CMP717
Image Processing

Semantic Segmentation

Erkut Erdem
Hacettepe University
Computer Vision Lab (HUCVL)

Semantic Segmentation

- Joint recognition & segmentation
  - segmenting all the objects in a given image and identifying their visual categories
- aka scene parsing or image parsing
- Early studies aim at segmenting out a single object of a known category
  - Borenstein & Ullman, 2002, Liebe & Schiele, 2003, etc.
- More recent work depends on CNNs
  - Farabet et al., 2013, Pinheiro and Collobert, 2014, Long et al., 2015, Noh et al., 2015

Computer Vision Tasks

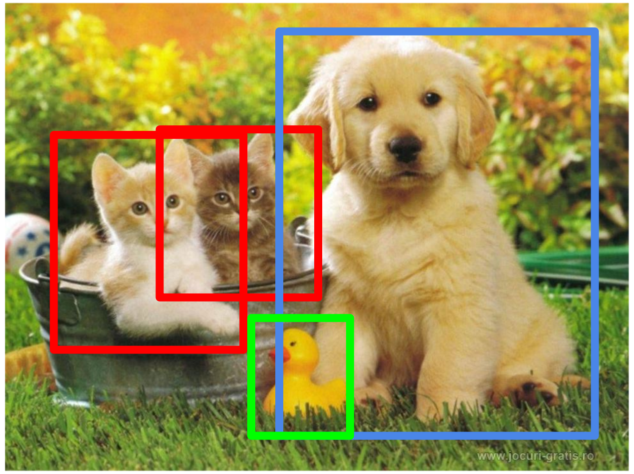
Classification

CAT

Classification + Localization

CAT

Object Detection



CAT, DOG, DUCK

Semantic Segmentation

CAT, DOG, DUCK

Single object

Multiple objects

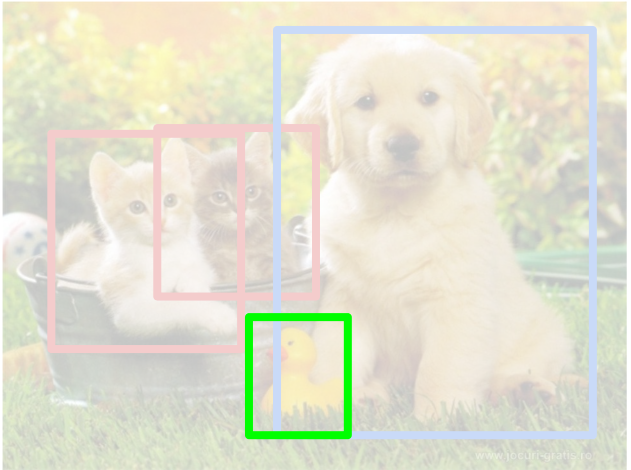
F.-F. Li, A. Karpathy and J. Johnson

Computer Vision Tasks

Classification

Classification
+ Localization

Object
Detection



Semantic
Segmentation

Today

Semantic Segmentation

Label every pixel in the image with a category label

Don't differentiate instances (cows)

Classic computer vision problem

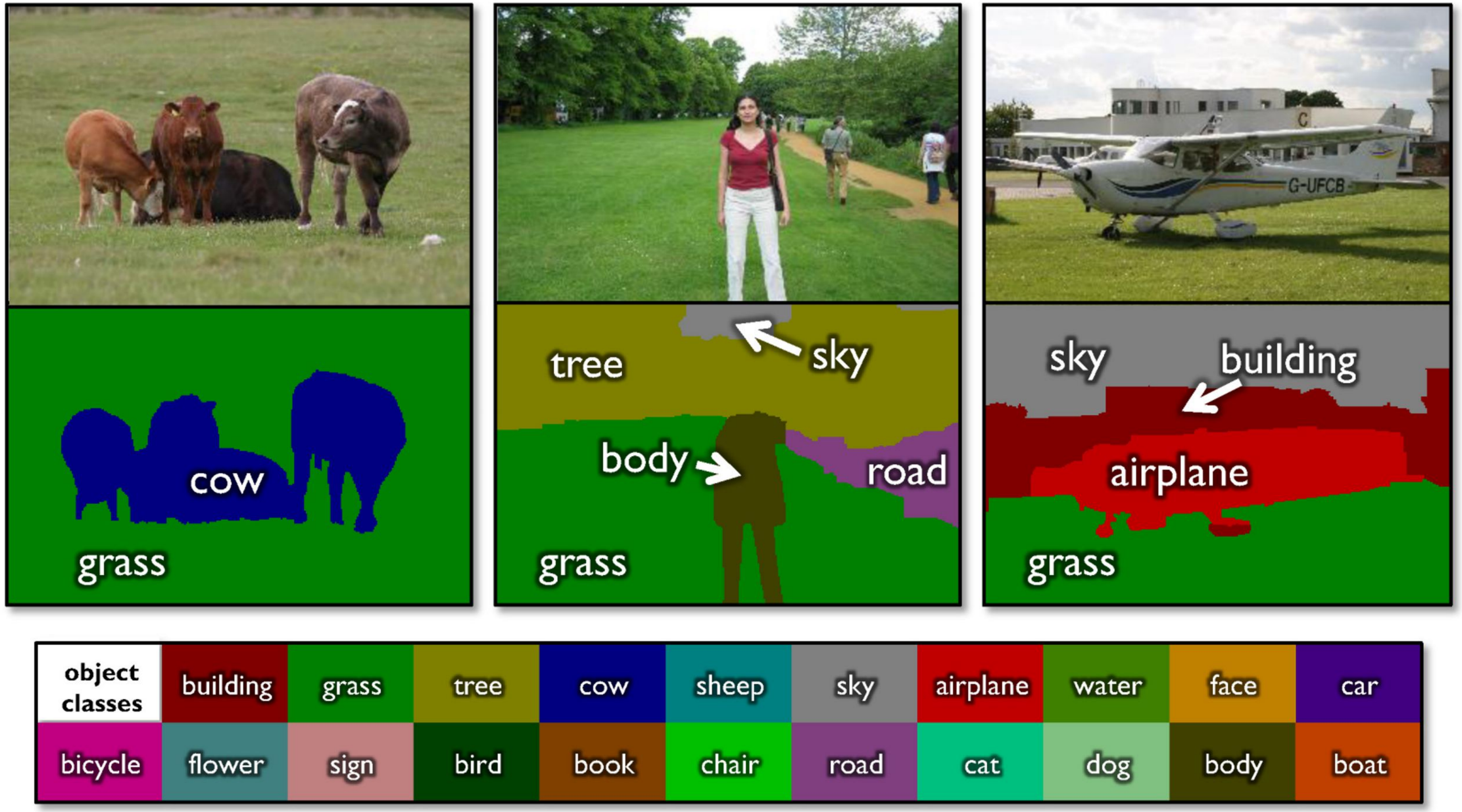


Figure credit: Shotton et al, "TextonBoost for Image Understanding: Multi-Class Object Recognition and Segmentation by Jointly Modeling Texture, Layout, and Context", IJCV 2007

Instance Segmentation

Detect instances, give category, label pixels

“simultaneous detection and segmentation” (SDS)

Lots of recent work (MS-COCO)

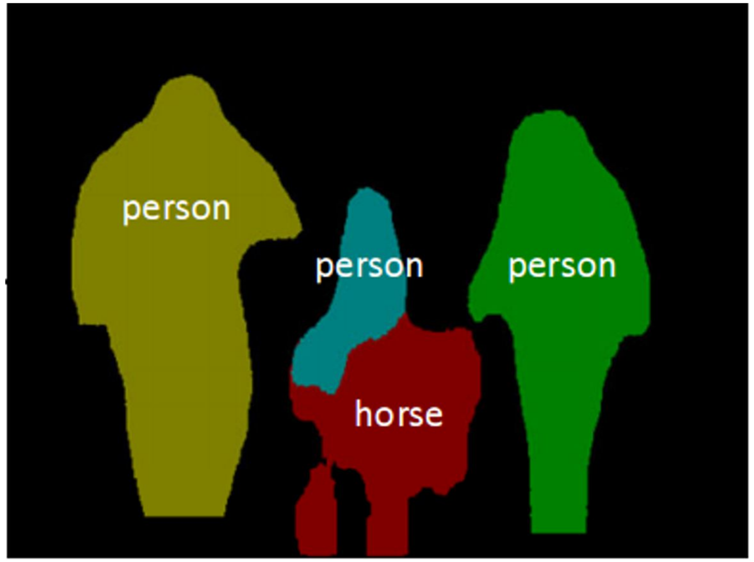
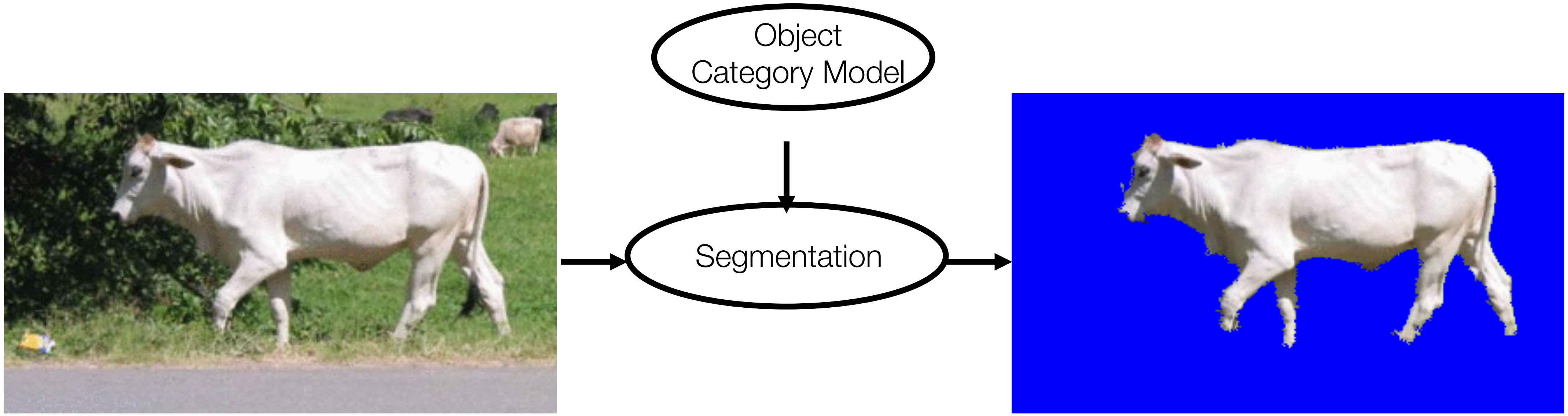


Figure credit: Dai et al, “Instance-aware Semantic Segmentation via Multi-task Network Cascades”, arXiv 2015

Early Studies of Semantic Segmentation

- Given an image and object category, to segment the object



Cow Image

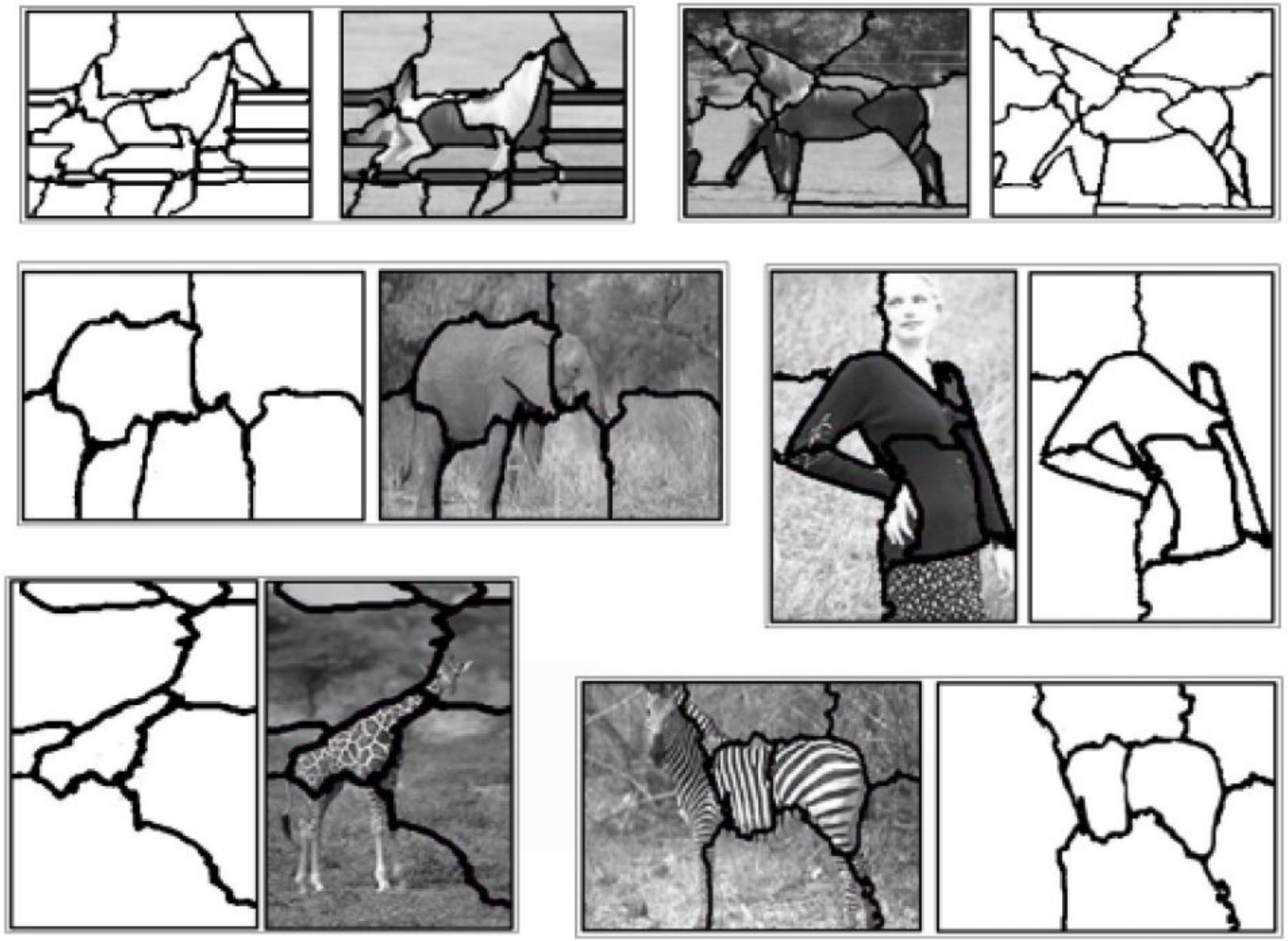
Segmented Cow

- Segmentation should (ideally) be
  - shaped like the object e.g. cow-like
  - obtained efficiently in an unsupervised manner
  - able to handle self-occlusion

M. P. Kumar

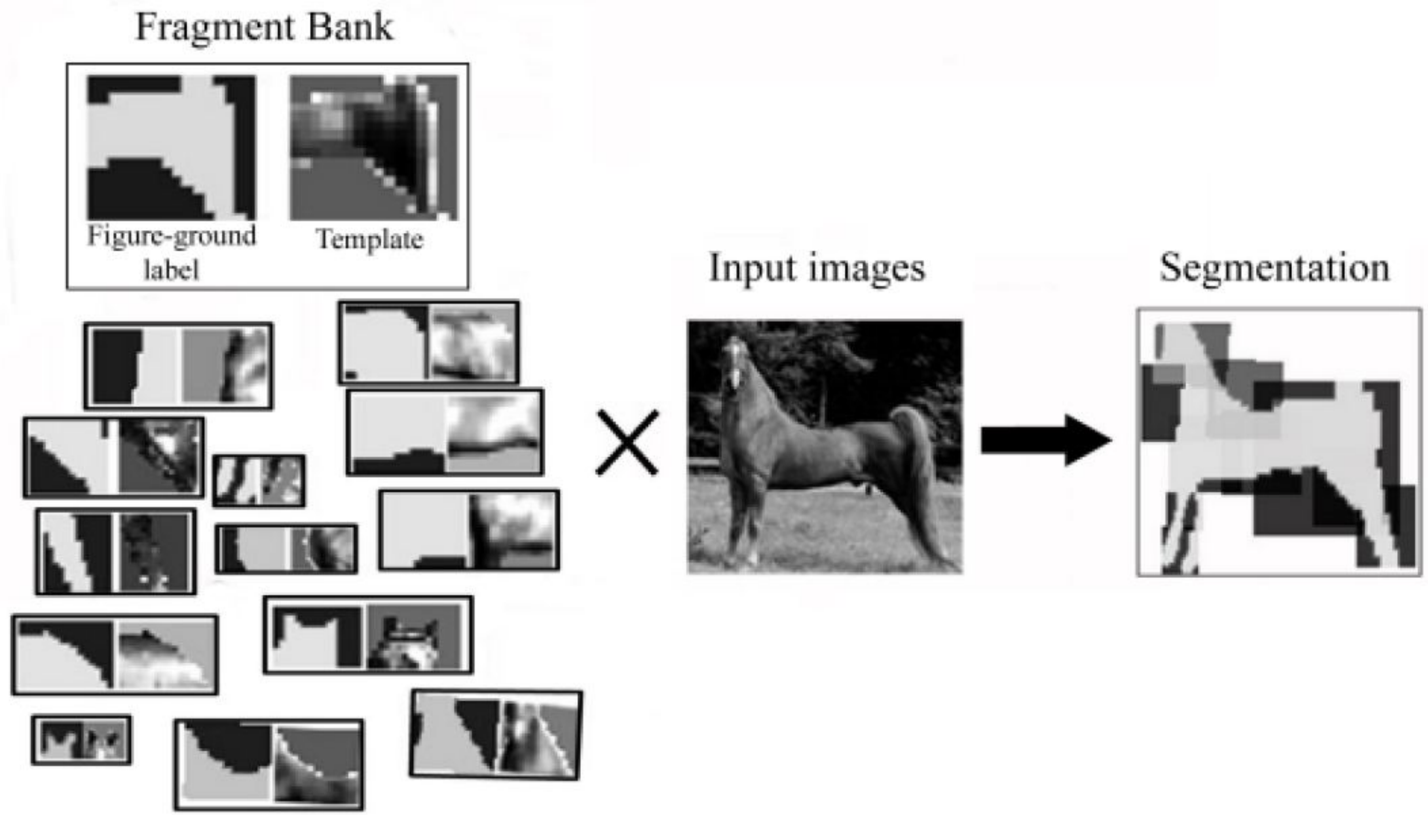
Examples of Bottom-up Segmentation

Using Normalized Cuts, Shi & Malik, 1997



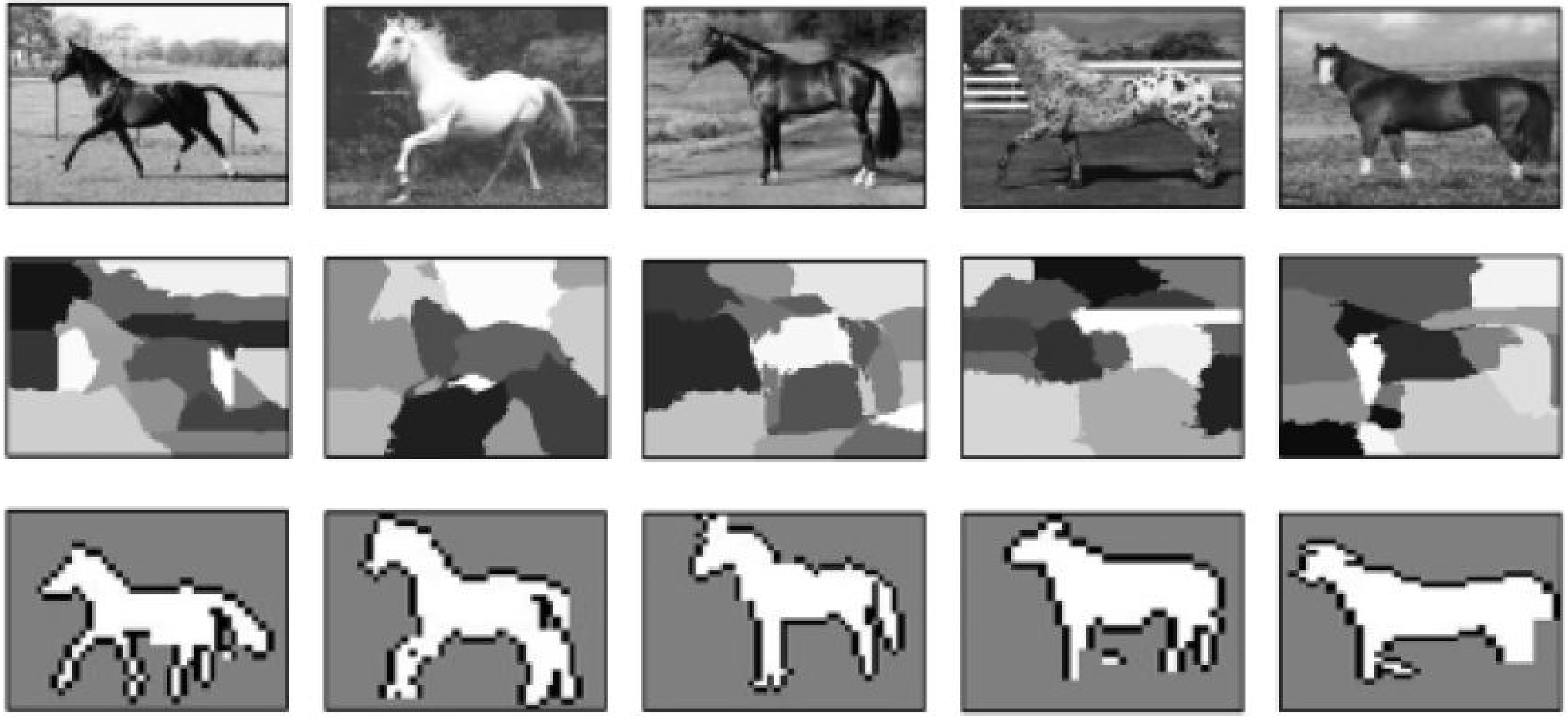
R. Fergus

Jigsaw approach: Borenstein and Ullman, 2002



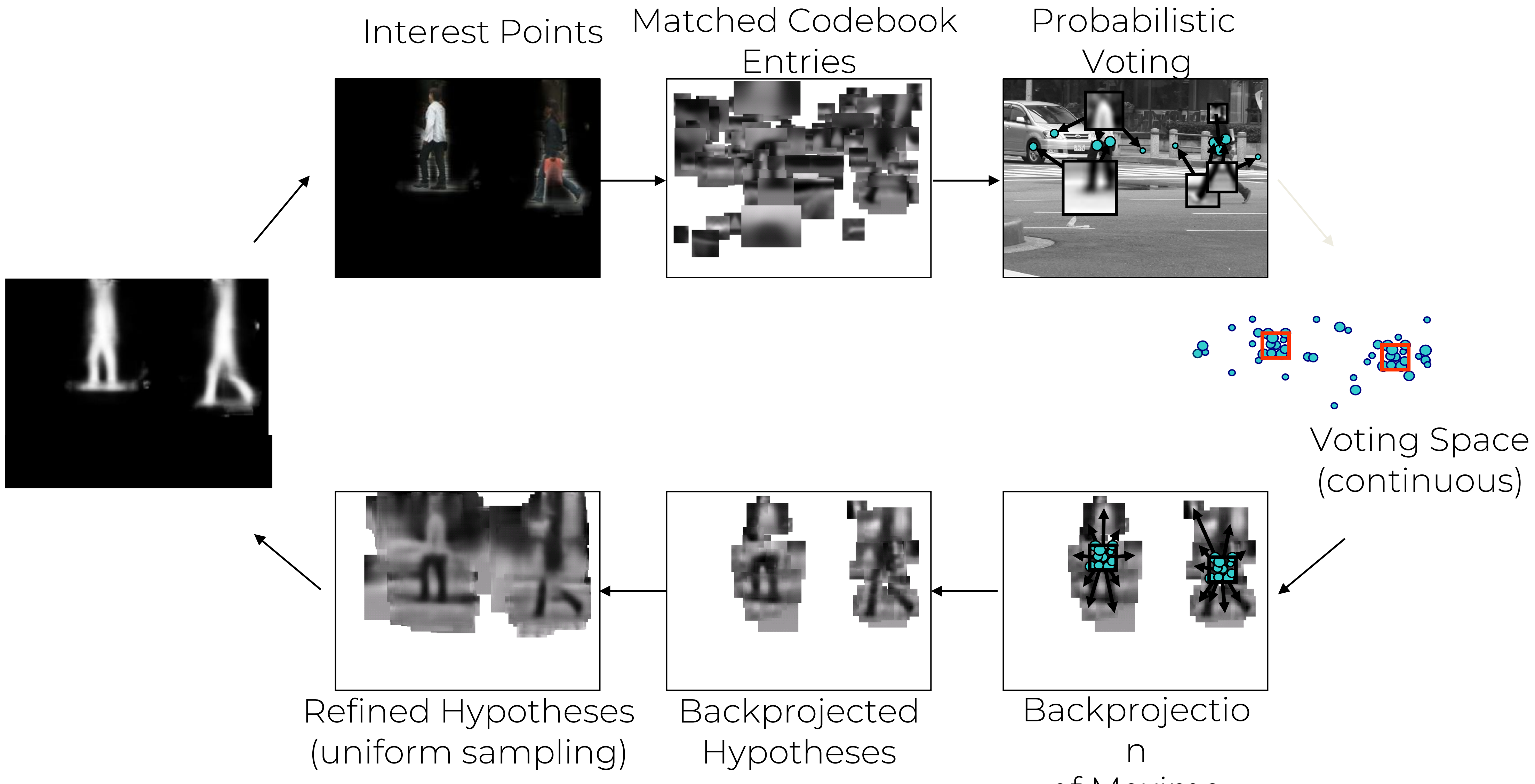
R. Fergus

Jigsaw approach: Borenstein and Ullman, 2002



R. Fergus

Implicit Shape Model - Liebe and Schiele, 2003



R. Fergus

Random Fields for segmentation

I = Image pixels (observed)
h = foreground/background labels (hidden) – one label per pixel
θ = Parameters

$$\underbrace{p(h \mid I, \theta)}_{\text{Posterior}}$$

Random Fields for segmentation

I = Image pixels (observed)
h = foreground/background labels (hidden) – one label per pixel
θ = Parameters

$$\underbrace{p(h \mid I, \theta)}_{\text{Posterior}} \propto \underbrace{p(I, h \mid \theta)}_{\text{Joint}} = \underbrace{p(I \mid h, \theta)}_{\text{Likelihood}} \underbrace{p(h \mid \theta)}_{\text{Prior}}$$

