Selective Search for Object Recognition

J.R.R. Uijlings*[1,2], K.E.A. van de Sande†[2], T. Gevers[2], and A.W.M. Smeulders[2]

[1]University of Trento, Italy
[2]University of Amsterdam, the Netherlands

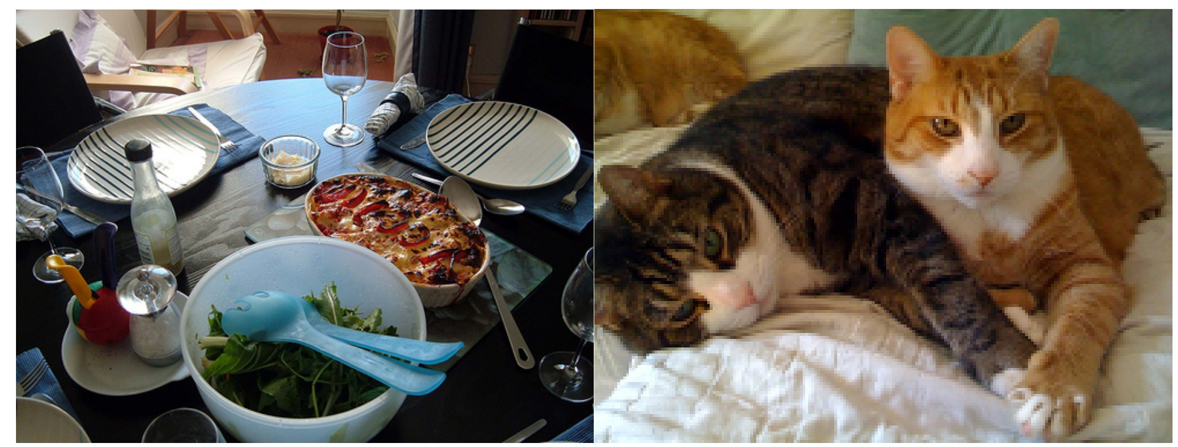
Technical Report 2012, submitted to IJCV

Presented by Song Cao

Computer vision seminar, 5/2/2013

Goal: generating possible object locations

- Why is this hard?
- High variety of reasons of forming an object
    - (a) varied scales
    - (b) color
    - (c) texture
    - (d) enclosure



(a) (b)

(c) (d)

Solution - Diversify

- Two ends of the spectrum
  - Exhaustive Search (sliding window)
    - Examples: DPM, branch and bound
    - Pros: **capture all possible locations**
    - Cons: class dependent, limited to objects, **too many proposals**
  - Segmentation
    - Data-driven, exploit image structure for proposals

Key Questions

- 1. How do we use segmentation?
- 2. What is good diversification strategy?
- 3. How effective is selective search (**small** set of **high-quality** locations)?

1. How do we use segmentation?

- Fast segmentation algorithm based on pairwise region comparison (by Felzenszwalb etal.) -> initial regions
- Greedily group regions together by selecting the pair with highest **similarity**
- Until the whole image become a single region
- Generates a hierarchy of bounding boxes

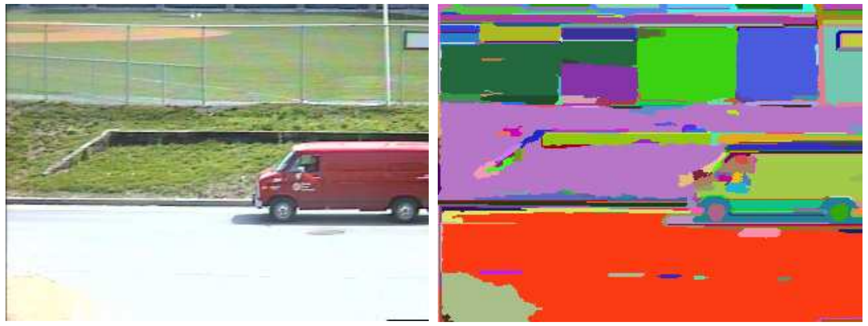


Figure 2: A street scene (320 × 240 color image), and the segmentation results produced by our algorithm ($\sigma = 0.8$, $k = 300$).

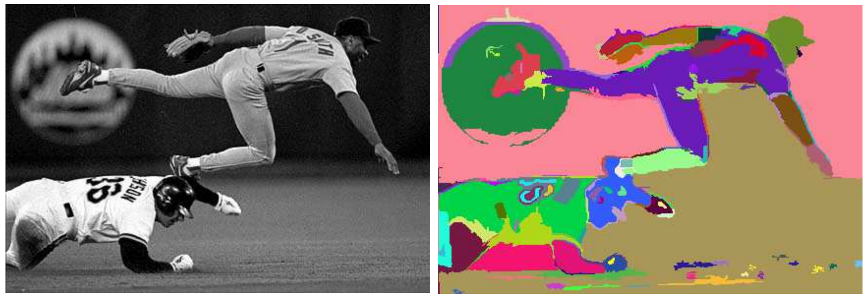


Figure 3: A baseball scene (432 × 294 grey image), and the segmentation results produced by our algorithm ($\sigma = 0.8$, $k = 300$).

Figure 4: An indoor scene (image 320 × 240, color), and the segmentation results produced by our algorithm ($\sigma = 0.8$, $k = 300$).

