3 Object Detection

BVM 2018 Tutorial: Advanced Deep Learning Methods

Paul F. Jaeger, Division of Medical Image Computing

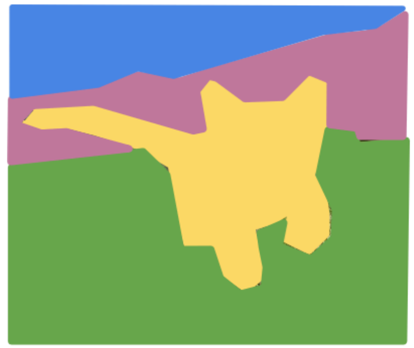
What is object detection?

classification

CAT

(1 label per image)

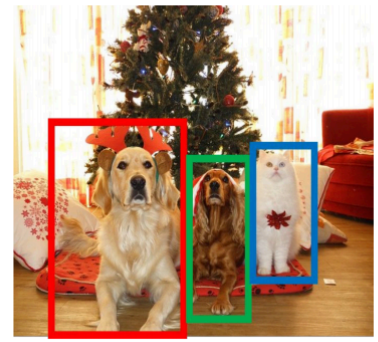
segmentation



GRASS, CAT, TREE, SKY

(1 label per pixel)

obj. detection



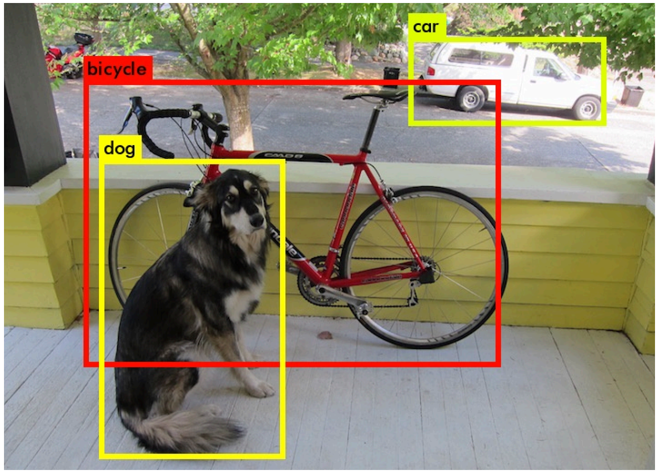
DOG, DOG, CAT

(1 label per object)

Source: CS231n: Convolutional Neural Networks for Visual Recognition. Fei-Fei Li, Justin Johnson, Serena Young, 2017.

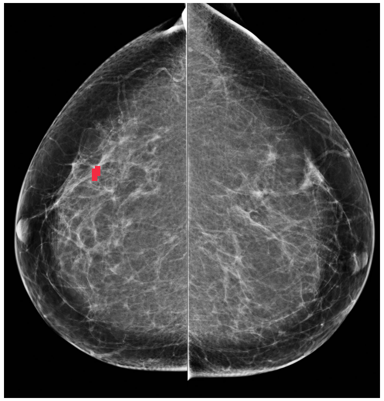
Is my problem a detection problem?

1. Locate and classify multiple instances in an image:



Source: You only look once, Redmon et al., 2015

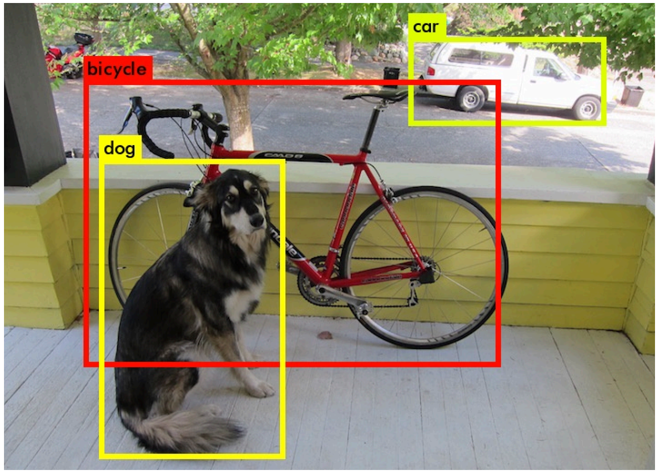
2. Whole image classification with little training data and one or more labeled regions of interest (ROIs) (~*supervised attention*):



Source: The Radiology Assistant : Bi-RADS for Mammography and Ultrasound 2013

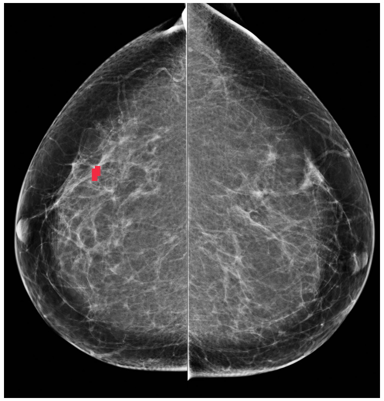
Is my problem a detection problem?

