Object detection: where then what

D.A. Forsyth

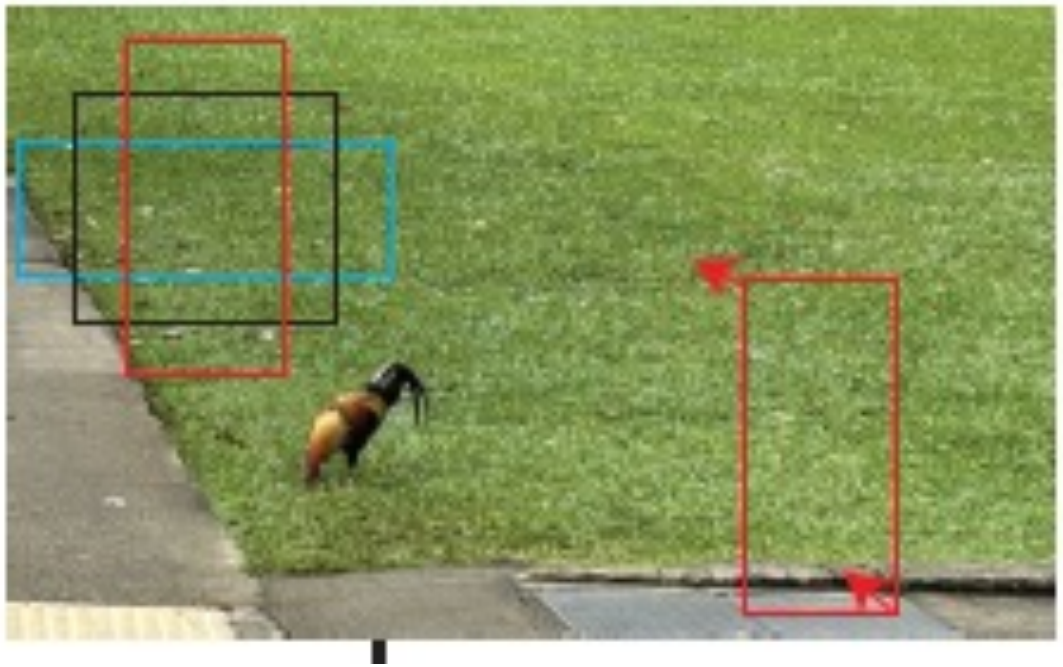
University of Illinois at Urbana Champaign