1. How do we use segmentation?

Algorithm 1: Hierarchical Grouping Algorithm

Input: (colour) image
Output: Set of object location hypotheses L

Obtain initial regions $R = \{r_1, \cdots, r_n\}$ using [13]
Initialise similarity set $S = \emptyset$
foreach *Neighbouring region pair* (r_i, r_j) **do**
  Calculate similarity $s(r_i, r_j)$
  $S = S \cup s(r_i, r_j)$

while $S \neq \emptyset$ **do**
  Get highest similarity $s(r_i, r_j) = \max(S)$
  Merge corresponding regions $r_t = r_i \cup r_j$
  Remove similarities regarding $r_i : S = S \setminus s(r_i, r_*)$
  Remove similarities regarding $r_j : S = S \setminus s(r_*, r_j)$
  Calculate similarity set S_t between r_t and its neighbours
  $S = S \cup S_t$
  $R = R \cup r_t$

Extract object location boxes L from all regions in R

Evaluation Metric

- Average Best Overlap (ABO)

$$\text{ABO} = \frac{1}{|G^c|} \sum_{g_i^c \in G^c} \max_{l_j \in L} \text{Overlap}(g_i^c, l_j).$$

$$\text{Overlap}(g_i^c, l_j) = \frac{\text{area}(g_i^c) \cap \text{area}(\mathrm{l_j})}{\text{area}(g_i^c) \cup \text{area}(\mathrm{l_j})}.$$

(a) Bike: 0.863 (b) Cow: 0.874 (c) Chair: 0.884 (d) Person: 0.882 (e) Plant: 0.873

- Mean Average Best Overlap (MABO)

Hierarchy v.s. Flat

threshold k in [13]	MABO	# windows
Flat [13] $k = 50, 150, \cdots, 950$	0.659	387
Hierarchical (this paper) $k = 50$	0.676	395
Flat [13] $k = 50, 100, \cdots, 1000$	0.673	597
Hierarchical (this paper) $k = 50, 100$	0.719	625

Table 2: A comparison of multiple flat partitionings against hierarchical partitionings for generating box locations shows that for the hierarchical strategy the Mean Average Best Overlap (MABO) score is consistently higher at a similar number of locations.

- Hierarchical strategy works better than multiple flat partitions
- Hierarchy - natural and effective

2. What is good diversification strategy?

2.1 Using a variety of color spaces

colour channels	R	G	B	I	V	L	a	b	S	r	g	C	H
Light Intensity	-	-	-	-	-	-	+/-	+/-	+	+	+	+	+
Shadows/shading	-	-	-	-	-	-	+/-	+/-	+	+	+	+	+
Highlights	-	-	-	-	-	-	-	-	-	-	-	+/-	+

colour spaces	RGB	I	Lab	rgI	HSV	rgb	C	H
Light Intensity	-	-	+/-	2/3	2/3	+	+	+
Shadows/shading	-	-	+/-	2/3	2/3	+	+	+
Highlights	-	-	-	-	1/3	-	+/-	+

Table 1: The invariance properties of both the individual colour channels and the colour spaces used in this paper, sorted by degree of invariance. A "+/-" means partial invariance. A fraction 1/3 means that one of the three colour channels is invariant to said property.

2. What is good diversification strategy?

2.1 Using a variety of color spaces

Similarities	MABO	# box
C	0.635	356
T	0.581	303
S	0.640	466
F	0.634	449
C+T	0.635	346
C+S	0.660	383
C+F	0.660	389
T+S	0.650	406
T+F	0.638	400
S+F	0.638	449
C+T+S	0.662	377
C+T+F	0.659	381
C+S+F	0.674	401
T+S+F	0.655	427
C+T+S+F	0.676	395

Colours	MABO	# box
HSV	0.693	463
I	0.670	399
RGB	0.676	395
rgI	0.693	362
Lab	0.690	328
H	0.644	322
rgb	0.647	207
C	0.615	125

Thresholds	MABO	# box
50	0.676	395
100	0.671	239
150	0.668	168
250	0.647	102
500	0.585	46
1000	0.477	19

Table 3: Mean Average Best Overlap for box-based object hypotheses using a variety of segmentation strategies. (C)olour, (S)ize, and (F)ill perform similar. (T)exture by itself is weak. The best combination is as many diverse sources as possible.

2. What is good diversification strategy?

2.1 Using a variety of color spaces

colour channels	R	G	B	I	V	L	a	b	S	r	g	C	H
Light Intensity	-	-	-	-	-	-	+/-	+/-	+	+	+	+	+
Shadows/shading	-	-	-	-	-	-	+/-	+/-	+	+	+	+	+
Highlights	-	-	-	-	-	-	-	-	-	-	-	+/-	+

colour spaces	RGB	I	Lab	rgI	HSV	rgb	C	H
Light Intensity	-	-	+/-	$2/3$	$2/3$	+	+	+
Shadows/shading	-	-	+/-	$2/3$	$2/3$	+	+	+
Highlights	-	-	-	-	$1/3$	-	+/-	+

Table 1: The invariance properties of both the individual colour channels and the colour spaces used in this paper, sorted by degree of invariance. A "+/-" means partial invariance. A fraction $1/3$ means that one of the three colour channels is invariant to said property.