1. Locate and classify multiple instances in an image:



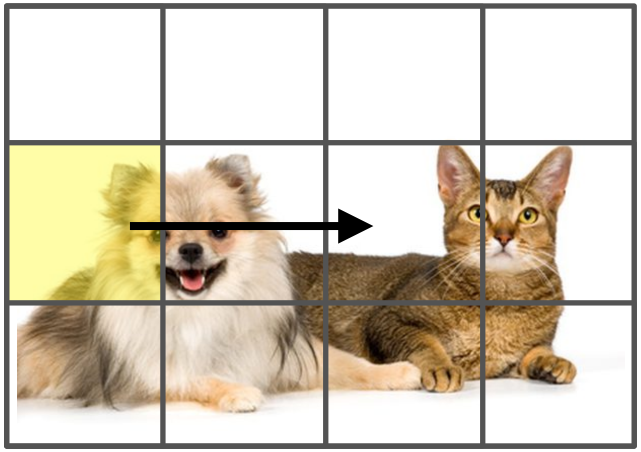
Source: You only look once, Redmon et al., 2015

2. Whole image classification with little training data and one or more labeled regions of interest (ROIs) (~*supervised attention*):



Source: The Radiology Assistant : Bi-RADS for Mammography and Ultrasound 2013

Detection via Sliding window Classification

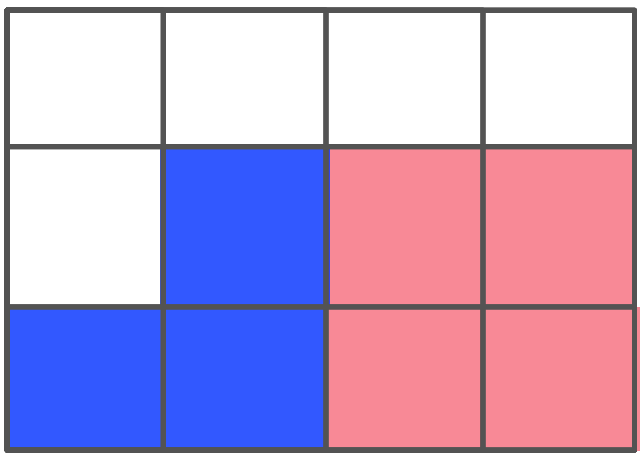


Dog ?

Cat ?

Downsides:

- Need to apply huge amount of windows and scales!
- fixed patch size might not match variably sized objects.

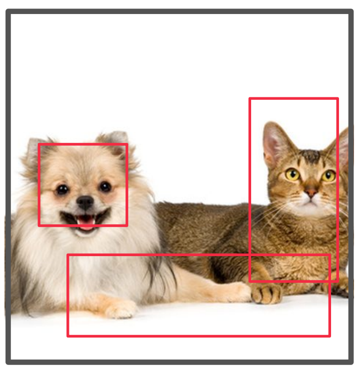


Output heatmap with detected dog and cat patches.

Two shot detection networks

Idea: Apply classifier only on sparse regions proposed by a preceding model!

1. Region Proposal Network



2. Classification Network

Cat 0.3

Dog 0.7

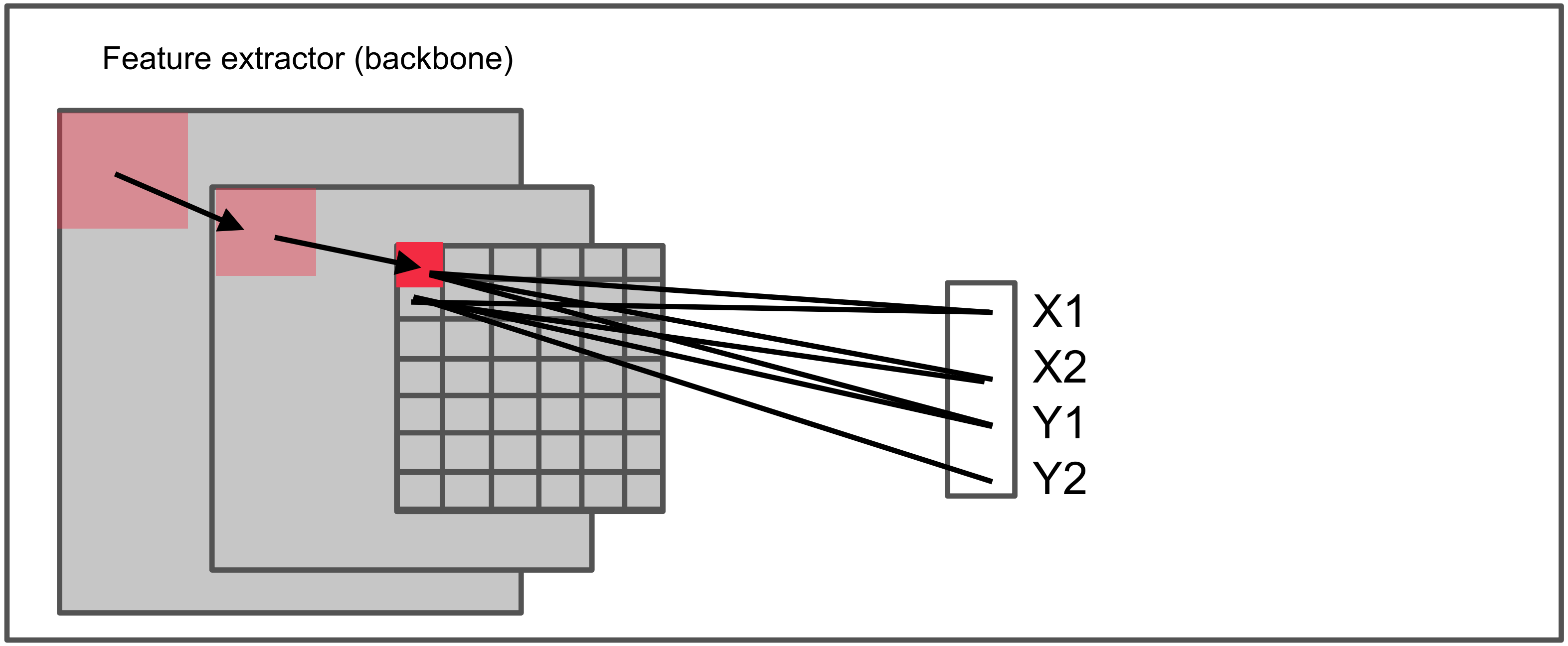
Cat 0.7

Dog 0.3

Can be trained end-to-end!

