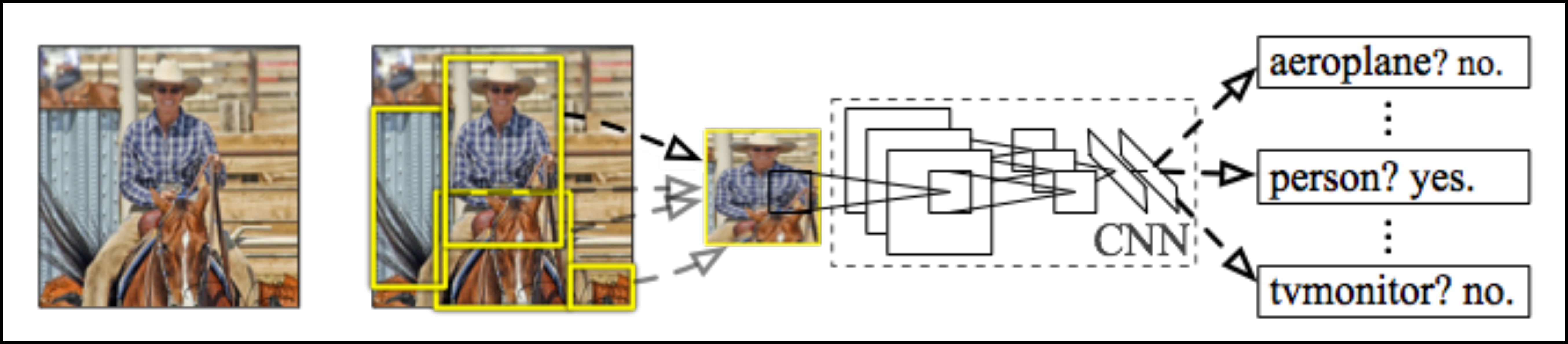
Compute CNN features

227 x 227

a. Crop

b. Scale (anisotropic)

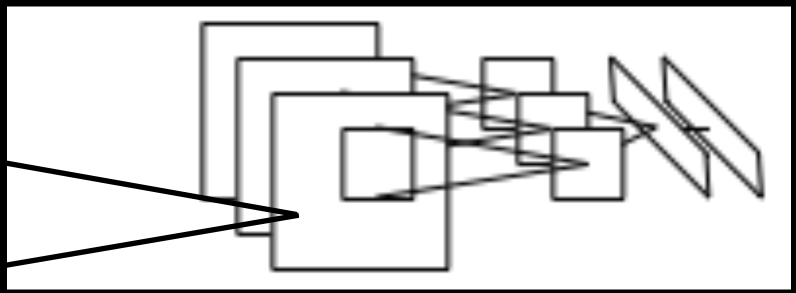
R-CNN at test time: Step 2



Input image

Extract region proposals (~2k / image)

Compute CNN features

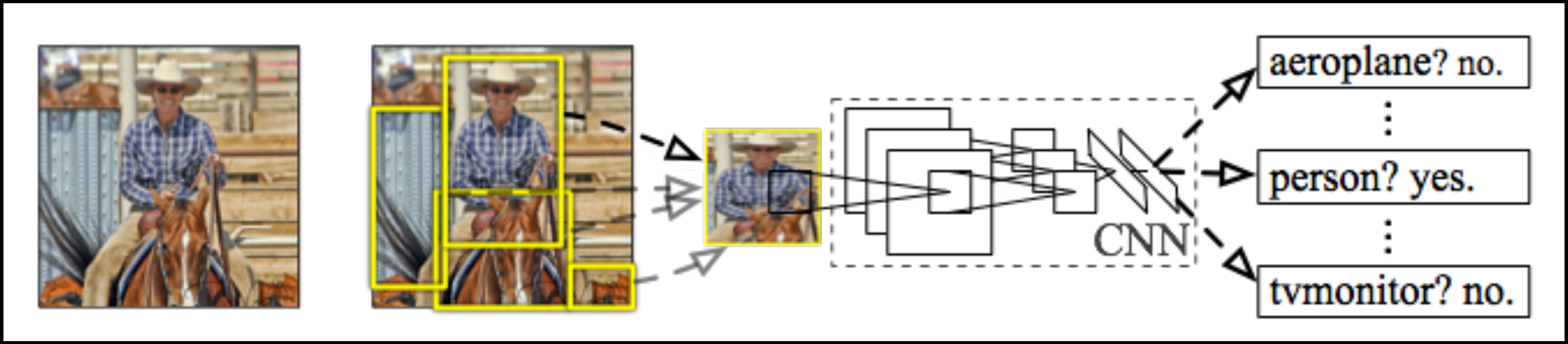


Crop

b. Scale (anisotropic)

c. Forward propagate
Output: “fc_7” features

R-CNN at test time: Step 3

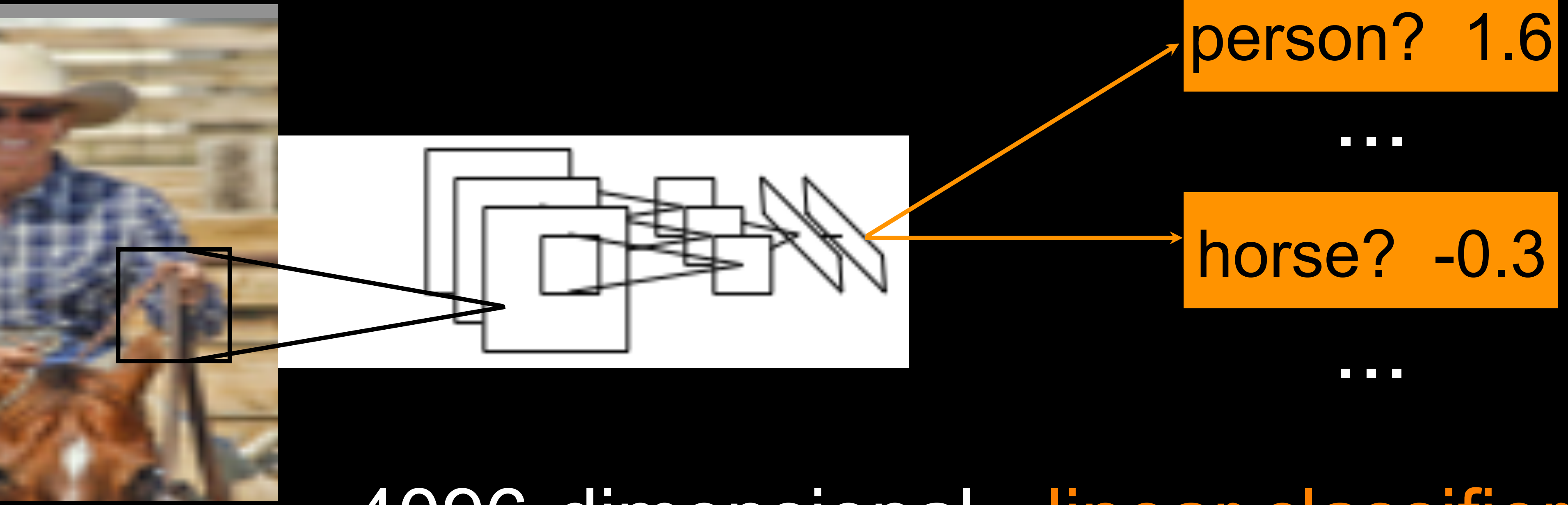


Input image

Extract region proposals (~2k / image)

Compute CNN features

Classify regions



proposal

4096-dimensional fc_7 feature vector

linear classifiers (SVM or softmax)

Step 4: Object proposal refinement

Linear regression

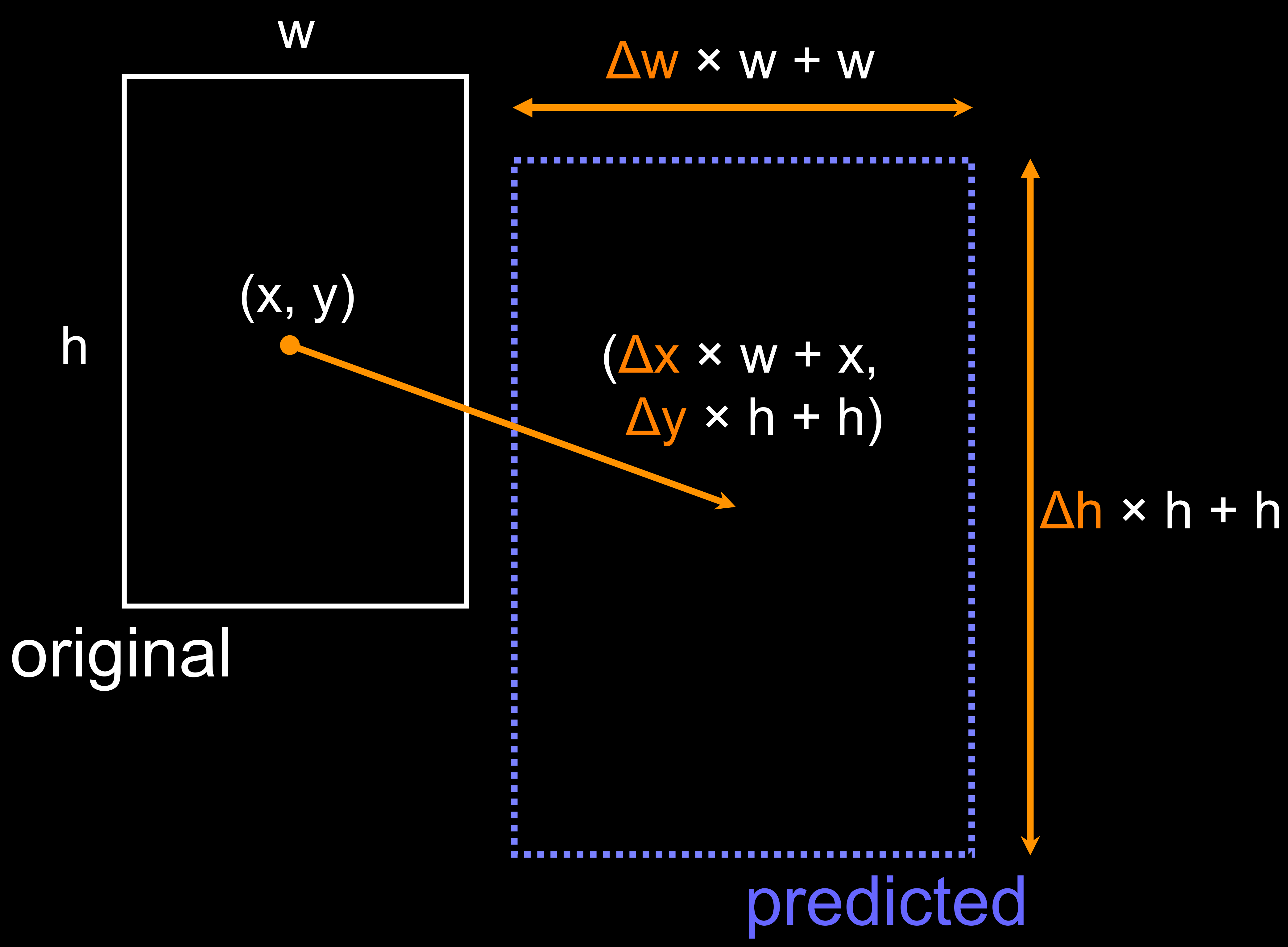
on CNN features

Original proposal

Predicted object bounding box

Bounding-box regression

Bounding-box regression



R-CNN results on PASCAL

	VOC 2007	VOC 2010
DPM v5 (Girshick et al. 2011)	33.7%	29.6%
UVA sel. search (Uijlings et al. 2013)		35.1%
Regionlets (Wang et al. 2013)	41.7%	39.7%
SegDPM (Fidler et al. 2013)		40.4%

Reference systems

metric: mean average precision (higher is better)

R-CNN results on PASCAL

	VOC 2007	VOC 2010
DPM v5 (Girshick et al. 2011)	33.7%	29.6%
UVA sel. search (Uijlings et al. 2013)		35.1%
Regionlets (Wang et al. 2013)	41.7%	39.7%
SegDPM (Fidler et al. 2013)		40.4%
R-CNN	54.2%	50.2%
R-CNN + bbox regression	58.5%	53.7%

metric: mean average precision (higher is better)

Top bicycle FPs (AP = 72.8%)