Faster RCNN

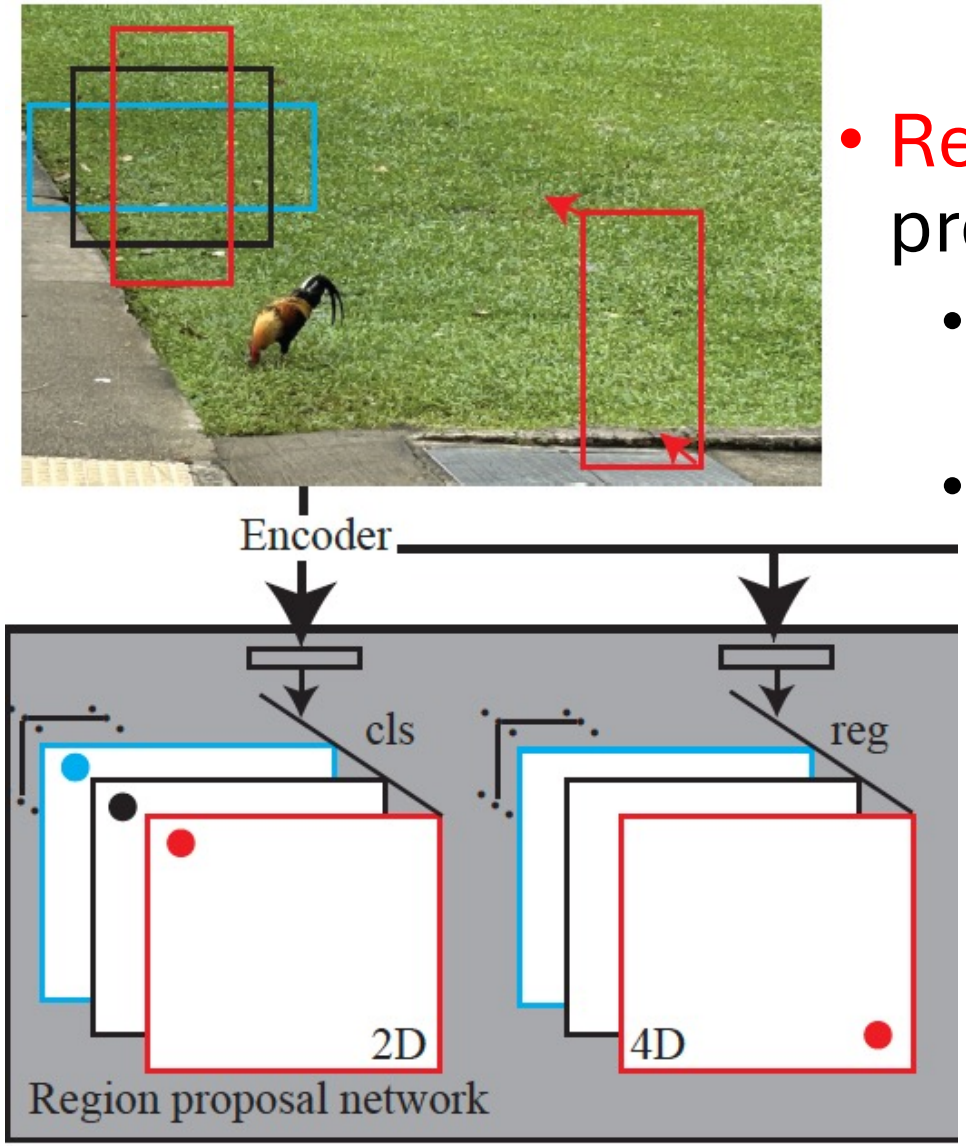
- Ideas:
  - Sample the world of boxes
  - Decide which boxes are worth looking at
  - Score boxes
  - Cleanup
    - Threshold
    - Non-maximum suppression
    - Bounding box regression
- Build all this on top of an image encoder

Sampling the world of boxes

- 4 dimensional
  - boxes are axis aligned (eg top left, bottom right)
- Samples don’t need to be uniform across dims
- Sampling:
  - Even grid of centers across image (S x T)
  - At each center, 3 scales and 3 aspect ratios