Region proposals: Via Localisation?

Localisation Network

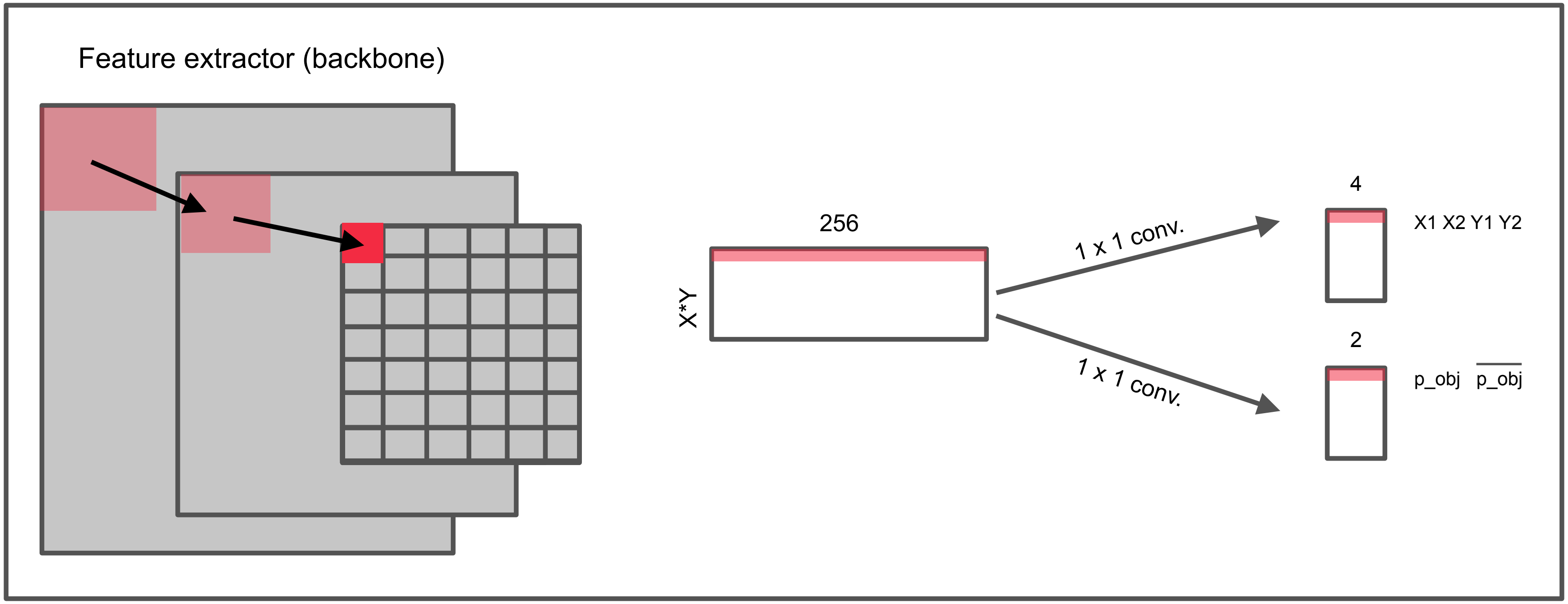


- too much pooling decreases spatial information.
- large feature maps need to be fully connected to reg. outputs.
- **Only 1 object per image possible!**

too many parameters!

Region proposals

Region Proposal Network (RPN)