False positive #15

bicycle (bg): ov=0.00 1–r=0.44

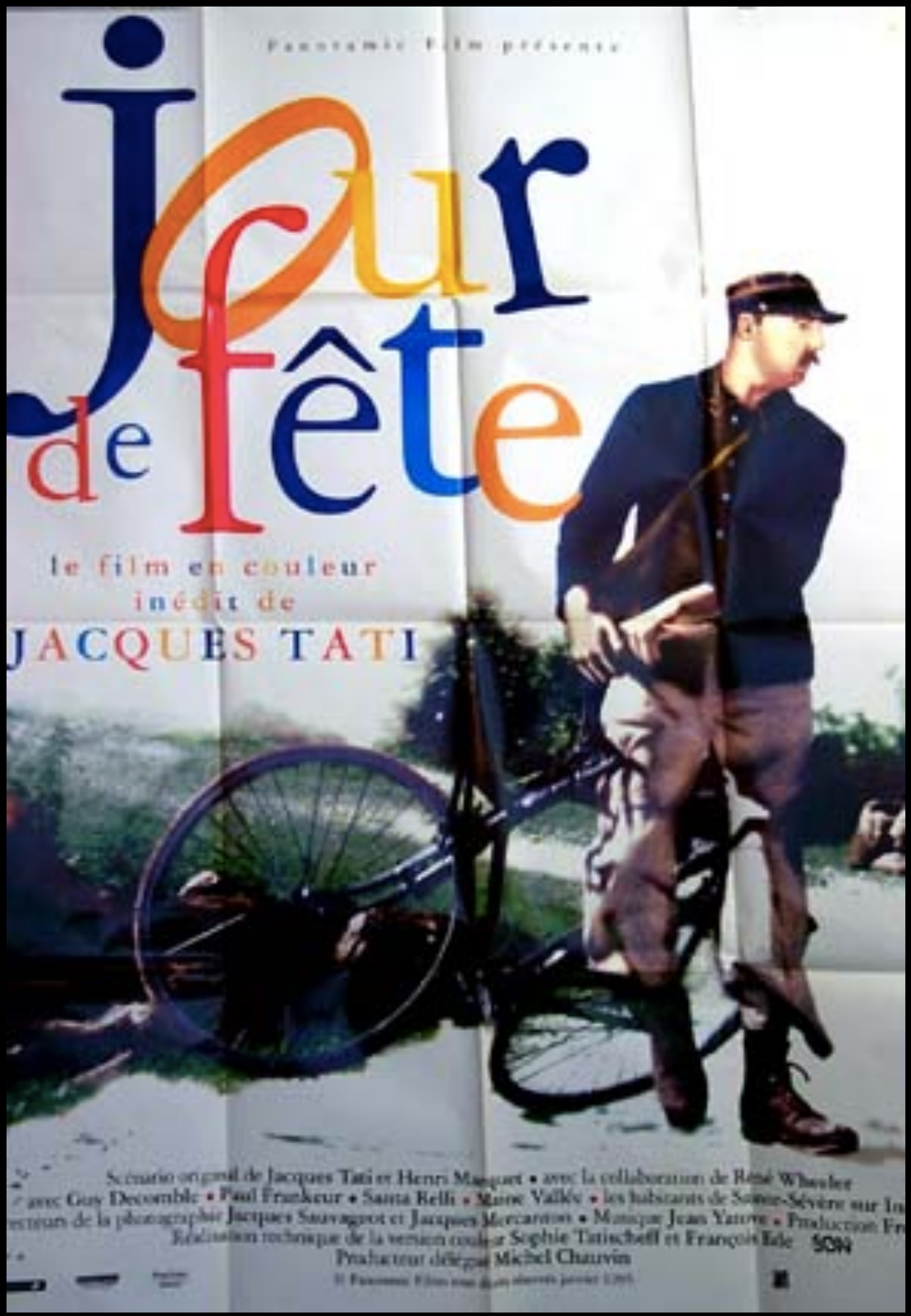
False positive #15

bicycle (bg): ov=0.00 1−r=0.44

(zoom)

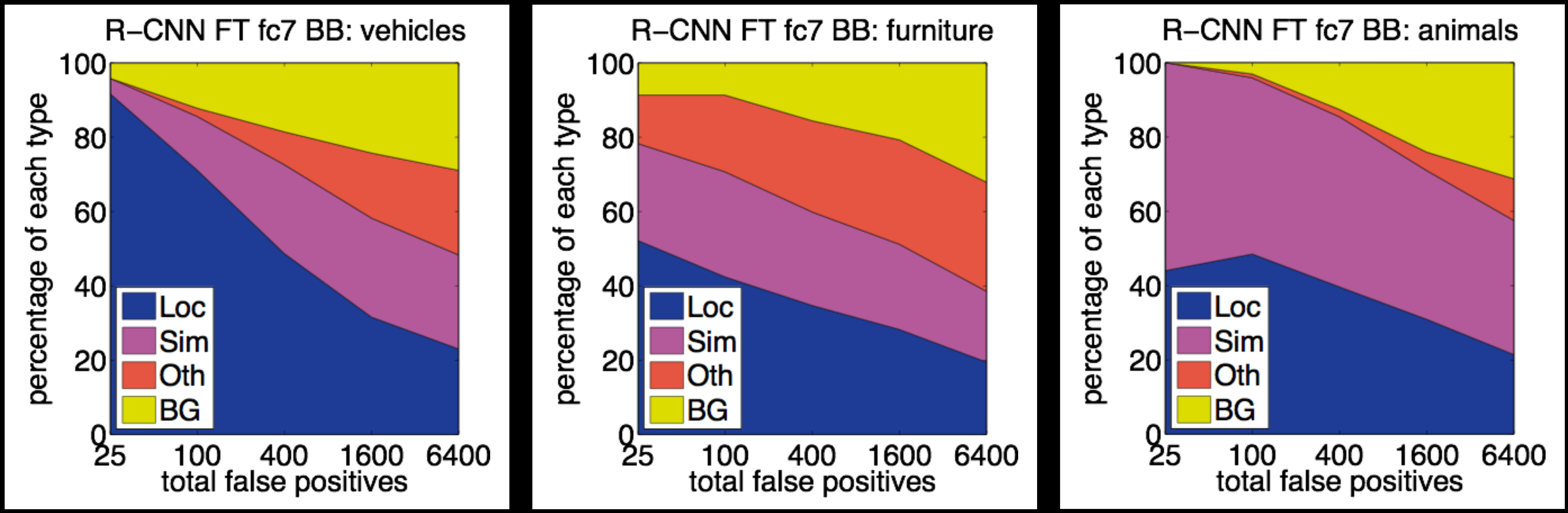
Unannotated bicycle

False positive #15



1949 French comedy by Jacques Tati

False positive type distribution



Loc = localization

Sim = similar classes

Oth = other / dissimilar classes

BG = background

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

Training R-CNN

Bounding-box labeled detection data is scarce

Key insight:

Use supervised pre-training on a data-rich auxiliary task and transfer to detection

ImageNet LSVR Challenge

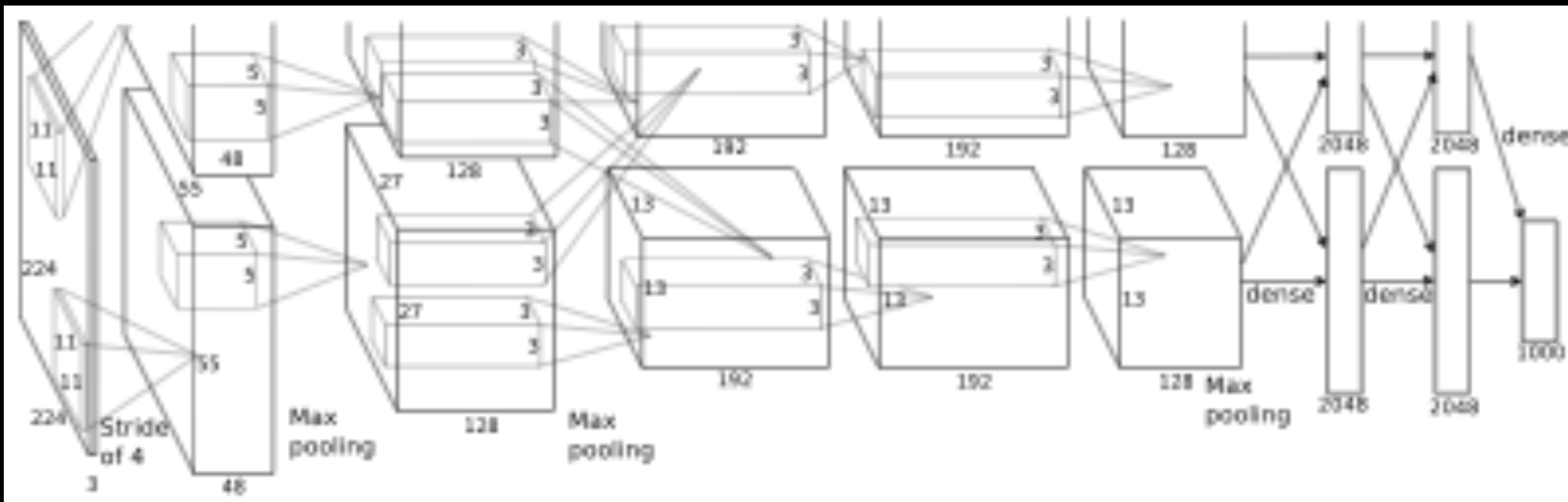
- Image classification
  (not detection)
- 1000 classes
  (vs. 20)
- 1.2 million training labels
  (vs. 25k)

bus anywhere?

[Deng et al. CVPR' 09]

ILSVRC 2012 winner

"SuperVision" Convolutional Neural Network (CNN)



input ⟵ 5 convolutional layers ⟶ fully connected

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

Impressive ImageNet results!

1000-way image classification

	Top-5 error
Fisher Vectors (ISI)	26.2%
5 SuperVision CNNs	16.4%
7 SuperVision CNNs	15.3%

now: ~12%

metric: c s better)

But... does it generalize to other datasets and tasks?

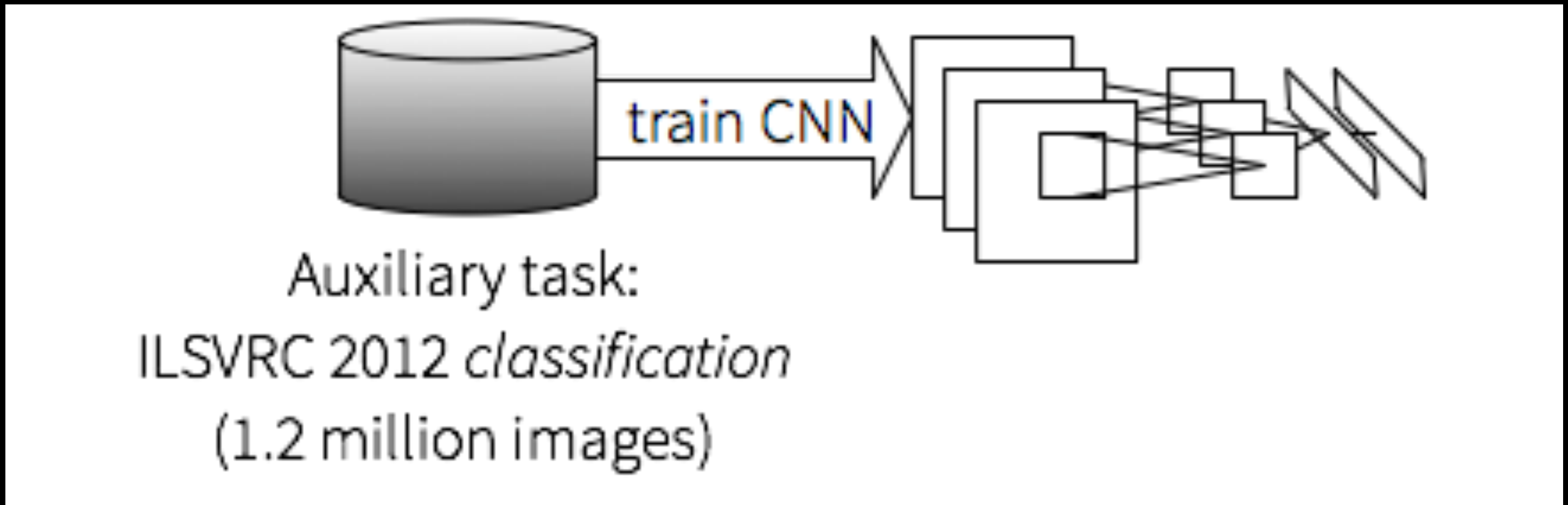
[See also: DeCAF. Donahue et al., ICML 2014.]