1. Generative approach models joint
   → Markov random field (MRF)

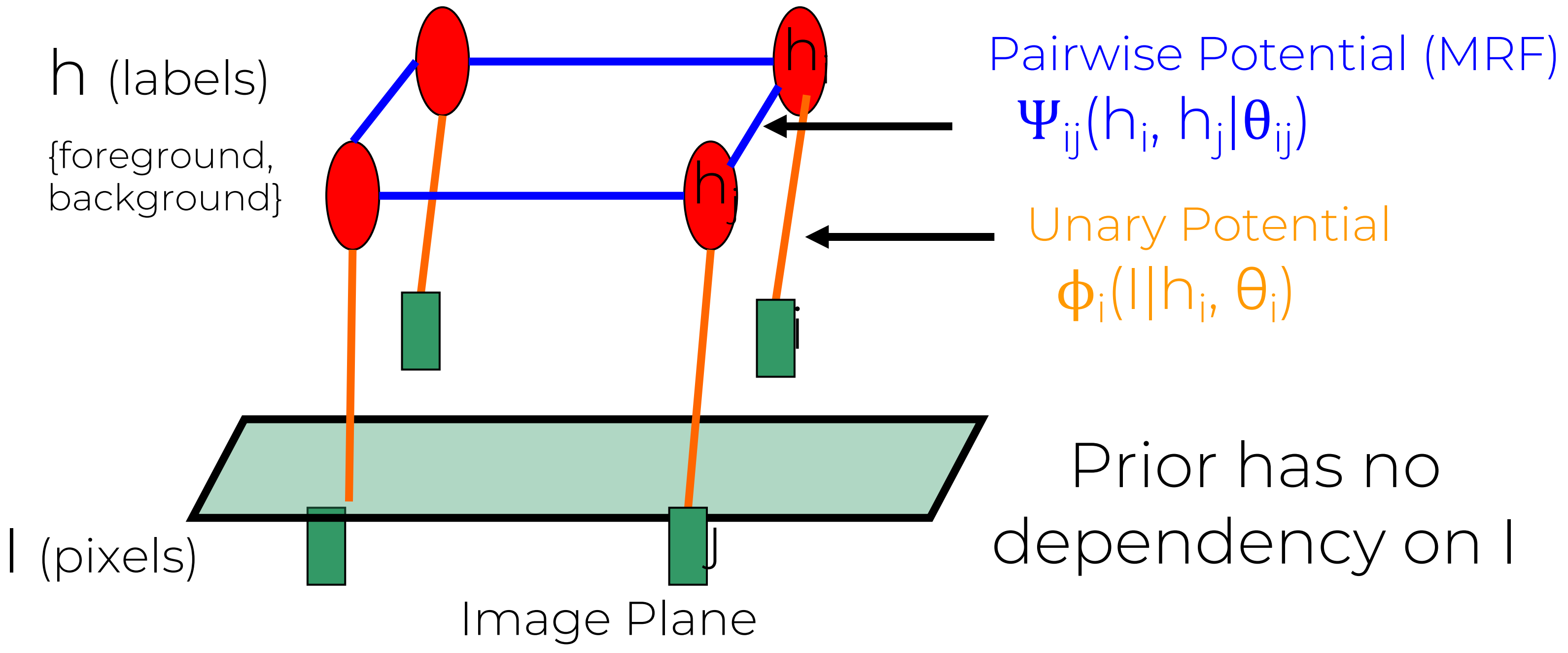
2. Discriminative approach models posterior directly
   → Conditional random field (CRF)

R. Fergus

Generative Markov Random Field

$$p(h, I \mid \theta) = \boxed{p(I \mid h, \theta)}\,\boxed{p(h \mid \theta)}$$

$$= \frac{1}{Z(\theta)}\left[\underbrace{\prod_i \phi_i(I \mid h_i, \theta_i)}_{\text{Likelihood}}\ \underbrace{\prod_{ij} \psi_{ij}(h_i, h_j \mid \theta_{ij})}_{\text{MRF Prior}}\right]$$



Prior has no dependency on I

R. Fergus

Conditional Random Field

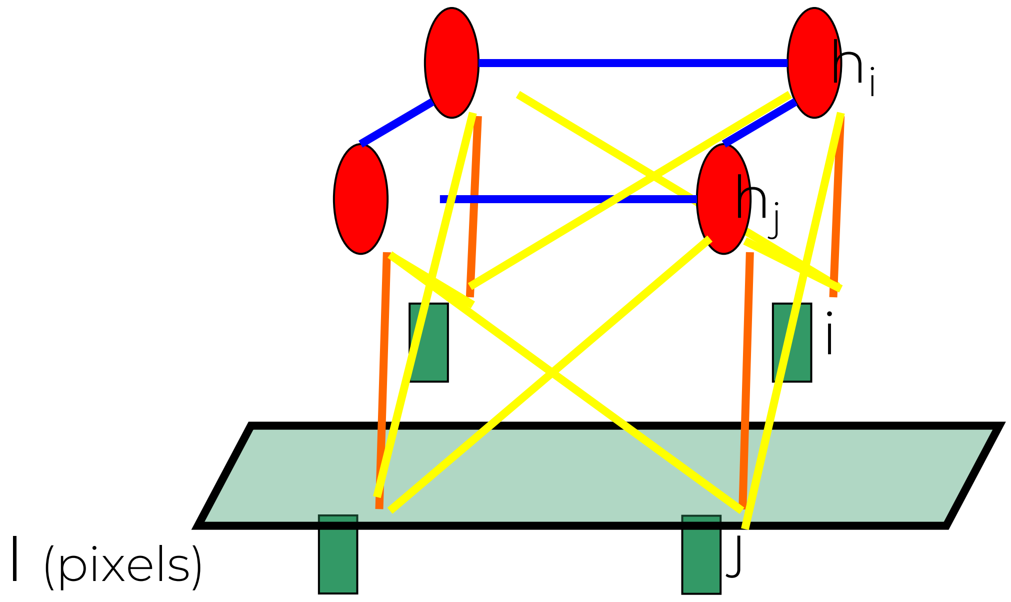
Discriminative approach

Lafferty, McCallum and Pereira 2001

$$p(h \mid I,\theta) = \frac{1}{Z(I,\theta)}\left[\prod_i \underbrace{\phi_i(h_i, I \mid \theta_i)}_{\text{Unary}} \prod_{ij} \underbrace{\psi_{ij}(h_i, h_j, I \mid \theta_{ij})}_{\text{Pairwise}}\right]$$

- Dependency on I allows introduction of pairwise terms that make use of image.
- For example, neighboring labels should be similar only if pixel colors are similar → Contrast term

e.g Kumar and Hebert 2003



R. Fergus

Conditional Random Fields for Segmentation

- Segmentation map x
- Image I

$$E(x;I) = \nu \underbrace{\sum_{i,j} w_{ij}|x(i) - x(j)|}_{\text{Low-level pairwise term}} + \underbrace{\sum_k \lambda_k |x - x_{F_k,I}|}_{\text{High-level local term}}$$

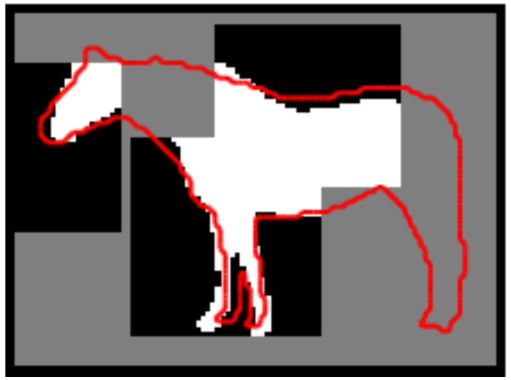
Pixel-wise similarity

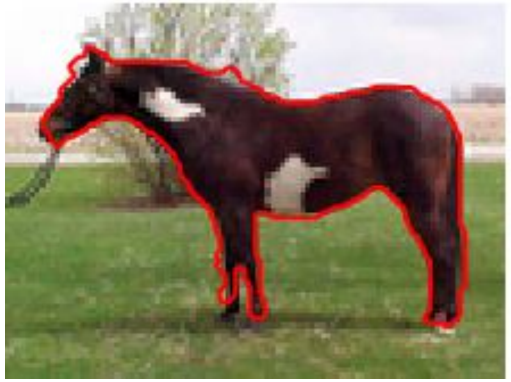
Levin & Weiss [ECCV 2006]

$$E(h;I) = \sum_i \lambda_i \left| h - h_{F_i,I} \right| + \sum_{ij} w(i,j) \left| h_i - h_j \right|$$

Consistency with fragments segmentation

Segmentation alignment with image edges



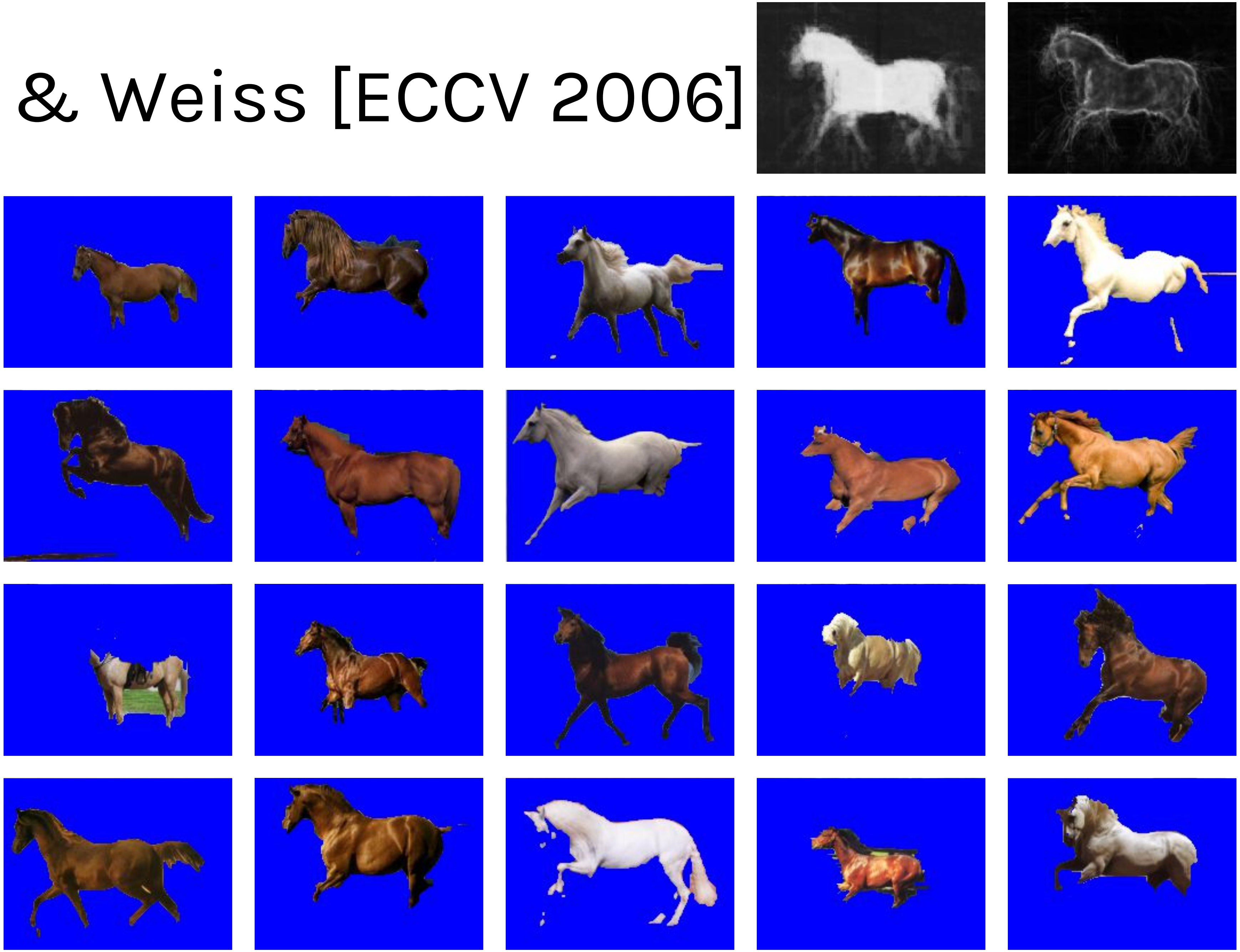


Resulting min-cut segmentation

R. Fergus

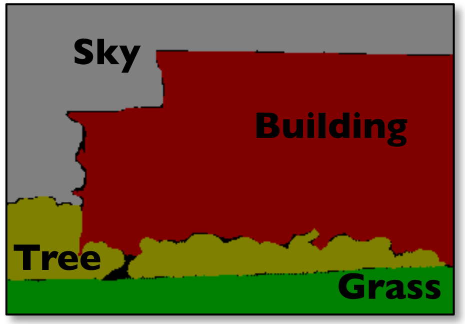
Levin & Weiss [ECCV 2006]

Levin & Weiss [ECCV 2006]



Semantic Segmentation

Joint Object recognition & segmentation

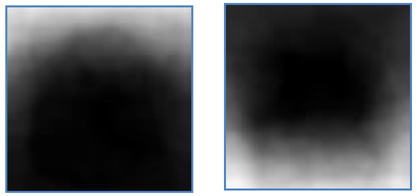


$$E(x,\omega) = \sum_i \theta_i(\omega, x_i) + \sum_i \theta_i(x_i) + \sum_i \theta_i(x_i) + \sum_{i,j} \theta_{ij}(x_i,x_j)$$

(color) (location) (class) (edge aware Ising prior)

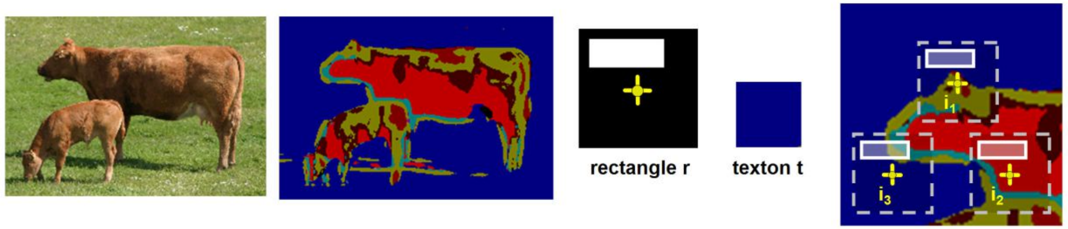
$x_i \in \{1,\dots,K\}$ for K object classes

Location



sky grass

Class (boosted textons)



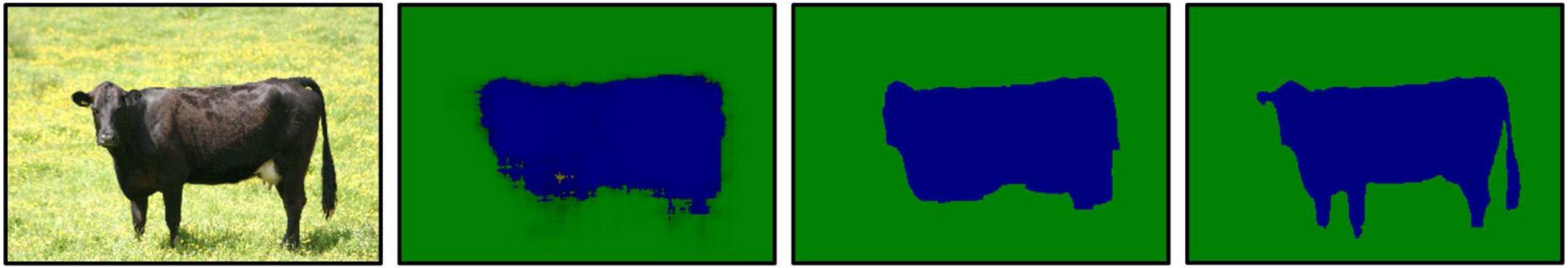
(a) Input image (b) Texton map (c) Feature pair = (r,t) (d) Superimposed rectangles

[TextonBoost; Shotton et al, '06]

C. Rother

Semantic Segmentation

Joint Object recognition & segmentation



(a) (b) 69.6% (c) 70.3% (d) 72.2%

Class + location | + edges | + color

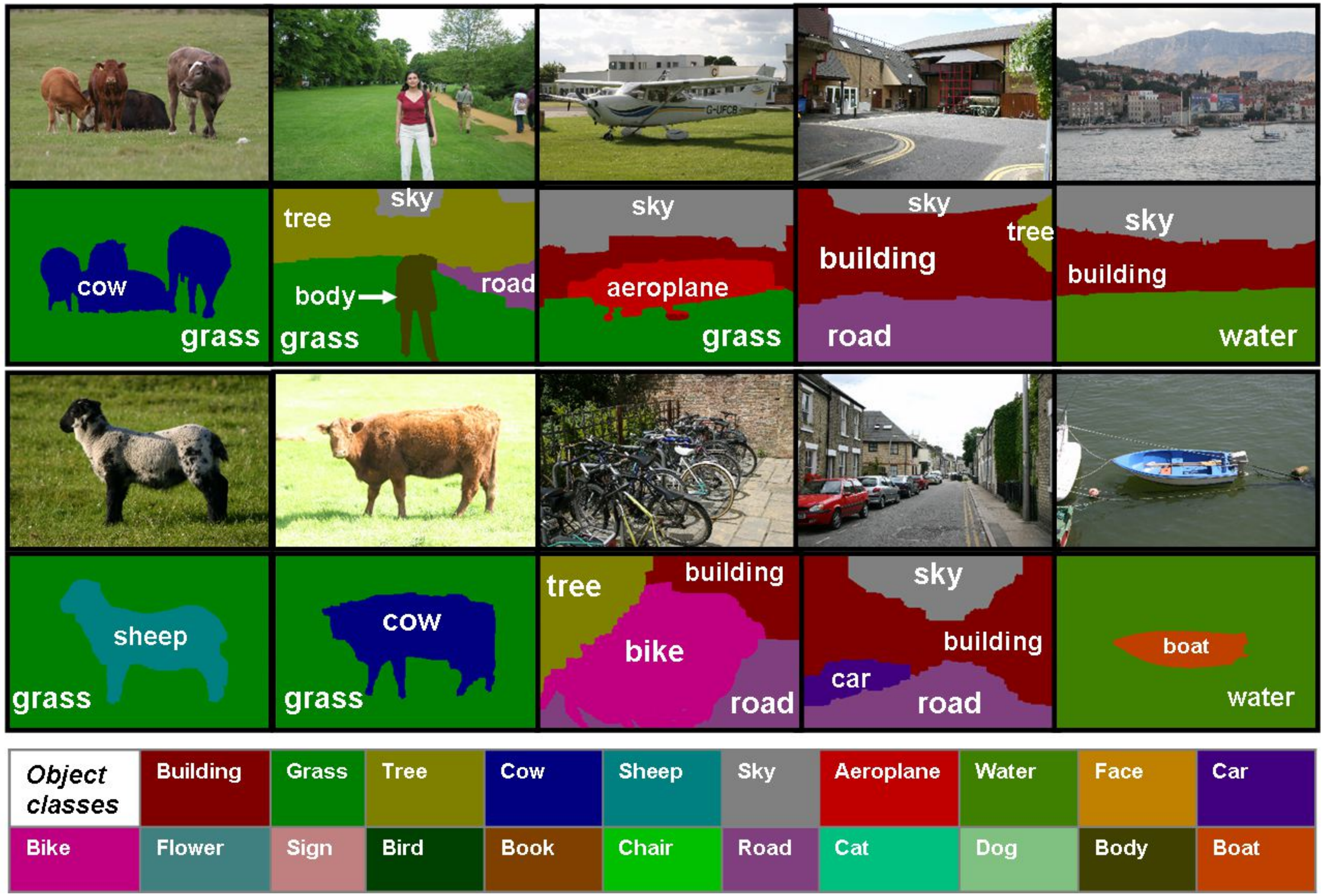
[TextonBoost; Shotton et al, '06]

C. Rother

Semantic Segmentation

Joint Object recognition & segmentation

Good results …



[TextonBoost; Shotton et al, '06]

C. Rother

Semantic Segmentation

Joint Object recognition & segmentation

Failure cases...

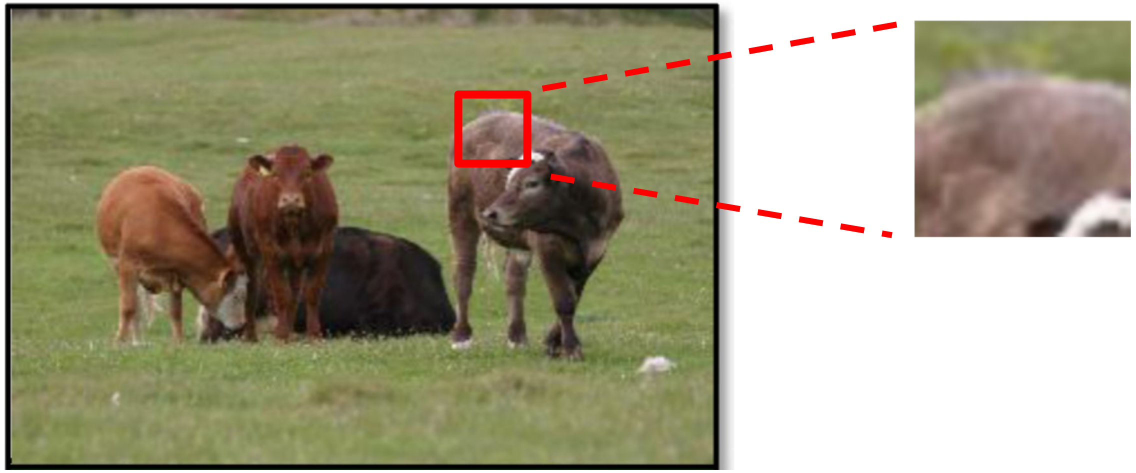
[TextonBoost; Shotton et al, '06]