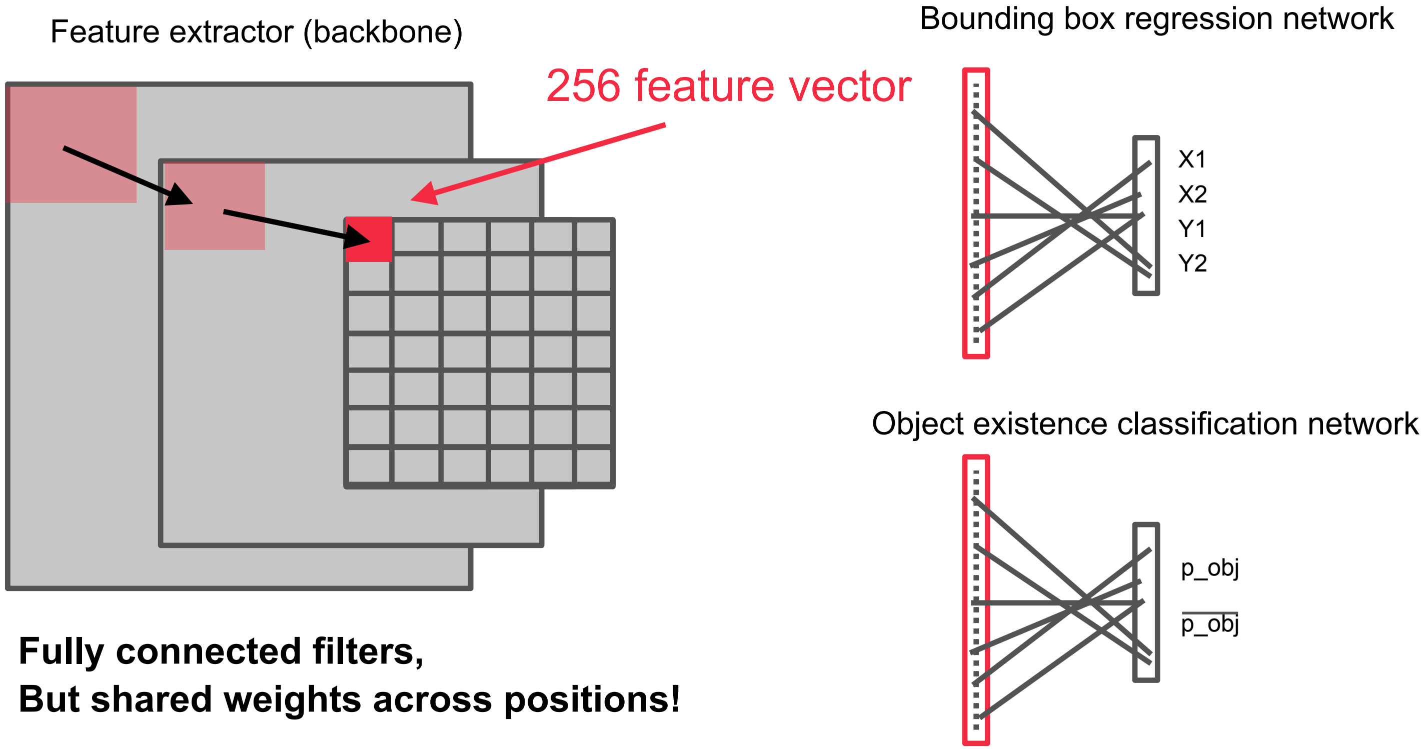


- Fully convolutional architecture. Fewer paramters!
- **multiple objects per image position possible!**
- Additional object existence classification enables RPN to assign a score to all proposals.

Region proposals

Passing the feature vectors of all positions to a fully connected network successively,
Is the same as performing a 1 x 1 convolution!

Region Proposal Network

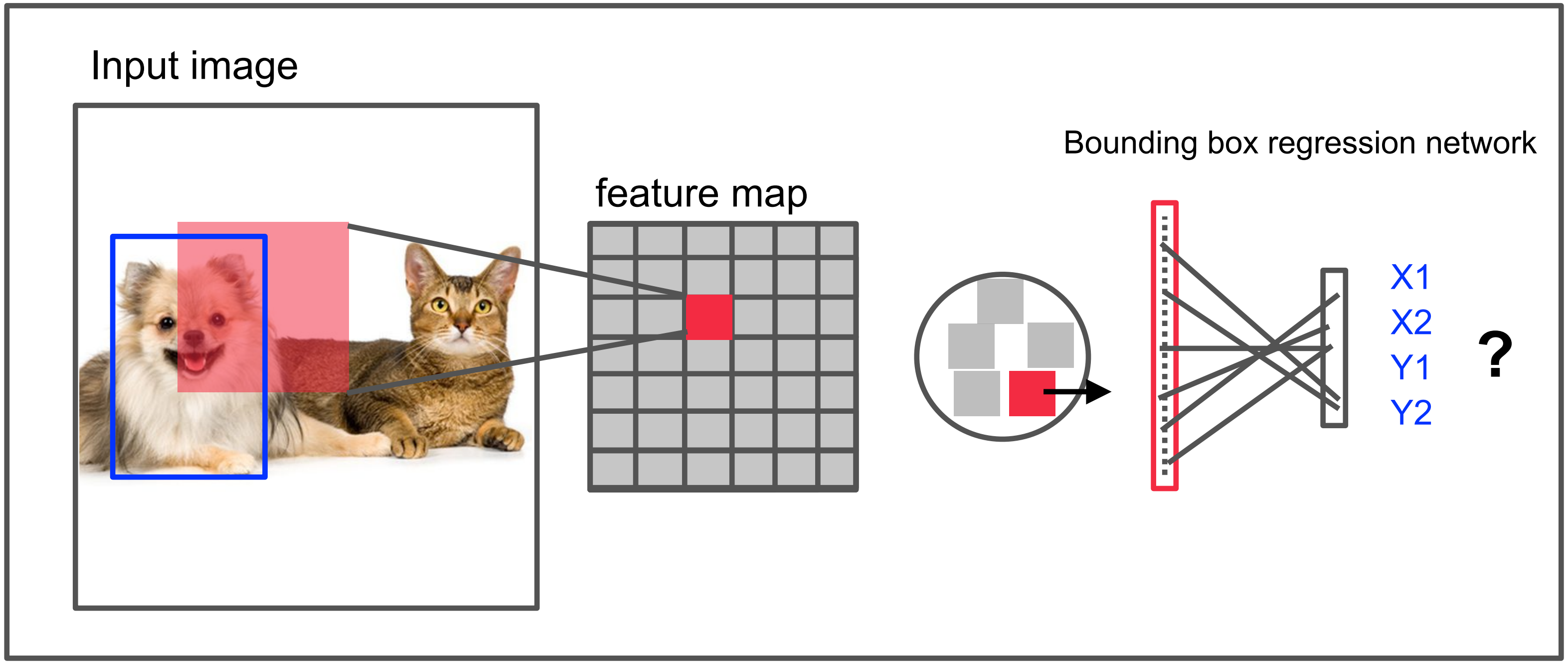


Fully connected filters,
But shared weights across positions!

