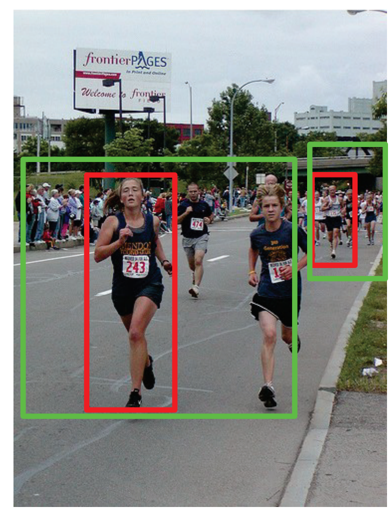
Modelling Action In Context

Action Recognition: Still Images vs Video

What defines "context" of actions?

- Person's pose
- What are surrounding objects
- What interaction with objects
- What other people



How to quantify context as action "cues"?

- Difficult to locate – no fixed position
- Variable – differs among instances of the same action class

Action Recognition as Multi-Instance Learning

- Learning to automatically select the most informative/relevant action cue for each individual instance of an action class

- Consider an action is described by
  1. the presence of a person – a primary region(s)
  2. the context – a surrounding region(s) in proximity of the primary region, as most informative visual cues define the action of interest

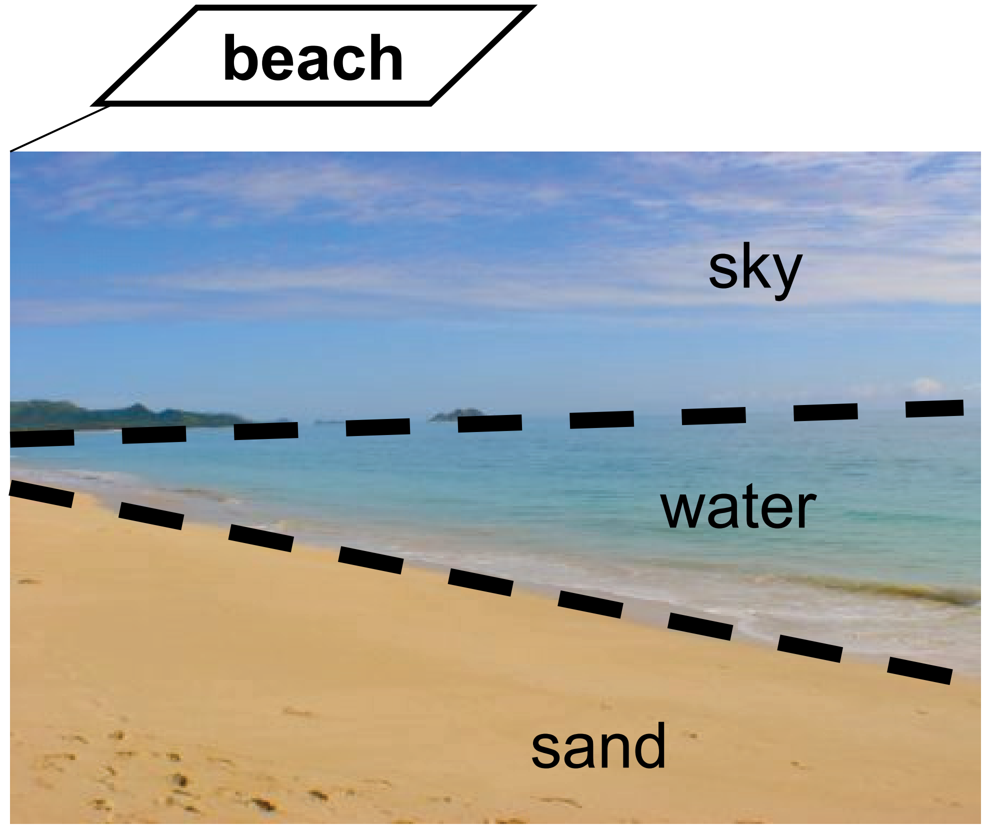
  "Multi-Instance Learning"

Why Multi-Instance Learning (MIL)