C. Rother

Deep Semantic Segmentation

Deep Semantic Segmentation

Extract patch

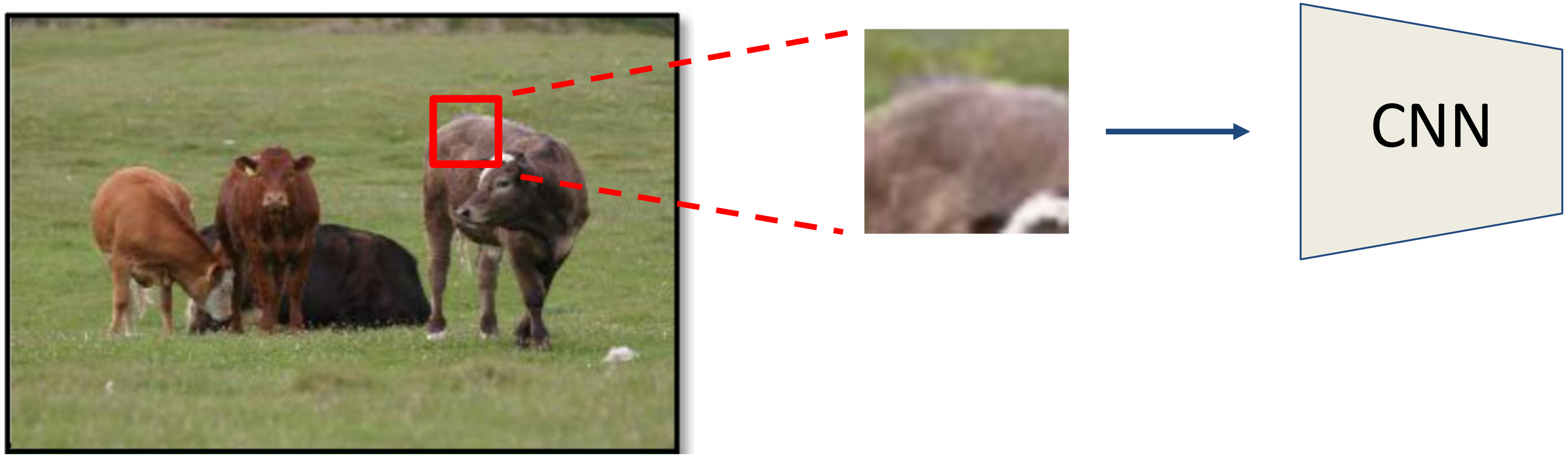


F.-F. Li, A. Karpathy and J. Johnson

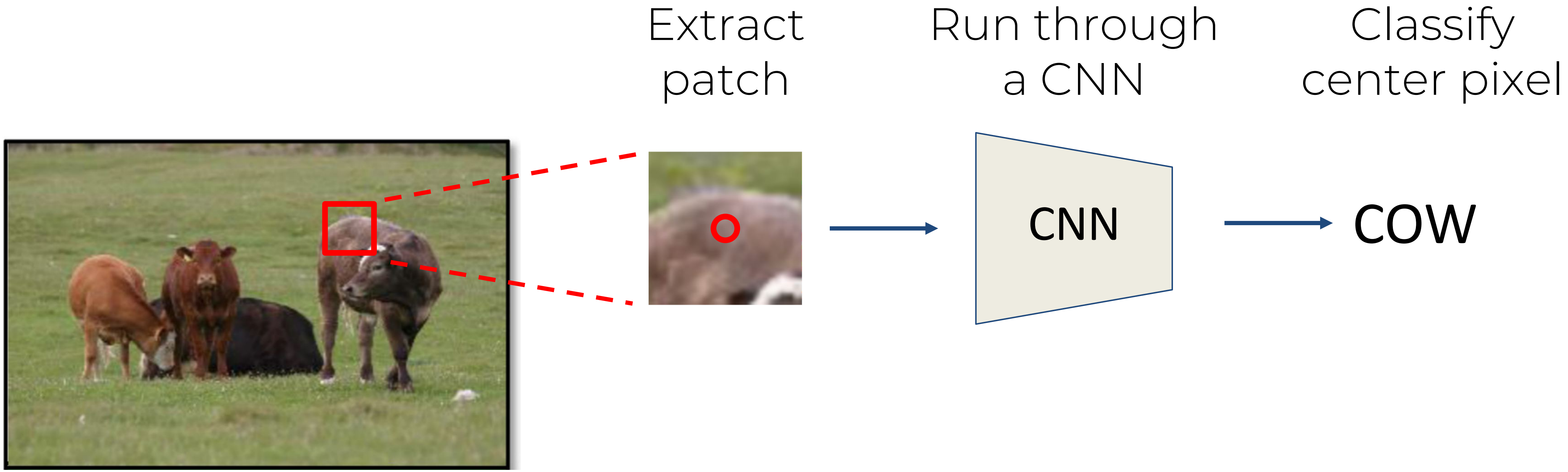
Deep Semantic Segmentation

Extract patch

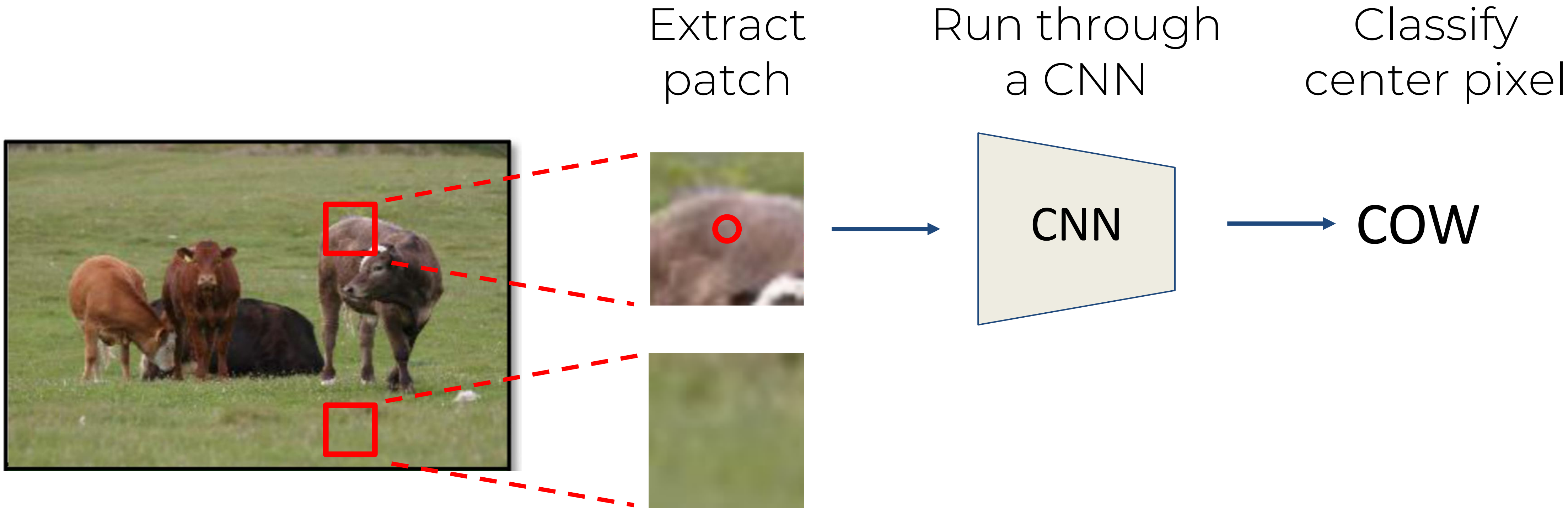
Run through a CNN



Deep Semantic Segmentation

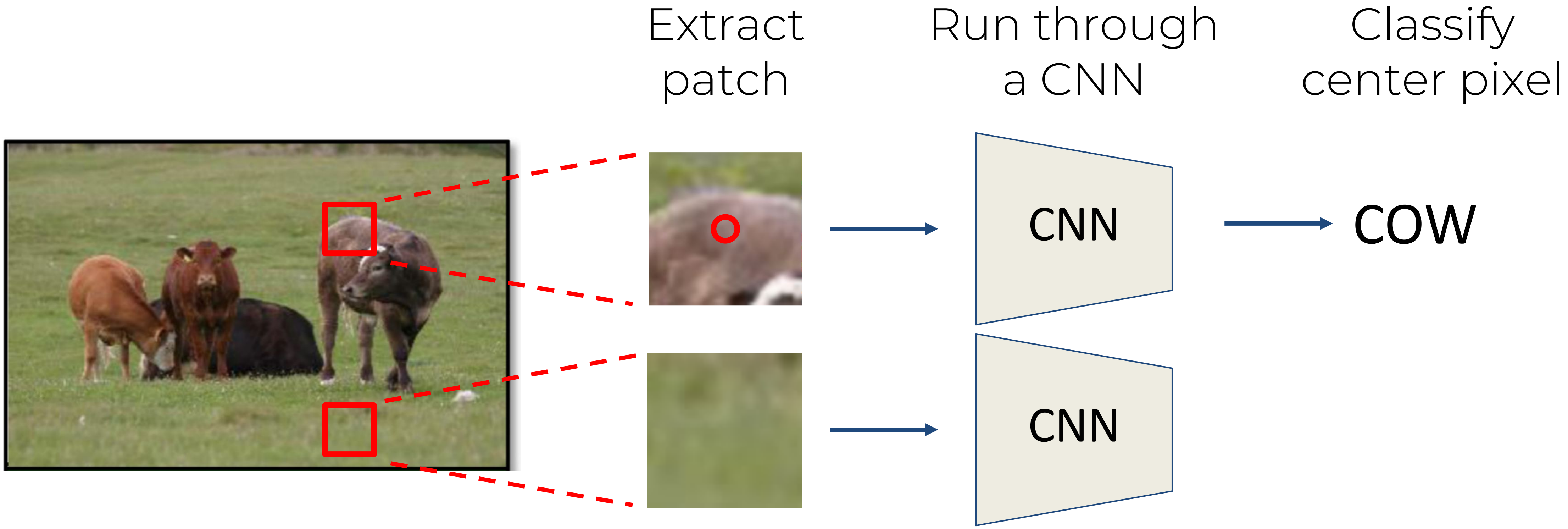


Deep Semantic Segmentation



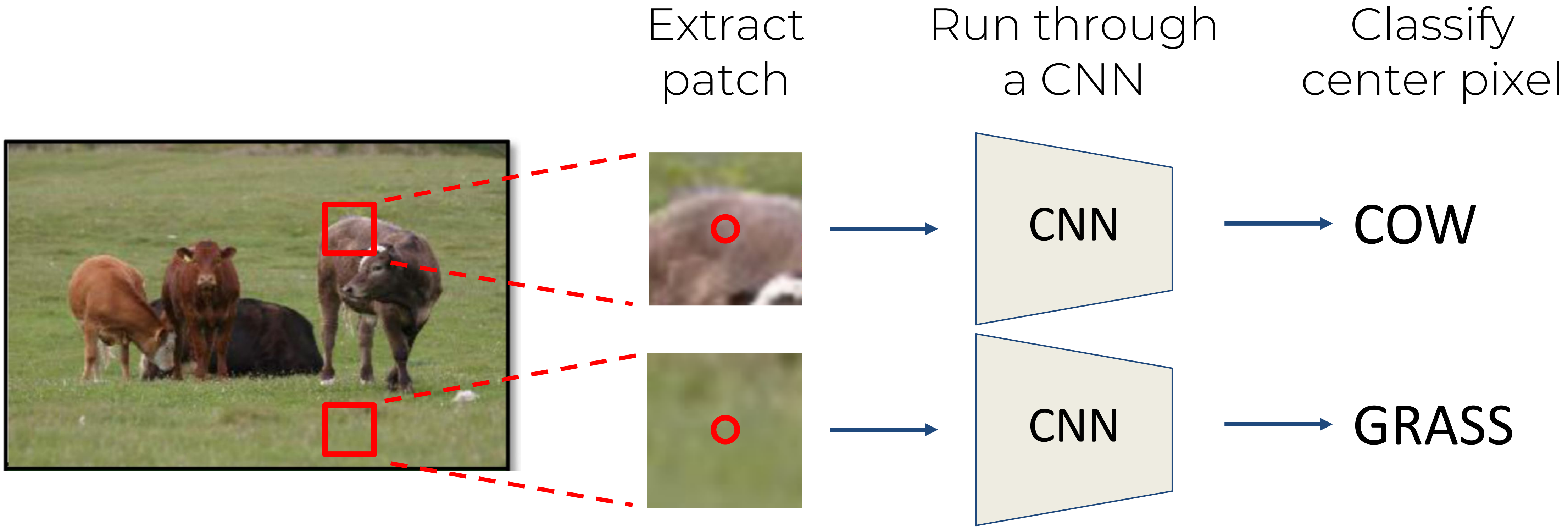
F.-F. Li, A. Karpathy and J. Johnson

Deep Semantic Segmentation



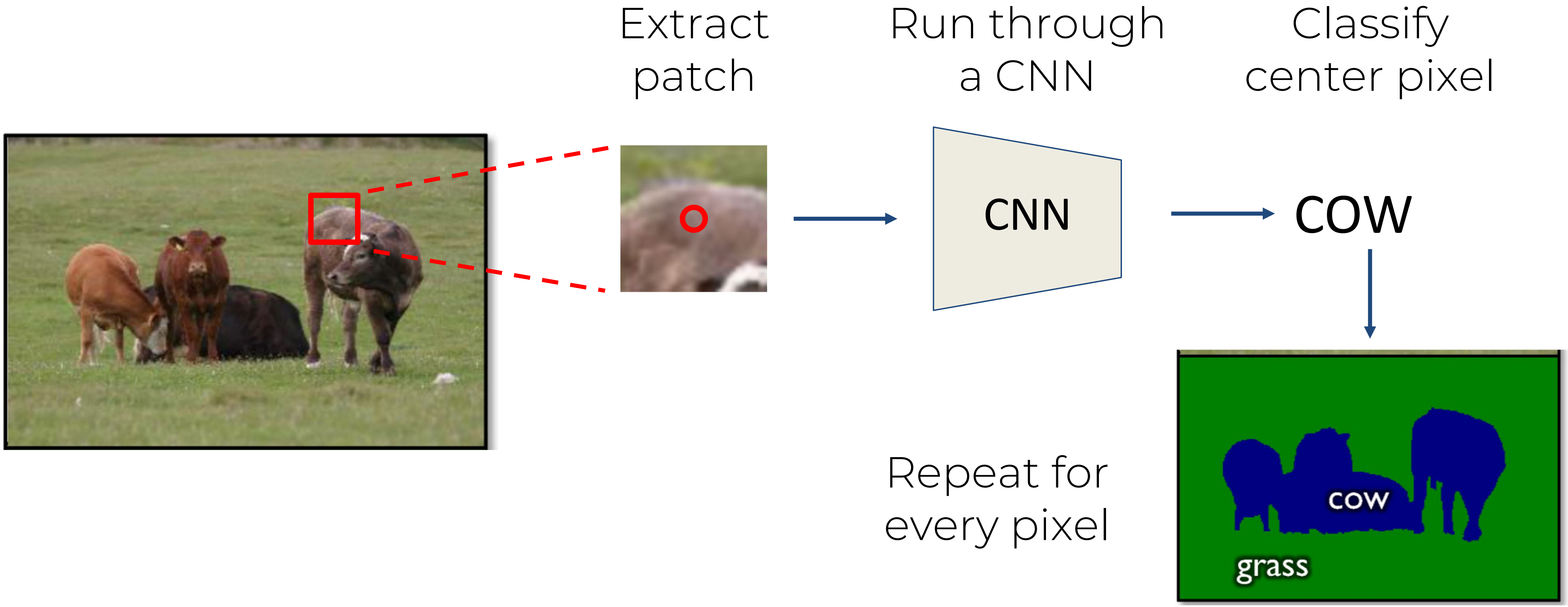
F.-F. Li, A. Karpathy and J. Johnson

Deep Semantic Segmentation



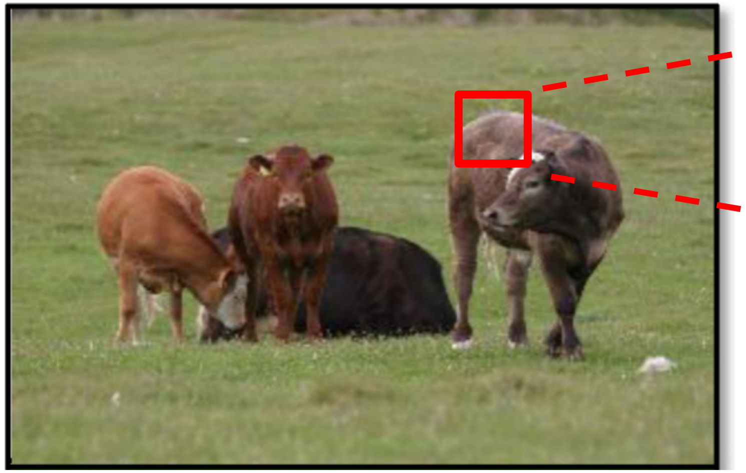
F.-F. Li, A. Karpathy and J. Johnson

Deep Semantic Segmentation



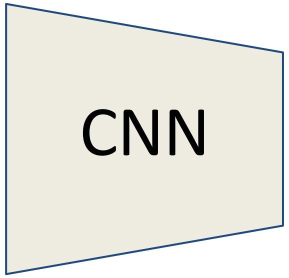
F.-F. Li, A. Karpathy and J. Johnson

Deep Semantic Segmentation



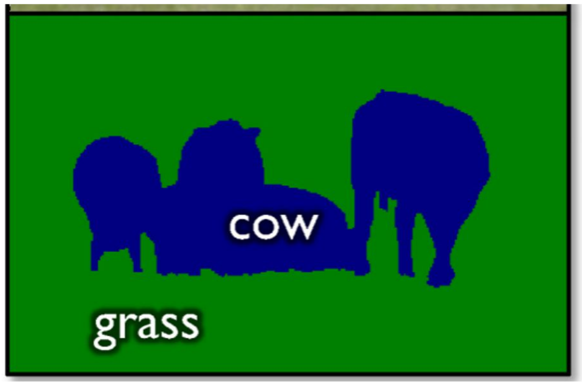
Extract patch

Run through a CNN



Classify center pixel

COW

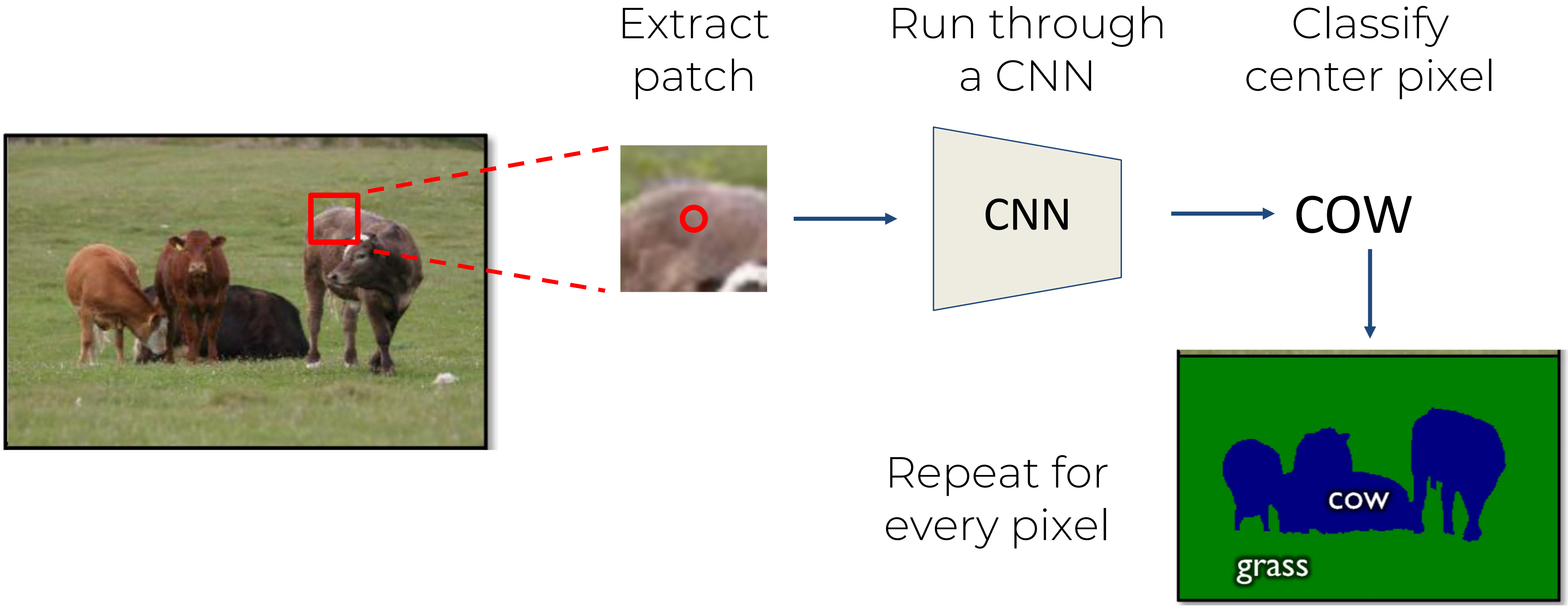


Repeat for every pixel

Problem: Very inefficient!
Not reusing shared features
between overlappoing patches.

F.-F. Li, A. Karpathy and J. Johnson

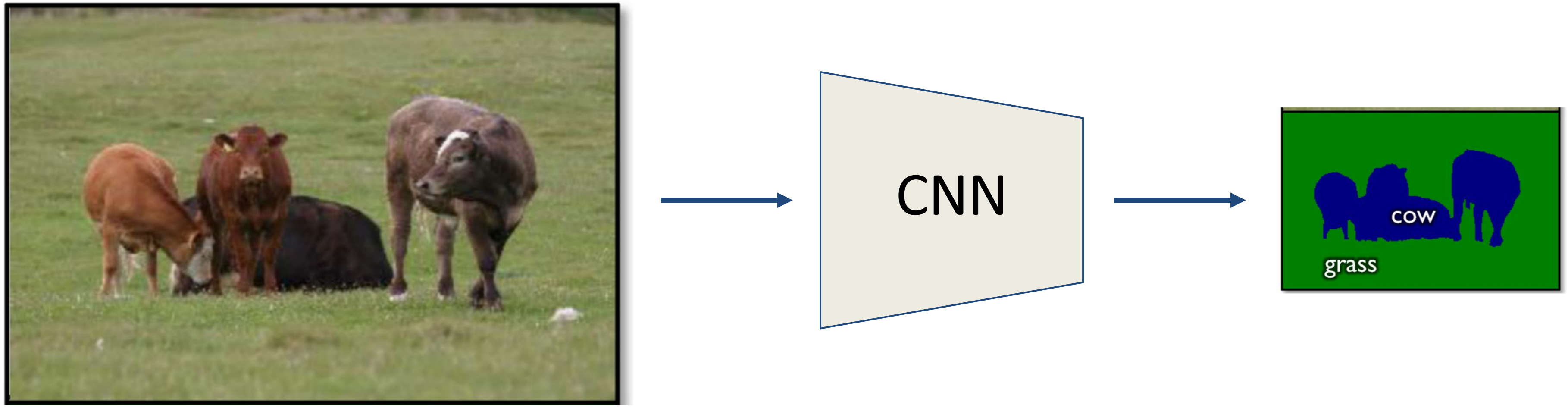
Deep Semantic Segmentation



F.-F. Li, A. Karpathy and J. Johnson

Semantic Segmentation Idea: Fully Convolutional

Run "fully convolutional" network to get all pixels at once

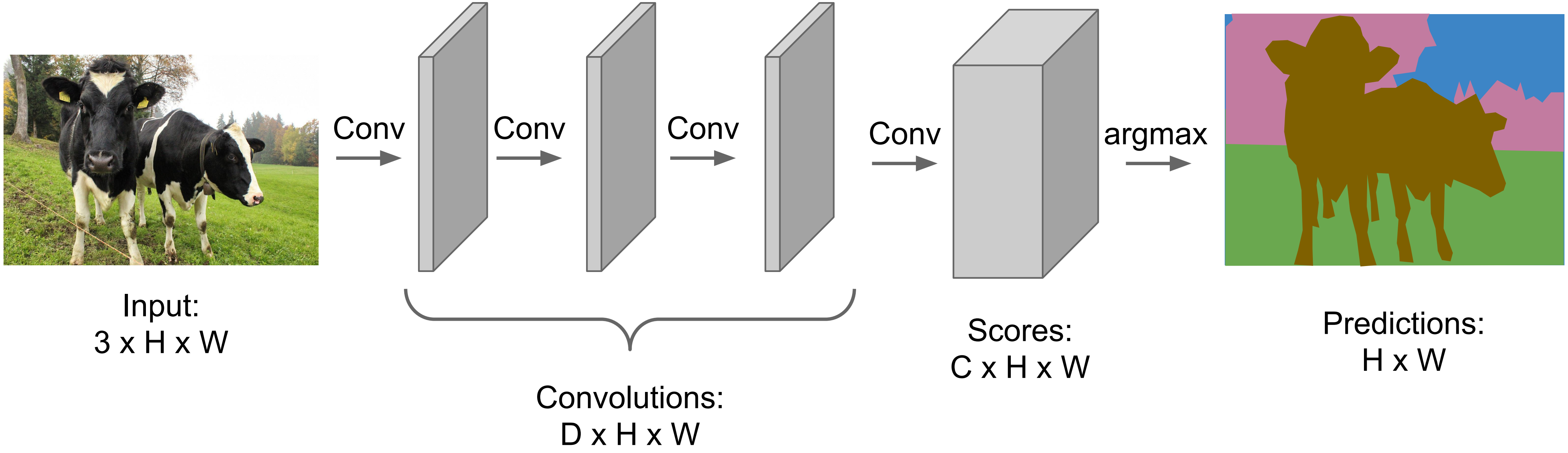


Smaller output due to pooling

F.-F. Li, A. Karpathy and J. Johnson

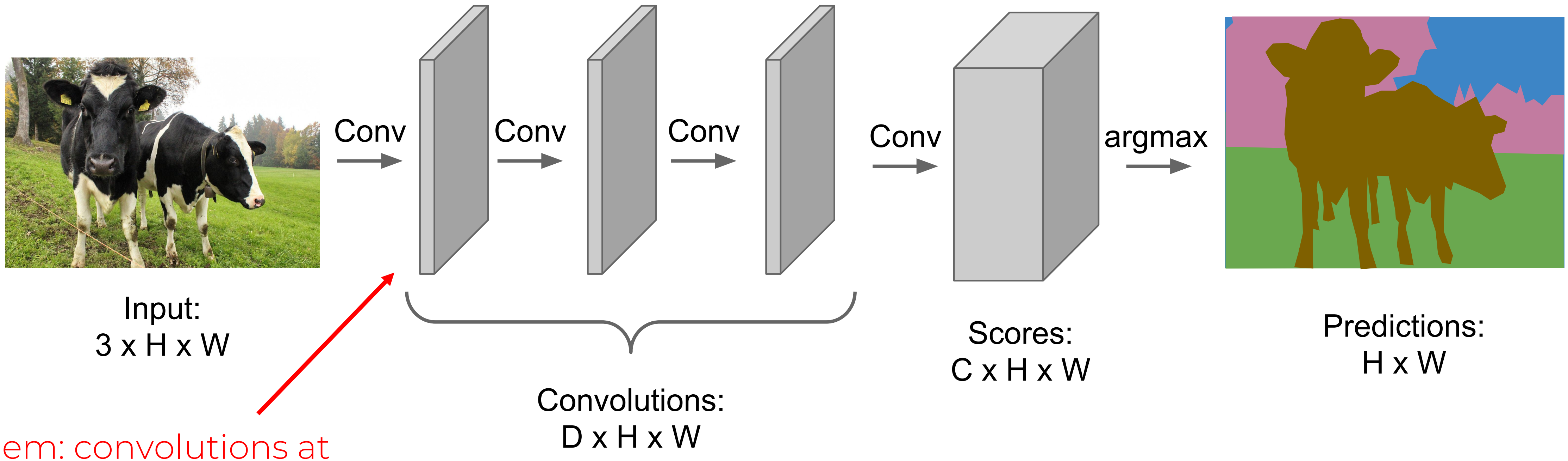
Semantic Segmentation Idea: Fully Convolutional

- Design a network as a bunch of convolutional layers to make predictions for pixels all at once!