Anchor Boxes (Reference Boxes)

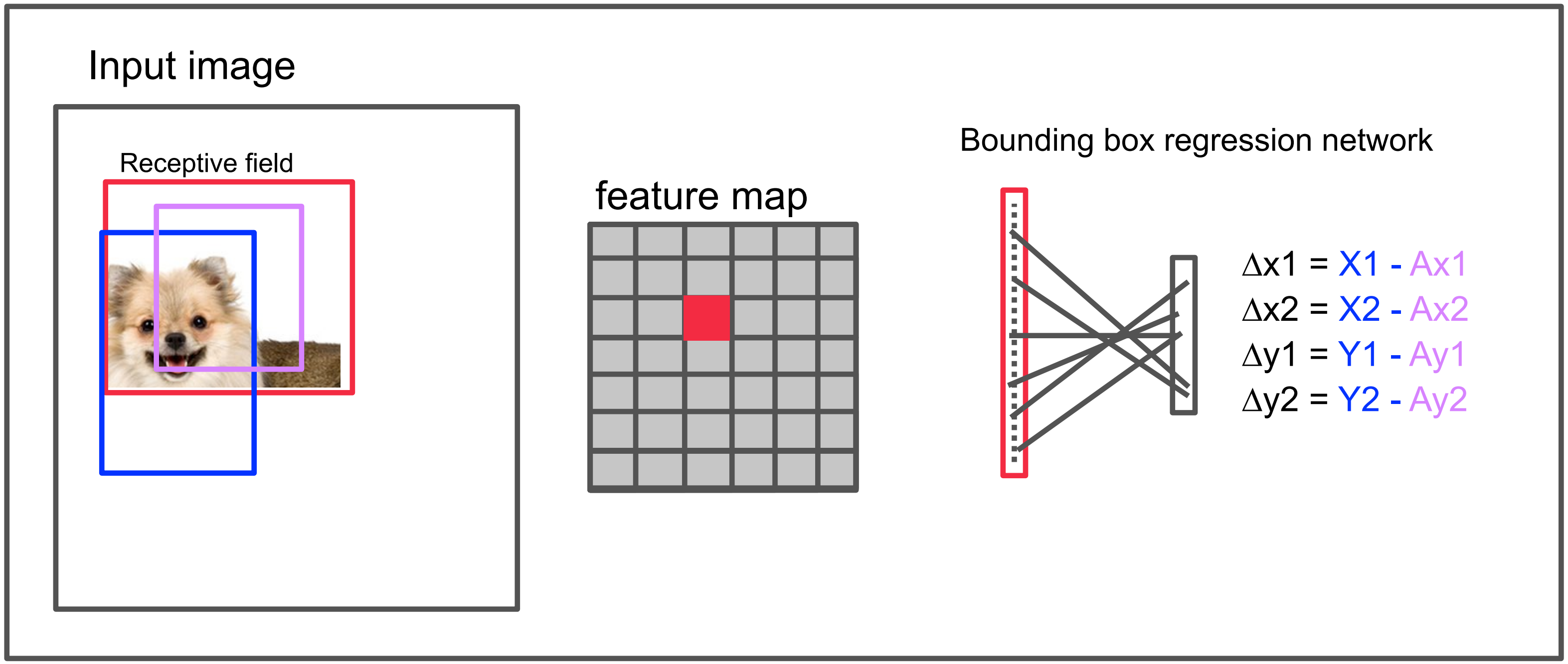
How can input coordinates be regressed if the spatial information is lost in fully conv. architecutre?

- Learned information is contained in kernel weights, which are shared across positions
- kernel does not “know where it is”, input coordinates have no meaning.



Anchor Boxes (Reference Boxes)

Solution: Encode position information into the target coordinates via Anchor Boxes!