Spirited debate at ECCV 2012

R-CNN training: Step 1

Supervised pre-training

Train a SuperVision CNN* for the 1000-way ILSVRC image classification task

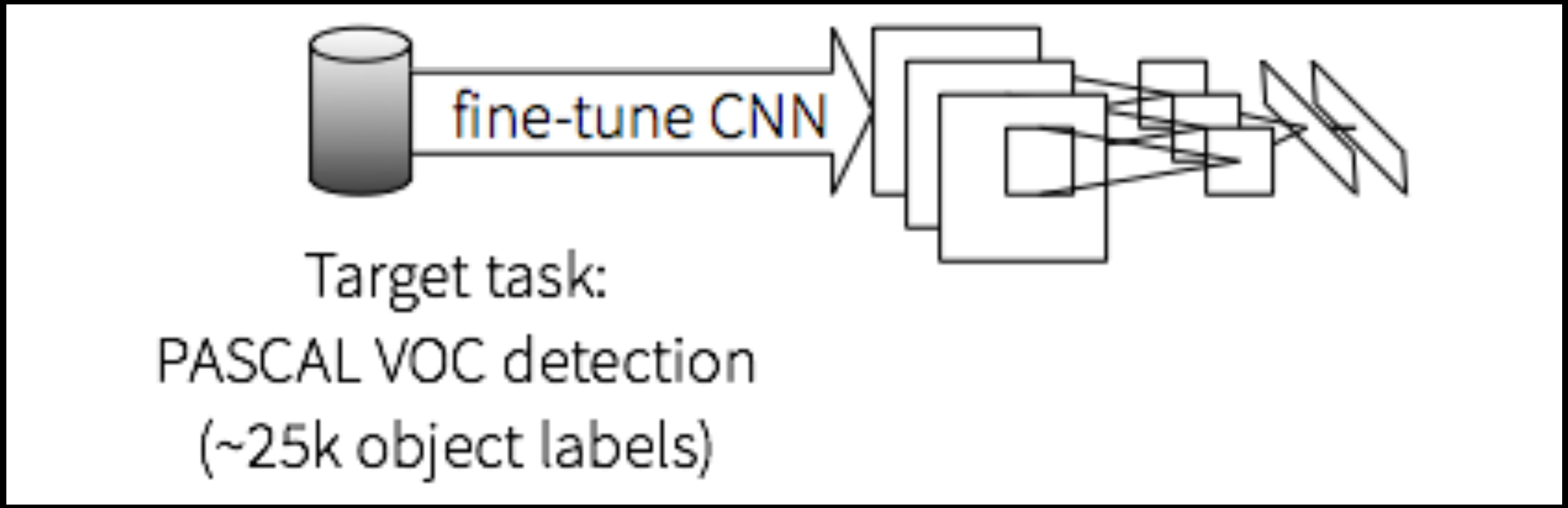


*Network from Krizhevsky, Sutskever & Hinton. NIPS 2012
Also called "AlexNet"

R-CNN training: Step 2

Fine-tune the CNN for detection

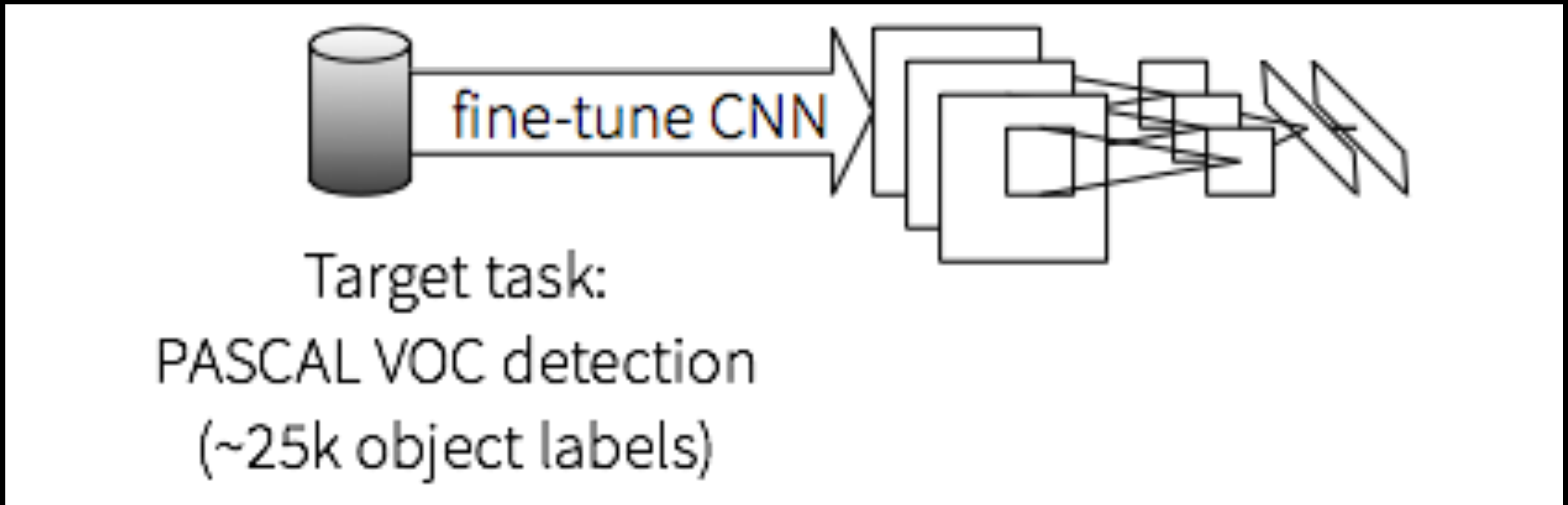
Transfer the representation learned for ILSVRC classification to PASCAL (or ImageNet detection)



R-CNN training: Step 2

Fine-tune the CNN for detection

Transfer the representation learned for ILSVRC classification to PASCAL (or ImageNet detection)



Try Caffe! http://caffe.berkeleyvision.org

- Clean & fast CNN library in C++ with Python and MATLAB interfaces
- Used by R-CNN for training, fine-tuning, and feature computation

R-CNN training: Step 3

Train detection SVMs
(With the softmax classifier from fine-tuning mAP decreases from 54% to 51%)

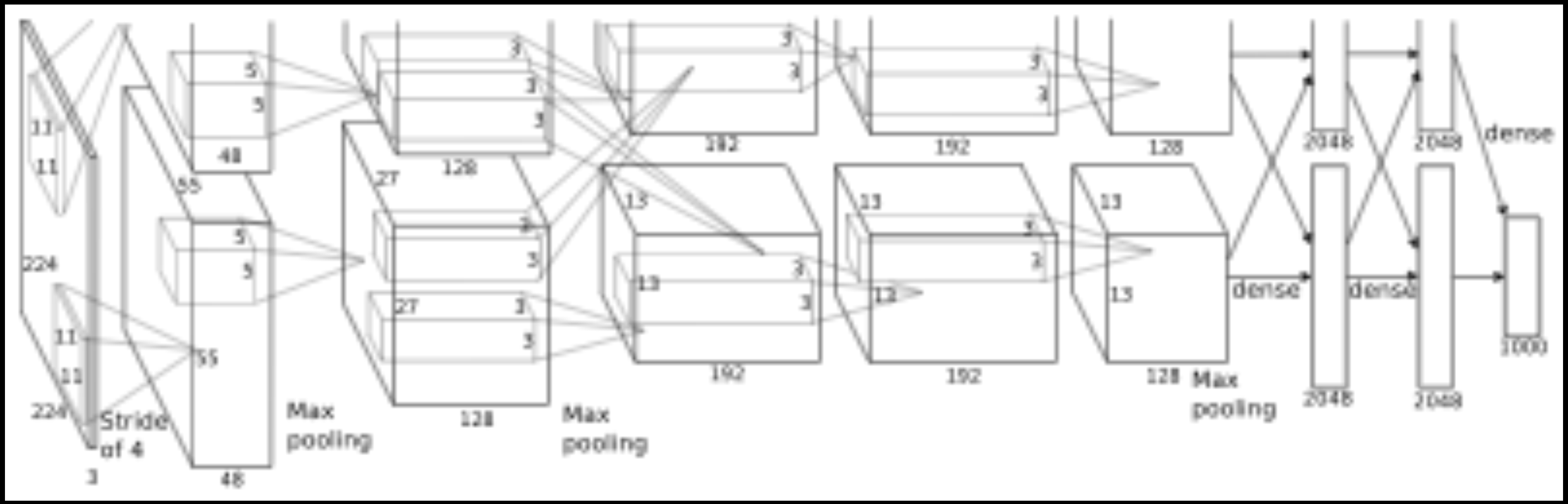
Compare with fine-tuned R-CNN

	VOC 2007	VOC 2010
Regionlets (Wang et al. 2013)	41.7%	39.7%
SegDPM (Fidler et al. 2013)		40.4%
R-CNN $pool_5$	44.2%	
R-CNN fc_6	46.2%	
R-CNN fc_7	44.7%	
R-CNN FT $pool_5$	47.3%	
R-CNN FT fc_6	53.1%	
R-CNN FT fc_7	54.2%	50.2%

fine-tuned

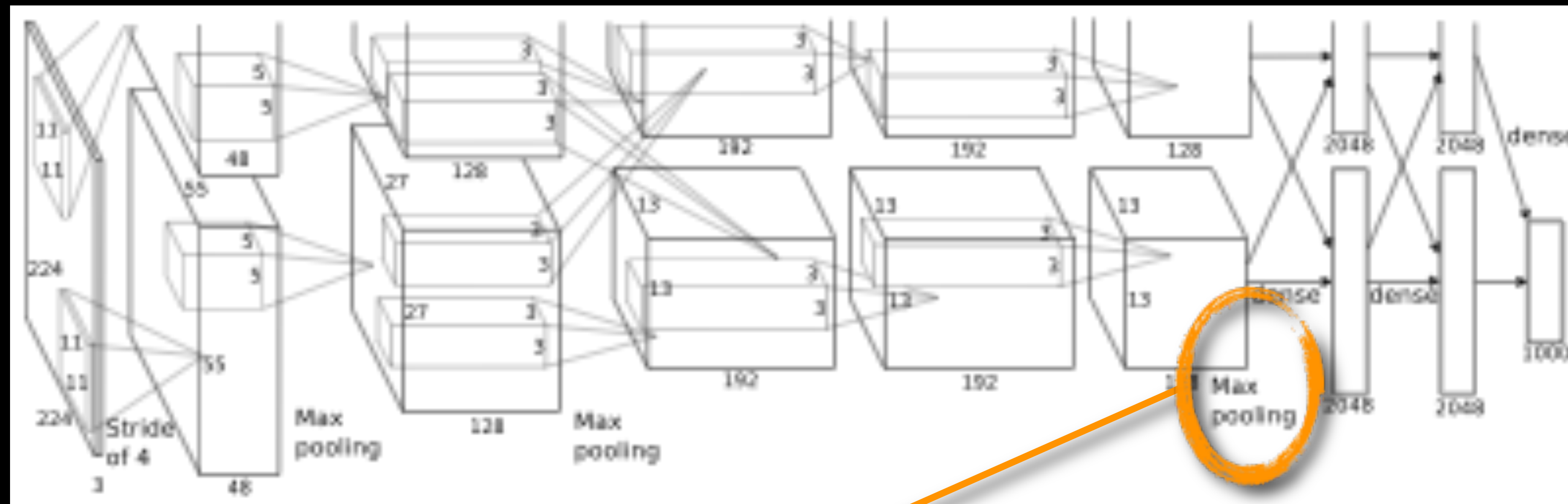
metric: mean average precision (higher is better)

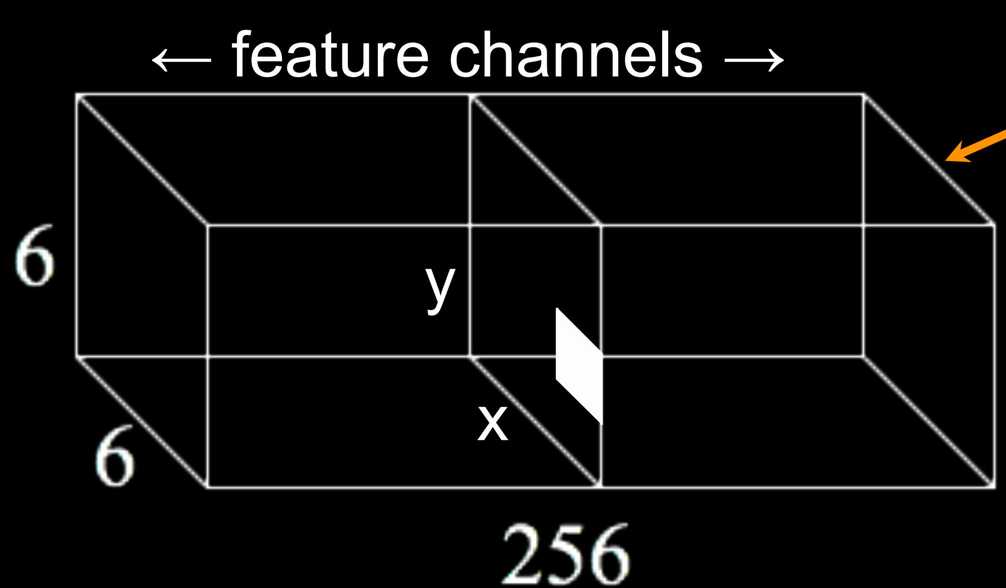
What did the network learn?



What did the network learn?

"stimulus"