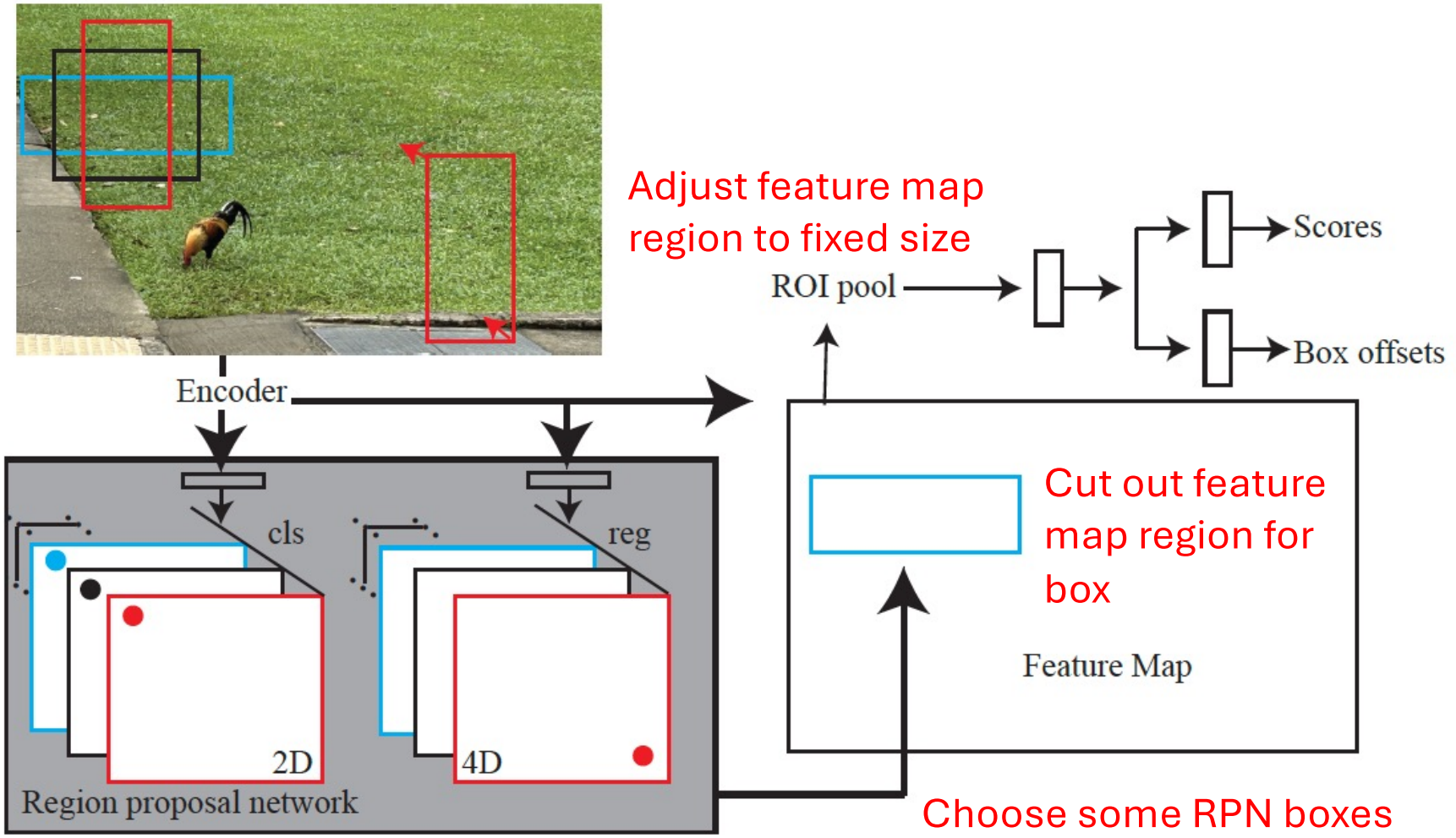


Which boxes are worth looking at

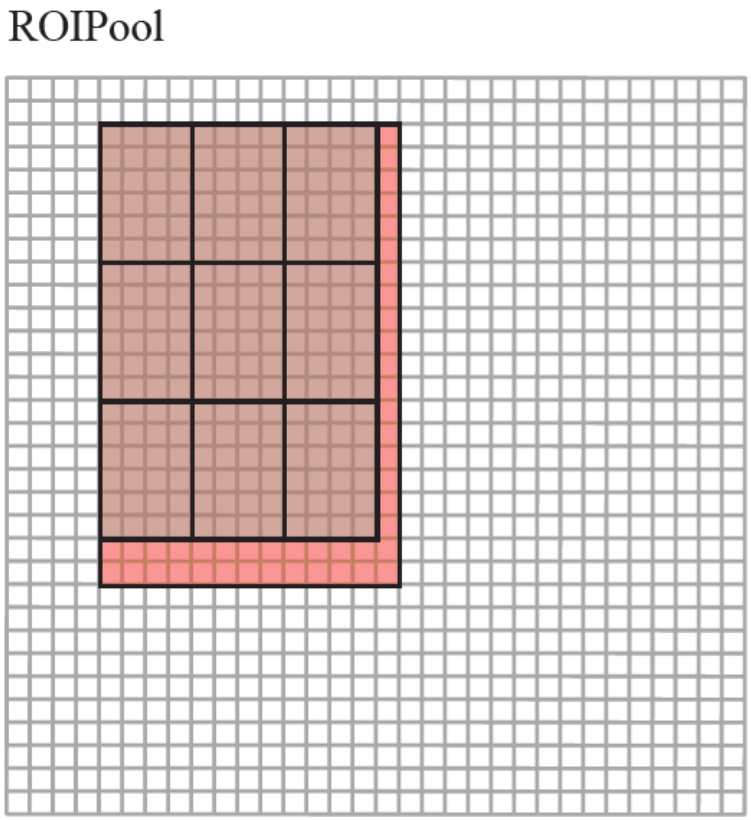


- Region proposal network produces:
  - (2x9)xSxT block of box scores (cls)
    - objectness score
  - (4x9)xSxT block of offsets (reg)
    - mild improvement to box corners to get score