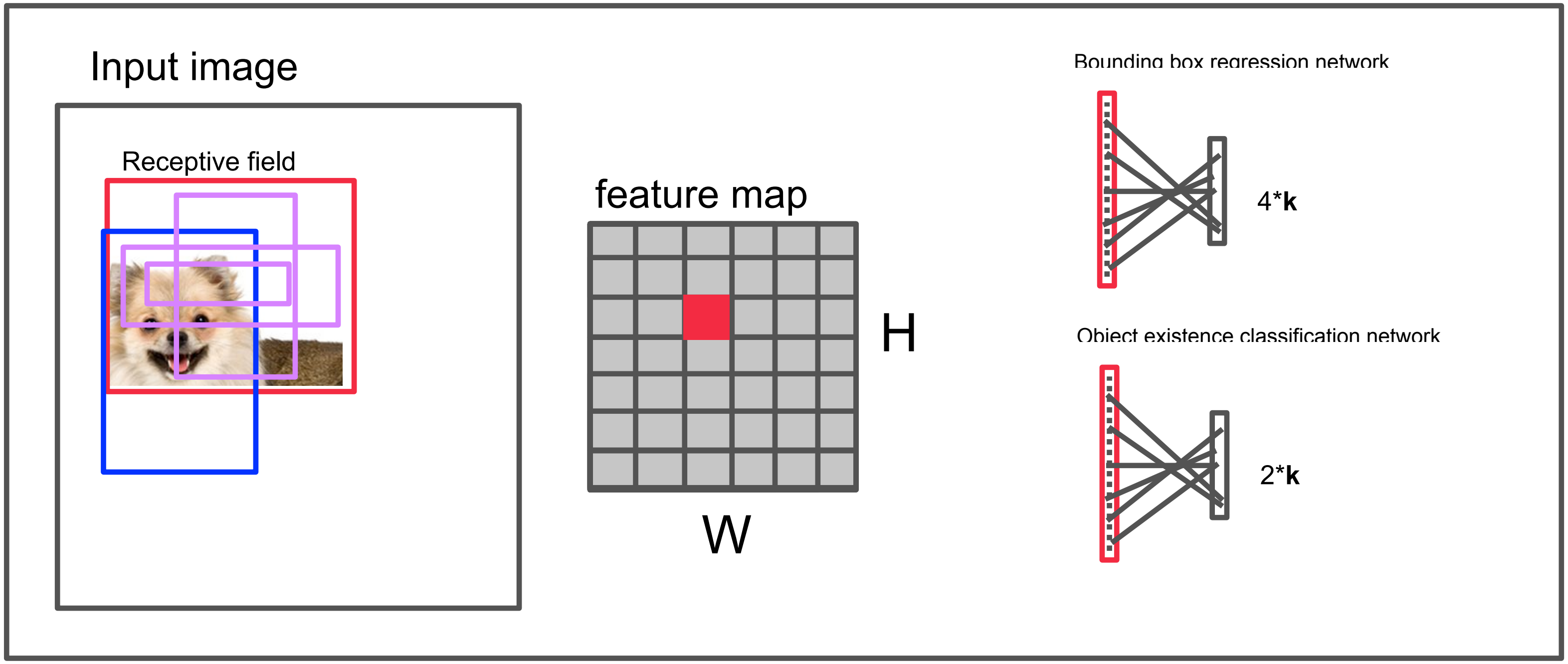


- For each feature map position: Pre-define an anchor box centered in the corresponding receptive field region of the input image.
- Enables Network to predict box coordinates relative to the respective feature map position.
- For proposal generation (and during test time) simply unfold absolute coordinates: $x1 = \Delta x1 + Ax1$

Anchor Boxes (Reference Boxes)

Learning separate weights for different scales and ratios improves performance.

- assigning multiple (**k**) anchors per position!

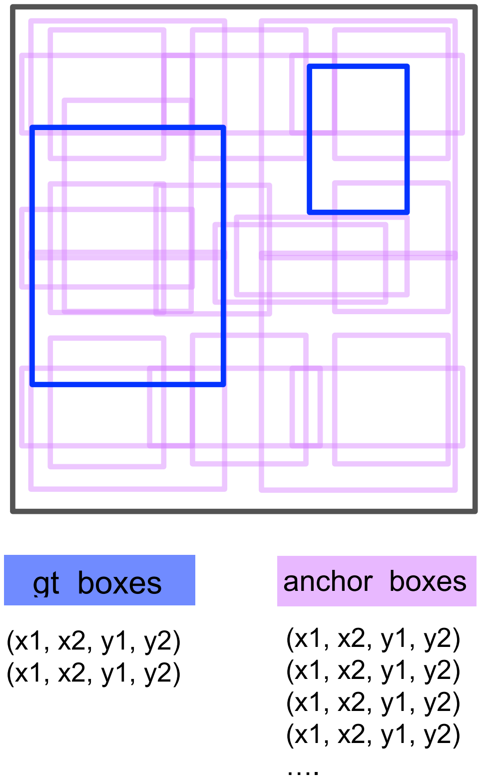


- in total the RPN proposes one region per anchor: H*W*k ~ thousands.

RPN training

How to regress multiple ground truth bounding boxes per image?

Input image



IoU matching

	a1	a2	a3
gt_box 1	0.1	0.8	0.5
gt_box 2	0.75	0.2	0.0

Anchor target scores

	a1	a2	a3
gt_box 1	-1	1	0
gt_box 2	1	-1	-1

IoU	Score
< 0.3	-1
0.3 - 0.7	0
> 0.7	1

RPN loss function

$$L(\{p_i\}, \{t_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*).$$

- every gt_box gets assigned at least one anchor.
- Every anchor gets assigned at most one gt_box.
- Compute Δ target coordinates for positive anchors.
- Sample pos. and neg. anchors for loss according to desired ratio (e.g. 1:3).

Non-maximum surpression (NMS)

RPN puts out a list of thousands of proposals. Filter them by NMS:

Proposals

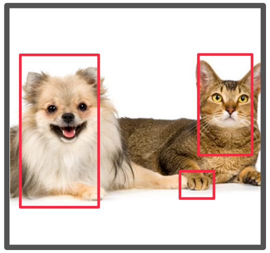
(x1, x2, y1, y2, p_score)
(x1, x2, y1, y2, p_score)
(x1, x2, y1, y2, p_score)
(x1, x2, y1, y2, p_score)
….

NMS

- Compute IoU matrix for all proposals
- Define threshold x.
- For proposal clusters with IoU > x, keep only the one with the highest p_score

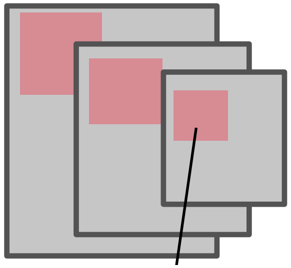
SOTA Two shot detector: Faster RCNN

1. RPN

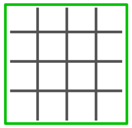


RPN outputs: proposals on input image.