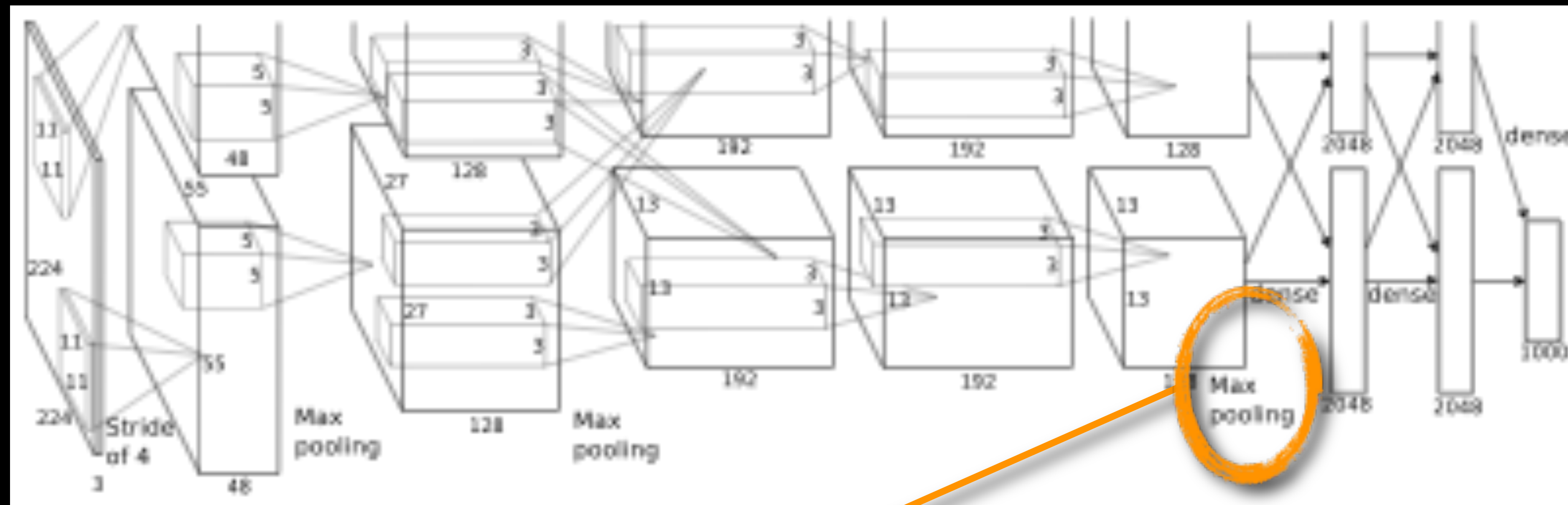
pool$_5$ feature map

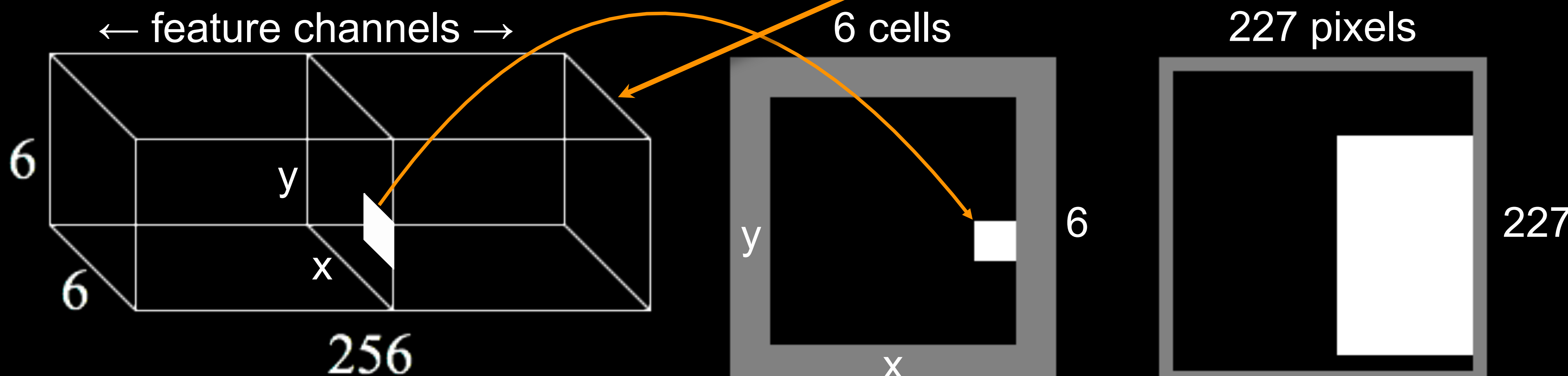
y

x

What did the network learn?

“stimulus”