Bag -> An image

Instance -> A ‘region’ in the image

Label -> {‘beach’ ‘not beach’}



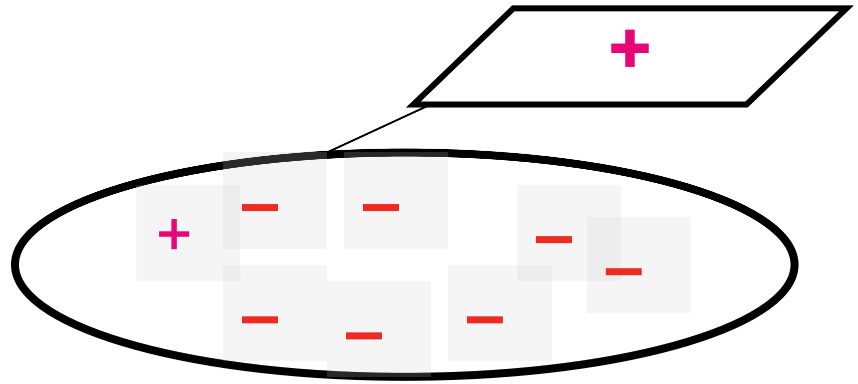
http://www.adrhi.com/Waimanalo-Beach.jpg

(slides by Anjali Koppal)

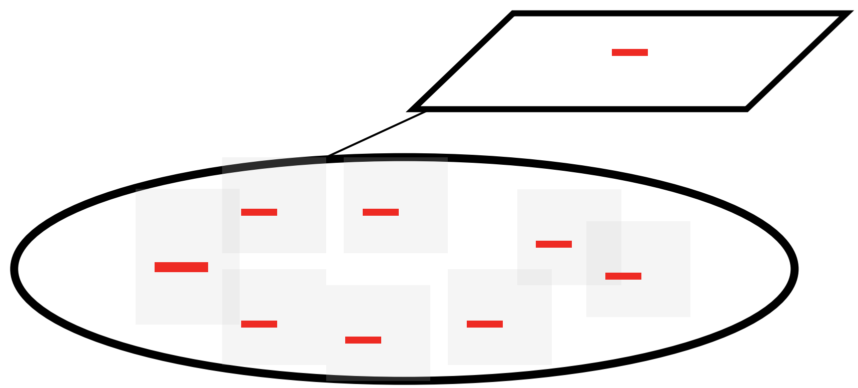
Weak Labelling

MIL bag labelling:

A bag is positive if at least one of its instances is predicted to be positive.



A bag is negative if all its instances are predicted to be negative.



(slides by Anjali Koppal)

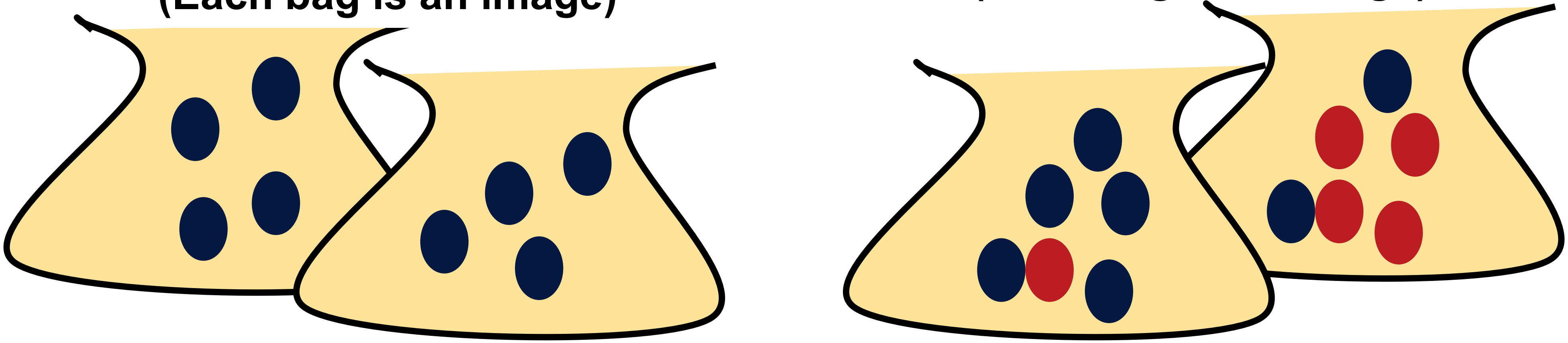
What is Multiple Instance Learning?

Instead of each sample (instance) individually labelled, the training data is set of labelled "bags" and each bag is given a label:

- A bag contains many instances (with unknown labels), e.g. a bag is an image.
- An instance in a bag is a feature vector, e.g. each instance is a part/region.
- Binary case: A bag is labeled as "positive" if AT LEAST ONE of its instances is positive; labelled "negative" if ALL its instances are negative.