Semantic Segmentation Idea: Fully Convolutional

- Design a network as a bunch of convolutional layers to make predictions for pixels all at once!

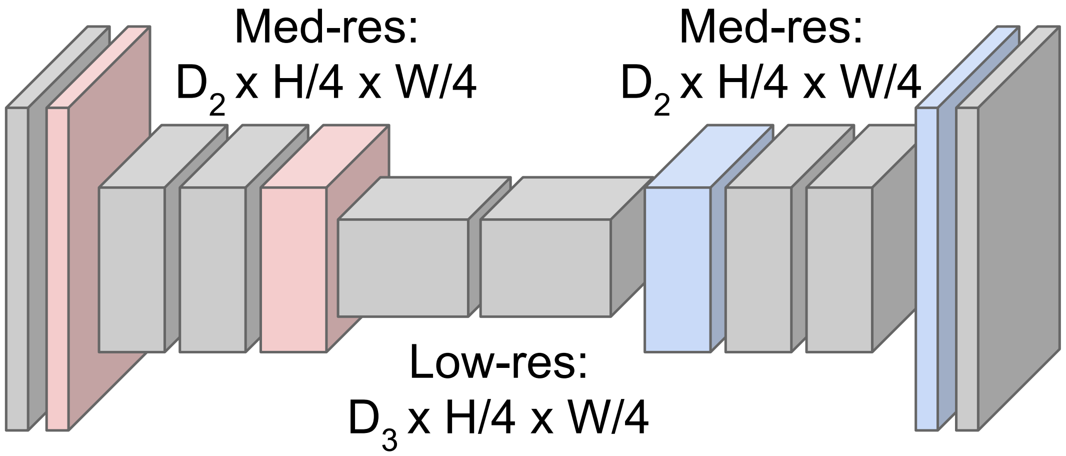


Problem: convolutions at original image resolution will be very expensive …

Semantic Segmentation Idea: Fully Convolutional

- Design network as a bunch of convolutional layers, with downsampling and upsampling inside the network!

Input:
3 x H x W



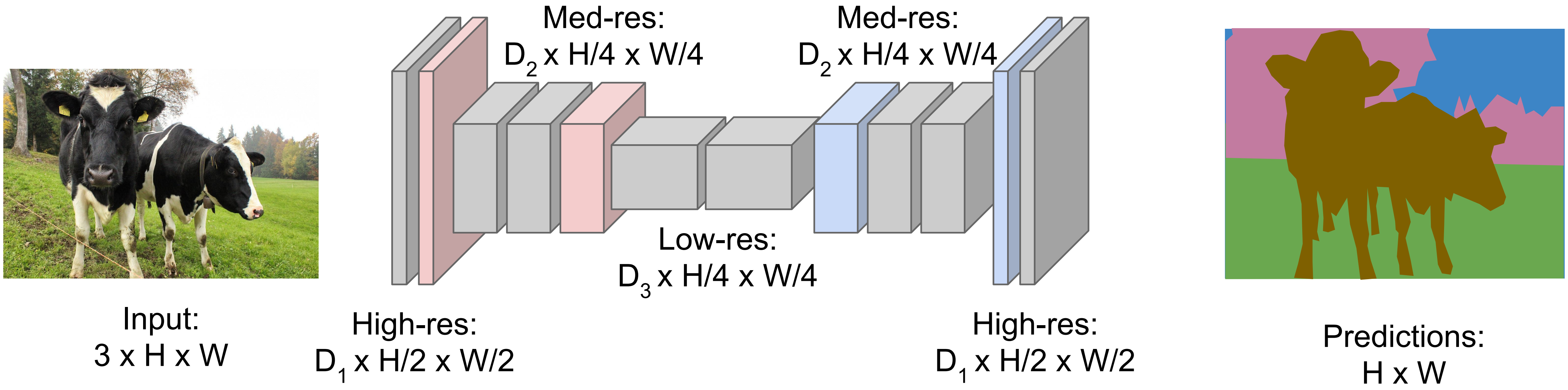
High-res:
D_1 x H/2 x W/2

High-res:
D_1 x H/2 x W/2

Predictions:
H x W

Semantic Segmentation Idea: Fully Convolutional

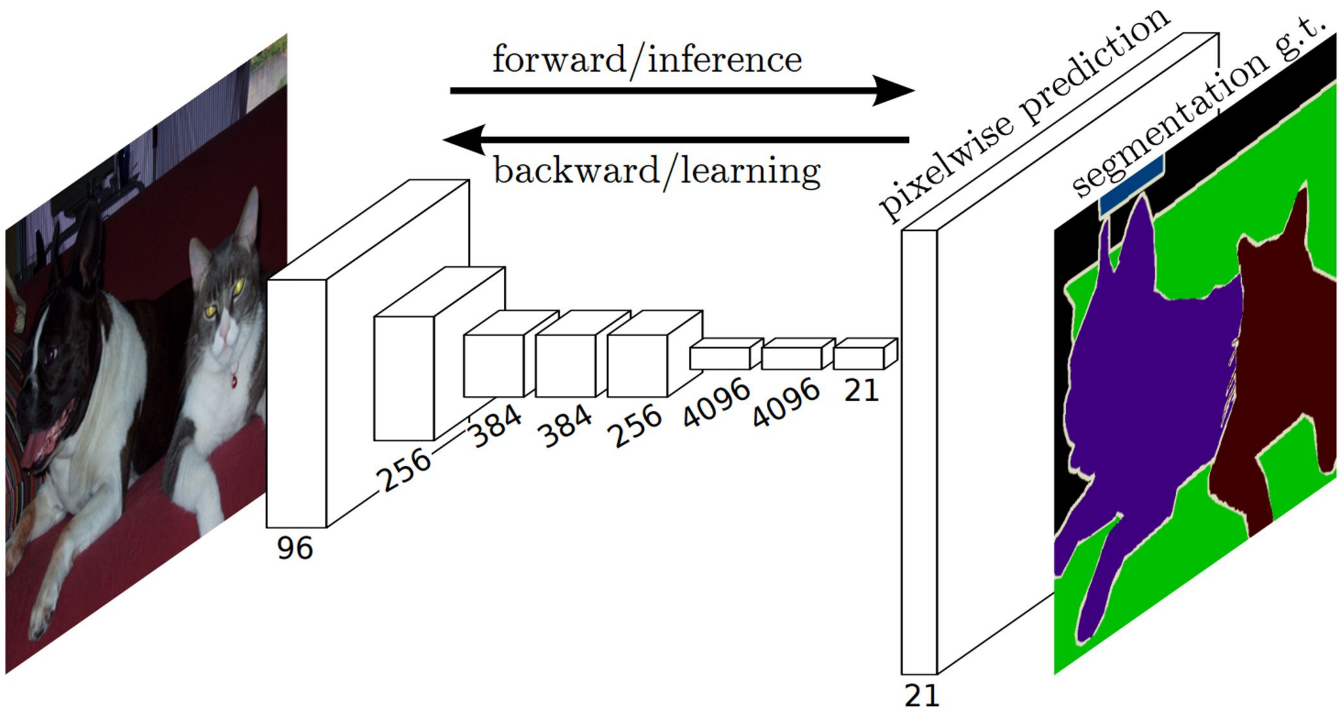
- Design network as a bunch of convolutional layers, with downsampling and upsampling inside the network!



Downsampling: Pooling, strided convolution

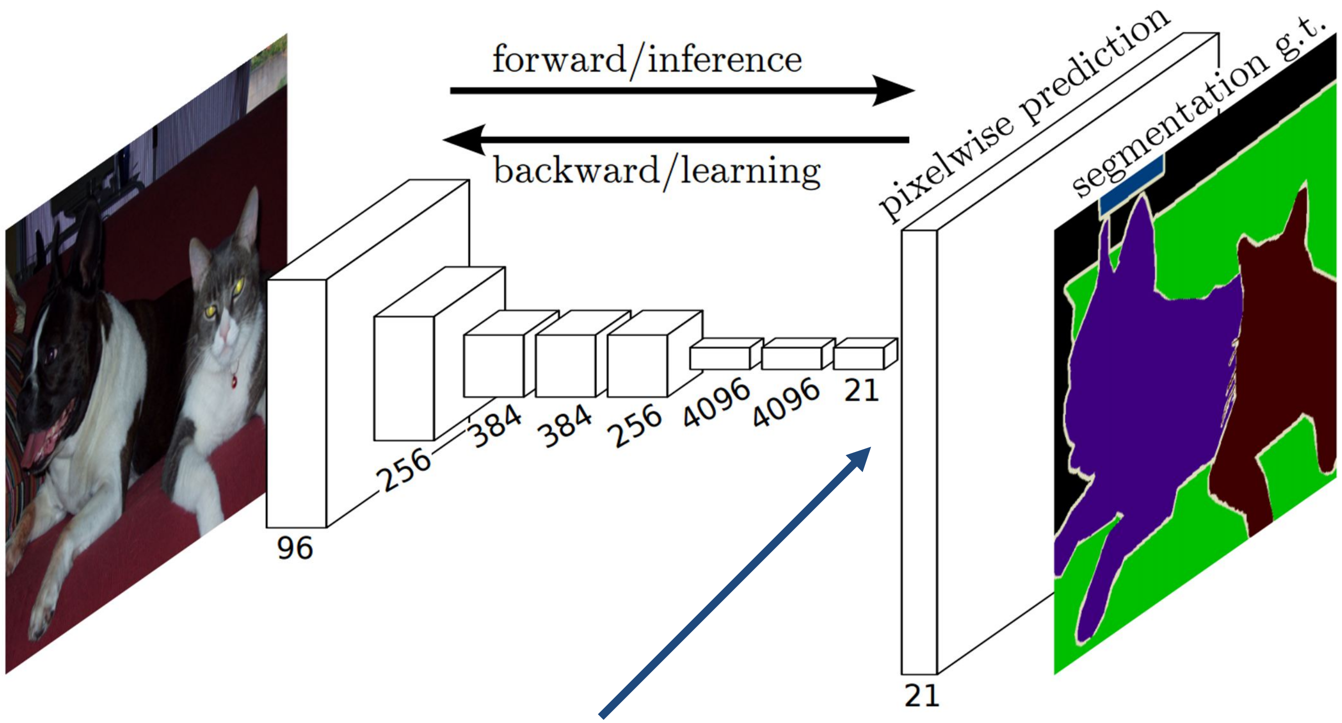
Upsampling: ???

Semantic Segmentation: Upsampling



Long, Shelhamer, and Darrell, “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015

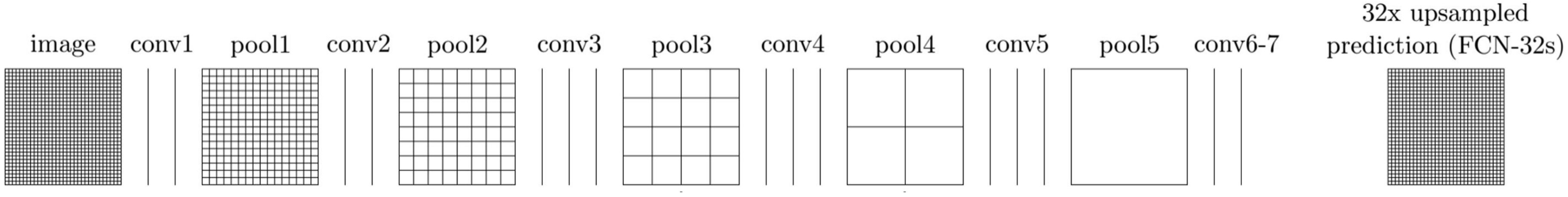
Semantic Segmentation: Upsampling



Learnable
upsampling!

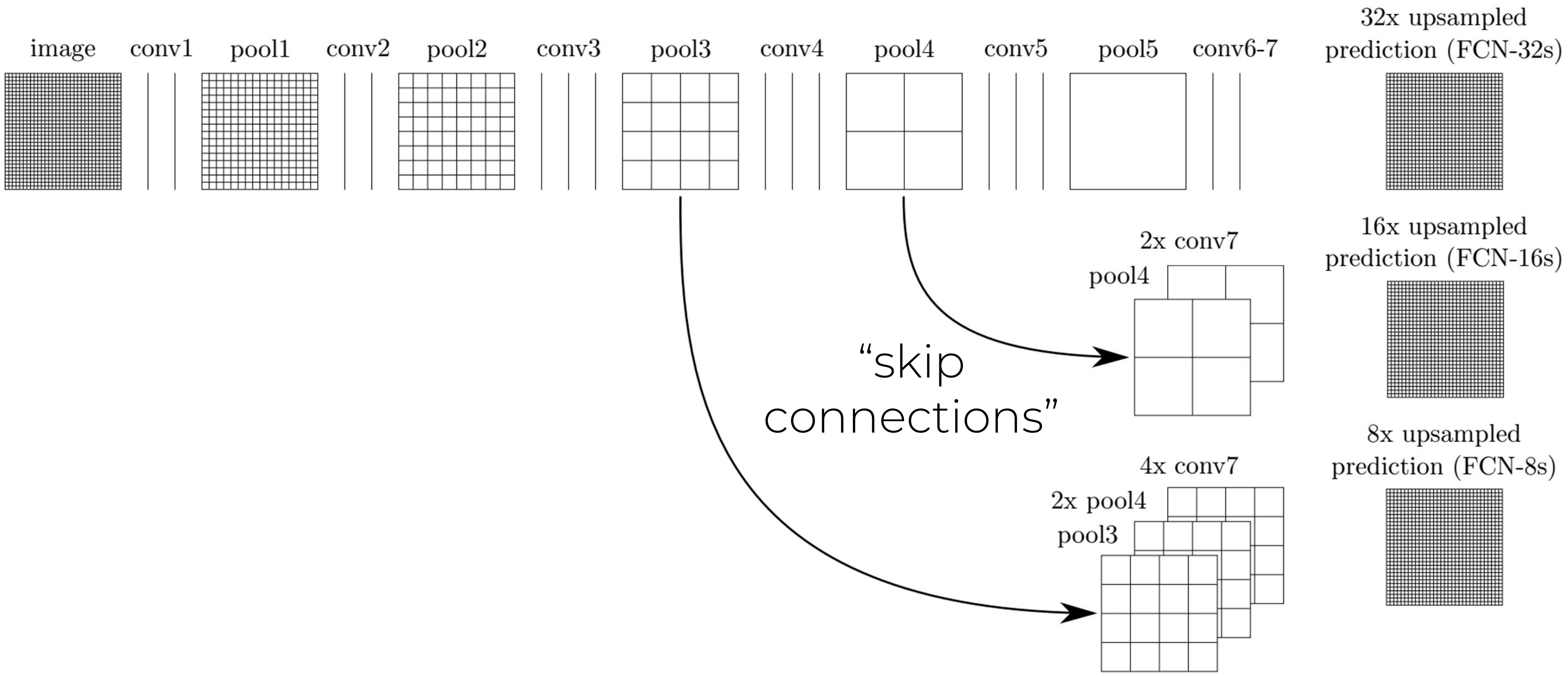
Long, Shelhamer, and Darrell, “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015

Semantic Segmentation: Upsampling



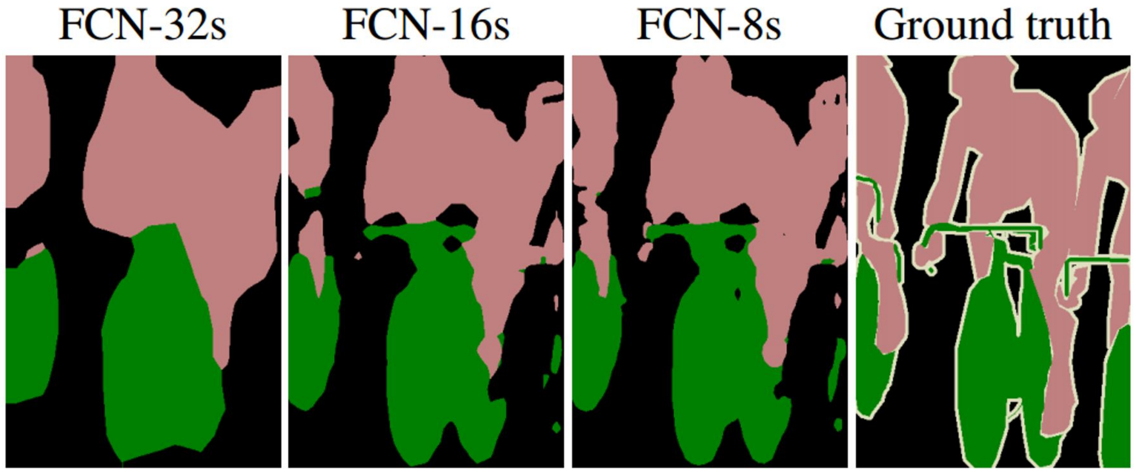
Long, Shelhamer, and Darrell, “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015

Semantic Segmentation: Upsampling

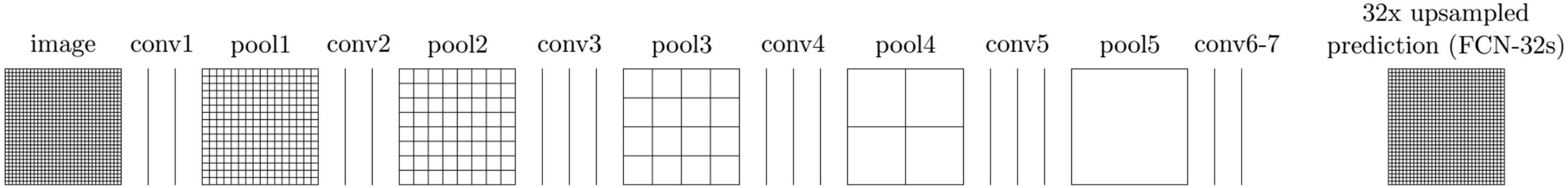


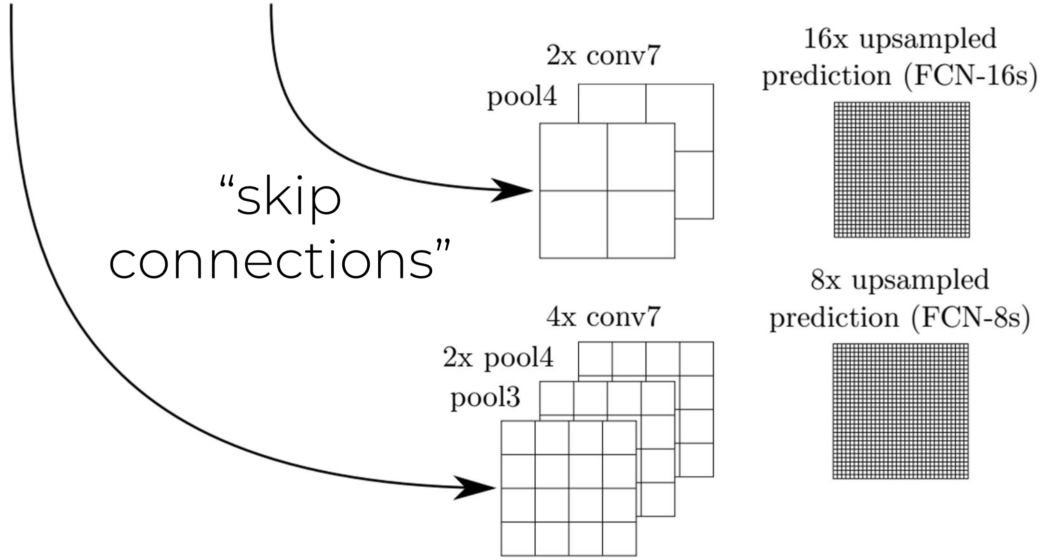
Long, Shelhamer, and Darrell, "Fully Convolutional Networks for Semantic Segmentation", CVPR 2015

Semantic Segmentation: Upsampling



Skip connections = Better results

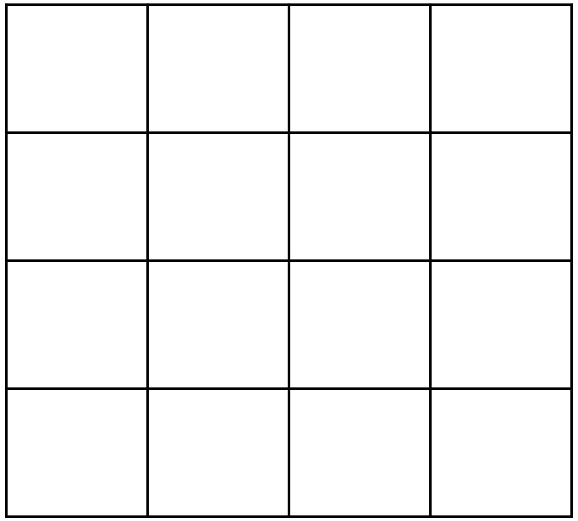




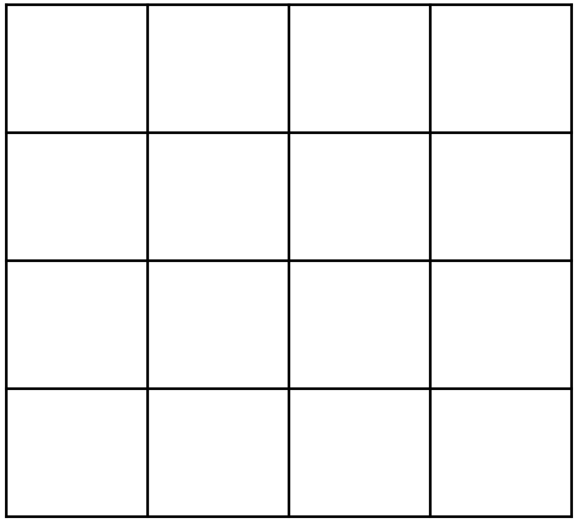
Long, Shelhamer, and Darrell, "Fully Convolutional Networks for Semantic Segmentation", CVPR 2015

Learnable Upsampling: "Deconvolution"

Typical 3 x 3 convolution, stride 1 pad 1



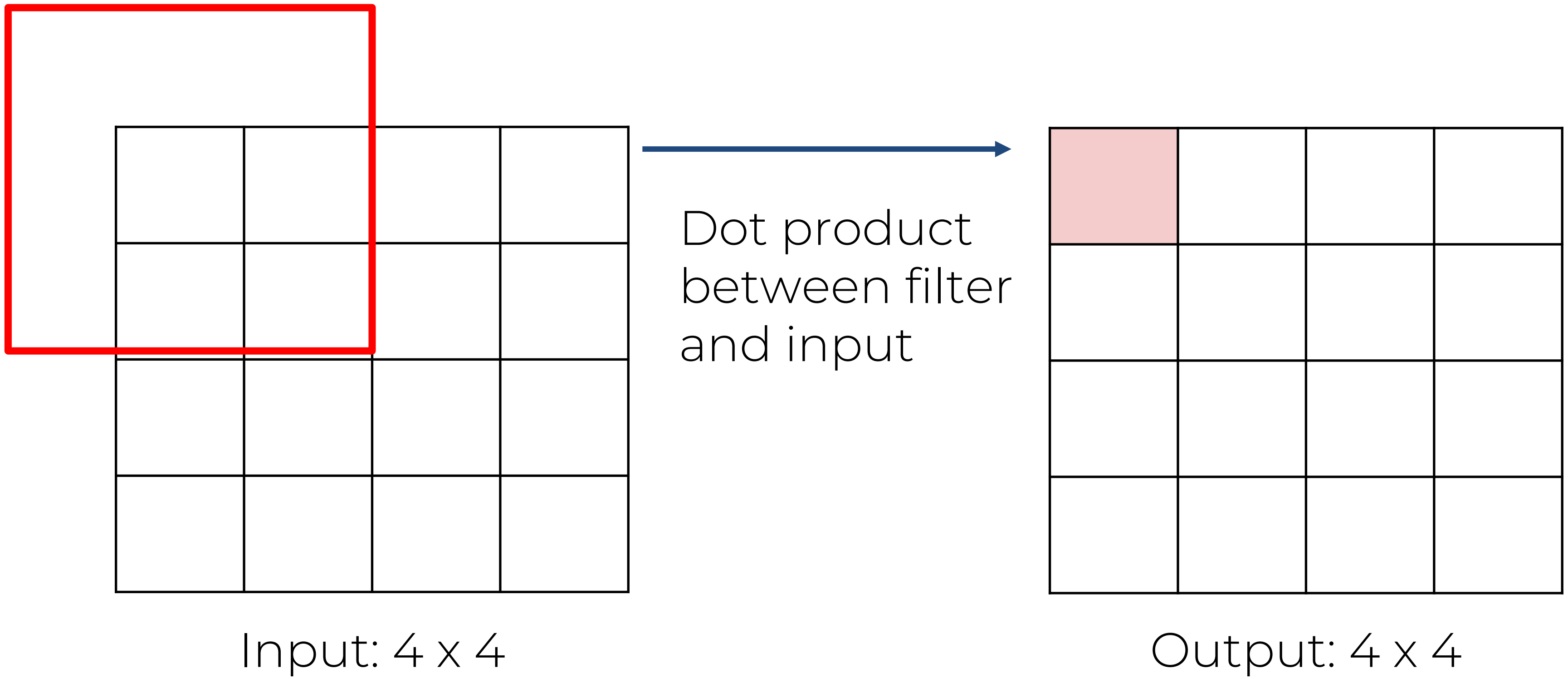
Input: 4 x 4



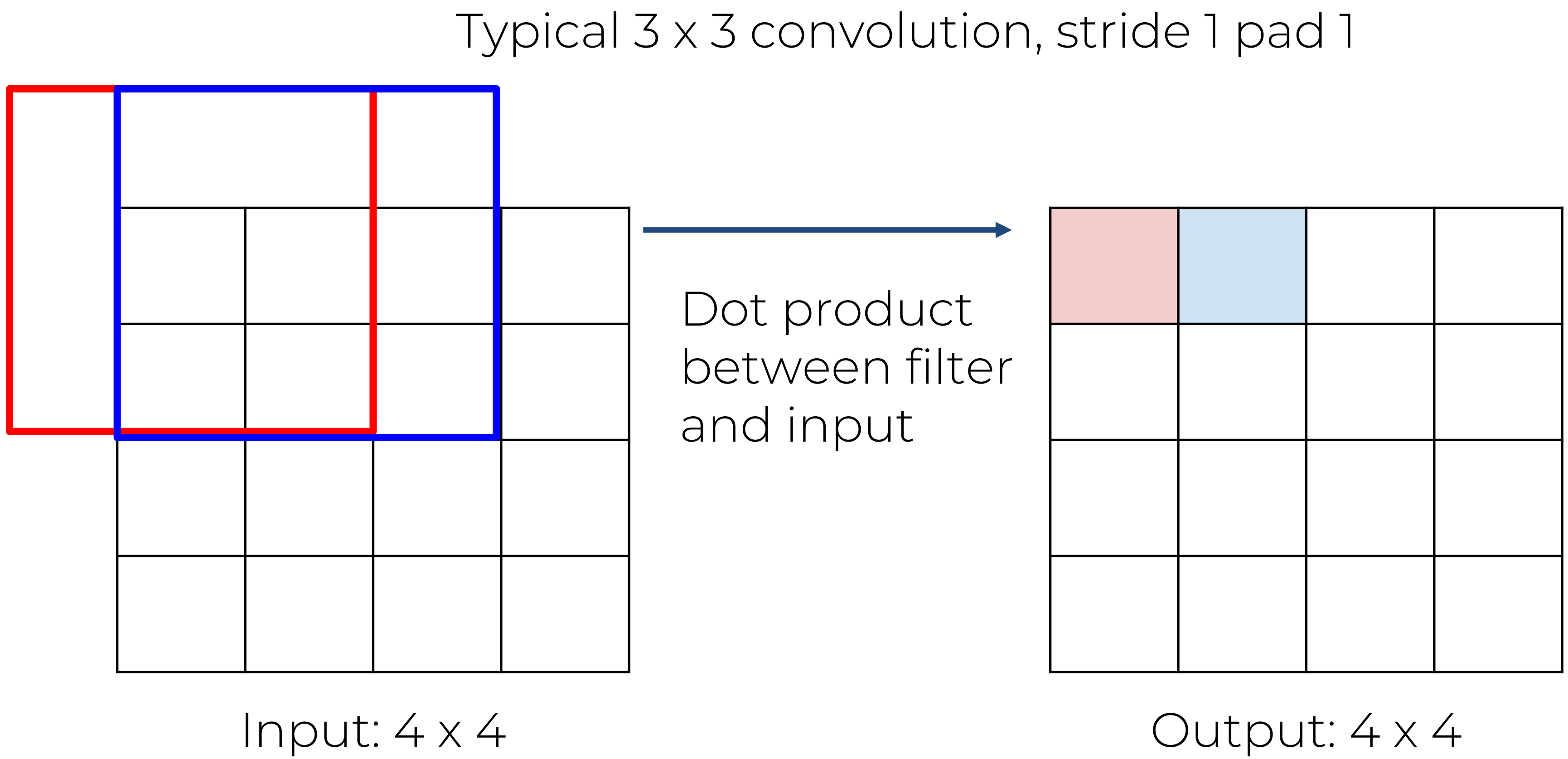
Output: 4 x 4

Learnable Upsampling: "Deconvolution"