2. What is good diversification strategy?

2.2 Using four different similarity measures

$$s_{colour}(r_i, r_j) = \sum_{k=1}^{n} \min(c_i^k, c_j^k).$$

$$s_{texture}(r_i, r_j) = \sum_{k=1}^{n} \min(t_i^k, t_j^k).$$

$$s_{size}(r_i, r_j) = 1 - \frac{\text{size}(r_i) + \text{size}(\mathrm{r_j})}{\text{size}(im)},$$

$$fill(r_i, r_j) = 1 - \frac{\text{size}(BB_{ij}) - \text{size}(r_i) - \text{size}(r_i)}{\text{size}(im)}$$

- Size score encourages small regions to merge early
- Fill score encourage overlapping regions to avoid holes

$$\begin{aligned} s(r_i, r_j) \quad = \quad & a_1 s_{colour}(r_i, r_j) + a_2 s_{texture}(r_i, r_j) + \\ & a_3 s_{size}(r_i, r_j) + a_4 s_{fill}(r_i, r_j), \end{aligned}$$

2. What is good diversification strategy?

- 2.3 Varying starting regions (given by Felzenszwalb etal.)
  - Using different color spaces
  - Varying the threshold parameter k
- Combining diversification strategies

Version	Diversification Strategies	MABO	# win	# strategies	time (s)
Single Strategy	HSV C+T+S+F $k = 100$	0.693	362	1	0.71
Selective Search Fast	HSV, Lab C+T+S+F, T+S+F $k = 50, 100$	0.799	2147	8	3.79
Selective Search Quality	HSV, Lab, rgI, H, I C+T+S+F, T+S+F, F, S $k = 50, 100, 150, 300$	0.878	10,108	80	17.15

3. How effective is selective search?

- Bounding box quality evaluation
  - VOC 2007 TEST Set
- Object recognition performance
  - VOC 2010 detection task

3. How effective is selective search?

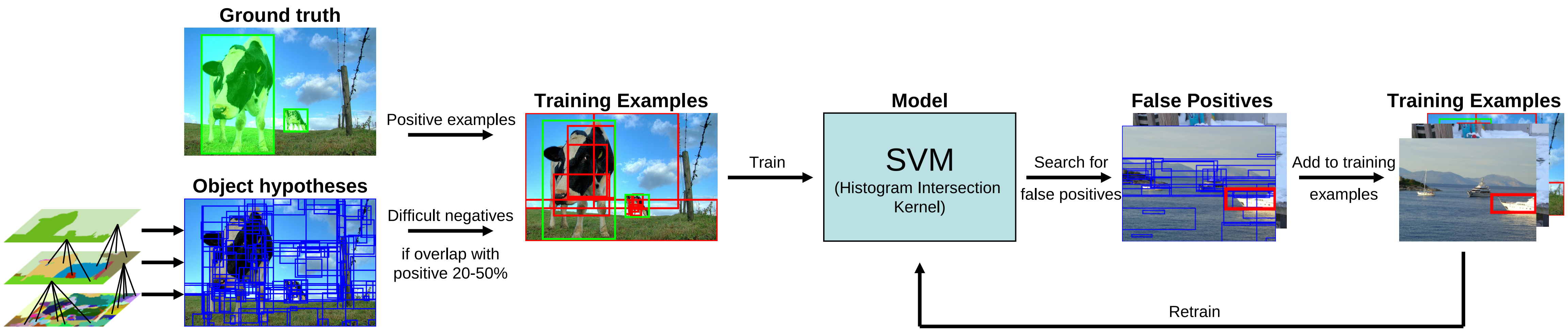
- Bounding box quality evaluation

method	recall	MABO	# windows
Arbelaez *et al*. [3]	0.752	0.649 ± 0.193	418
Alexe *et al*. [2]	0.944	0.694 ± 0.111	1,853
Harzallah *et al*. [16]	0.830	-	200 per class
Carreira and Sminchisescu [4]	0.879	0.770 ± 0.084	517
Endres and Hoiem [9]	0.912	0.791 ± 0.082	790
Felzenszwalb *et al*. [12]	0.933	0.829 ± 0.052	100,352 per class
Vedaldi *et al*. [34]	0.940	-	10,000 per class
Single Strategy	0.840	0.690 ± 0.171	289
Selective search "Fast"	0.980	0.804 ± 0.046	2,134
Selective search "Quality"	0.991	0.879 ± 0.039	10,097

Table 5: Comparison of recall, Mean Average Best Overlap (MABO) and number of window locations for a variety of methods on the Pascal 2007 TEST set.

3. How effective is selective search?

- Evaluation on object recognition
- Selective search + SIFT + bag-of-words + SVMs



3. How effective is selective search?

- Evaluation on object recognition
- Selective search + SIFT + bag-of-words + SVMs

System	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow
NLPR	.533	**.553**	**.192**	**.210**	.300	.544	.467	.412	**.200**	.315
MIT UCLA [38]	.542	.485	.157	.192	.292	**.555**	.435	.417	.169	.285
NUS	.491	.524	.178	.120	.306	.535	.328	.373	.177	.306
UoCTTI [12]	.524	.543	.130	.156	**.351**	.542	**.491**	.318	.155	.262
This paper	**.562**	.424	.153	.126	.218	.493	.368	**.461**	.129	**.321**

table	dog	horse	motor	person	plant	sheep	sofa	train	tv
.207	.303	.486	.553	.465	.102	.344	.265	**.503**	.403
.267	.309	.483	.550	.417	.097	.358	.308	.472	.408
.277	.295	**.519**	**.563**	.442	.096	.148	.279	.495	.384
.135	.215	.454	.516	**.475**	.091	.351	.194	.466	.380
.300	**.365**	.435	.529	.329	**.153**	**.411**	**.318**	.470	**.448**

3. How effective is selective search?

- SIFT based feature enabled by this method
- Performs well on non-rigid object categories