NEGATIVE BAGS
(Each bag is an image)

POSITIVE BAGS
(Each bag is an image)



(slides by Jeremy Bolton)

Learning from Bags – Weakly Supervised Learning

- Supervised learning
  - Learn a classifier given a set of training samples *X* and corresponding label *Y* ***for every sample***

$$X = \{x_1, ..., x_n\}, Y = \{y_1, ..., y_n\}$$

- MIL: Weakly supervised learning
  - Learn a classifier given ***multiple bags*** of samples and labels for the ***bags***.
  - Why "weakly" – only a label for a bag without labelling every sample in the bag
  - Learning with uncertain labels (noisy teacher) or weak labels (lazy teacher)

$$X_i = \{x_{i1}, \ldots, x_{in_i}\}, Y_i = 1 \text{ or } 0, \{y_{i1} = ?, \ldots, y_{in_i} = ?\}$$

(slides by Jeremy Bolton)

Objectives of Multiple Instance Learning

- Given:
  - A set of *I* bags
    - Labeled + or -

$$\mathbf{B} = \{B_1^+, ..B_i^+, B_{i+1}^-, ..., B_I^-\}$$

  - The i^{th} bag is a set of instance feature vectors J_i in some feature space

$$B_i = \{x_{i1}, ..., x_{iJ_i}\}$$

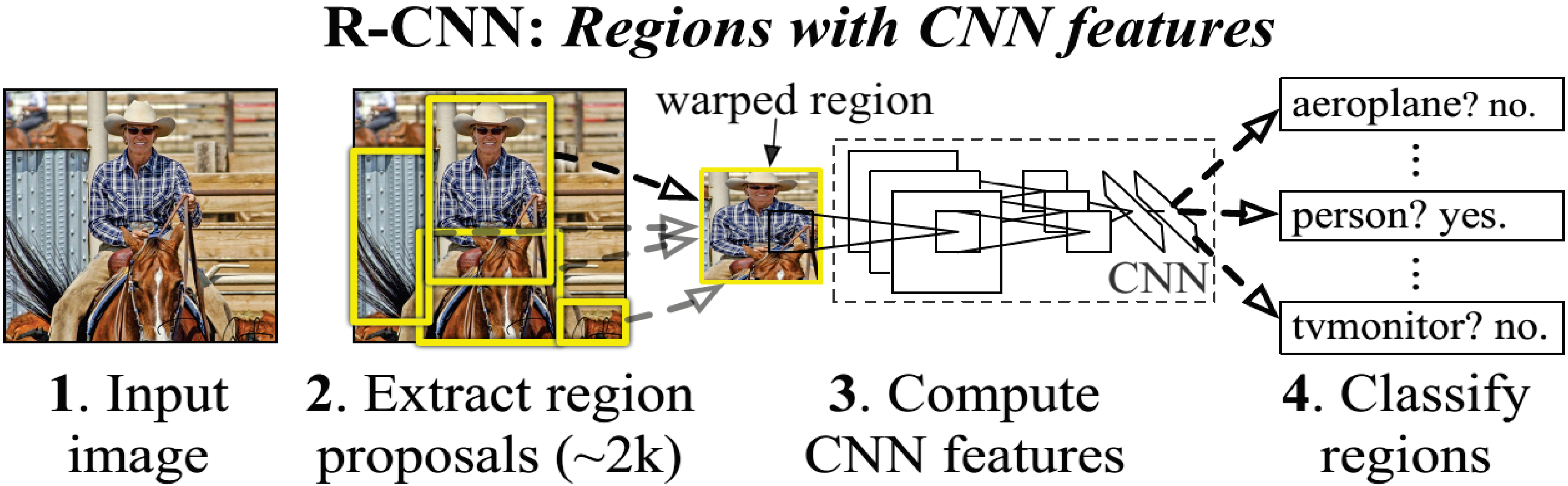
  - Assignment of labels

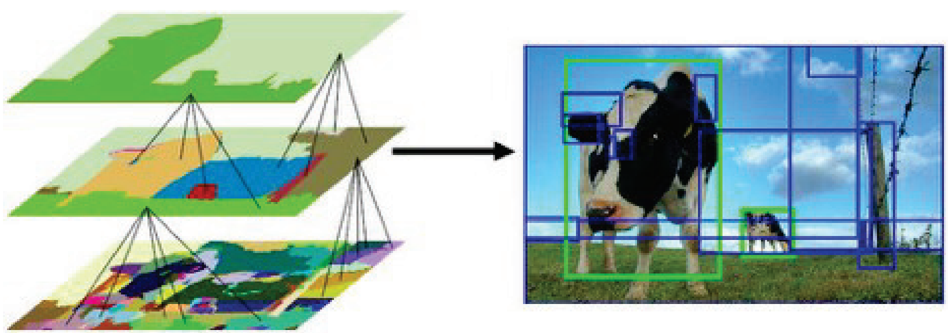
$$B_i^+ \Rightarrow \exists j : label(x_{ij}) = 1$$

$$B_i^- \Rightarrow \forall j, label(x_{ij}) = 0$$

- Learning objective:
  1. Select/correlate instances (feature vectors) common to the positive bags that is not observed in the negative bags