Typical 3 x 3 convolution, stride 1 pad 1



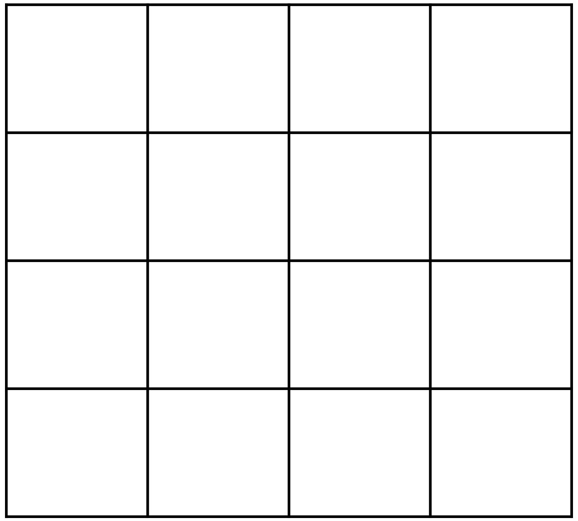
Learnable Upsampling: "Deconvolution"



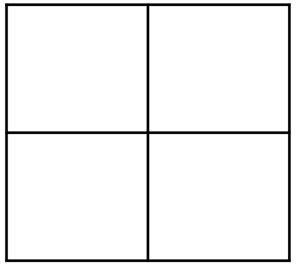
F.-F. Li, A. Karpathy and J. Johnson

Learnable Upsampling: "Deconvolution"

Typical 3 x 3 convolution, stride 2 pad 1



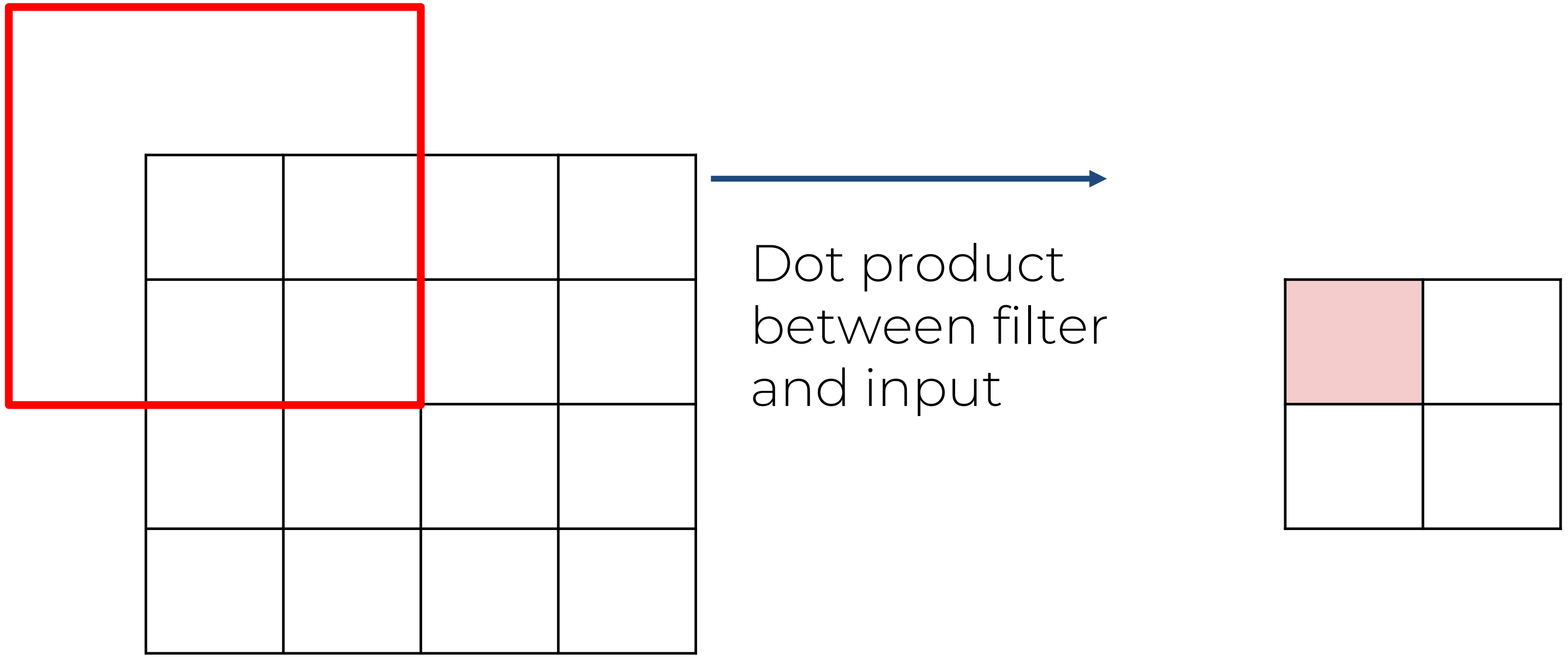
Input: 4 x 4



Output: 2 x 2

Learnable Upsampling: "Deconvolution"

Typical 3 x 3 convolution, **stride 2** pad 1

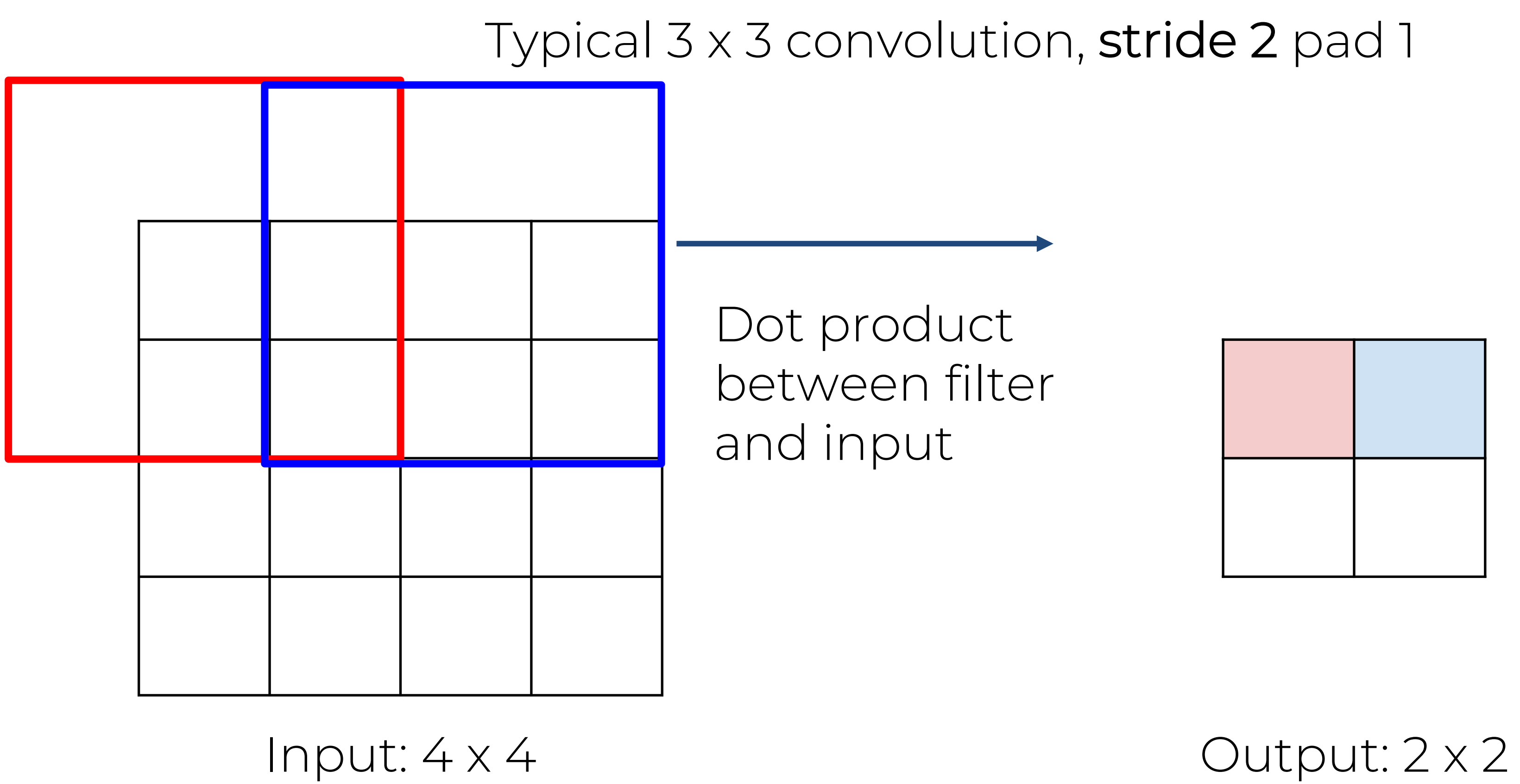


Input: 4 x 4

Output: 2 x 2

F.-F. Li, A. Karpathy and J. Johnson

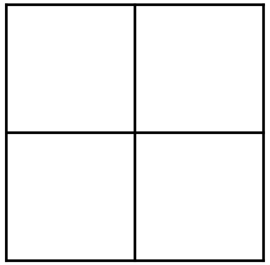
Learnable Upsampling: “Deconvolution”



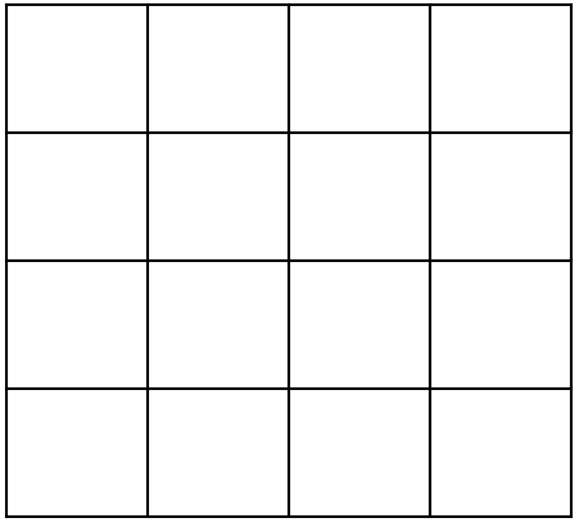
F.-F. Li, A. Karpathy and J. Johnson

Learnable Upsampling: “Deconvolution”

3 x 3 deconvolution, stride 2 pad 1

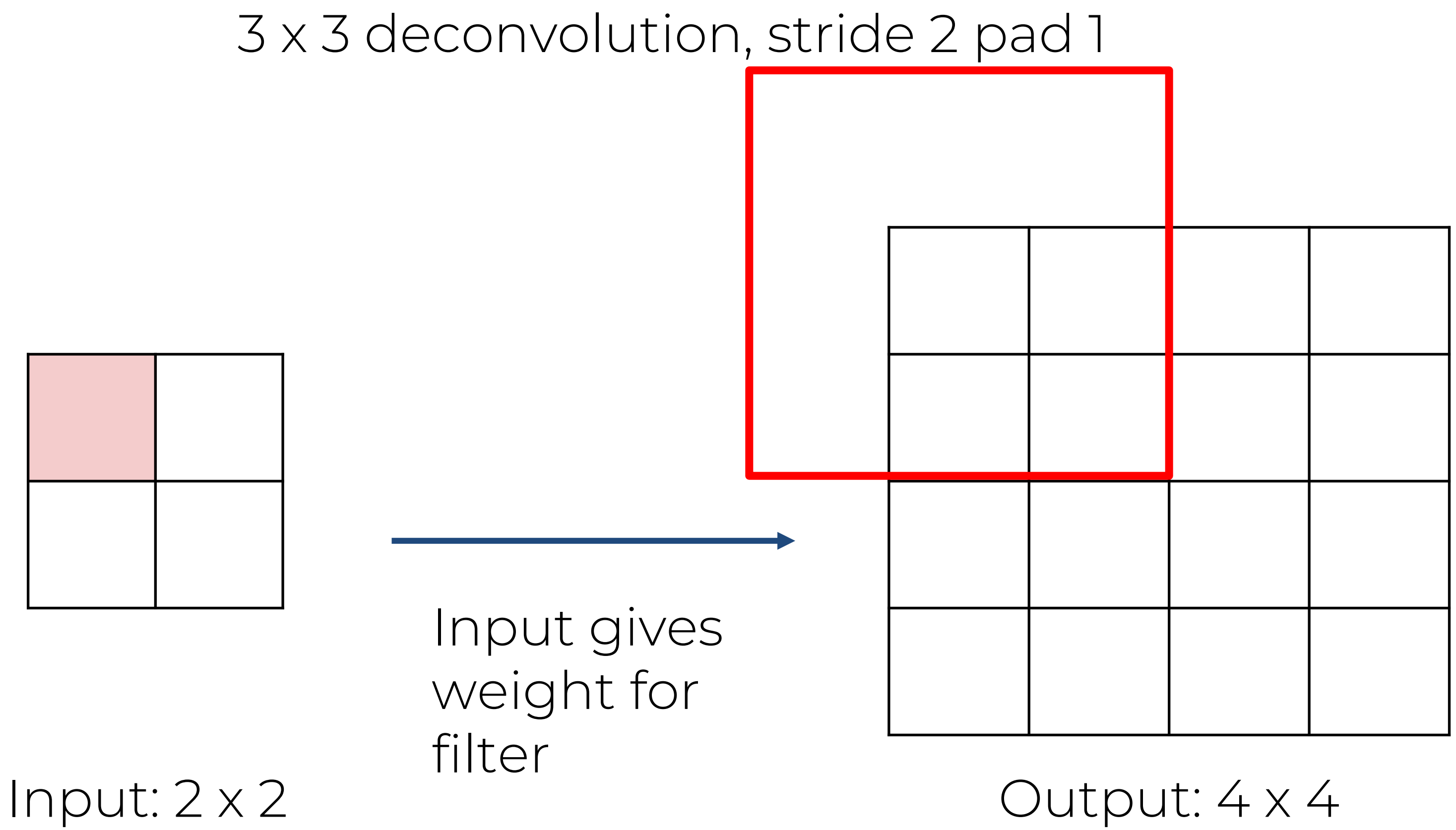


Input: 2 x 2

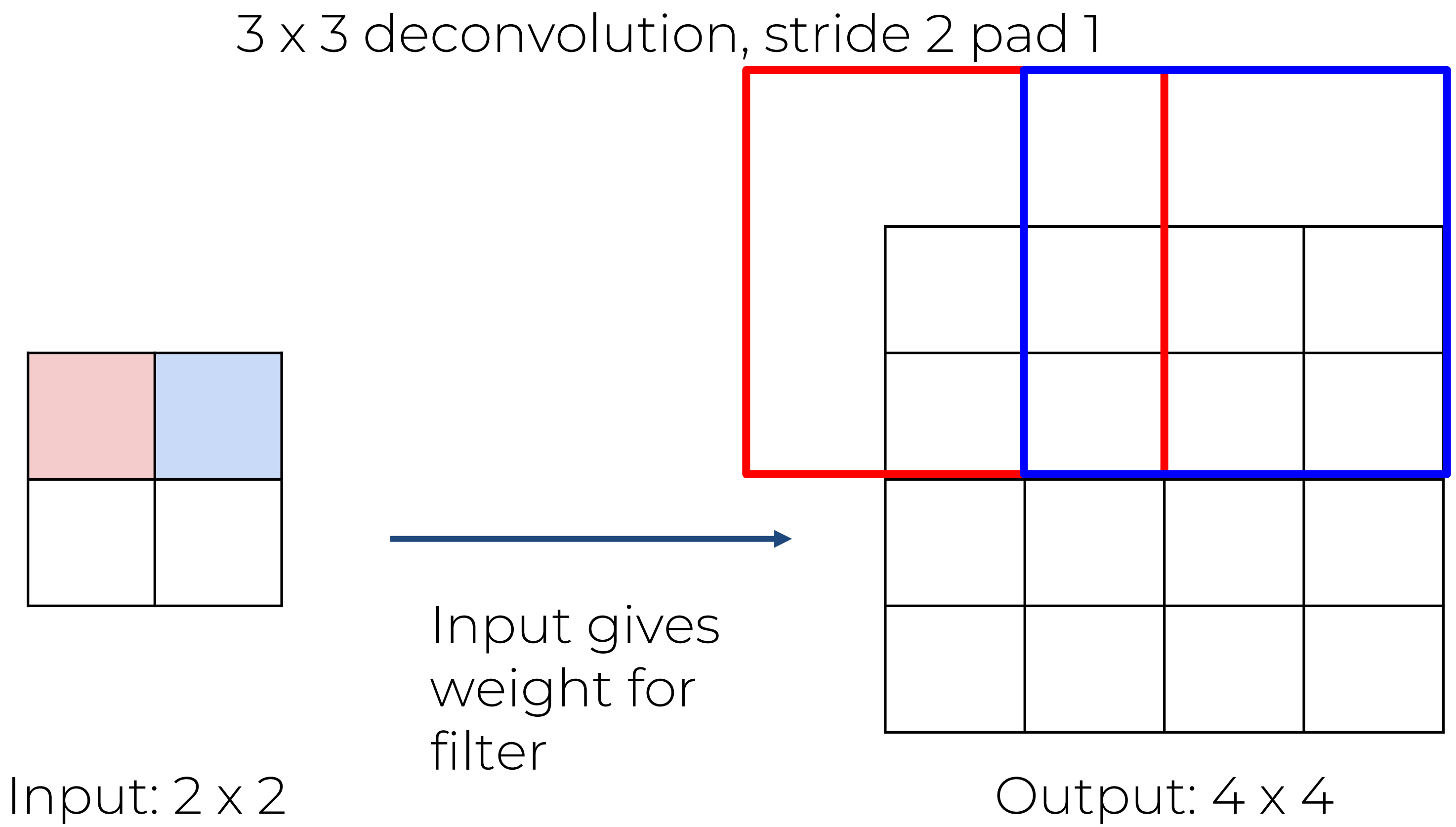


Output: 4 x 4

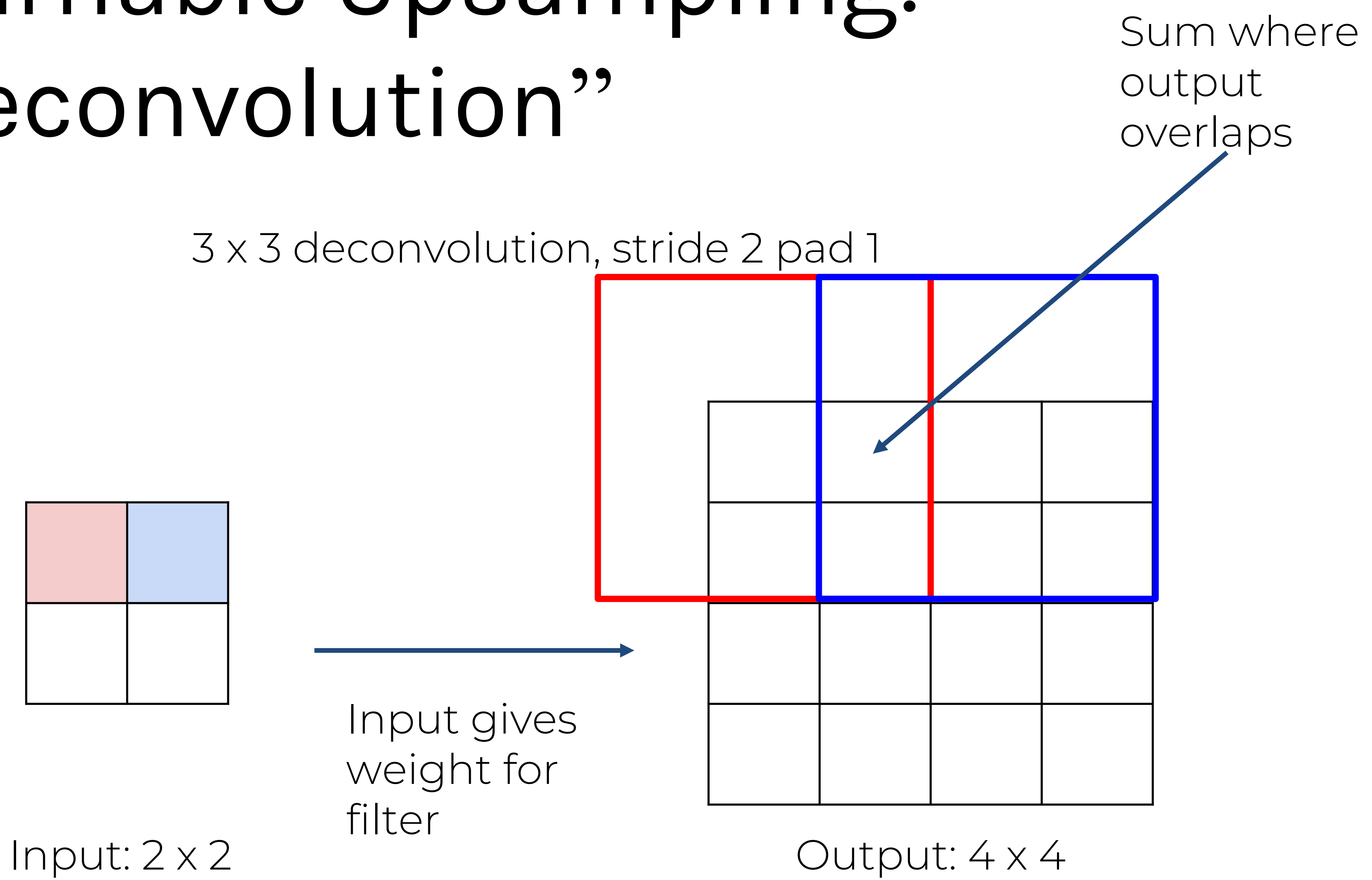
Learnable Upsampling: "Deconvolution"