System	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow
NLPR	.533	**.553**	**.192**	**.210**	.300	.544	.467	.412	**.200**	.315
MIT UCLA [38]	.542	.485	.157	.192	.292	**.555**	.435	.417	.169	.285
NUS	.491	.524	.178	.120	.306	.535	.328	.373	.177	.306
UoCTTI [12]	.524	.543	.130	.156	**.351**	.542	**.491**	.318	.155	.262
This paper	**.562**	.424	.153	.126	.218	.493	.368	**.461**	.129	**.321**

table	dog	horse	motor	person	plant	sheep	sofa	train	tv
.207	.303	.486	.553	.465	.102	.344	.265	**.503**	.403
.267	.309	.483	.550	.417	.097	.358	.308	.472	.408
.277	.295	**.519**	**.563**	.442	.096	.148	.279	.495	.384
.135	.215	.454	.516	**.475**	.091	.351	.194	.466	.380
.300	**.365**	.435	.529	.329	**.153**	**.411**	**.318**	.470	**.448**

Rich feature hierarchies for accurate object detection and semantic segmentation

Tech report

Ross Girshick[1] Jeff Donahue[1,2] Trevor Darrell[1,2] Jitendra Malik[1]
[1]UC Berkeley and [2]ICSI
{rbg,jdonahue,trevor,malik}@eecs.berkeley.edu

Presented by Song Cao

Computer vision seminar, 5/2/2013

Background

- Deep learning (Convolutional Neural Network) is best performing image-classification method for *ImageNet (Krizhevsky et al. ECCV 2012)*
- Debate (War?)
- What about Object Recognition/Detection (PASCAL)?

They did it!

VOC 2010 test	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table
DPM HOG [19]	45.6	49.0	11.0	11.6	27.2	50.5	43.1	23.6	17.2	23.2	10.7
SegDPM [18]	56.4	48.0	24.3	21.8	**31.3**	**51.3**	47.3	48.2	16.1	29.4	19.0
UVA [36]	56.2	42.4	15.3	12.6	21.8	49.3	36.8	46.1	12.9	32.1	30.0
ours (R-CNN FT fc_7)	**65.4**	**56.5**	**45.1**	**28.5**	24.0	50.1	**49.1**	**58.3**	**20.6**	**38.5**	**31.1**

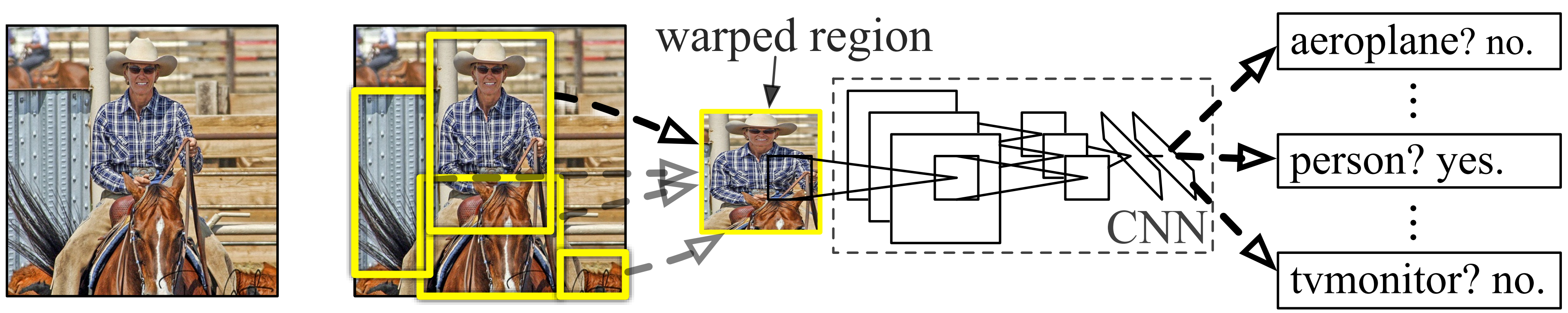
VOC 2010 test	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
DPM HOG [19]	20.5	42.5	44.5	41.3	8.7	29.0	18.7	40.0	34.5	29.6
SegDPM [18]	37.5	44.1	51.5	44.4	12.6	32.1	28.8	**48.9**	39.1	36.6
UVA [36]	36.5	43.5	52.9	32.9	15.3	41.1	**31.8**	47.0	44.8	35.1
ours (R-CNN FT fc_7)	**57.5**	**50.7**	**60.3**	**44.7**	**21.6**	**48.5**	24.9	48.0	**46.5**	**43.5**

- On PASCAL 2007 improves upon DPM by **40%**
- Faster than UVA

Object Recognition using Deep Learning

Image features are the engine of recognition.