- Model application:
  1. Infer a concept (bag label) to labelling individual instances
  2. Predict the class/concept of an unlabelled bag

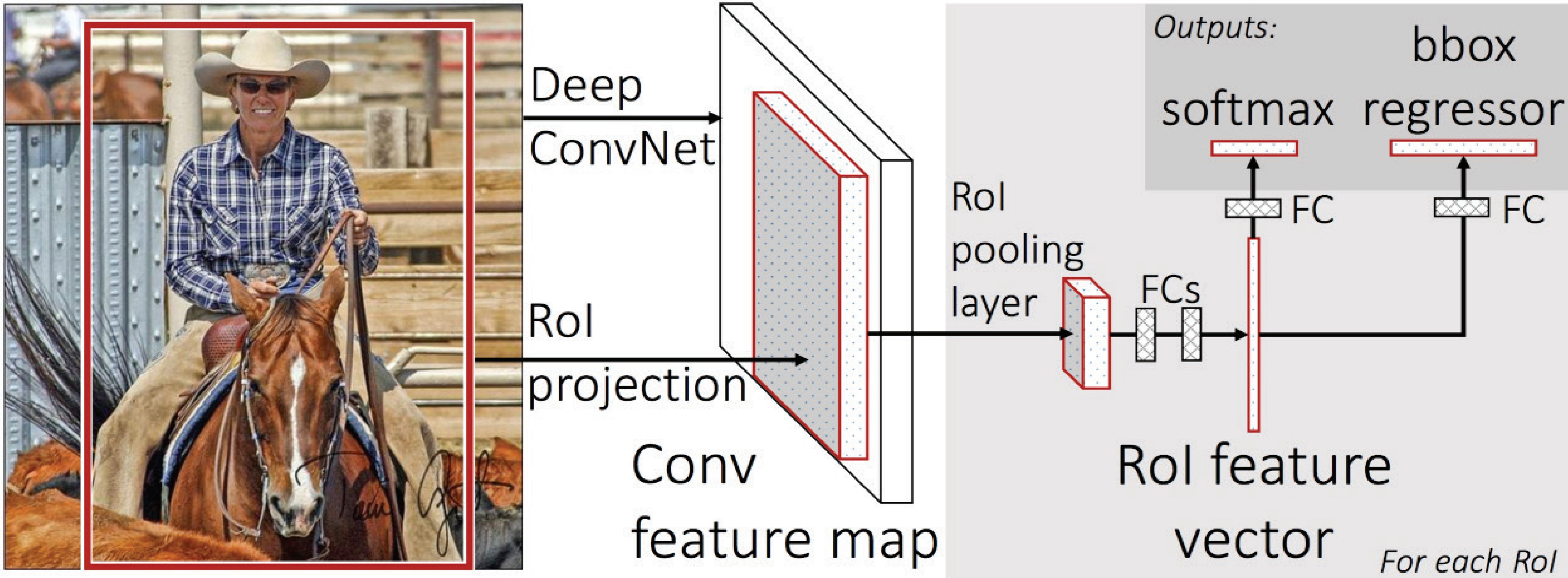




(region proposals – selective search)

The main problems of R-CNN is slow, as it performs a ConvNet forward pass for each object proposal without sharing computation / sharing features – redundant in computing features many times over.

Fast R-CNN [Girshick et al, 2015]