2. RoiPooling



Feed proposal through RPN backbone net (shared features!)



Max pool to fixed-sized grid

3a. Classification network

fully connected

Softmax probs across classes

Cat 0.7
Dog 0.2
Background 0.1

3b. BBox refinement network

fully connected

Δ coordinates between proposal and predicted gt_box (= RPN reg. error)

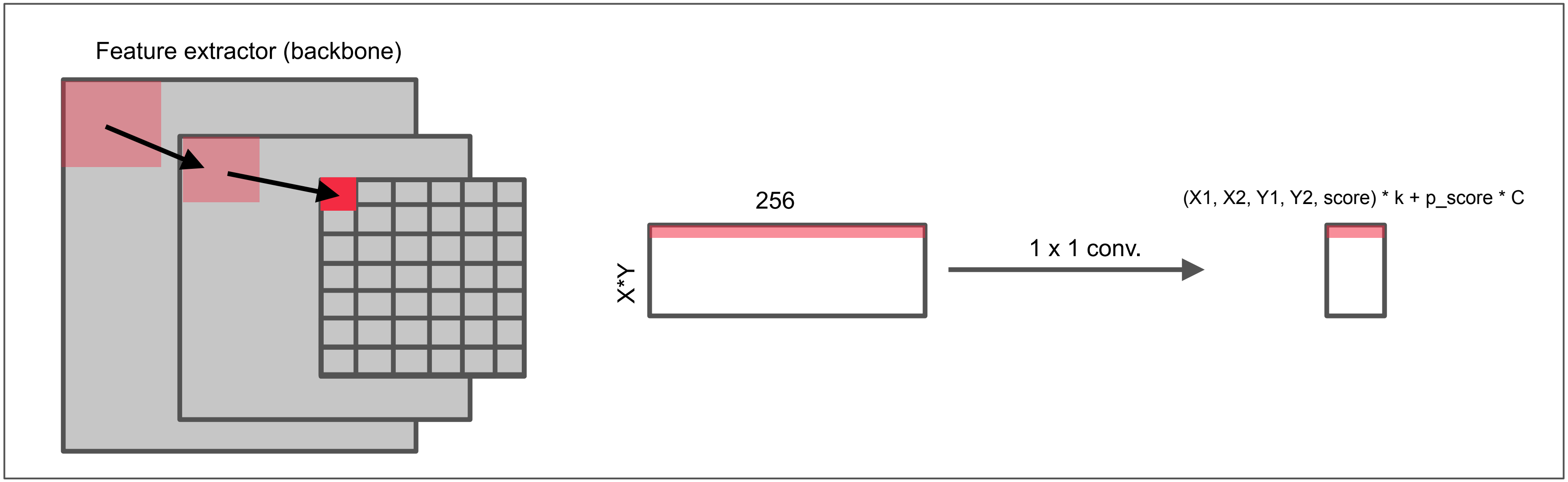
(Normalized and scaled invariant coordinates)

For final loss:

- sample proposals according to desired ratio
- activate regression loss only for foreground props.

REN, Shaoqing, et al. Faster r-cnn: Towards real-time object detection with region proposal networks. NIPS 2015

One shot detectors

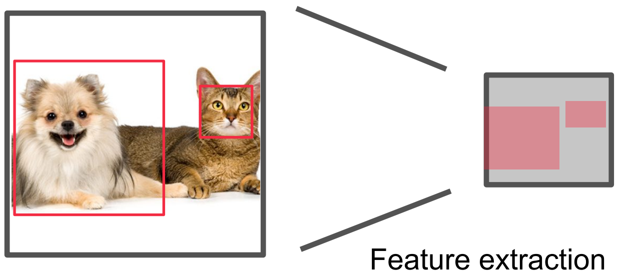


- Perform final classification right at RPN. No Sparse region selection ~ **Dense Detectors**.
- Trade off some accuracy for a significant test time speed up (questionable trade for most MIC problems) [1, 2]
- “Focal loss” claims the decrease in accuracy comes from the inefficient loss sampling (“hard negative mining”) and solves it by adapting the loss function [3]

1. Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." *CVPR*. 2016.
2. Liu, Wei, et al. "Ssd: Single shot multibox detector." *ECCV*. Springer, Cham, 2016.
3. Lin, Tsung-Yi, et al. "Focal loss for dense object detection." *arXiv preprint arXiv:1708.02002* (2017).

Feature Pyramid Networks (FPN)

Scaling semantics problem



Extracting features for all proposal scales at the same depth level leads to different semantics levels across proposals!

Solution: Extract features at multiple scales.