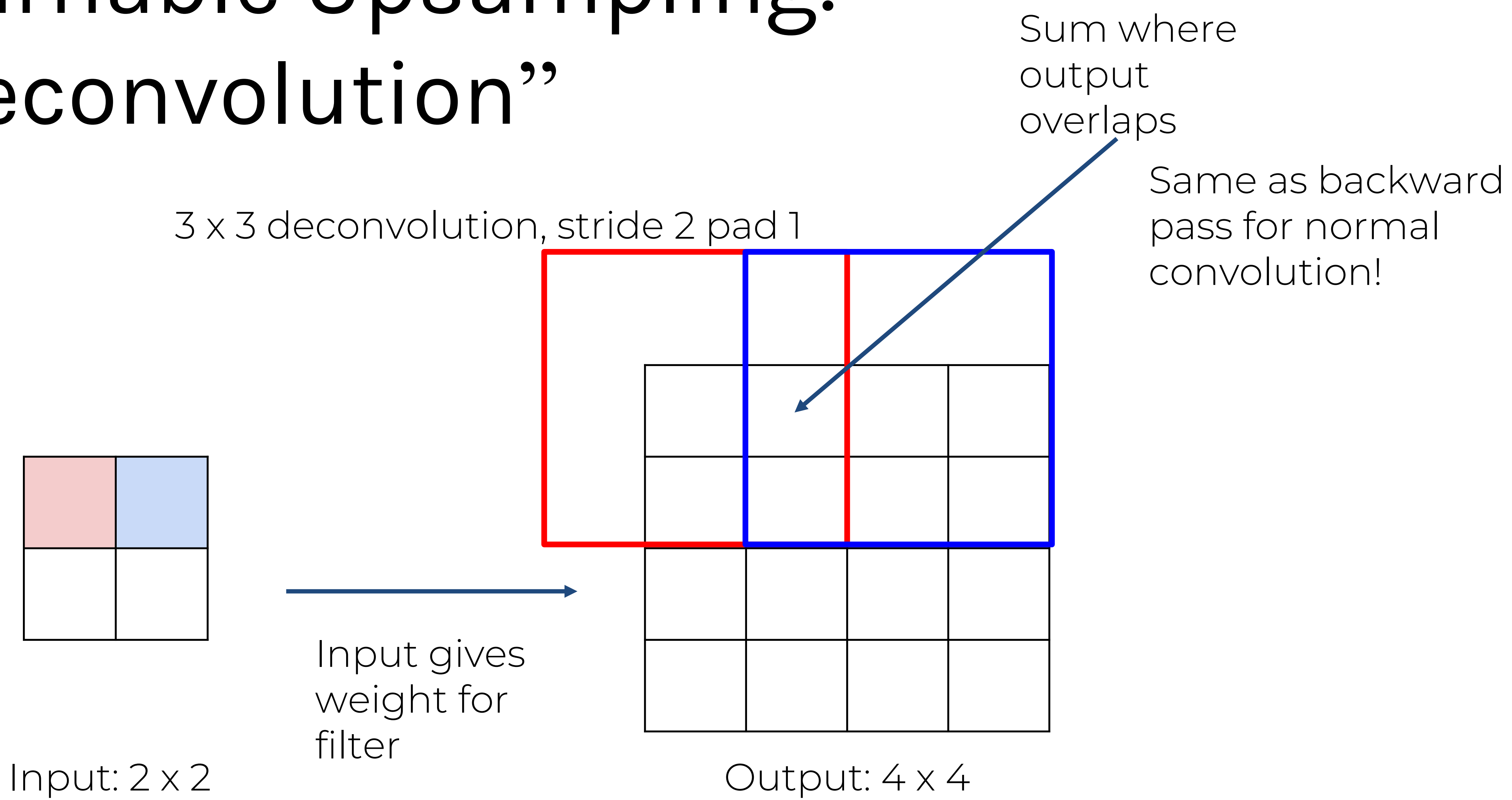
Learnable Upsampling: "Deconvolution"



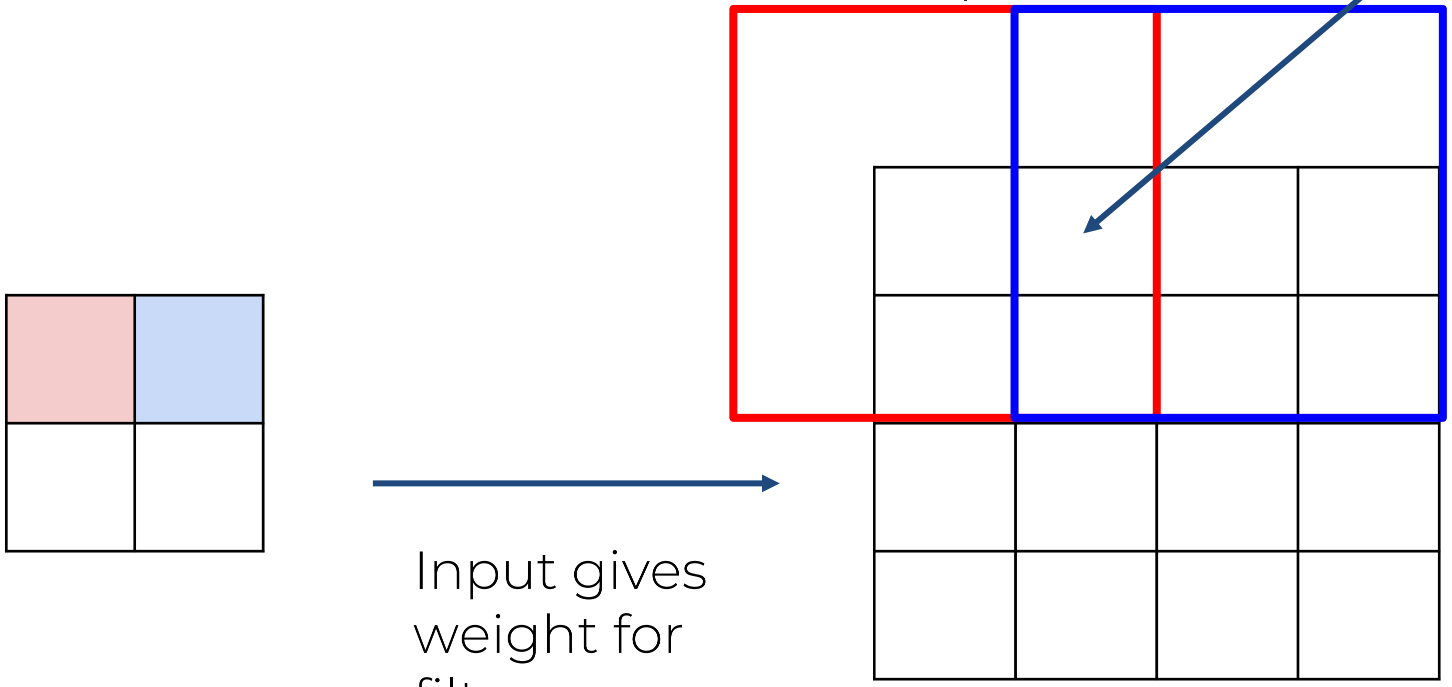
Learnable Upsampling: "Deconvolution"



Learnable Upsampling: "Deconvolution"



Learnable Upsampling: "Deconvolution"

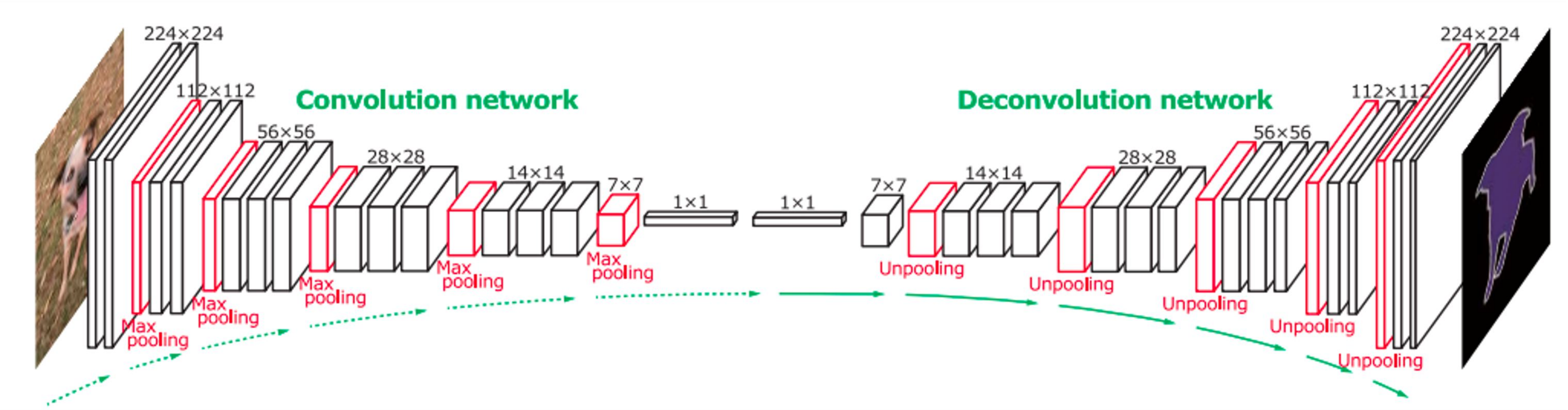


Same as backward pass for normal convolution!

"Deconvolution" is a bad name, already defined as "inverse of convolution"

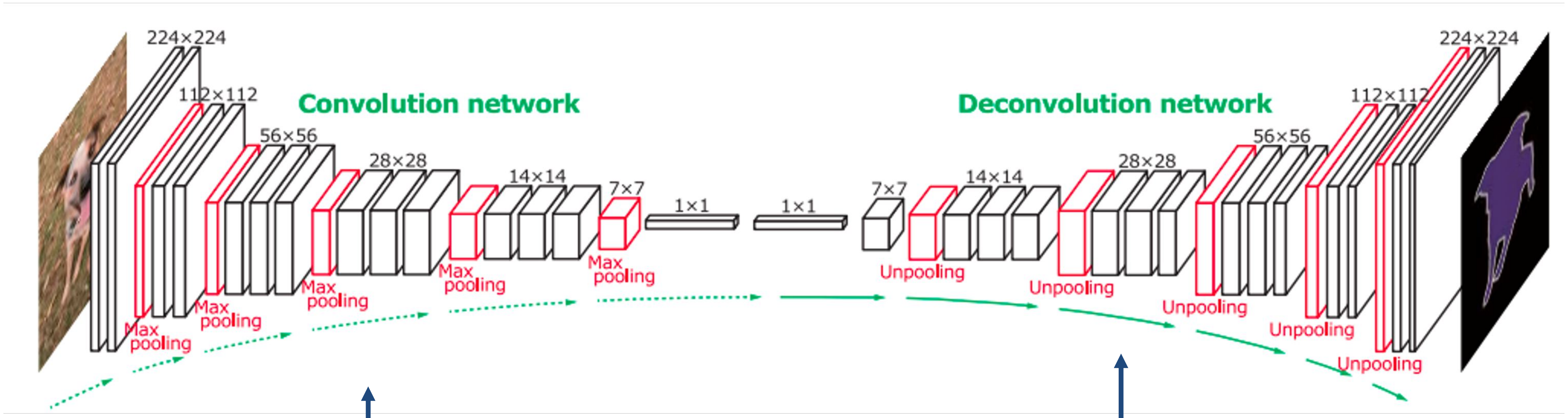
Better names:
convolution transpose,
backward strided convolution,
1/2 strided convolution,
upconvolution

Semantic Segmentation: Upsampling



Noh et al, "Learning Deconvolution Network for Semantic Segmentation", ICCV 2015

Semantic Segmentation: Upsampling



Normal VGG

"Upside down" VGG

6 days of training on Titan X…

Noh et al, "Learning Deconvolution Network for Semantic Segmentation", ICCV 2015

Instance Segmentation

Detect instances, give category, label pixels

"simultaneous detection and segmentation" (SDS)

Lots of recent work (MS-COCO)

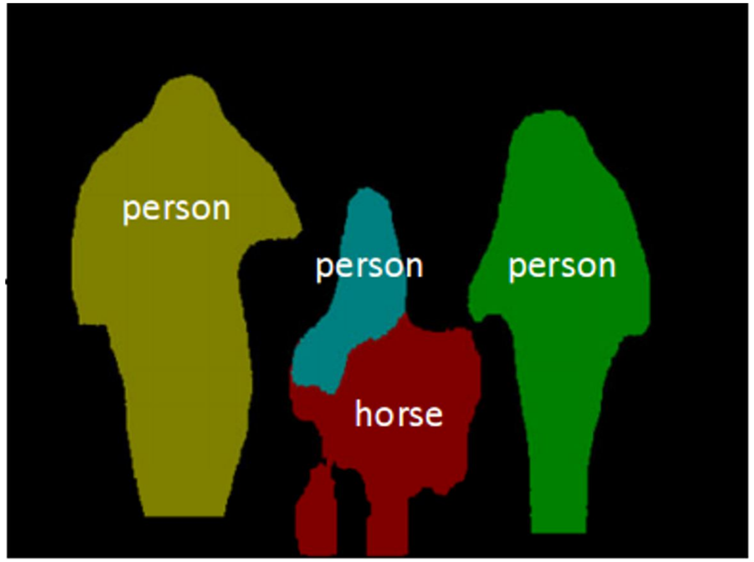
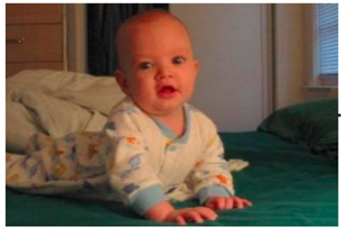


Figure credit: Dai et al, "Instance-aware Semantic Segmentation via Multi-task Network Cascades", arXiv 2015

Instance Segmentation

Similar to R-CNN, but with segments



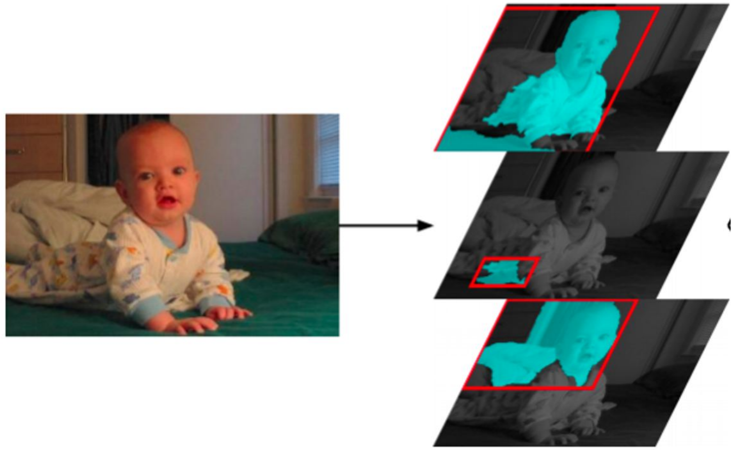
Hariharan et al, “Simultaneous Detection and Segmentation”, ECCV 2014