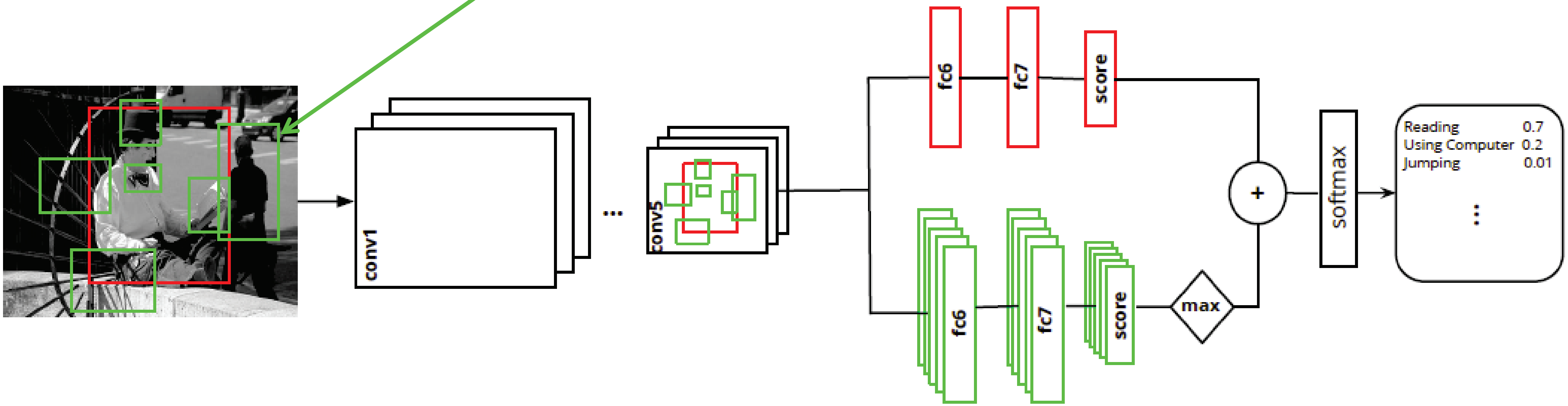


- Sharing features for all region proposals
- The most time consuming remains to be selective search

R*CNN: Context Model for Action Recognition

[Gkioxari et al, 2015]

$$R(r;I) = \{s \in S(I) : \text{overlap}(s,r) \in [l,u]\}$$ If $l = 0$ and $u = 1$ then $R(r;I) = S(I)$



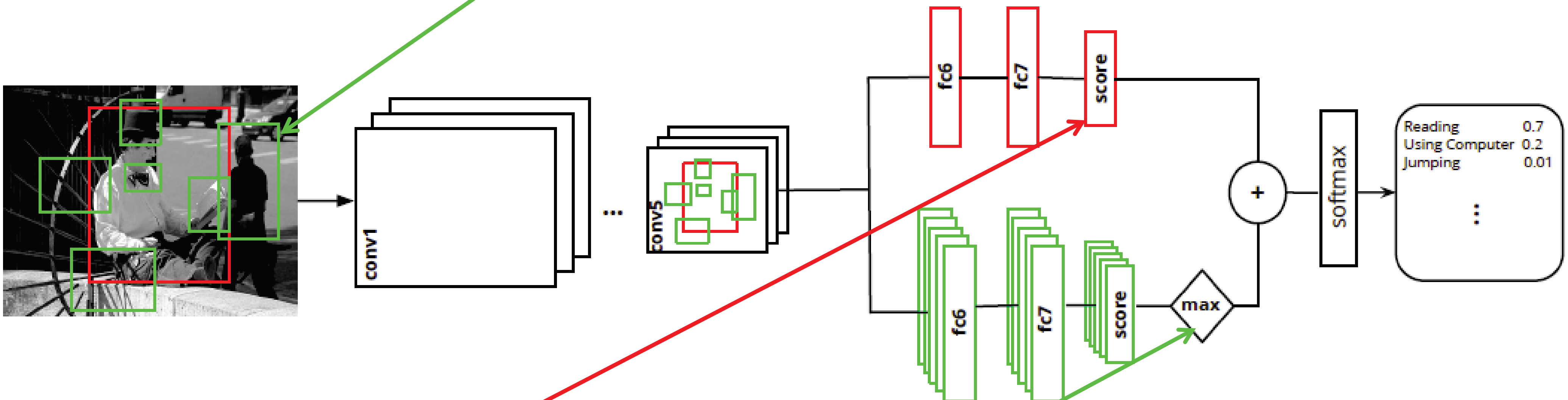
- Given image I, select the primary region to be the bounding box containing the person (red box) while region proposals define the set of candidate secondary regions (green boxes).

- For each action α, the most informative secondary region is selected (*max* operation) and its score is added to the primary. The *softmax* operation transforms scores into probabilities and forms the final prediction.