R-CNN: ***Regions with CNN features***

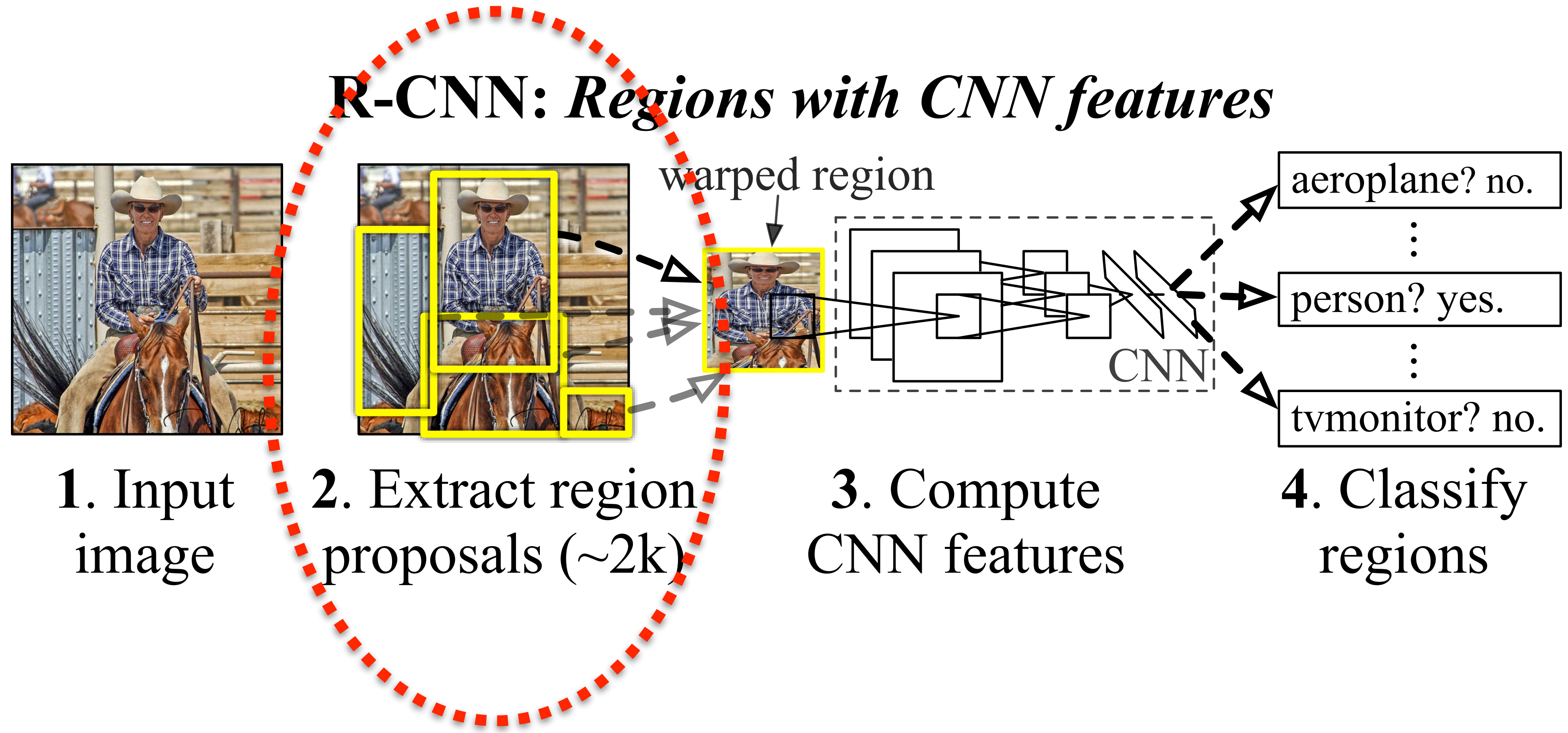


1. Input image **2**. Extract region proposals (~2k) **3**. Compute CNN features **4**. Classify regions

Region Proposal

Sliding window + CNN = High computational cost

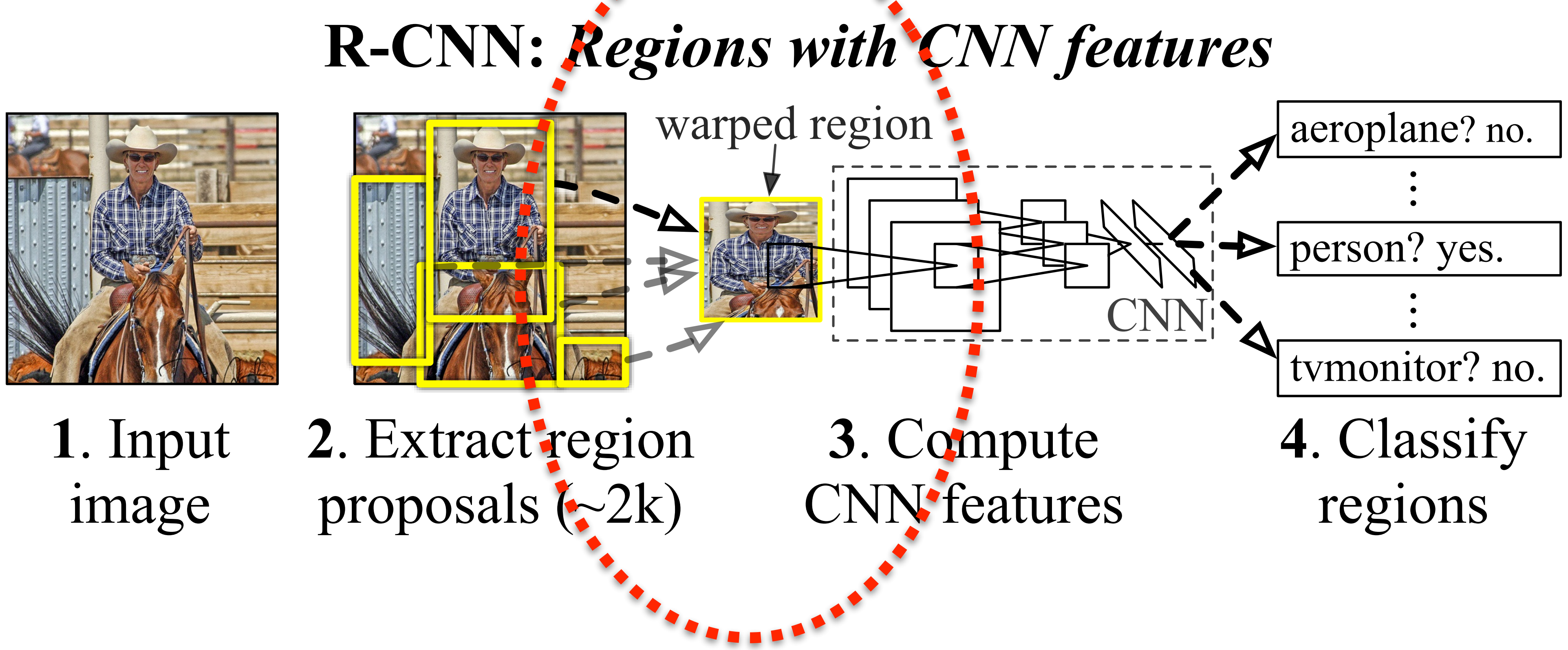
Selective Search!