Instance Segmentation

Similar to R-CNN, but with segments

Proposal Generation

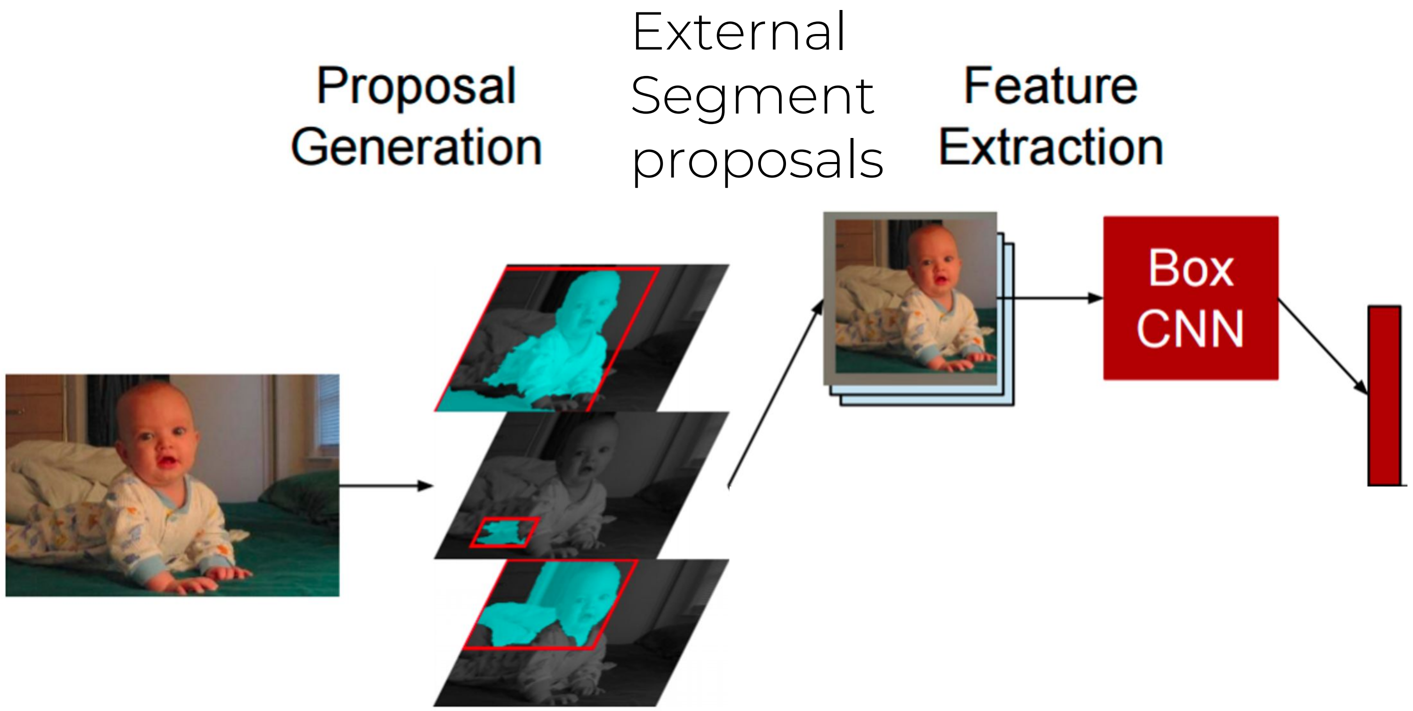
External Segment proposals



Hariharan et al, "Simultaneous Detection and Segmentation", ECCV 2014

Instance Segmentation

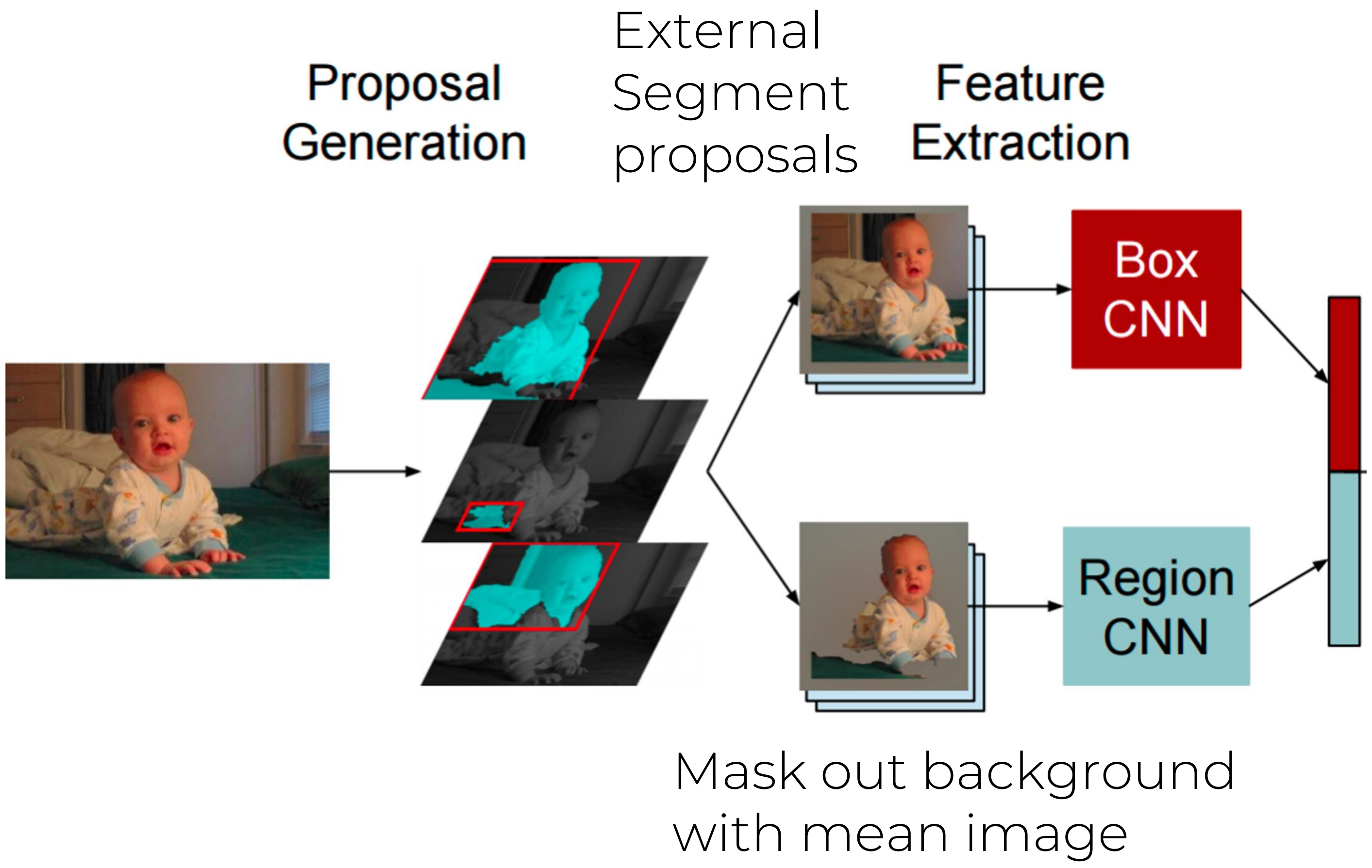
Similar to R-CNN, but with segments



Hariharan et al, "Simultaneous Detection and Segmentation", ECCV 2014

Instance Segmentation

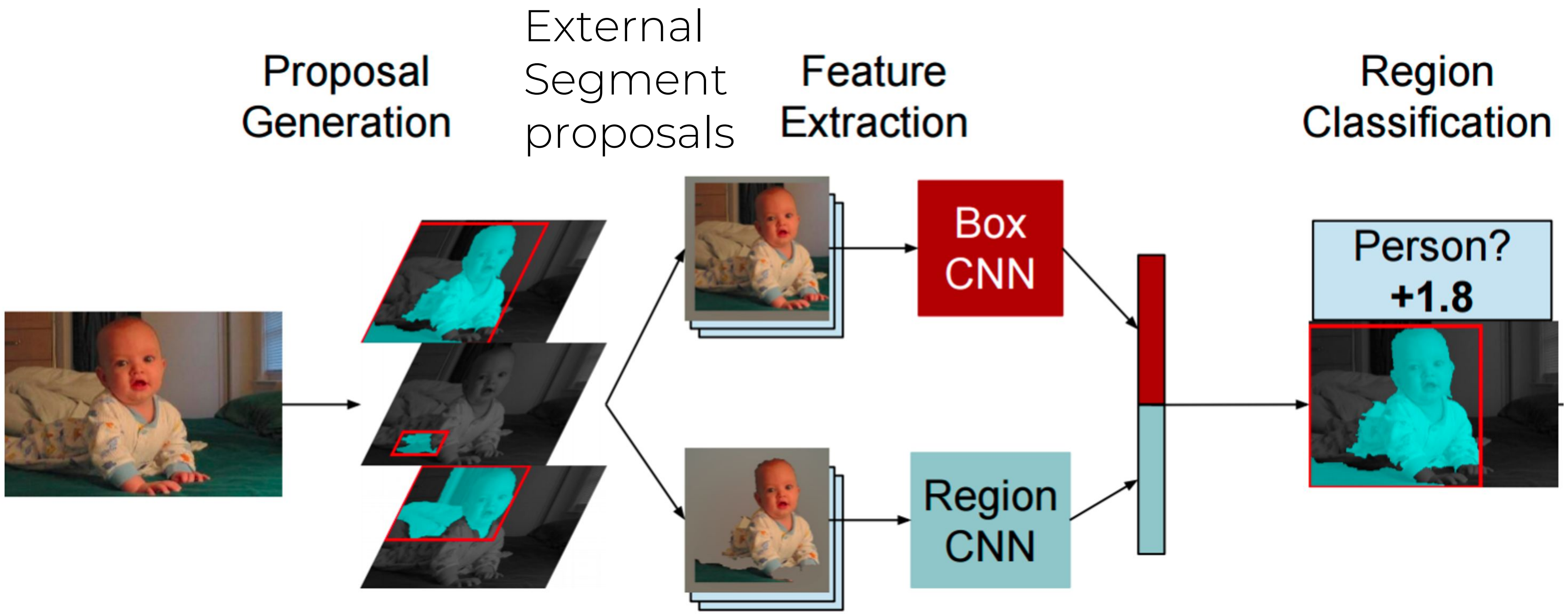
Similar to R-CNN, but with segments



Hariharan et al, “Simultaneous Detection and Segmentation”, ECCV 2014

Instance Segmentation

Similar to R-CNN, but with segments

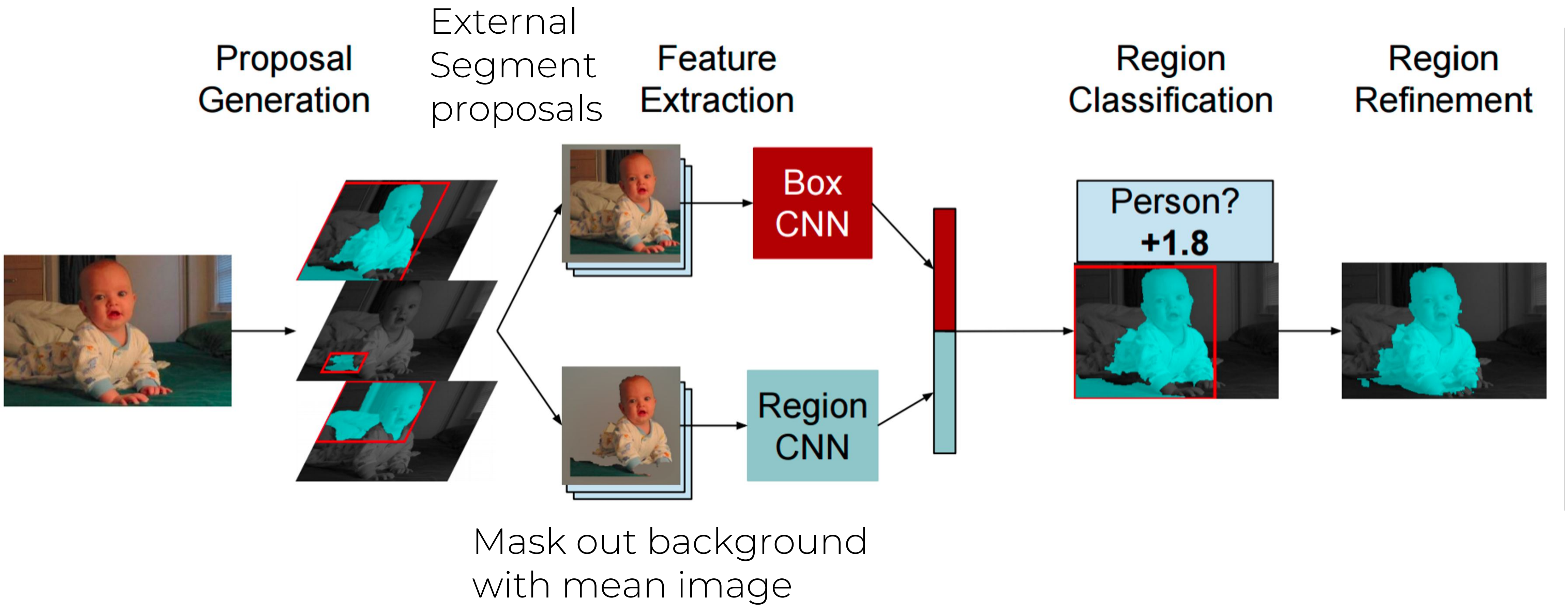


Mask out background with mean image

Hariharan et al, "Simultaneous Detection and Segmentation", ECCV 2014

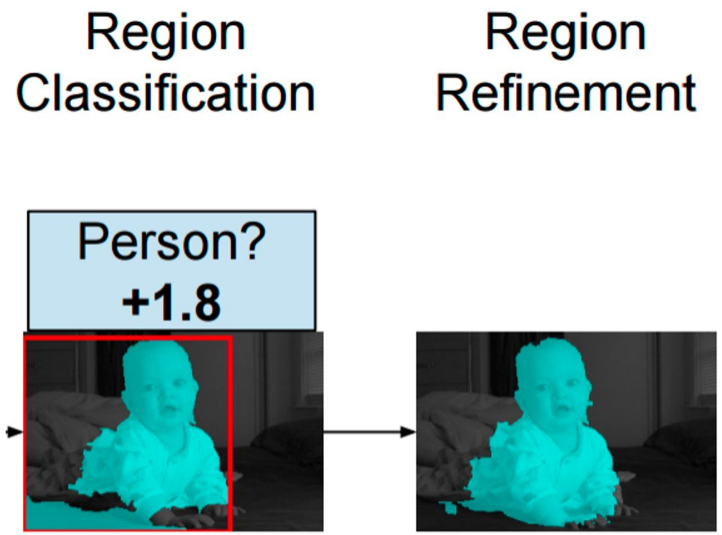
Instance Segmentation

Similar to R-CNN, but with segments



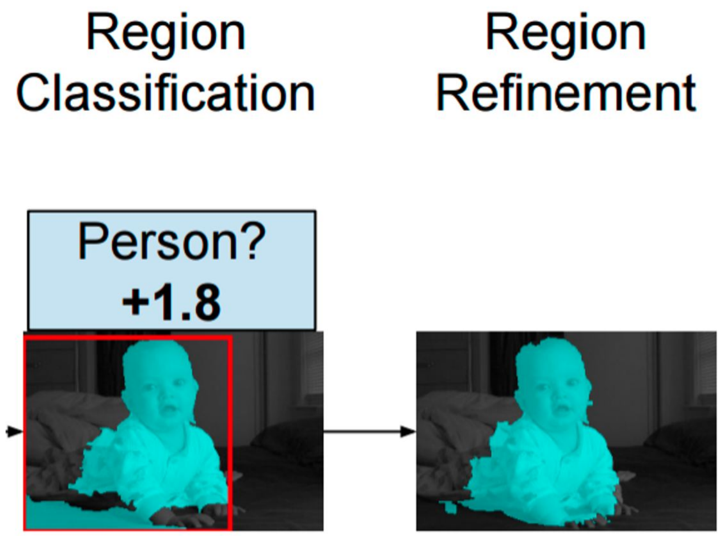
Hariharan et al, “Simultaneous Detection and Segmentation”, ECCV 2014

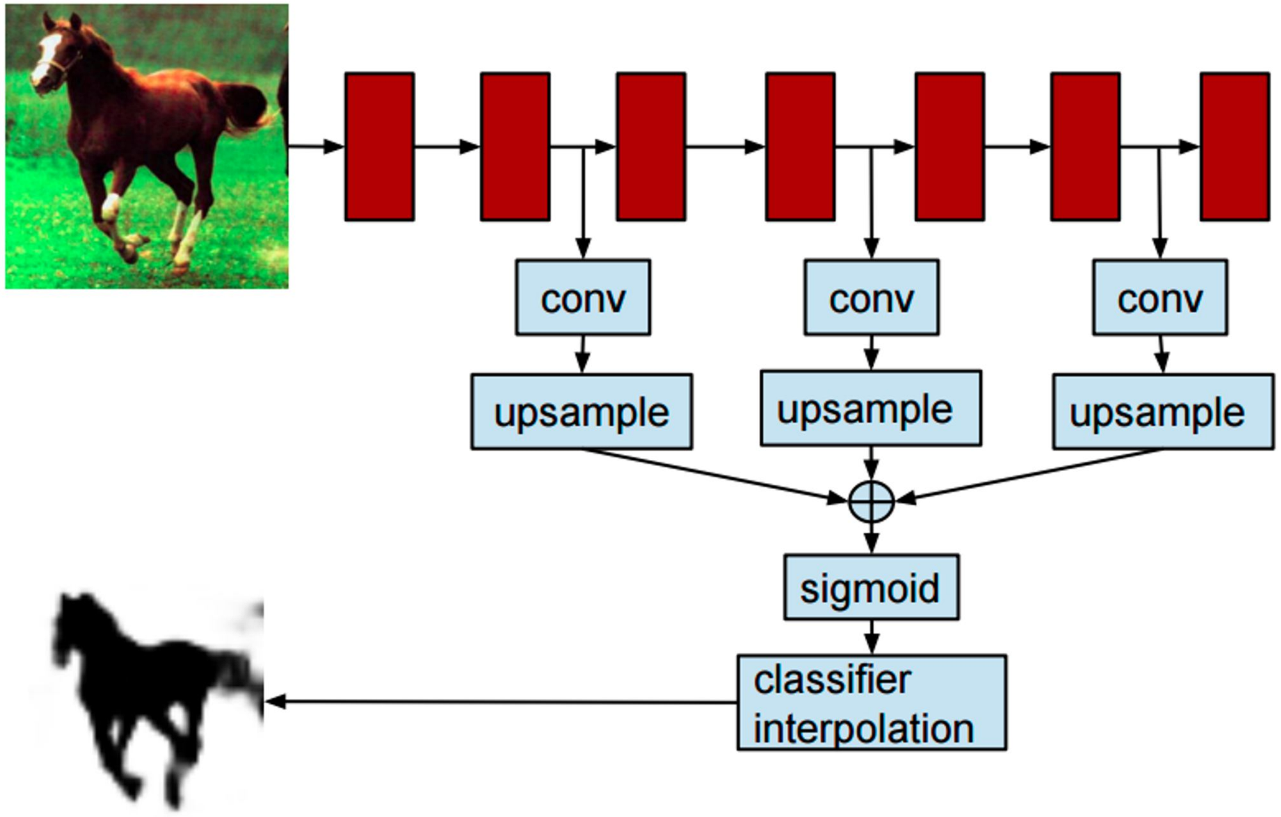
Instance Segmentation: Hypercolumns



Hariharan et al, "Hypercolumns for Object Segmentation and Fine-grained Localization", CVPR 2015

Instance Segmentation: Hypercolumns





Hariharan et al, “Hypercolumns for Object Segmentation and Fine-grained Localization”, CVPR 2015

Instance Segmentation: Cascades

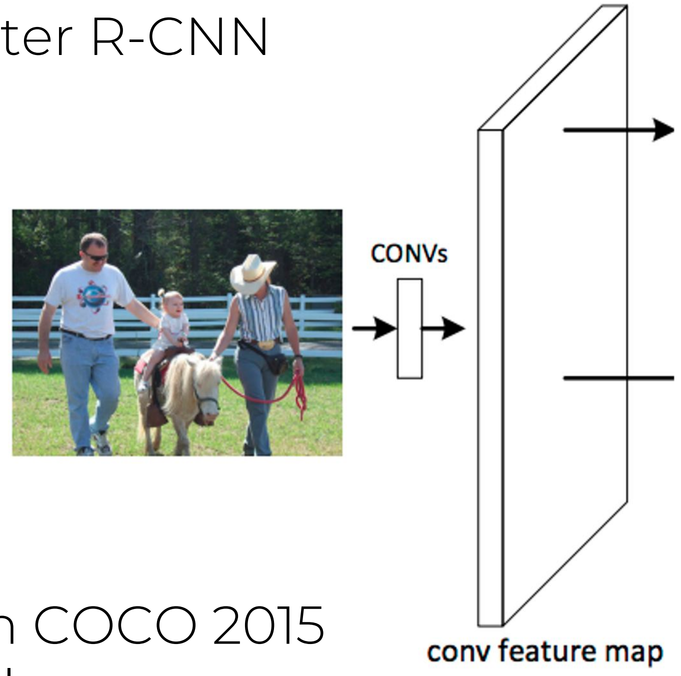
Similar to
Faster R-CNN

Won COCO 2015
challenge
(with ResNet)

Dai et al, "Instance-aware Semantic Segmentation via Multi-task Network Cascades", arXiv 2015

Instance Segmentation: Cascades

Similar to
Faster R-CNN



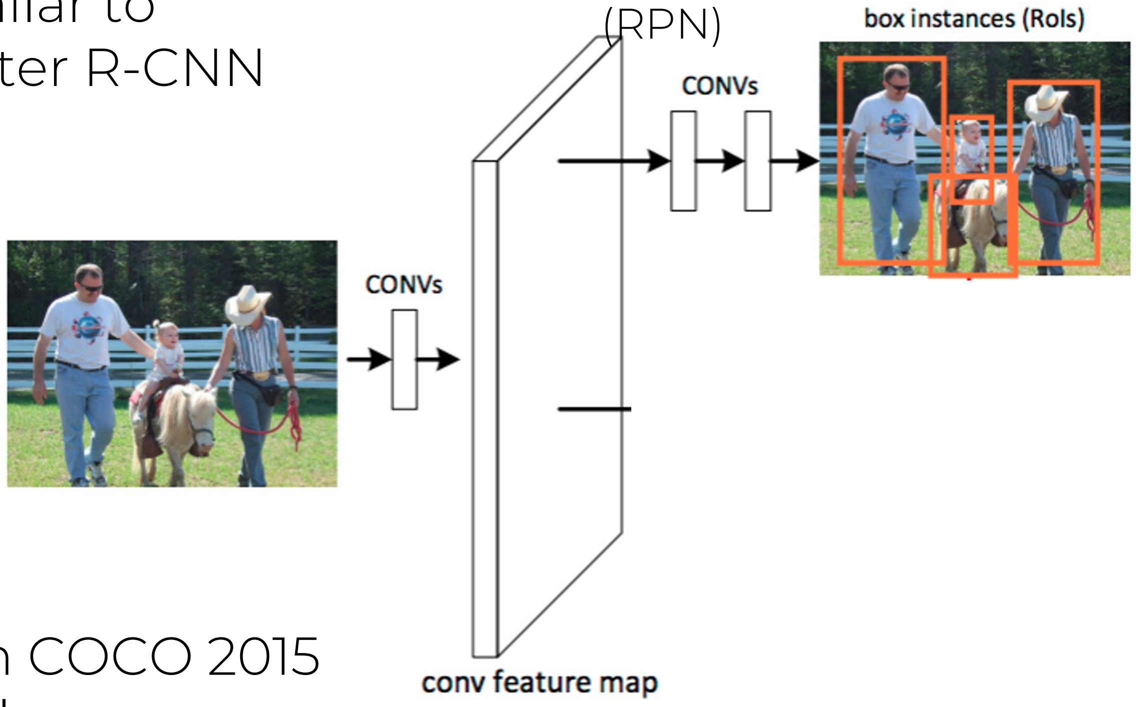
Won COCO 2015
challenge
(with ResNet)

Dai et al, "Instance-aware Semantic Segmentation via Multi-task Network Cascades", arXiv 2015

Instance Segmentation: Cascades

Similar to
Faster R-CNN

Region proposal network
(RPN)



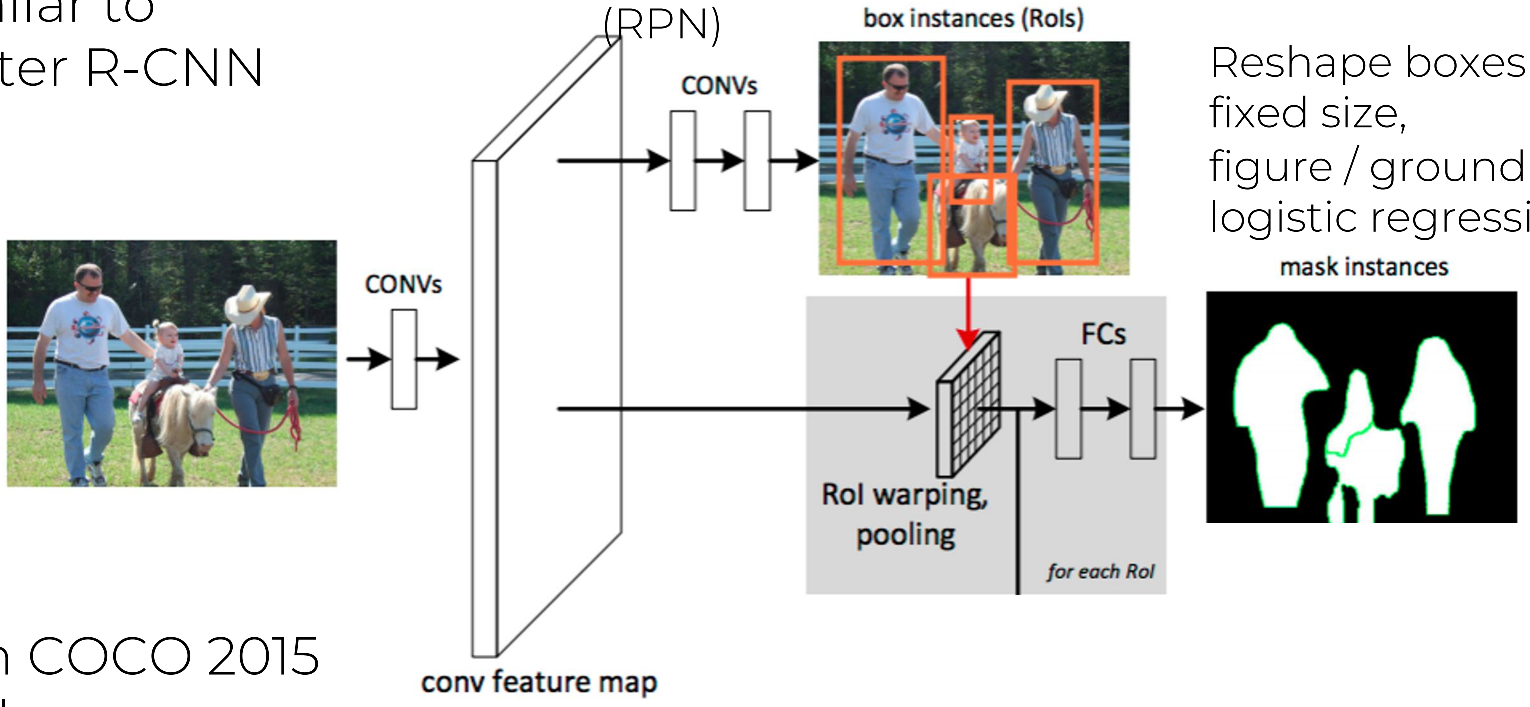
Won COCO 2015
challenge
(with ResNet)

Dai et al, "Instance-aware Semantic Segmentation via Multi-task Network Cascades",
arXiv 2015

Instance Segmentation: Cascades

Similar to
Faster R-CNN

Region proposal network
(RPN)



Reshape boxes to
fixed size,
figure / ground
logistic regression

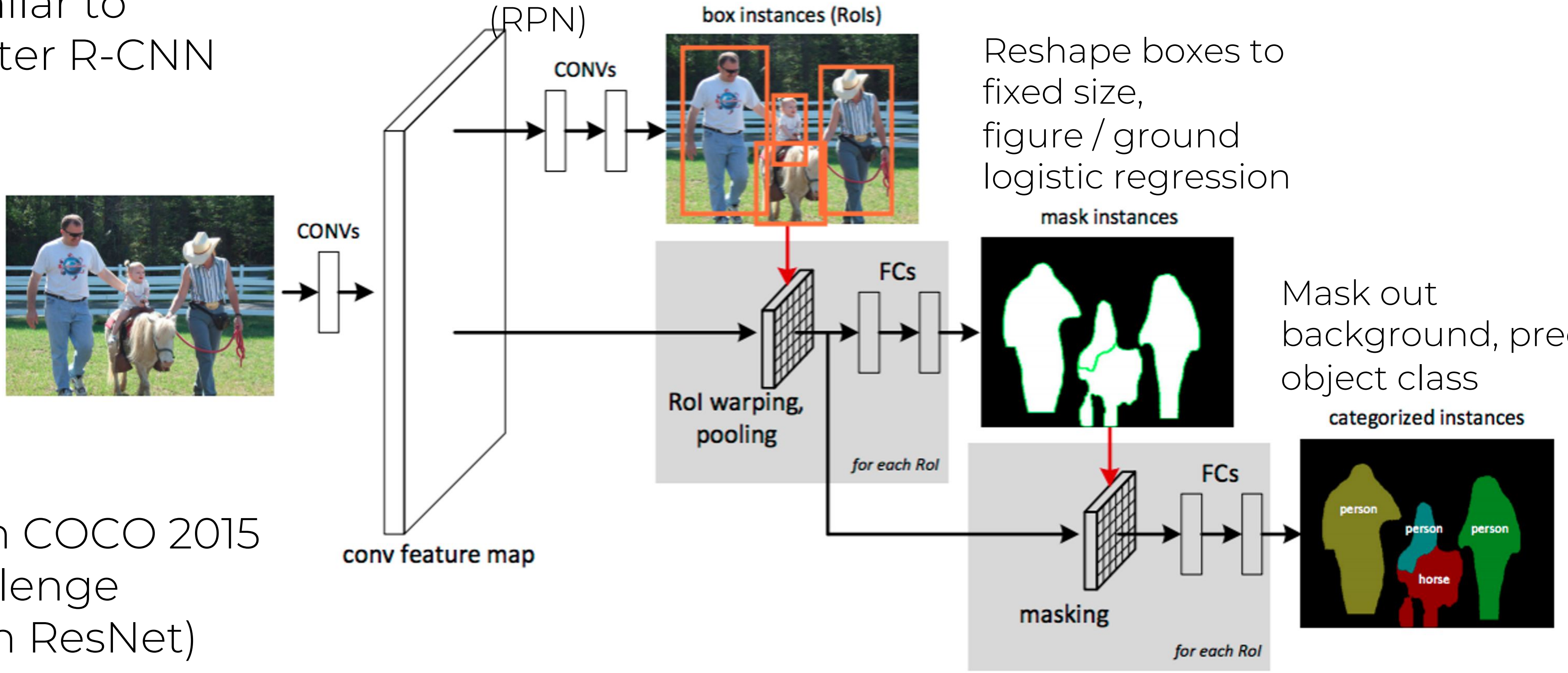
Won COCO 2015
challenge
(with ResNet)

Dai et al, "Instance-aware Semantic Segmentation via Multi-task Network Cascades",
arXiv 2015

Instance Segmentation: Cascades

Similar to
Faster R-CNN

Region proposal network (RPN)



Reshape boxes to
fixed size,
figure / ground
logistic regression

Mask out
background, predict
object class

Won COCO 2015
challenge
(with ResNet)

Dai et al, "Instance-aware Semantic Segmentation via Multi-task Network Cascades", arXiv 2015

Instance Segmentation: Cascades

Similar to
Faster R-CNN

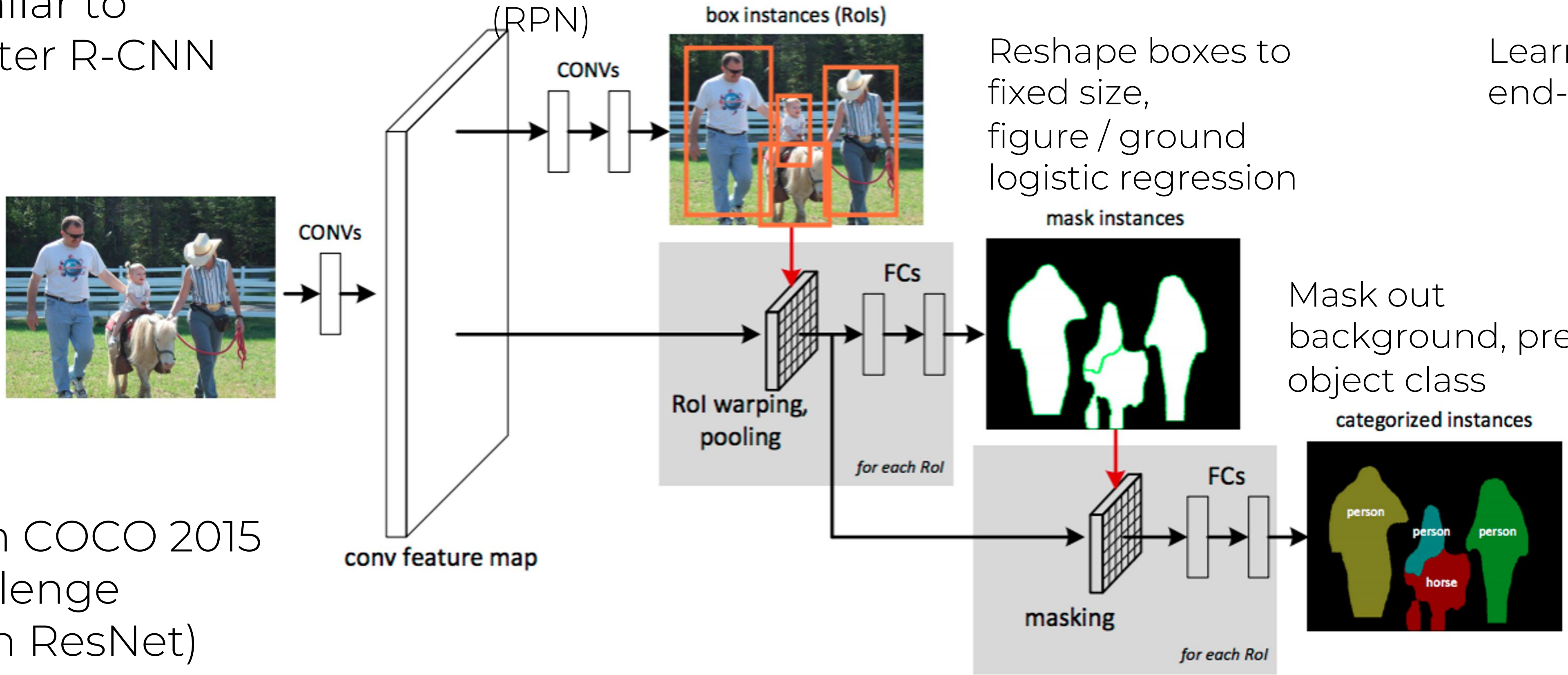
Won COCO 2015
challenge
(with ResNet)

Region proposal network
(RPN)

Reshape boxes to
fixed size,
figure / ground
logistic regression

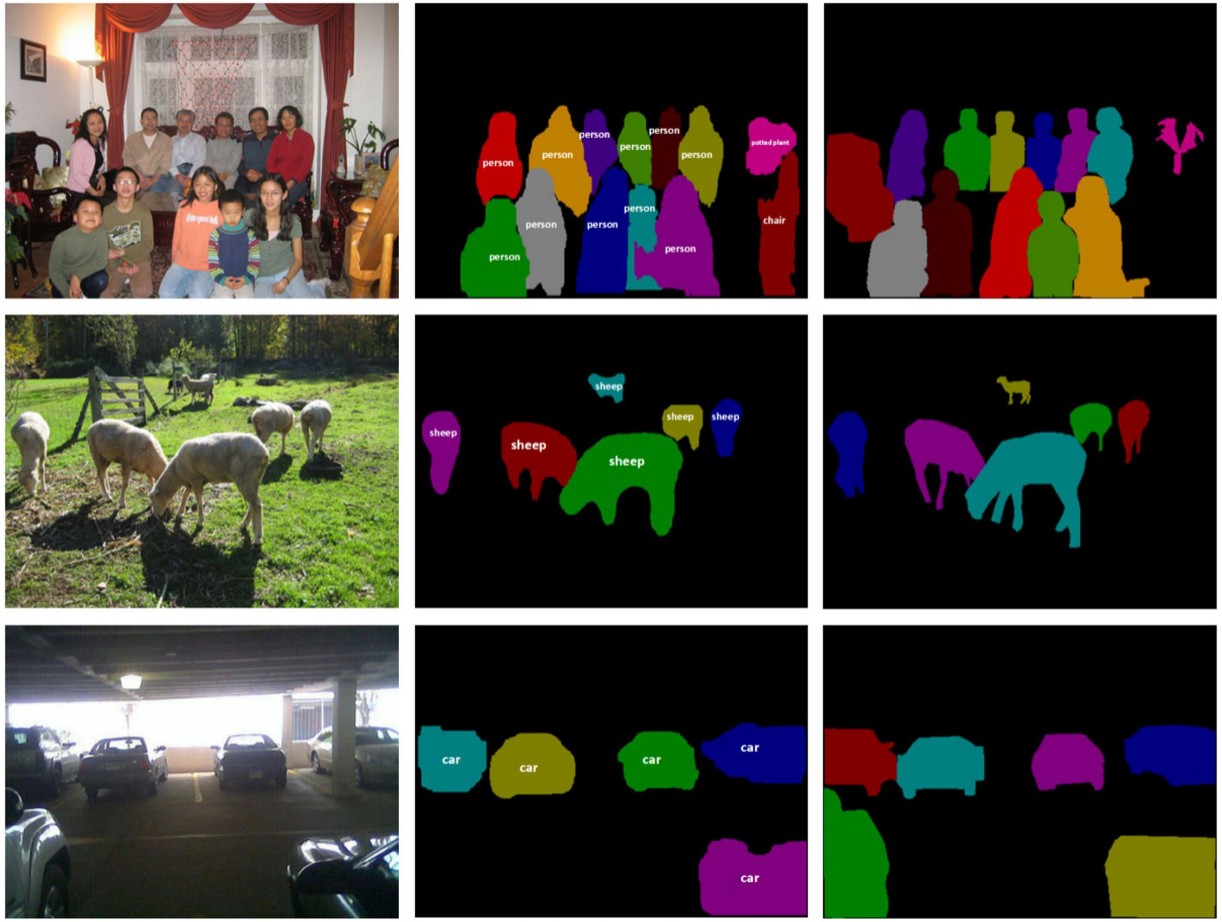
Mask out
background, predict
object class

Learn entire model
end-to-end!