R*CNN: Context Model for Action Recognition

[Gkioxari et al, 2015]

$$R(r; I) = \{s \in S(I) : \text{overlap}(s, r) \in [l, u]\} \quad \text{If } l = 0 \text{ and } u = 1 \text{ then } R(r; I) = S(I)$$



$$\text{score}(\alpha; I, r) = \boxed{\mathbf{w}_{\mathrm{P}}^{\alpha} \cdot \phi(r; I)} + \boxed{\max_{s \in R(r;I)} \mathbf{w}_{\mathrm{S}}^{\alpha} \cdot \phi(s; I)}$$

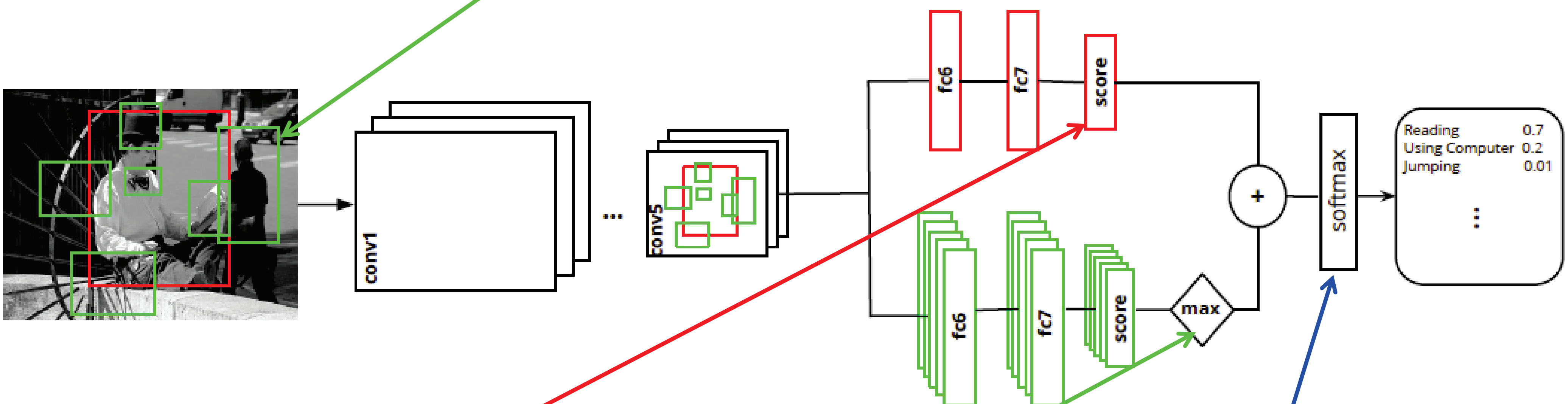
where $\phi(r; I)$ is a vector of features extracted from region r in I, while $\mathbf{w}_{\mathrm{P}}^{\alpha}$ and $\mathbf{w}_{\mathrm{S}}^{\alpha}$ are the primary and secondary weights for action α respectively. $R(r; I)$ defines the set of candidates for the secondary region.

w_P, w_S and φ are learned jointly

R*CNN: Context Model for Action Recognition

[Gkioxari et al, 2015]

$$R(r;I) = \{s \in S(I) : \text{overlap}(s,r) \in [l,u]\} \quad \text{If } l = 0 \text{ and } u = 1 \text{ then } R(r;I) = S(I)$$