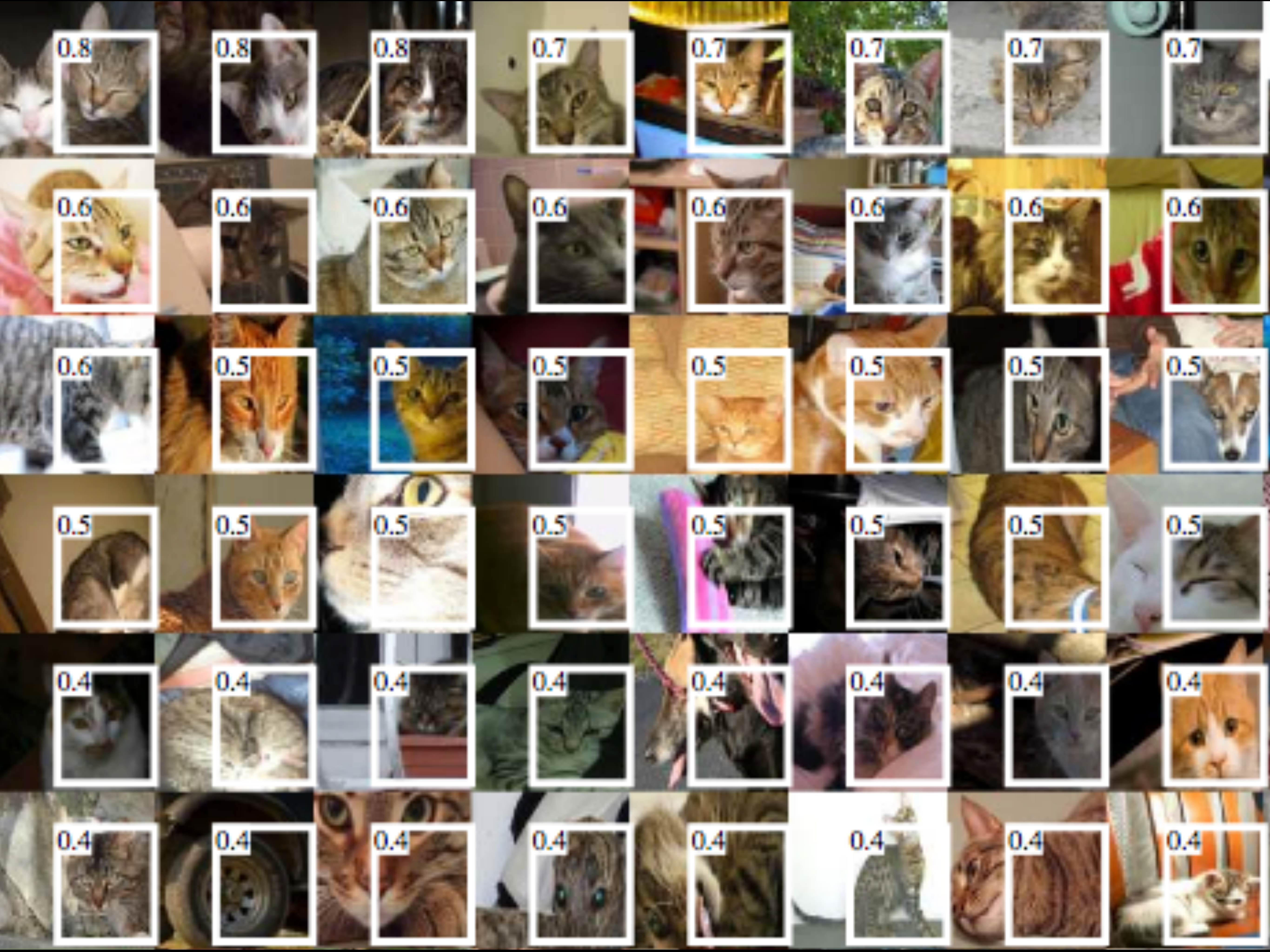


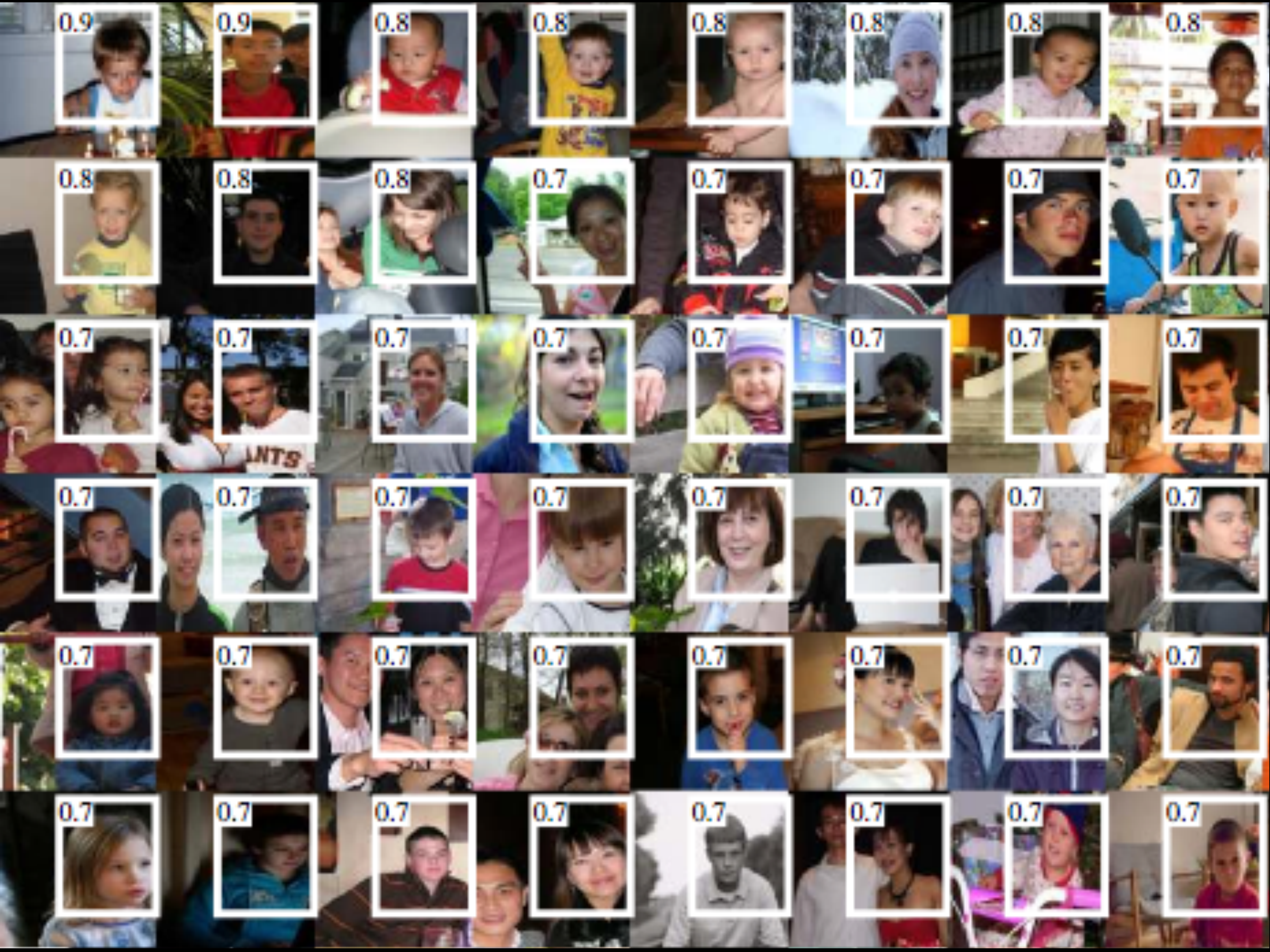
pool$_5$ feature map

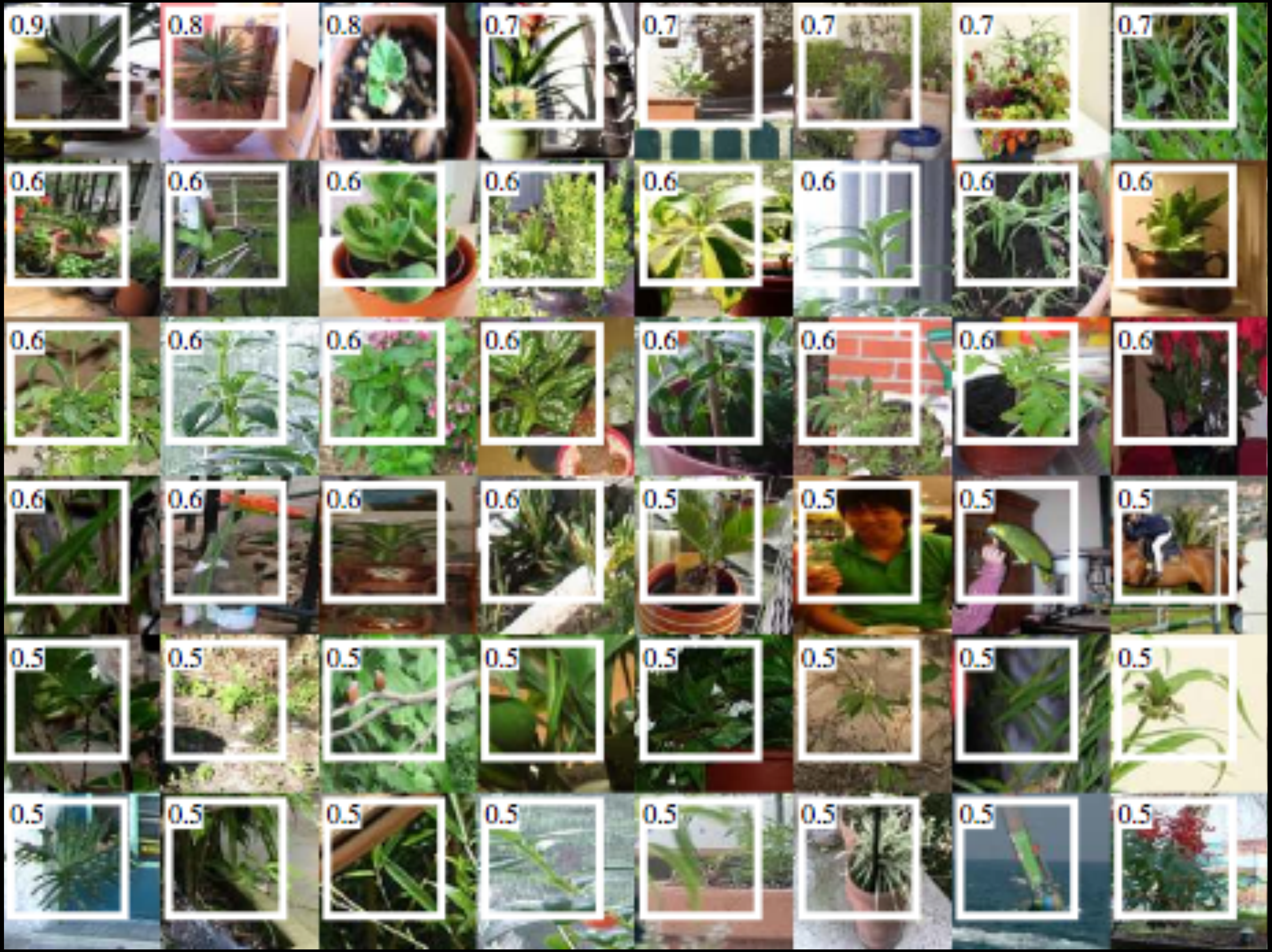
feature position

receptive field

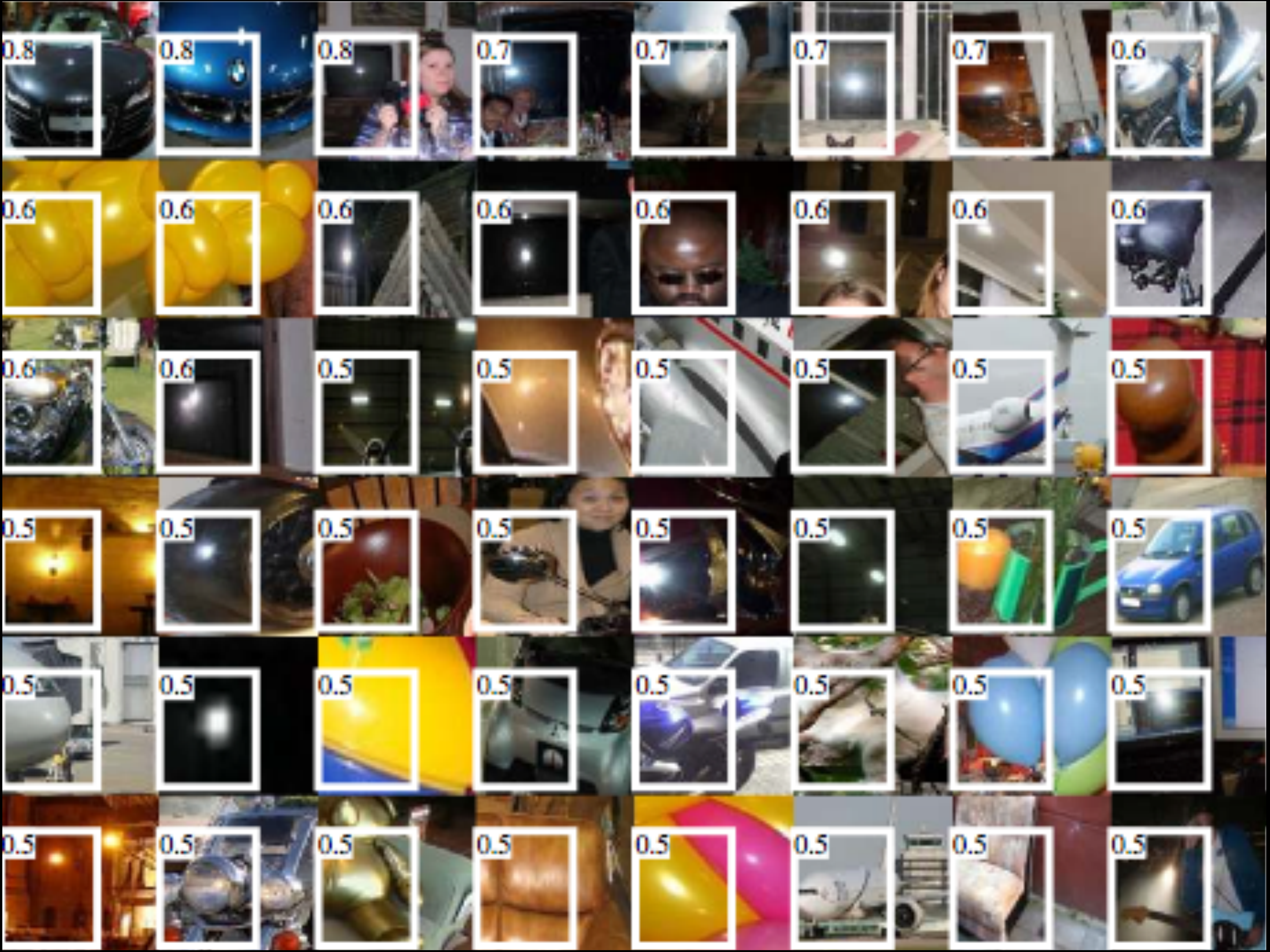
Visualize images that activate pool$_5$ a feature

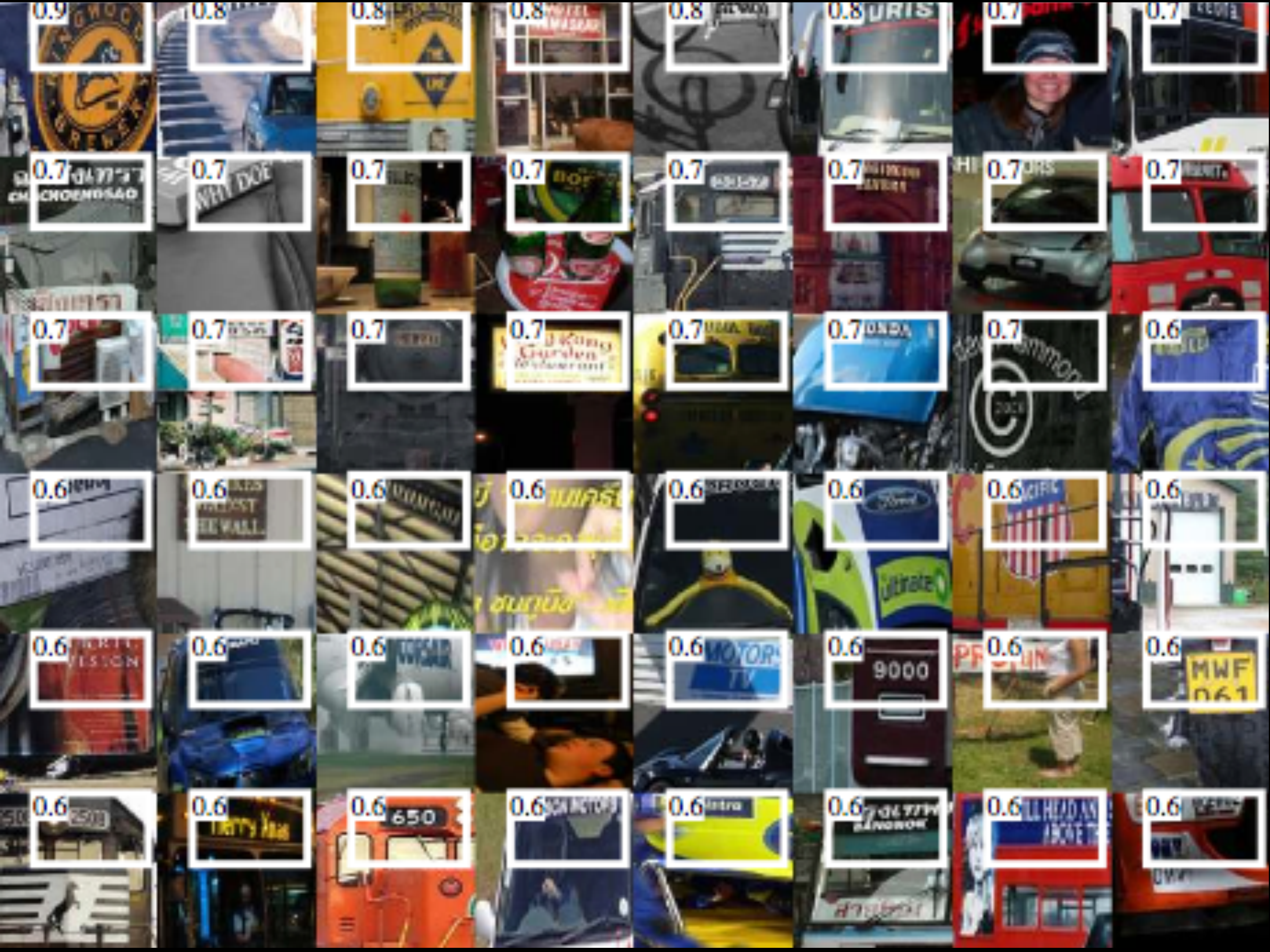

0.8
0.8
0.8
0.7
0.7
0.7
0.7
0.7
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.4
0.4
0.4
0.4
0.4
0.4
0.4
0.4
0.4
0.4
0.4
0.4
0.4
0.4
0.4
0.4


0.9 0.9 0.8 0.8 0.8 0.8 0.8 0.8
0.8 0.8 0.8 0.7 0.7 0.7 0.7 0.7
0.7 0.7 0.7 0.7 0.7 0.7 0.7 0.7
0.7 0.7 0.7 0.7 0.7 0.7 0.7 0.7
0.7 0.7 0.7 0.7 0.7 0.7 0.7 0.7
0.7 0.7 0.7 0.7 0.7 0.7 0.7 0.7



0.9
0.8
0.8
0.7
0.7
0.7
0.7
0.7
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.6
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5


0.8 0.8 0.8 0.7 0.7 0.7 0.7 0.6
0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6
0.6 0.6 0.5 0.5 0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5


0.9 0.8 0.8 0.8 0.8 0.8 0.7 0.7
0.7 0.7 0.7 0.7 0.7 0.7 0.7 0.7
0.7 0.7 0.7 0.7 0.7 0.7 0.7 0.6
0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6
0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6
0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6
9000
MWF
650

Take away

- Dramatically better PASCAL mAP

- R-CNN outperforms other CNN-based methods on ImageNet detection

- Detection speed is manageable (~11s / image)

- Scales very well with number of categories (30ms for 20 → 200 classes!)

- R-CNN is simple and completely open source

Semantic segmentation

CPMC segments
(Carreira & Sminchisescu)

full

fg

Semantic segmentation

CPMC segments
(Carreira & Sminchisescu)

full

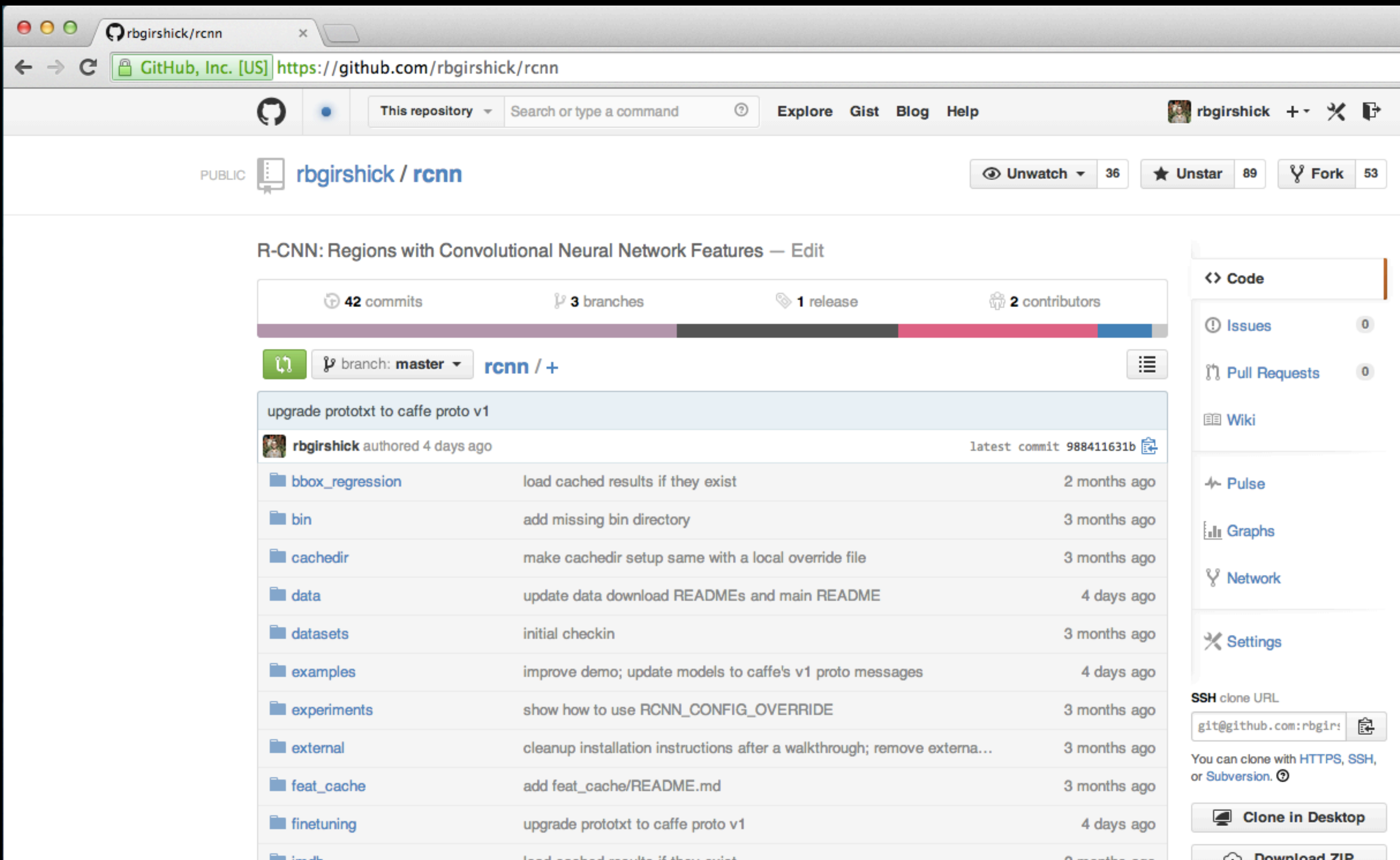
fg

	VOC 2011 test
Bonn second-order pooling (Carreira et al.)	**47.6%**
R-CNN fc_6 full+fg (no fine-tuning)	**47.9%**

Improved to 50.5% in our upcoming ECCV’ 14 work:
Simultaneous Localization and Detection. Hariharan et al.