Region Warping

Regardless of size and aspect ratio

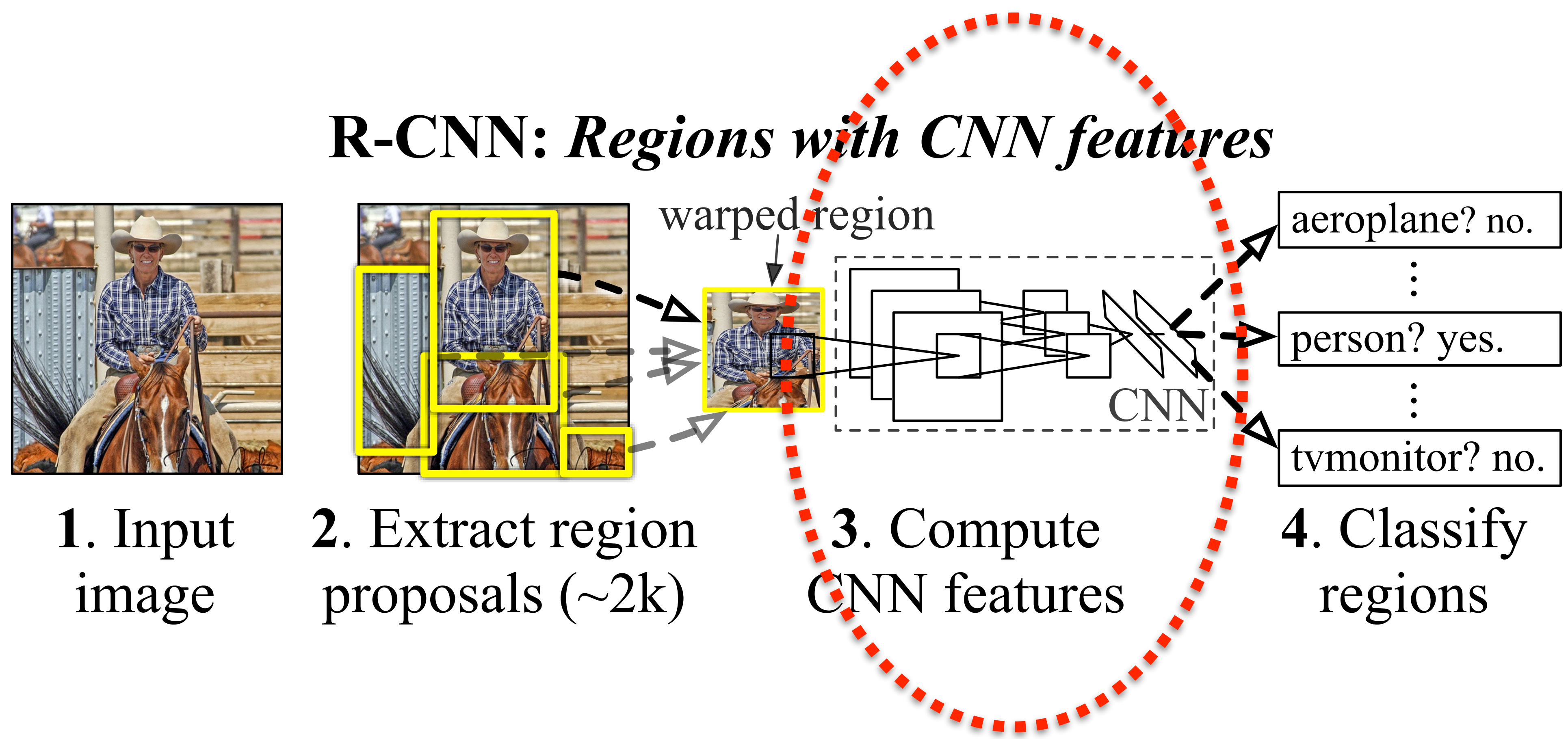
Warp to 224*224 patch



Feature Extraction

4096-dimensional feature vector

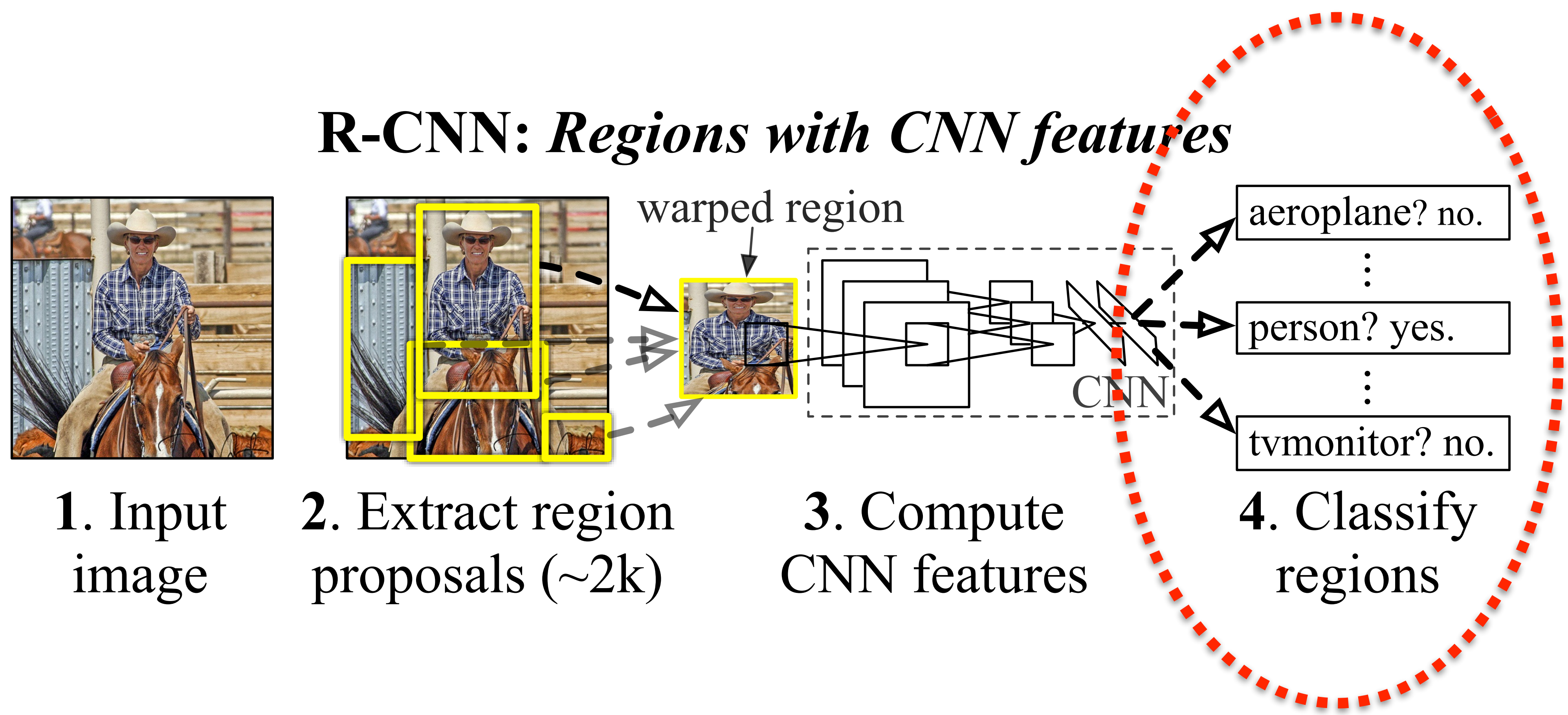
their own implementation of the CNN of (Krizhevsky et al. ECCV 2012)



Inference

Training + Testing using SVMs (with negative mining)

Efficient: shared CNN parameters + low dimensional features



CNN Training

- Pre-training + fine-tuning
- Overlap threshold to define positive/negative: 0.3
    - Performance is quite sensitive to this value
- What feature exactly did CNN learn?
- Visualization method: single out a unit and treat it as a detector