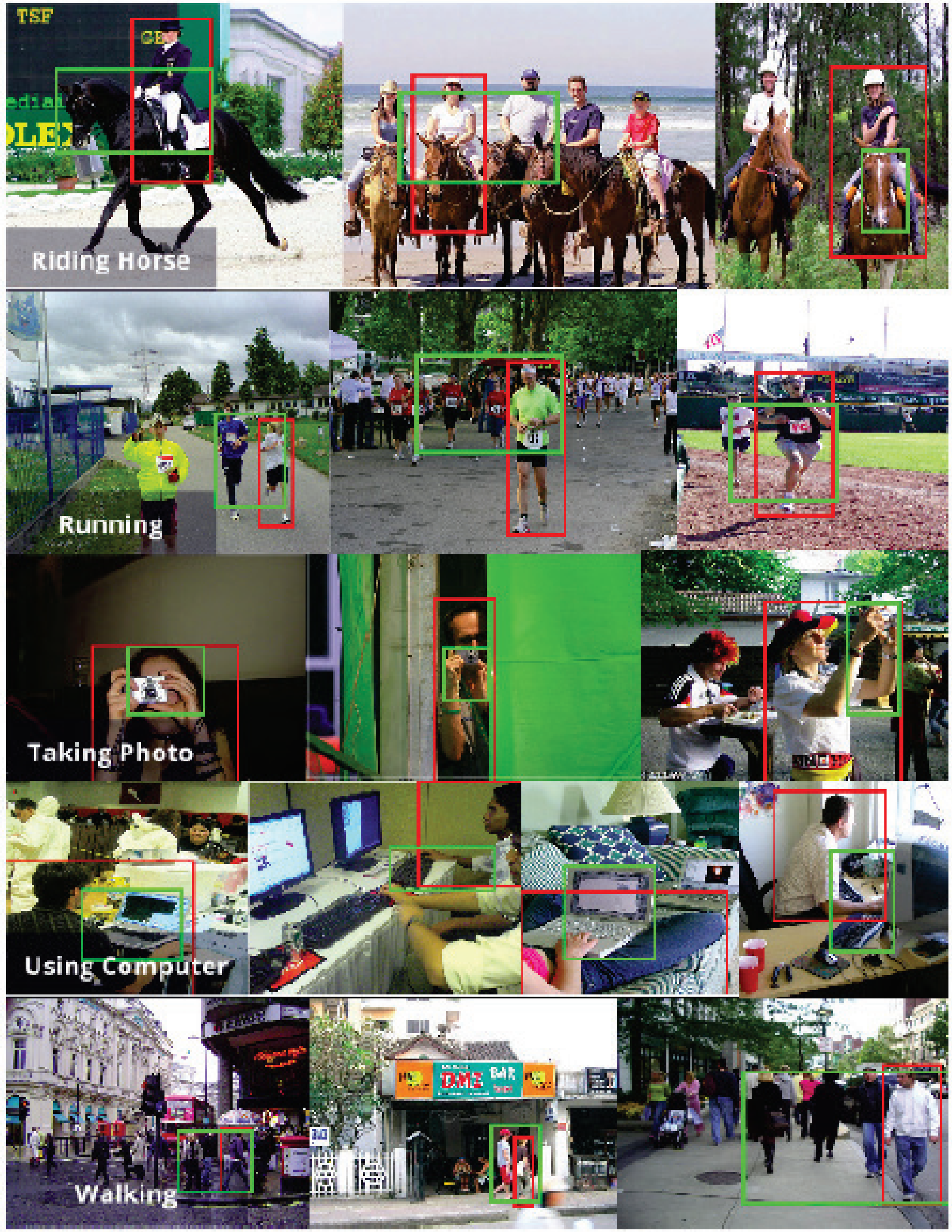
$$\text{score}(\alpha;I,r) = \boxed{\mathbf{w}_{\mathrm{P}}^{\alpha} \cdot \phi(r;I)} + \boxed{\max_{s \in R(r;I)} \mathbf{w}_{\mathrm{S}}^{\alpha} \cdot \phi(s;I)}$$

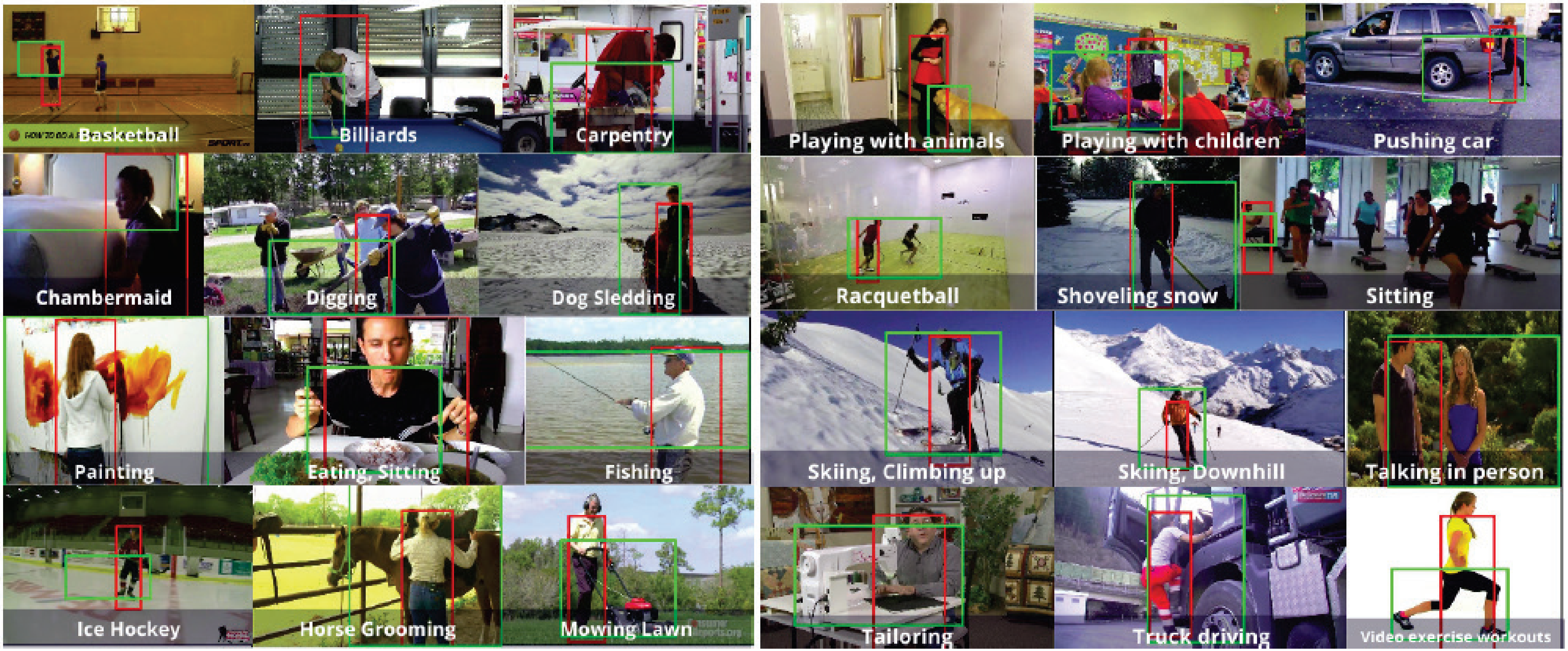
where $\phi(r;I)$ is a vector of features extracted from region r in I, while $\mathbf{w}_{\mathrm{P}}^{\alpha}$ and $\mathbf{w}_{\mathrm{S}}^{\alpha}$ are the primary and secondary weights for action α respectively. $R(r;I)$ defines the set of candidates for the secondary region.