Get the code and models!

bit.ly/rcnn-cvpr14



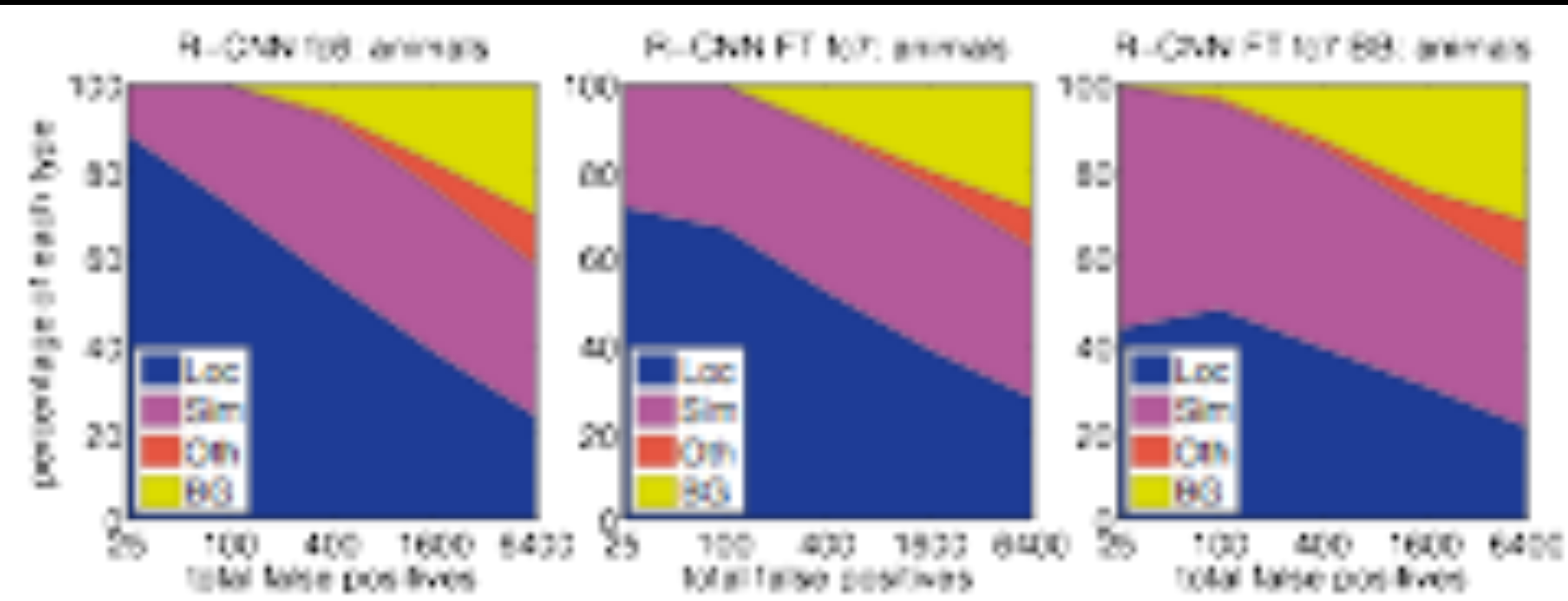
Supplementary slides follow

Pre-trained CNN + SVMs (no FT)

	VOC 2007	VOC 2010
Regionlets (Wang et al. 2013)	41.7%	39.7%
SegDPM (Fidler et al. 2013)		40.4%
R-CNN $pool_5$	44.2%	
R-CNN fc_6	46.2%	
R-CNN fc_7	44.7%	

metric: mean average precision (higher is better)

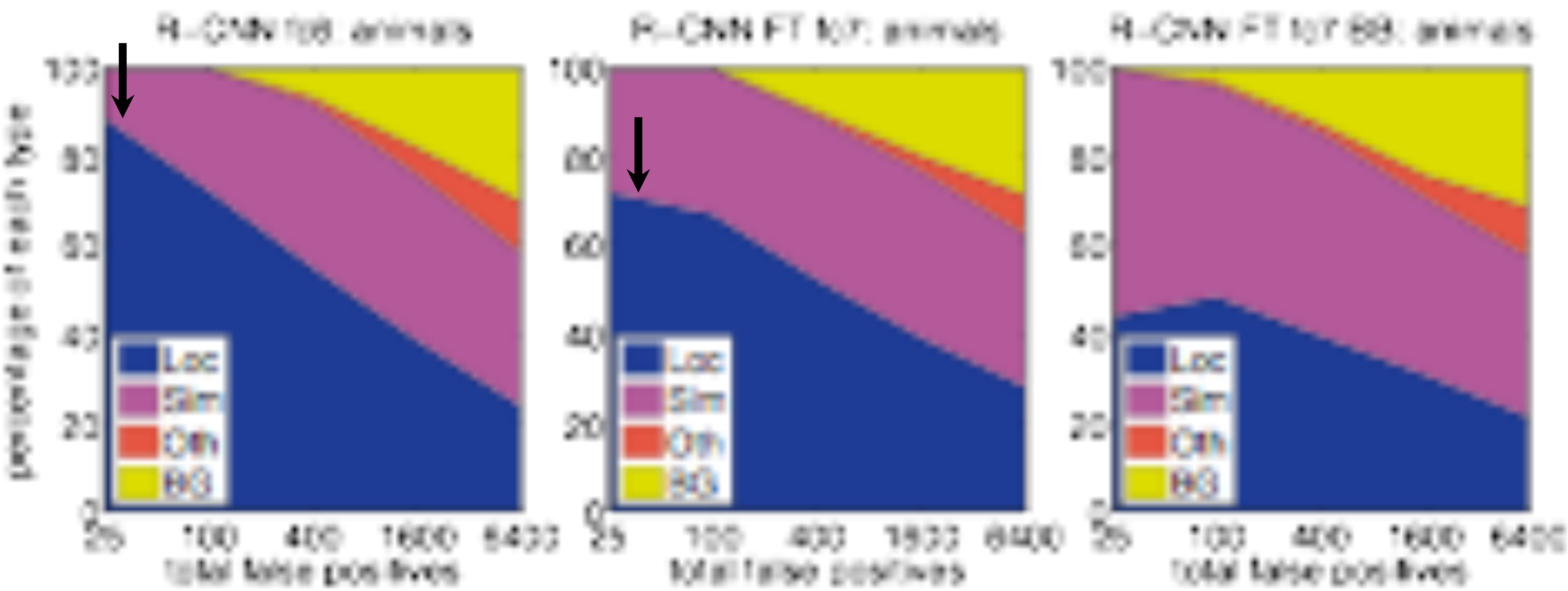
False positive analysis



No fine-tuning

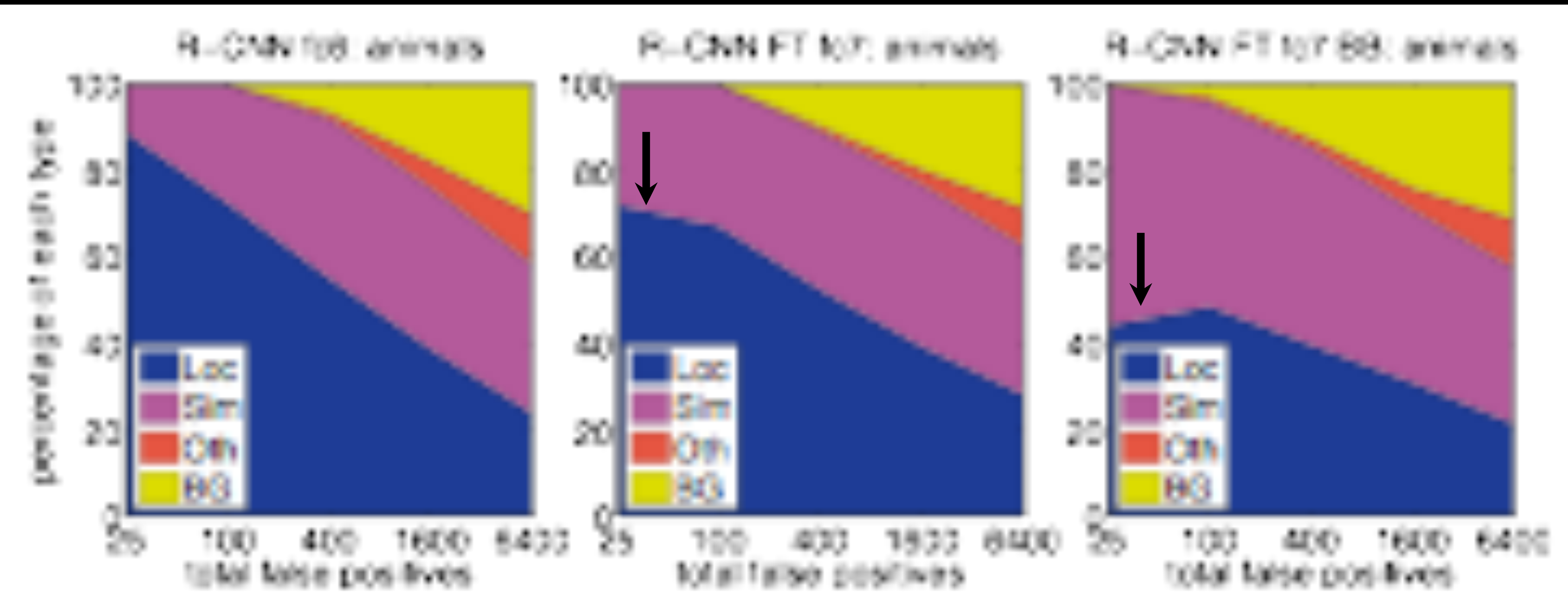
Analysis software: D. Hoiem, Y. Chodpathumwan, and Q. Dai. “Diagnosing Error in Object Detectors.” ECCV, 2012.