Scoring the boxes



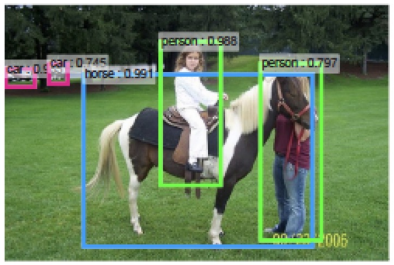
Adjusting the feature map region

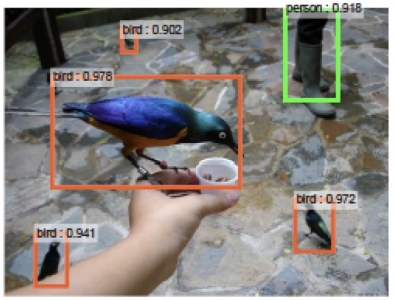


- Find corresponding window
  - by rounding
- Within subgrid, max-pool
  - subgrid by rounding

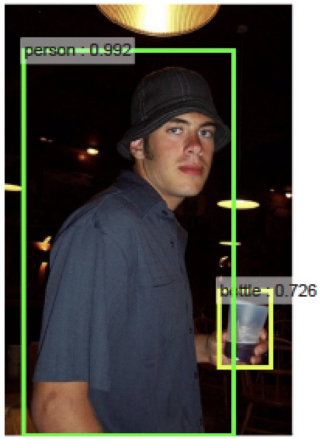
Figure 26.2 shows what happens when a 1024×1024 image produces a 32×32 feature map which is ROIPooled to a 9 dimensional vector for a given ROI. The top left corner of the ROI is at $(75, 140)$ in the original image, and the size of the ROI is 650×430. In this case, ROIPool would round the corner location to $(2, 4) = (\lfloor 75/32 \rfloor, \lfloor 140/32 \rfloor)$, the size to $20 \times 13 = \lfloor 650/32 \rfloor \times \lfloor 430/32 \rfloor$.