$$P(\alpha|I,r) = \frac{\exp(\text{score}(\alpha;I,r))}{\sum_{\alpha' \in A} \exp(\text{score}(\alpha';I,r))}$$

AP (%)	Jumping	Phoning	Playing Instrument	Reading	Riding Bike	Riding Horse	Running	Taking Photo	Using Computer	Walking	mAP
RCNN	88.7	72.6	92.6	74.0	96.1	96.9	86.1	83.3	87.0	71.5	84.9
Random-RCNN	89.1	72.7	92.9	74.4	96.1	97.2	85.0	84.2	87.5	70.4	85.0
Scene-RCNN	88.9	72.5	93.4	75.0	95.6	98.1	88.6	83.2	90.4	71.5	85.7
R*CNN (0.0, 0.5)	89.1	80.0	**95.6**	81.0	**97.3**	98.7	85.5	**85.6**	93.4	71.5	87.8
R*CNN (0.2, 0.5)	88.1	75.4	94.2	80.1	95.9	97.9	85.6	84.5	92.3	**71.6**	86.6
R*CNN (0.0, 1.0)	**89.2**	77.2	94.9	**83.7**	96.7	**98.6**	87.0	84.8	93.6	70.1	87.6
R*CNN (0.2, 0.75)	88.9	79.9	95.1	82.2	96.1	97.8	**87.9**	85.3	94.0	71.5	**87.9**
R*CNN (0.2, 0.75, 2)	87.7	**80.1**	94.8	81.1	95.5	97.2	87.0	84.7	**94.6**	70.1	87.3

AP (average precision) on the PASCAL VOC Action 2012 val set. *RCNN* is the baseline approach, with the ground-truth region being the primary region. *Random-RCNN* is a network trained with primary the ground-truth region and secondary a random region. *Scene-RCNN* is a network trained with primary the ground-truth region and secondary the whole image. *R*CNN* (l, u) is our system where l, u define the lower and upper bounds of the allowed overlap of the secondary region with the ground truth. *R*CNN* (l, u, n_S) is a variant in which n_S secondary regions are used, instead of one.


Riding Horse
Running
Taking Photo
Using Computer
Walking



Basketball
Billiards
Carpentry
Chambermaid
Digging
Dog Sledding
Painting
Eating, Sitting
Fishing
Ice Hockey
Horse Grooming
Mowing Lawn
Playing with animals
Playing with children
Pushing car
Racquetball
Shoveling snow
Sitting
Skiing, Climbing up
Skiing, Downhill
Talking in person
Tailoring
Truck driving
Video exercise workouts

Faster R-CNN [Ren et al, 2015]

Region Proposal Network:

- Takes an image as input and outputs rectangular object proposals.
- By sharing conv layers with the classifier network, the Region Proposal Network is faster than selective search method (200ms per image)