False positive analysis



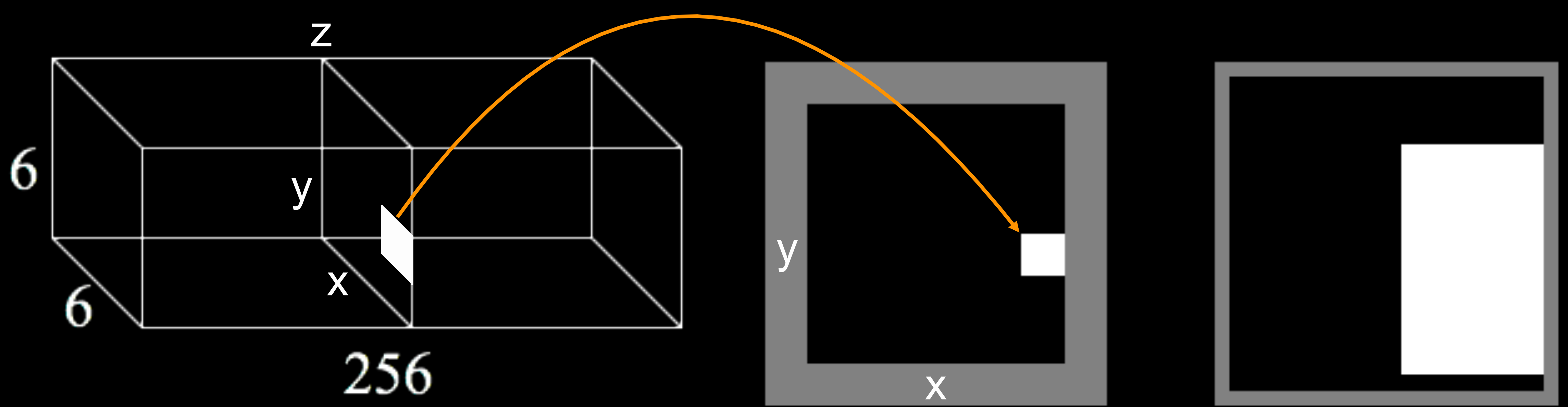
After fine-tuning

False positive analysis



After bounding-box regression

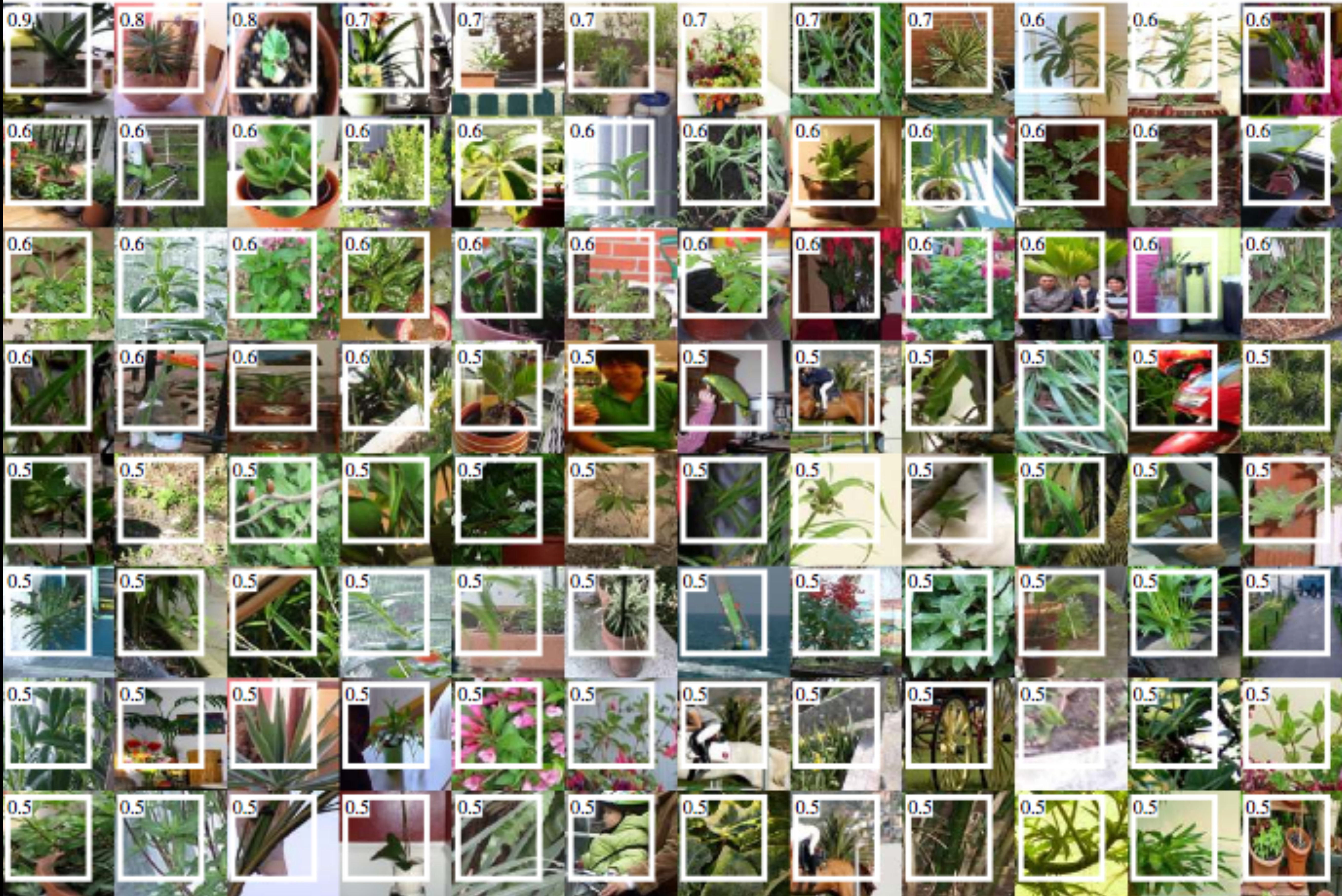
What did the network learn?



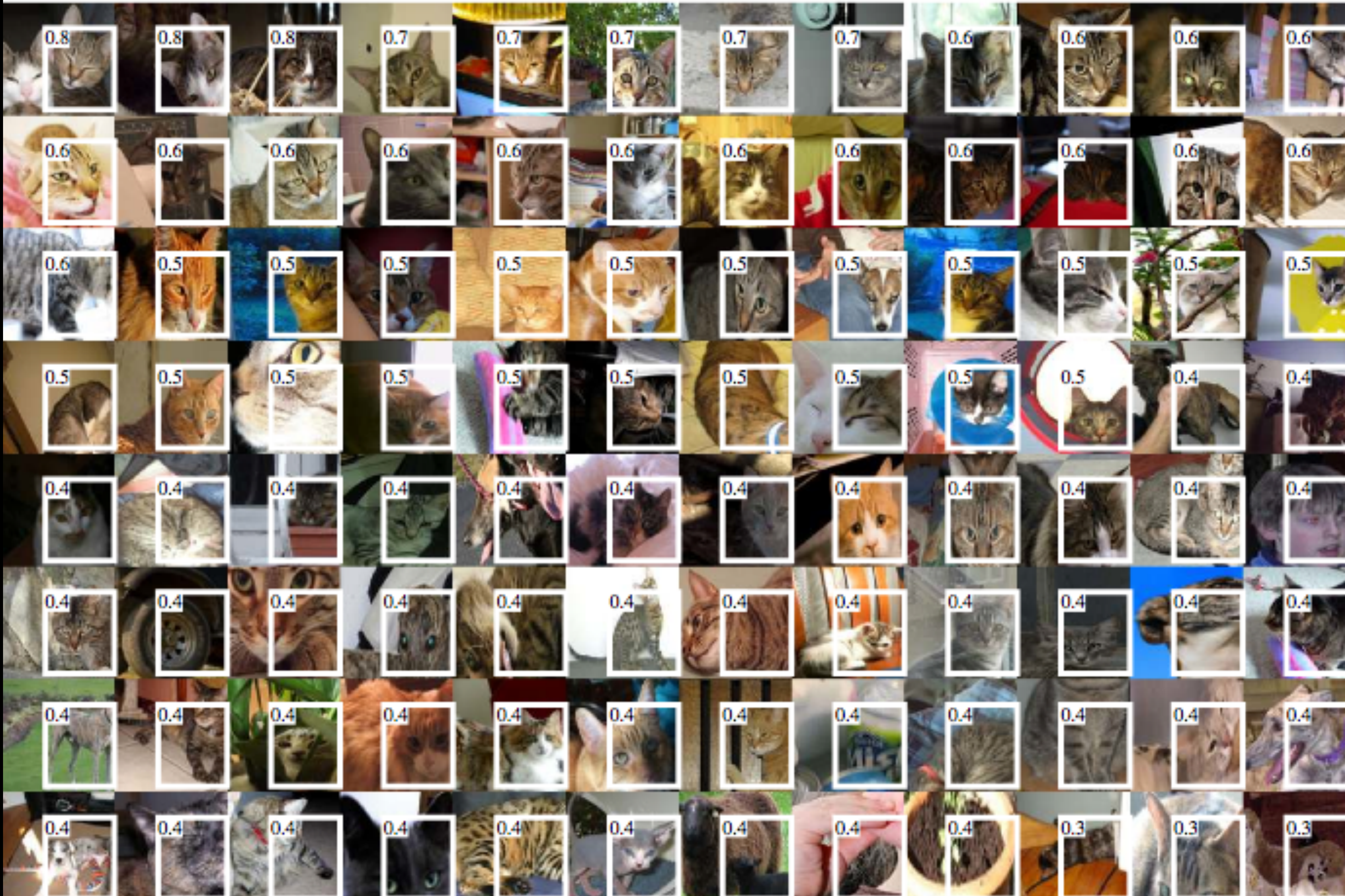
pool$_5$ “data cube” unit position receptive field

Visualize pool$_5$ units

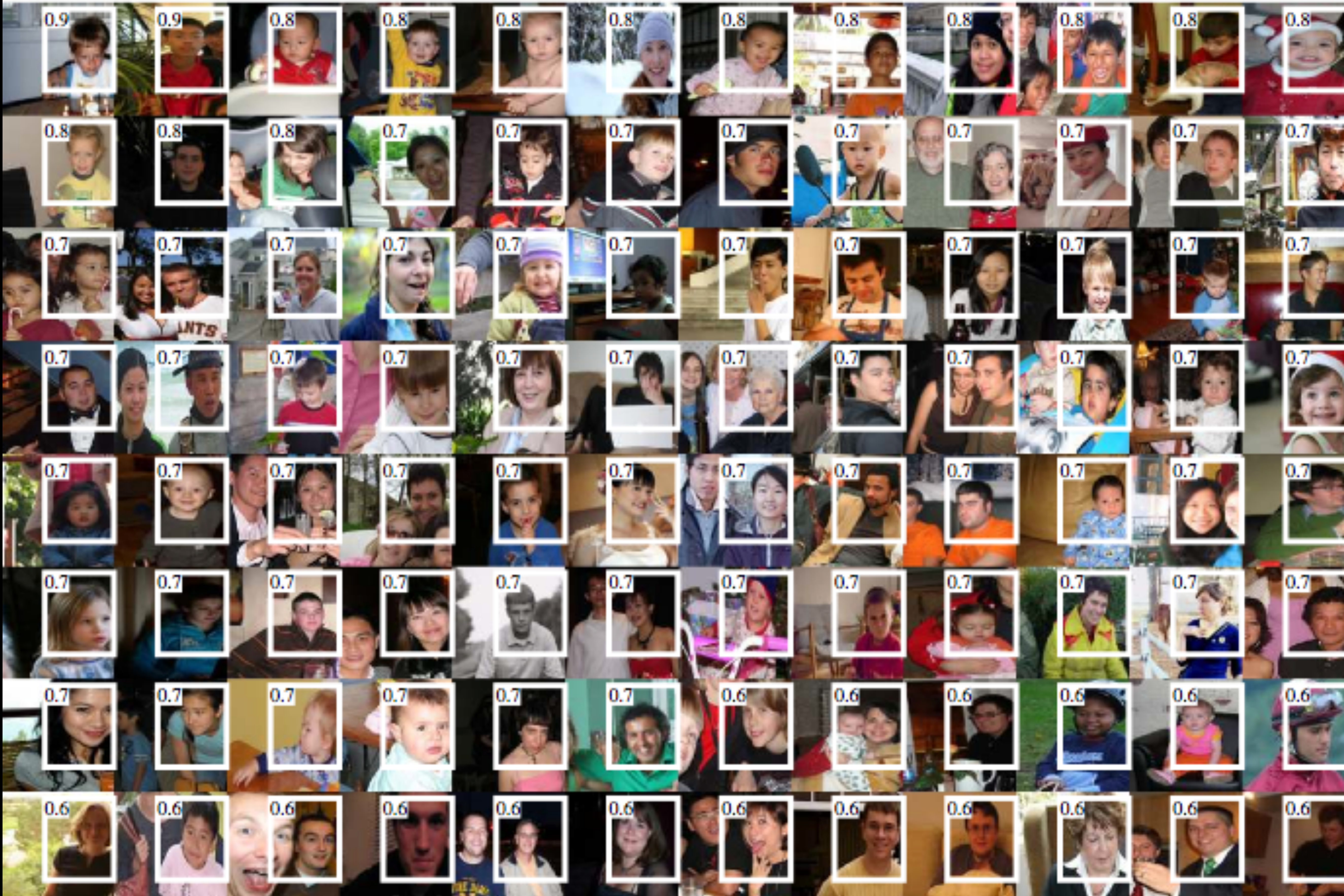
pool5 feature: (3,3,42) (top 1 – 96)



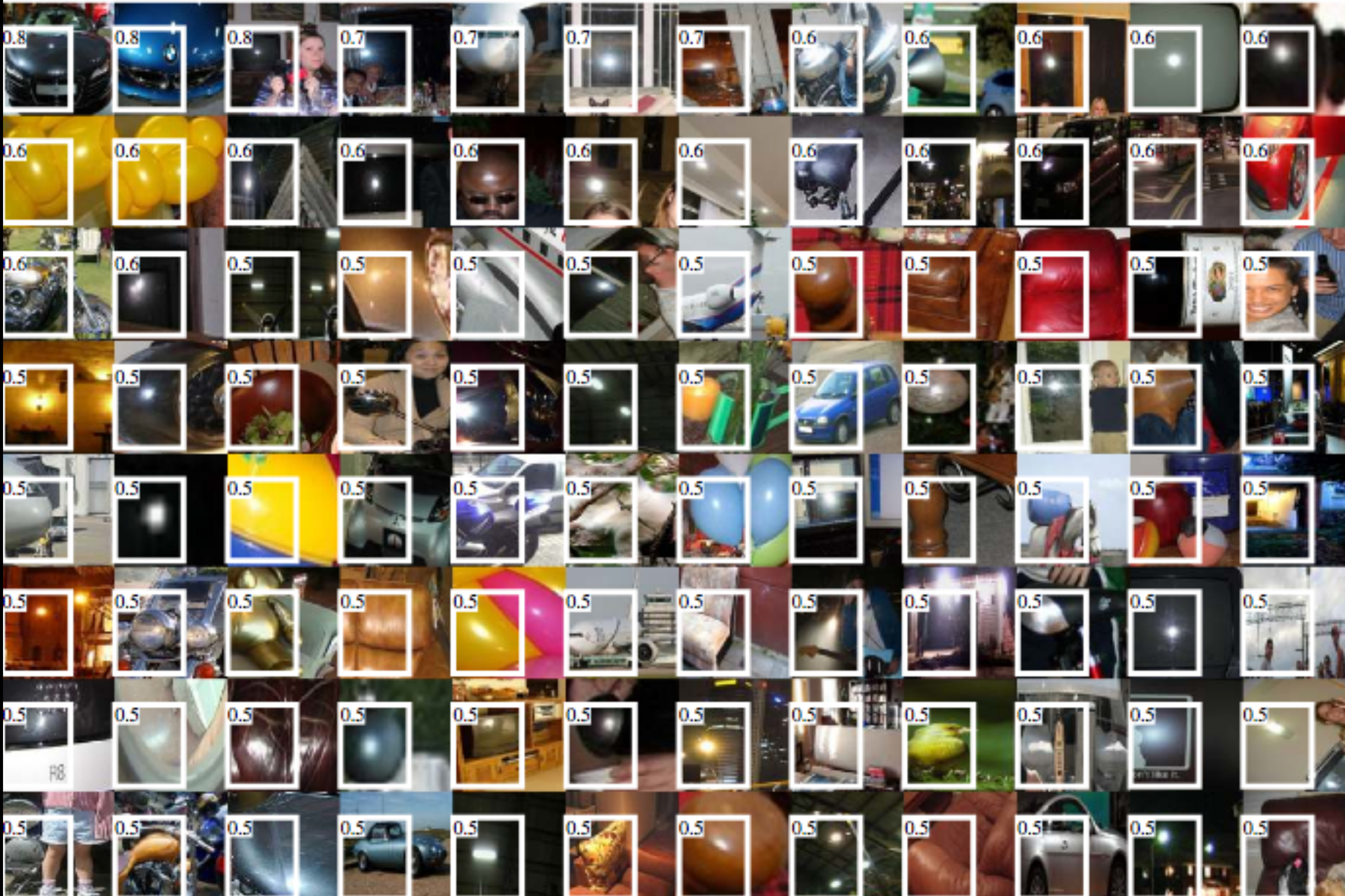
pool5 feature: (4,5,110) (top 1 – 96)