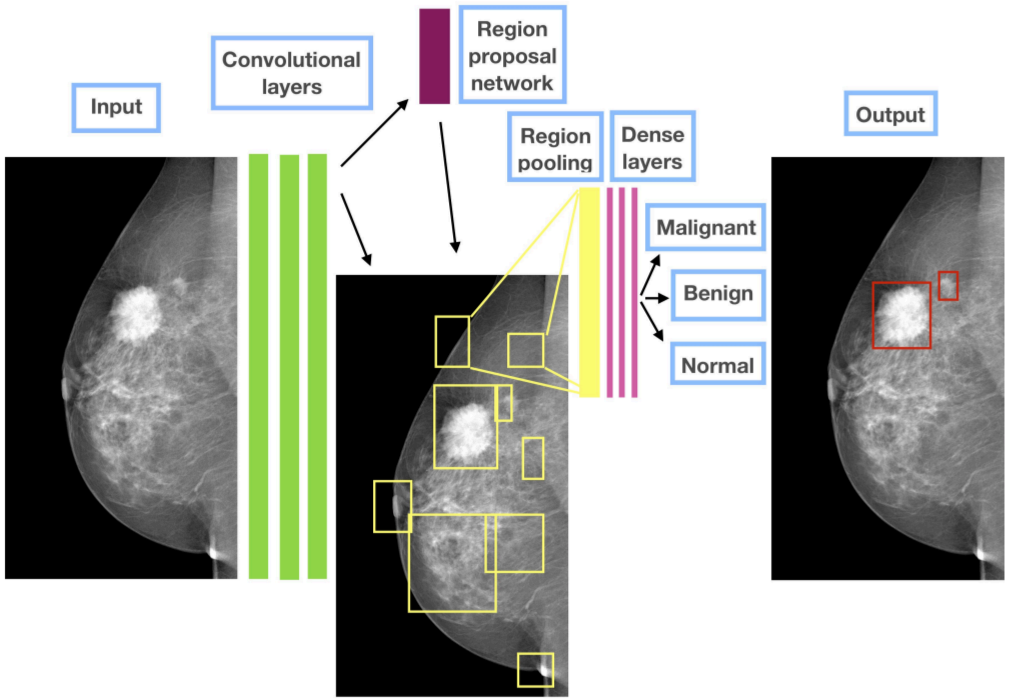
what we did so far. | no computational effort, but weak semantics. | computationally very expensive | FPN: way to go.

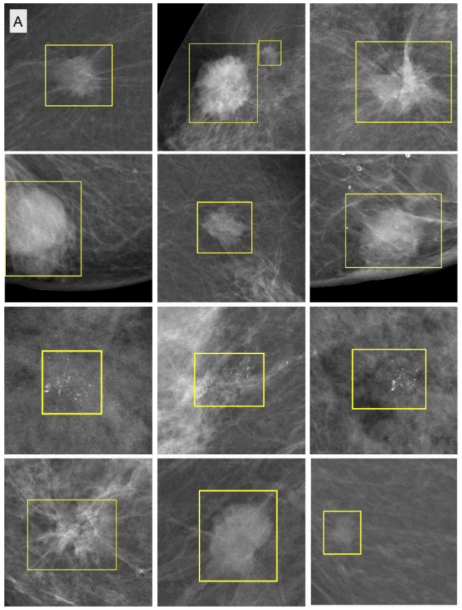
sota

- Performing object detection on multiple scales of the input image solves the scale invariance problem.
- Most sota object detectors use feature pyramids as backbone networks. (For Larger proposals features can be extracted from deeper layers).

Lin, Tsung-Yi, et al. "Feature pyramid networks for object detection." *CVPR*. Vol. 1. No. 2. 2017.

Faster RCNN in action



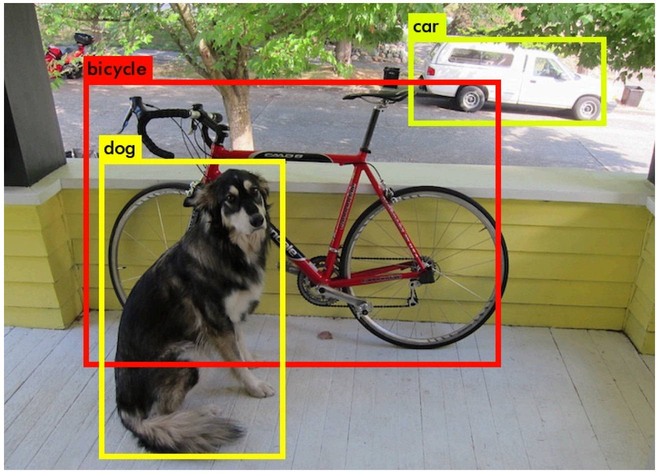


- 2nd position in the Digital Mammography DREAM Challenge with an AUC = 0.85. (~1000 teams)
- Very large images (~5000x5000) with mostly small regions of interest.
- Model generalises well across different data sets without finetuning!

Ribli, Dezső, et al. "Detecting and classifying lesions in mammograms with Deep Learning." *arXiv preprint arXiv:1707.08401* (2017).

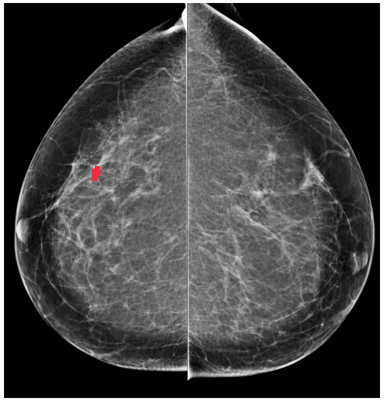
Is my problem a detection problem?

1. Locate and classify multiple instances in an image:



Source: You only look once, Redmon et al., 2015

2. Whole image classification with little training data and one or more labeled regions of interest (ROIs) (~*supervised attention*):



Source: The Radiology Assistant : Bi-RADS for Mammography and Ultrasound 2013

Questions?