Feature Visualization

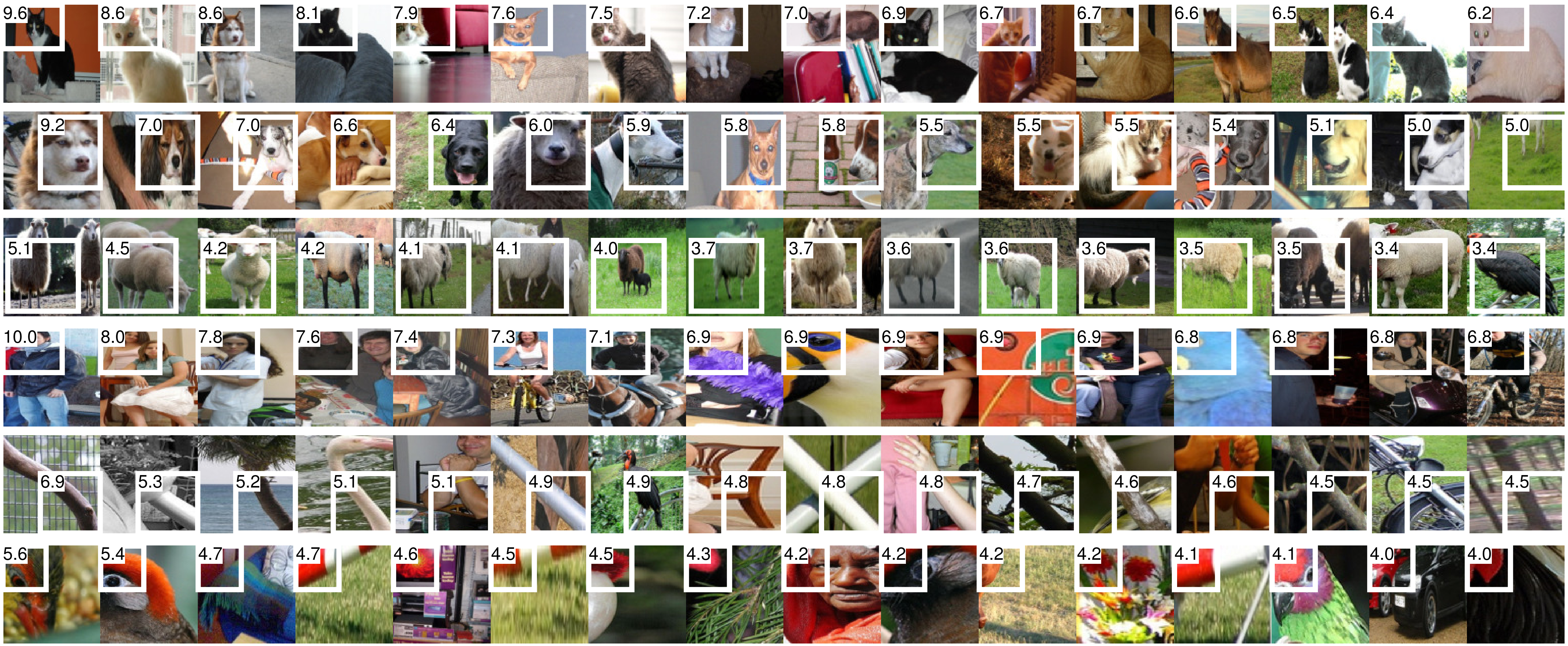


Figure 3: Top activations for six pool$_5$ units. Receptive fields and activation values are drawn in white. From top to bottom: (1) positive and (2) negative weight for cats; positive weight for (3) sheep and (4) person; selectivity for (5) diagonal bars and (6) red blobs.

Ablation Study

- Last three layers: $pool_5$, fc_6 and fc_7
- With or without fine-tuning
- $pool_5$ uses only 6% parameters (possible to use DPM on top)
- Color helps (40.1% -> 43.4% VOC 2007 on fc_6)

VOC 2007 test	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
R-CNN $pool_5$	49.3	58.0	29.7	22.2	20.6	47.7	56.8	43.6	16.0	39.7	37.7	39.6	49.6	55.6	37.5	20.6	40.5	37.4	47.8	51.3	40.1
R-CNN fc_6	56.1	58.8	34.4	29.6	22.6	50.4	58.0	52.5	18.3	40.1	41.3	46.8	49.5	53.5	39.7	23.0	46.4	36.4	50.8	59.0	43.4
R-CNN fc_7	53.1	58.9	35.4	29.6	22.3	50.0	57.7	52.4	19.1	43.5	40.8	43.6	47.6	54.0	39.1	23.0	42.3	33.6	51.4	55.2	42.6
R-CNN FT $pool_5$	55.6	57.5	31.5	23.1	23.2	46.3	59.0	49.2	16.5	43.1	37.8	39.7	51.5	55.4	40.4	23.9	46.3	37.9	49.7	54.1	42.1
R-CNN FT fc_6	61.8	62.0	38.8	35.7	29.4	52.5	**61.9**	53.9	22.6	49.7	40.5	**48.8**	49.9	**57.3**	44.5	**28.5**	50.4	**40.2**	54.3	61.2	47.2
R-CNN FT fc_7	**60.3**	**62.5**	**41.4**	**37.9**	29.0	**52.6**	61.6	**56.3**	**24.9**	**52.3**	**41.9**	48.1	54.3	57.0	**45.0**	26.9	**51.8**	38.1	**56.6**	**62.2**	**48.0**
DPM HOG [19]	33.2	60.3	10.2	16.1	27.3	54.3	58.2	23.0	20.0	24.1	26.7	12.7	**58.1**	48.2	43.2	12.0	21.1	36.1	46.0	43.5	33.7
DPM ST [29]	23.8	58.2	10.5	8.5	27.1	50.4	52.0	7.3	19.2	22.8	18.1	8.0	55.9	44.8	32.4	13.3	15.9	22.8	46.2	44.9	29.1
DPM HSC [32]	32.2	58.3	11.5	16.3	**30.6**	49.9	54.8	23.5	21.5	27.7	34.0	13.7	**58.1**	51.6	39.9	12.4	23.5	34.4	47.4	45.2	34.3

VOC Segmentation

- Segmentation by region classification
- Feature same as before + foreground mask

VOC 2011 test	bg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow
R&P [2]	83.4	46.8	18.9	36.6	31.2	42.7	57.3	47.4	44.1	8.1	39.4
O_2P [5]	**85.4**	**69.7**	22.3	45.2	**44.4**	46.9	66.7	57.8	56.2	**13.5**	**46.1**
ours (*full+fg* R-CNN fc_6)	84.2	66.9	**23.7**	**58.3**	37.4	**55.4**	**73.3**	**58.7**	**56.5**	9.7	45.5

VOC 2011 test	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mean
R&P [2]	**36.1**	36.3	49.5	48.3	50.7	26.3	47.2	22.1	42.0	43.2	40.8
O_2P [5]	32.3	41.2	**59.1**	55.3	51.0	**36.2**	50.4	**27.8**	46.9	**44.6**	47.6
ours (*full+fg* R-CNN fc_6)	29.5	**49.3**	40.1	**57.8**	**53.9**	33.8	**60.7**	22.7	**47.1**	41.3	**47.9**

Take aways

- Large CNN is highly effective in feature learning
- Classical computer vision tools and deep learning are partners, not enemies