Detection

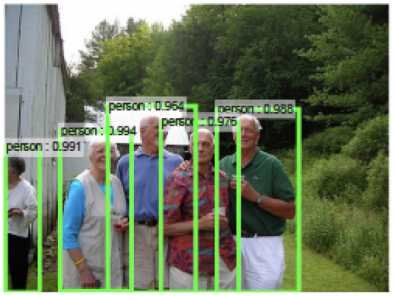

person : 0.988
car : 0.745
horse : 0.991
person : 0.797

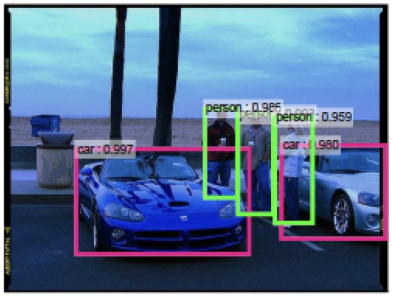

person : 0.918
bird : 0.902
bird : 0.978
bird : 0.972
bird : 0.941

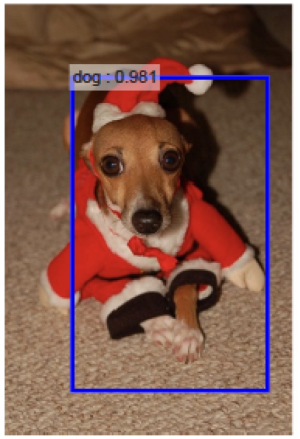

cow : 0.995
cow : 0.998


person : 0.992
bottle : 0.726

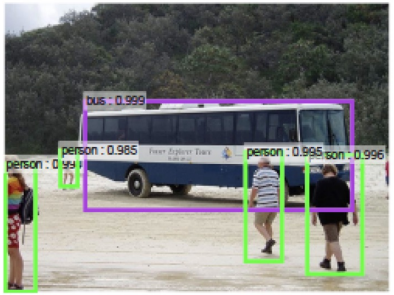

car : 0.999
person : 0.925

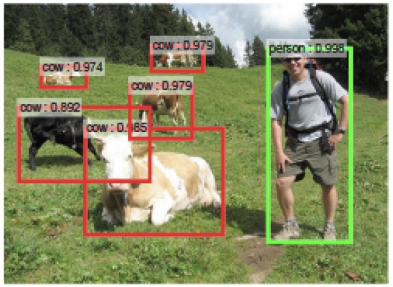

person : 0.964
person : 0.988
person : 0.994
person : 0.975
person : 0.991


person : 0.985
person : 0.959
car : 0.997
car : 0.980


dog : 0.981


person : 0.961
person : 0.958
person : 0.960
person : 0.757
dog : 0.697


bus : 0.999
person : 0.985
person : 0.995
son : 0.996


cow : 0.979
cow : 0.974
cow : 0.979
cow : 0.892
person : 0.936

Training requires care

- Data:
  - labelled boxes on images
- I) Train RPN using pretrained encoder
  - Finetune ImageNet encoder
  - Note class imbalance
    - there are more empty than object boxes
- II) Train classifier using pretrained encoder
  - Finetune different ImageNet encoder
  - Use RPN boxes (now two encoders!)
- III) Finetune RPN layers on fixed encoder from II
- IV) Finetune classifier, etc
  - using RPN, encoder from III

Mask RCNN

- Would like a better localization of object
- Get by:
  - modifying ROIPool
  - decoding result into mask as well.