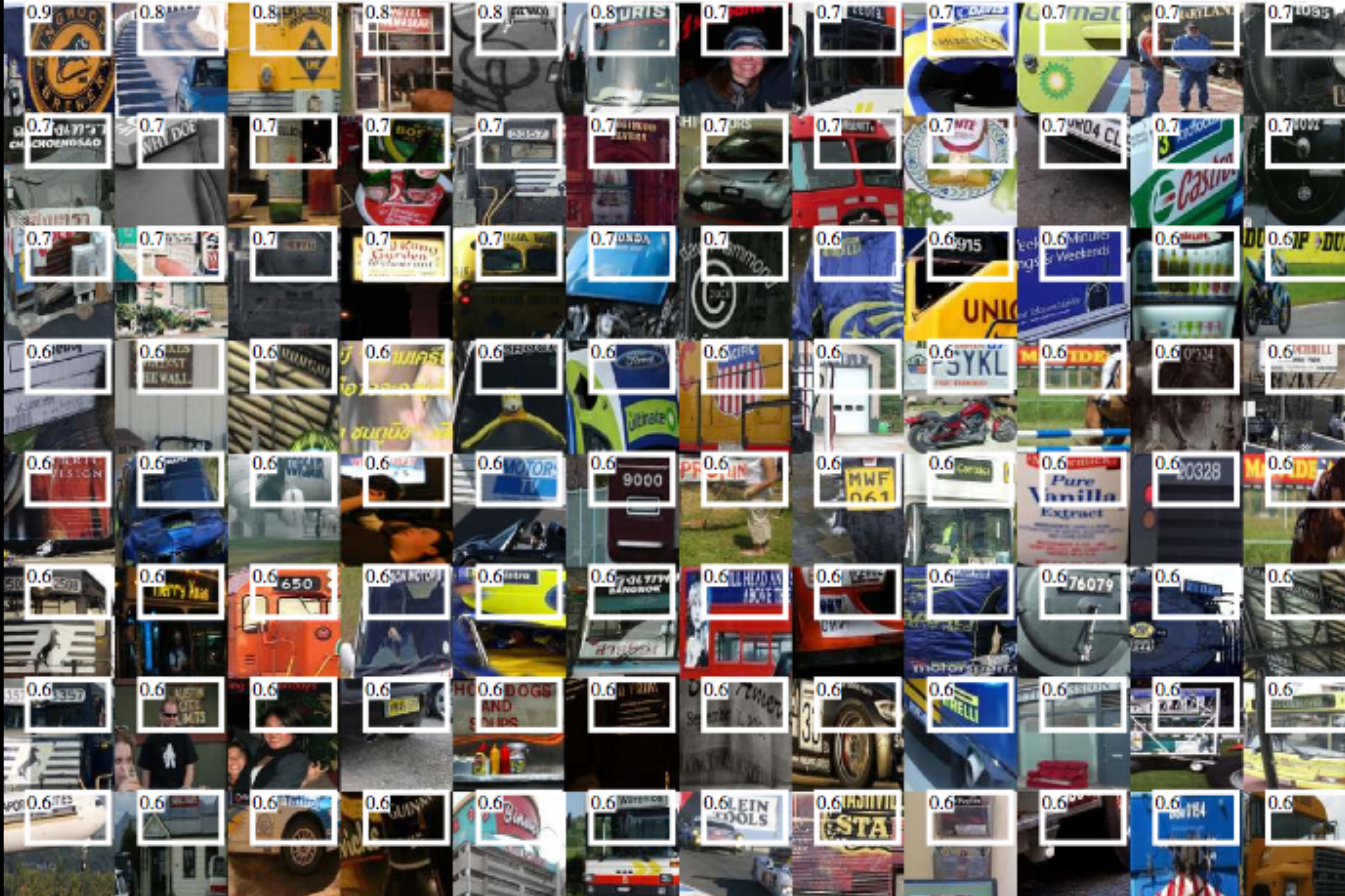
pool5 feature: (3,5,129) (top 1 – 96)



pool5 feature: (4,2,26) (top 1 – 96)

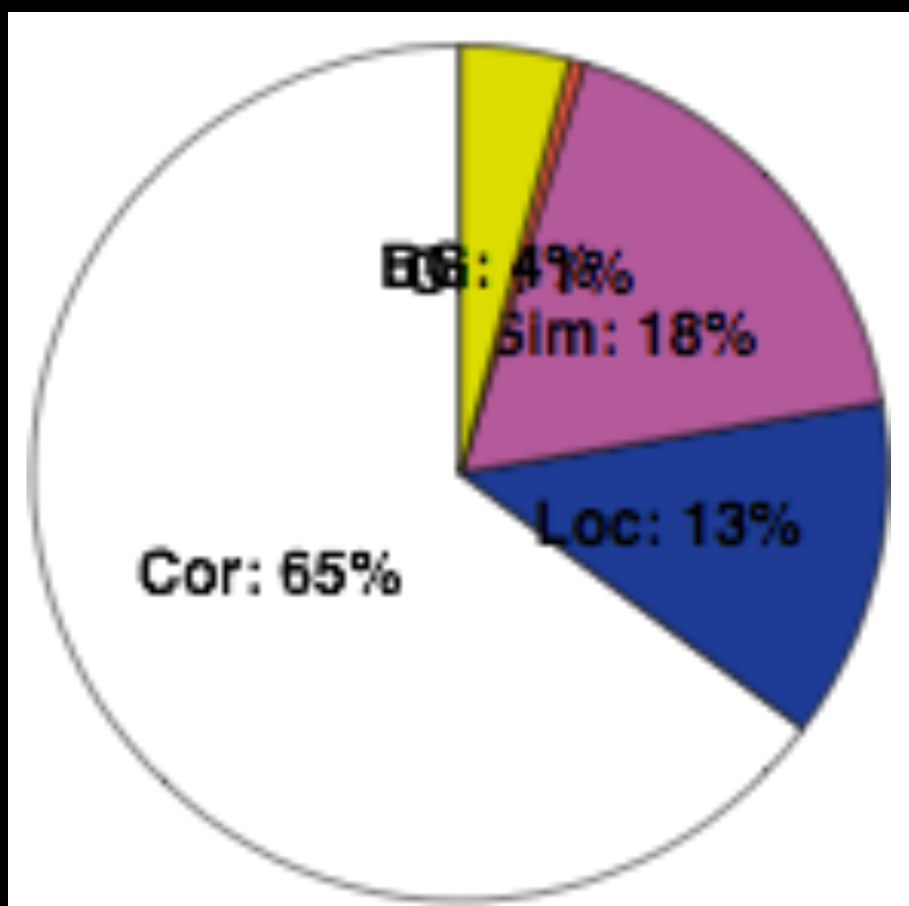


pool5 feature: (1,4,138) (top 1 – 96)



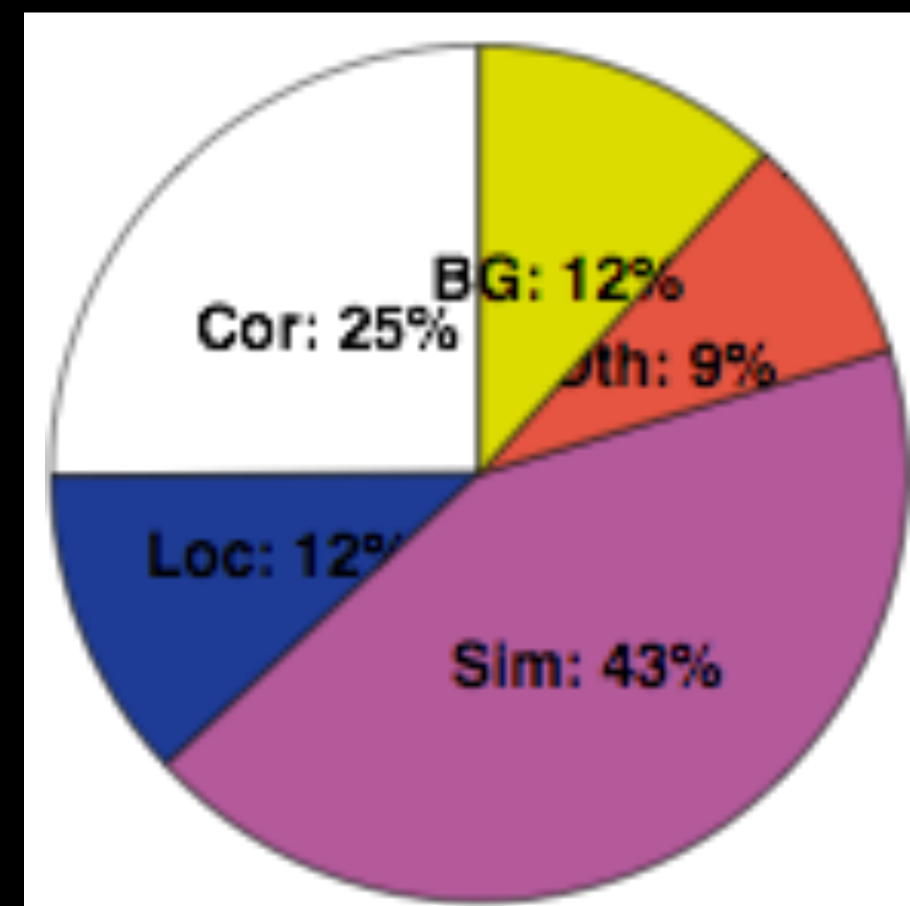
Comparison with DPM

R-CNN



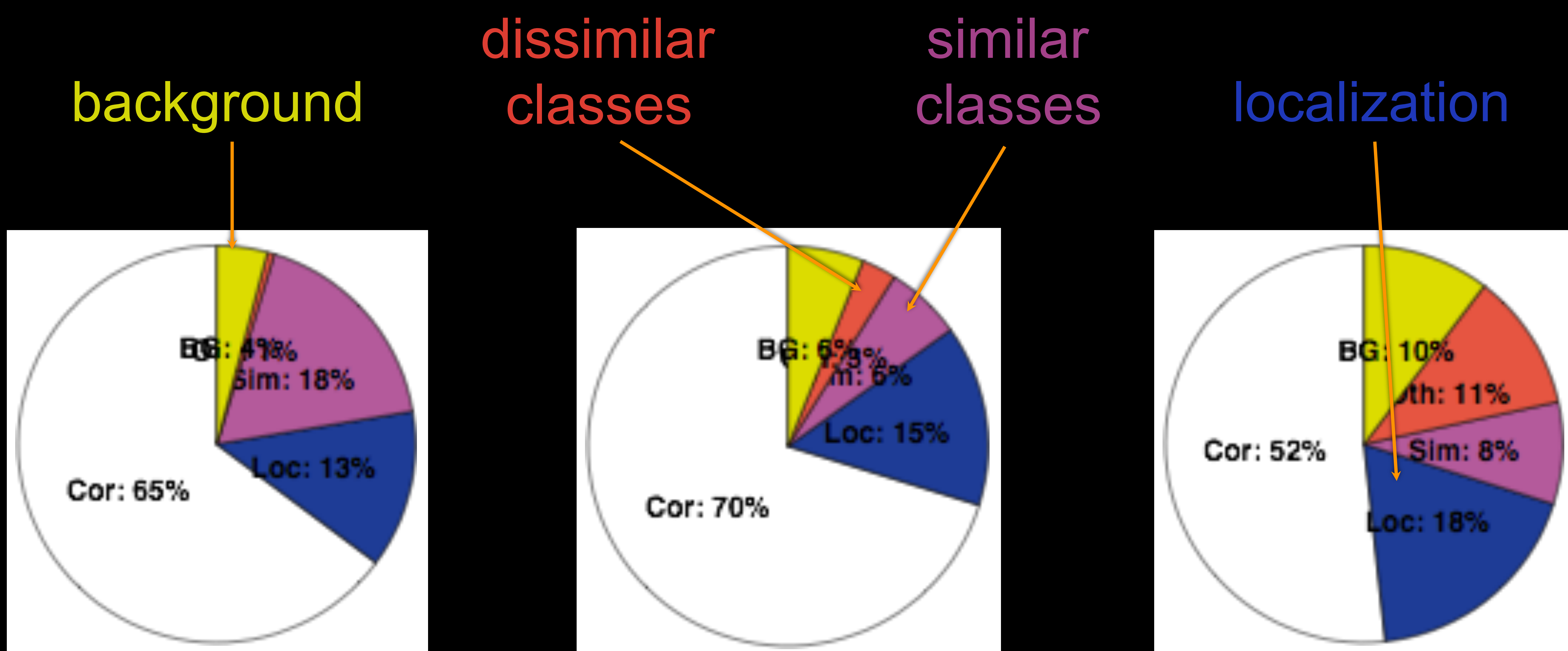
animals

DPM v5



animals

Localization errors dominate



animals vehicles furniture

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