ROIAlign preserves spatial detail

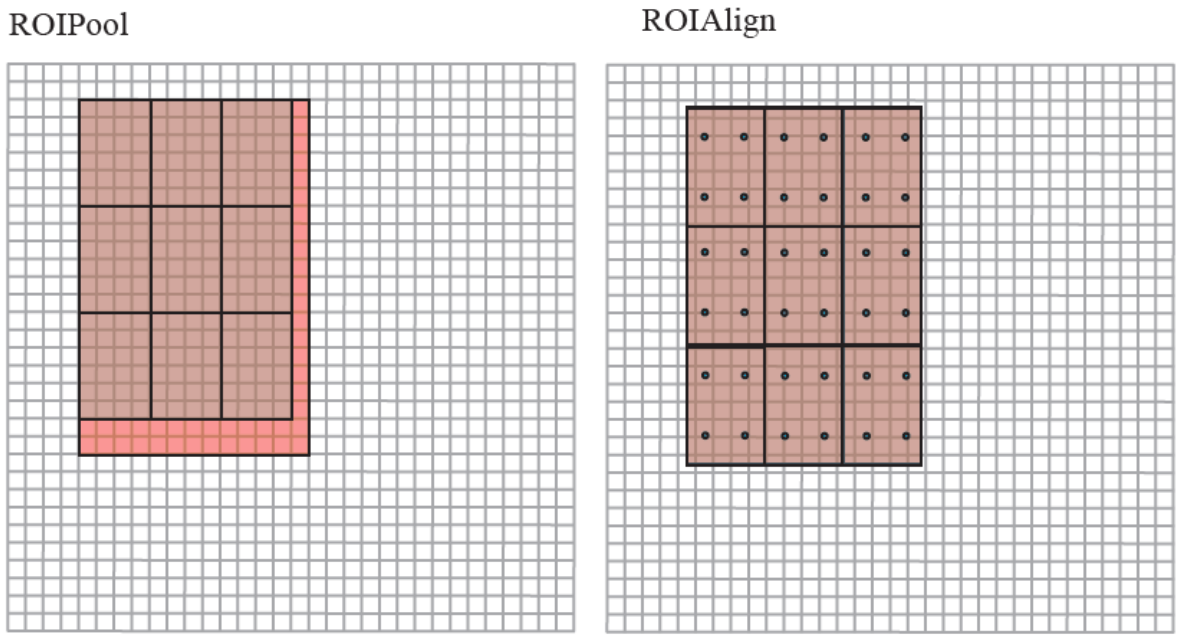
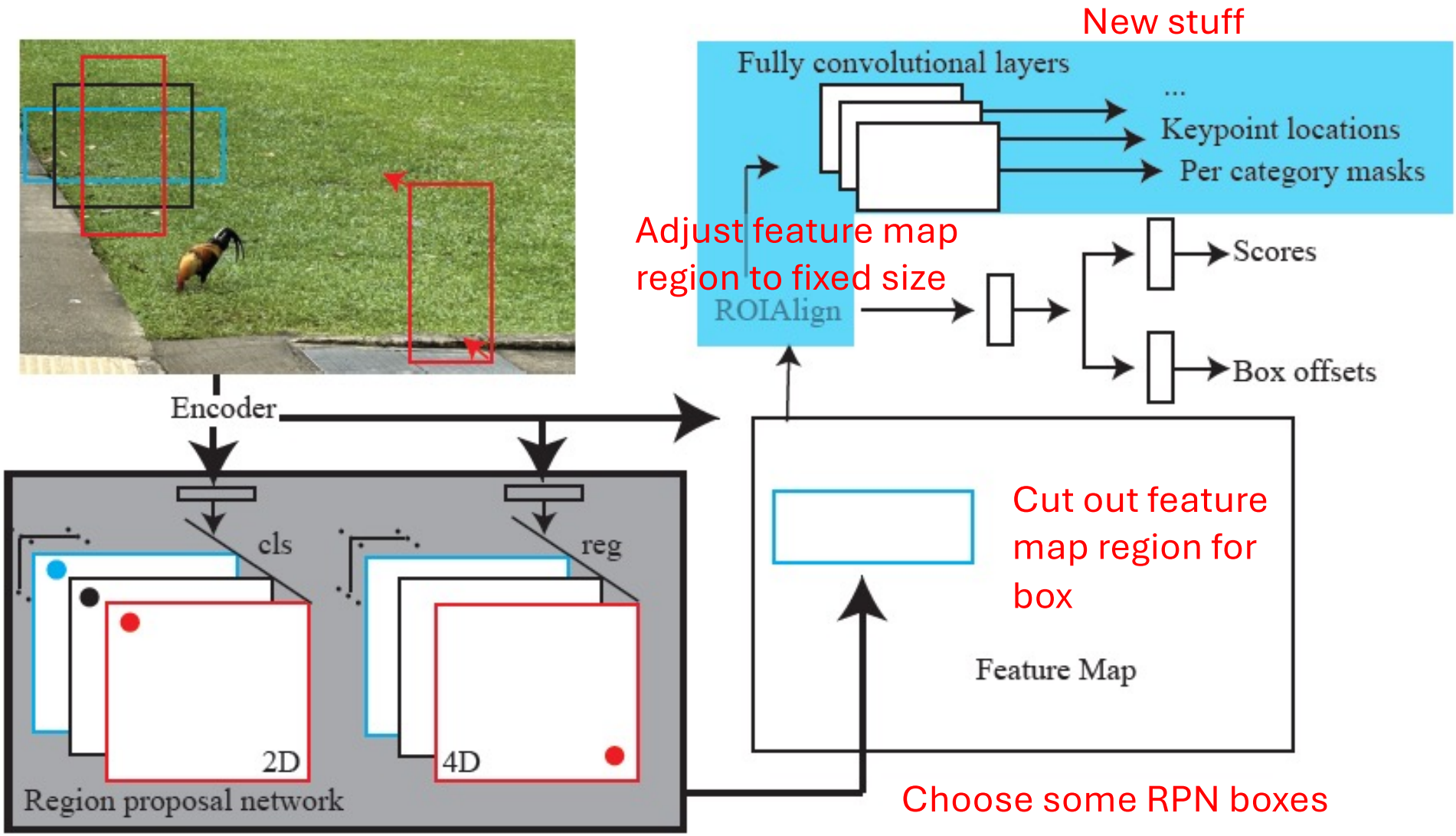
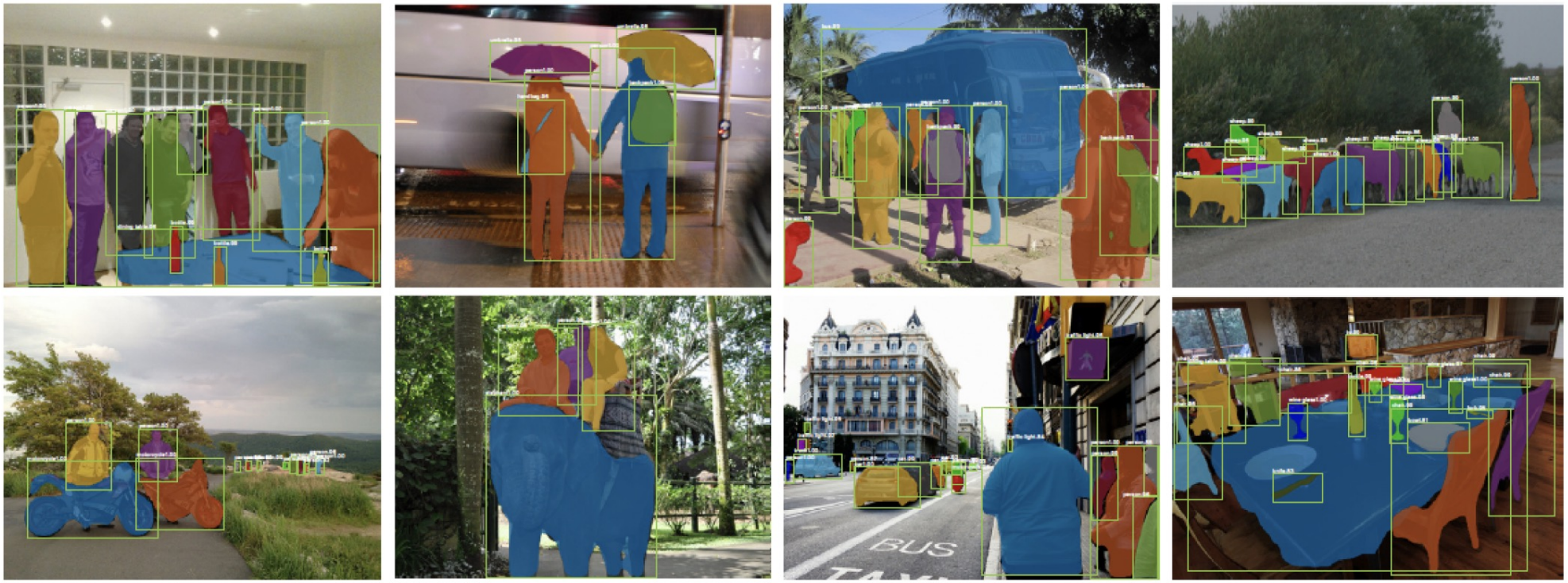


Figure 26.2 shows what happens when a 1024×1024 image produces a 32×32 feature map which is ROIPooled to a 9 dimensional vector for a given ROI. The top left corner of the ROI is at $(75, 140)$ in the original image, and the size of the ROI is 650×430. In this case, ROIPool would round the corner location to $(2, 4) = (\lfloor 75/32 \rfloor, \lfloor 140/32 \rfloor)$, the size to $20 \times 13 = \lfloor 650/32 \rfloor \times \lfloor 430/32 \rfloor$.

MaskRCNN


New stuff
Fully convolutional layers
...
Keypoint locations
Per category masks
Adjust feature map region to fixed size
ROIAlign
Scores
Box offsets
Encoder
cls
reg
2D
4D
Region proposal network
Cut out feature map region for box
Feature Map
Choose some RPN boxes

Detection with masks



- Evaluation:
  - you can compute IoU for masks, too…

Variants

- Predict more stuff from ROI
  - eg body keypoints

- Bolt onto a semantic segmenter
  - to get panoptic segmentation

Became Detectron, Detectron 2

https://detectron2.readthedocs.io/en/latest/

Things to think about…

- What other useful things could you predict with MaskRCNN?
- How do you guarantee performance?