Dai et al, "Instance-aware Semantic Segmentation via Multi-task Network Cascades", arXiv 2015

Instance Segmentation: Cascades



Predictions Ground truth

Dai et al, "Instance-aware Semantic Segmentation via Multi-task Network Cascades", arXiv 2015

Mask R-CNN: Model Overview

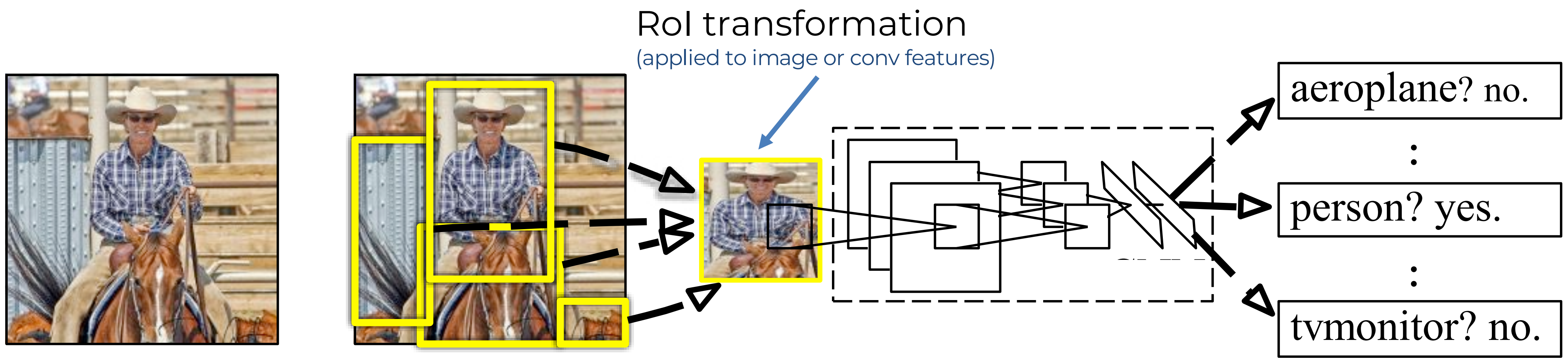
R-CNN-style detection system

1. Backbone architecture
2. Feature Pyramid Network (FPN)
3. Region Proposal Network (RPN)
4. Region of interest feature alignment (RoIAlign)
5. Multi-task network head
    - Box classifier
    - Box regressor
    - Mask predictor
    - Keypoint predictor

Modular composition of many recent ideas

R. Girshick

0. R-CNN-style Approach to Object Detection



Input image

Object / region proposals (RoIs)
(external or internal to network)

Deep Learning-based region classifier

Detection = Region classification + box regression + …

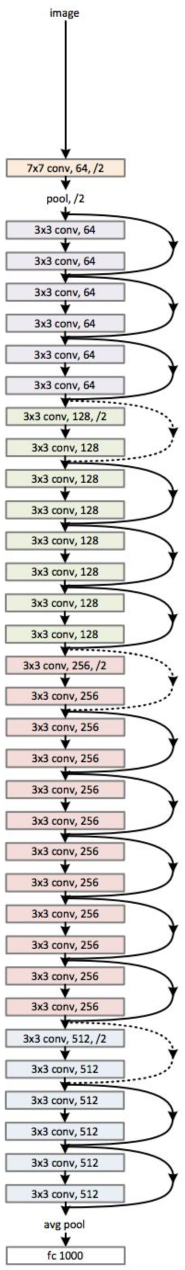
General formula for **R**egion-based **CNN** models

Girshick, Donahue, Darrell, Malik. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. CVPR 2014
Uijlings, van de Sande, Gevers, Smeulders. Selective Search for Object Recognition. IJCV 2013

R. Girshick

1. Backbone ConvNet

Example:
ResNet-34

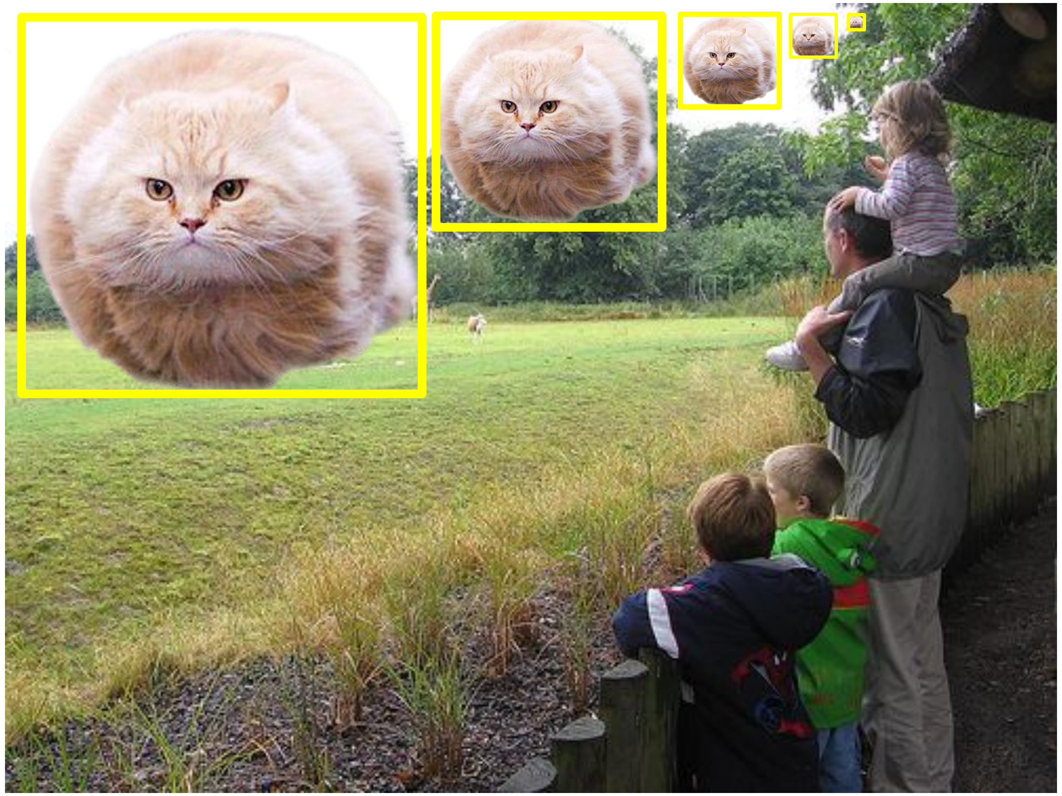


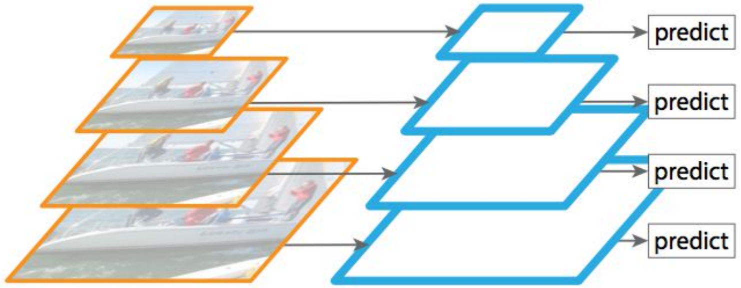
Use any standard ConvNet as a "backbone architecture"

- AlexNet, VGG, ResNet, Inception, Inception-ResNet, ResNeXt, ...
- Use "same" padding everywhere (preserves integer scales)
- Prefer fully convolutional networks (ignoring cls head; see next slide)
- Pre-train on ImageNet classification (or similar)

R. Girshick

2. Scale Invariant Detection

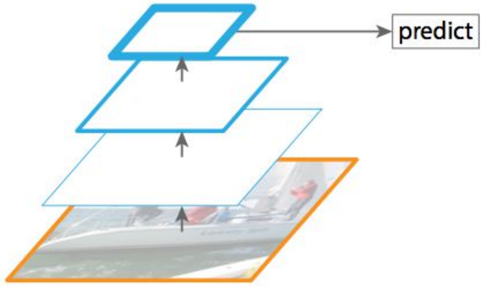




(a) Featurized image pyramid

SLOW!

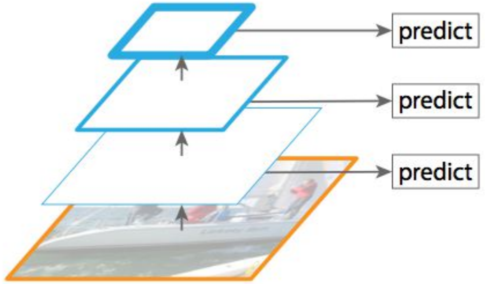
(Viola & Jones, HOG, DPM, multiscale Fast R-CNN, …)



(b) Single feature map

Do nothing; give up; Fast

(Fast R-CNN, YOLO, …)



(c) Pyramidal feature hierarchy

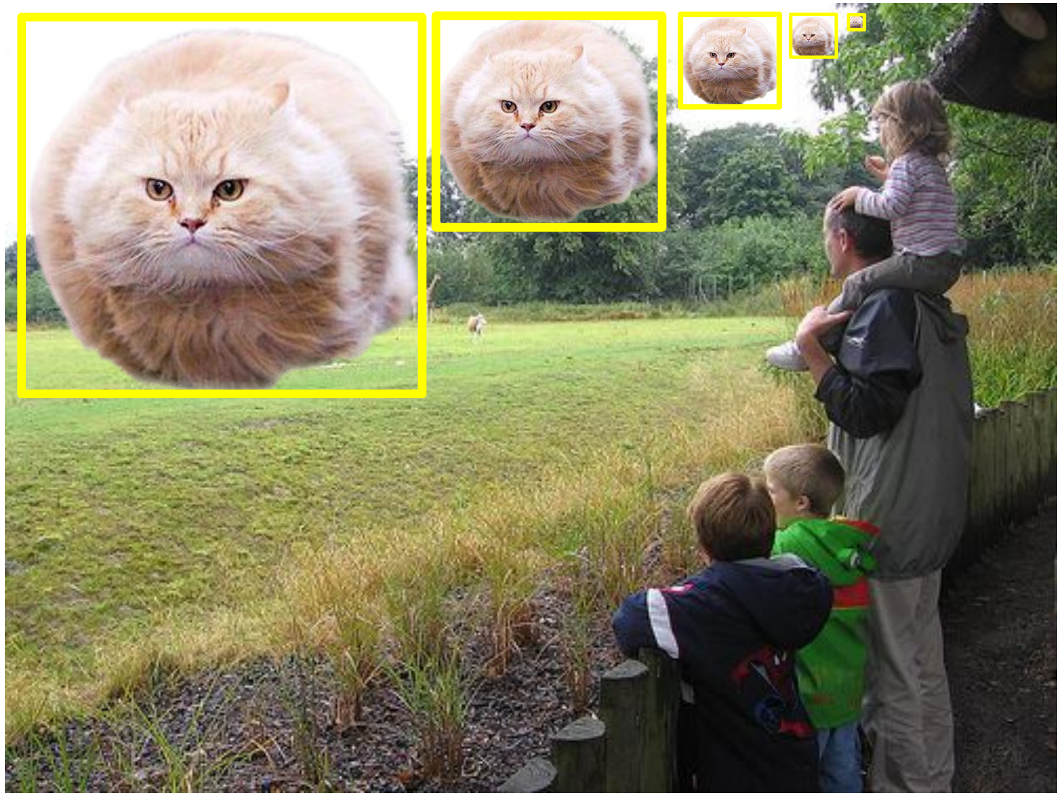
Fast!

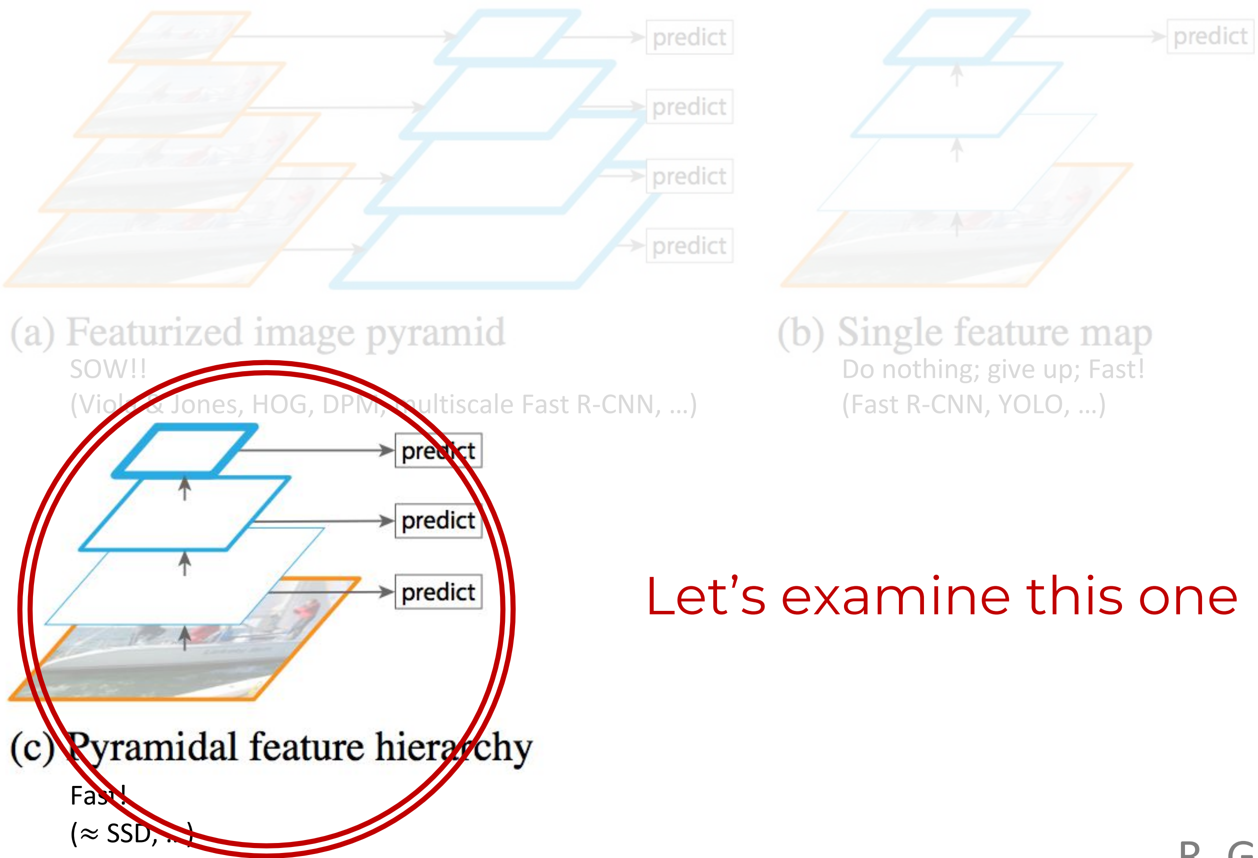
(≈ SSD, …)

Popular strategies for detecting objects over large scale changes

R. Girshick

2. Scale Invariant Detection

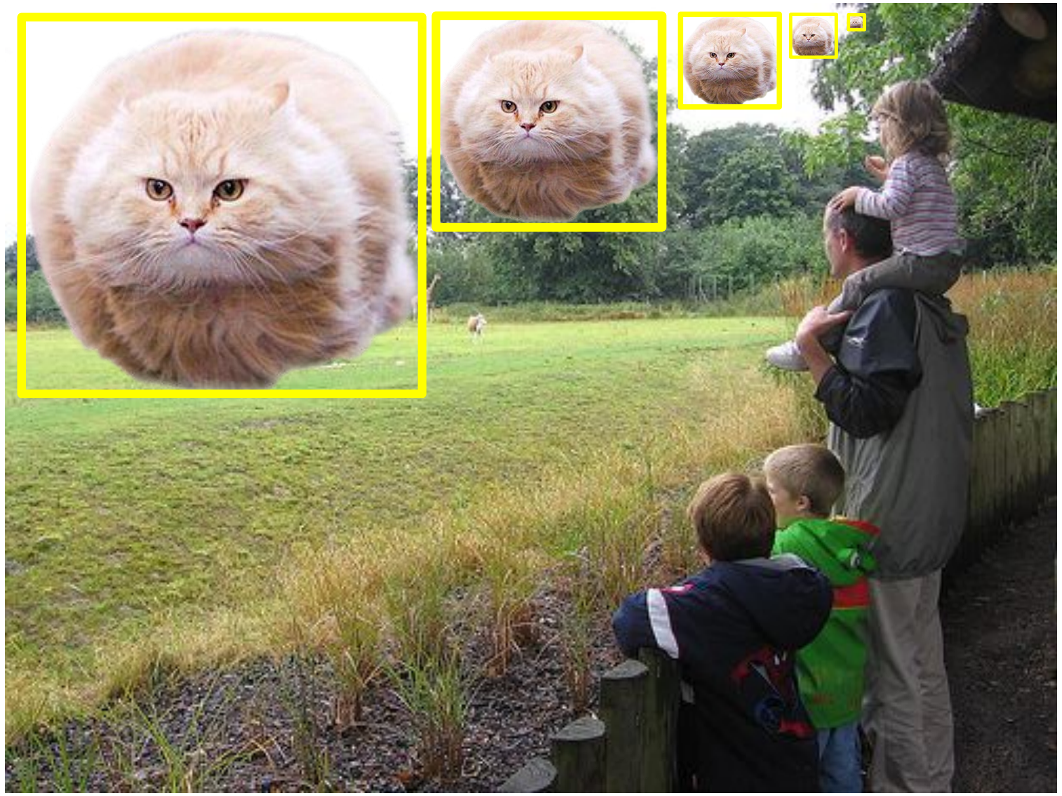


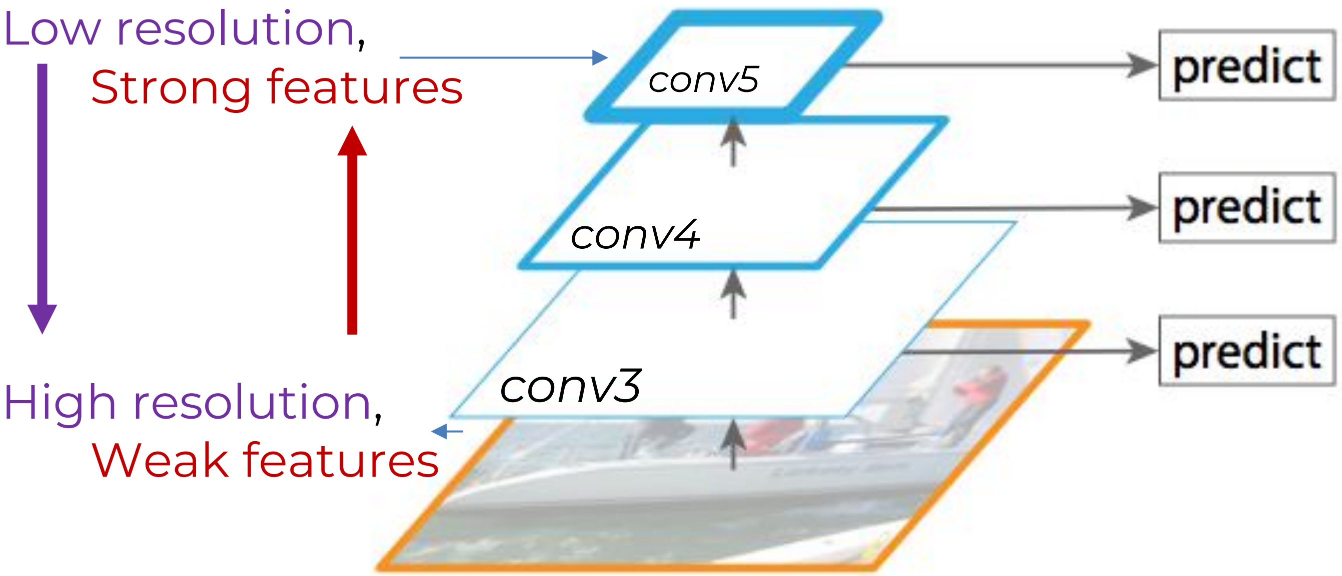


Let's examine this one

R. Girshick

Compromise Feature Quality, but Fast (“Free”)

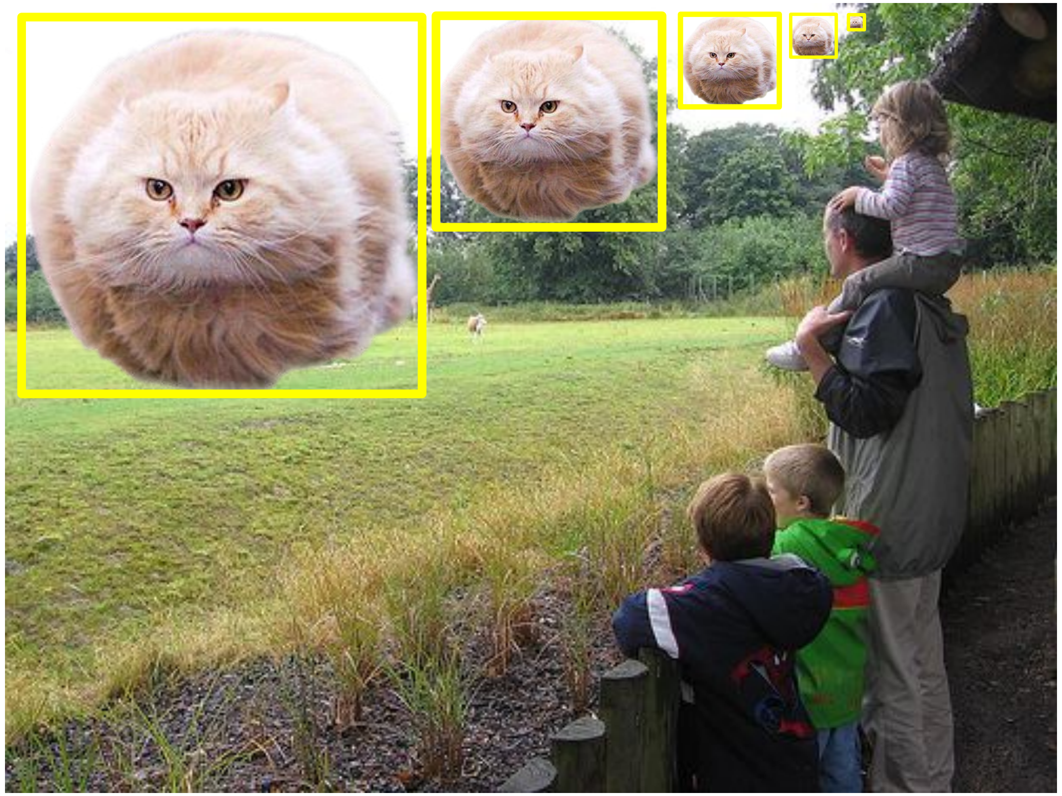


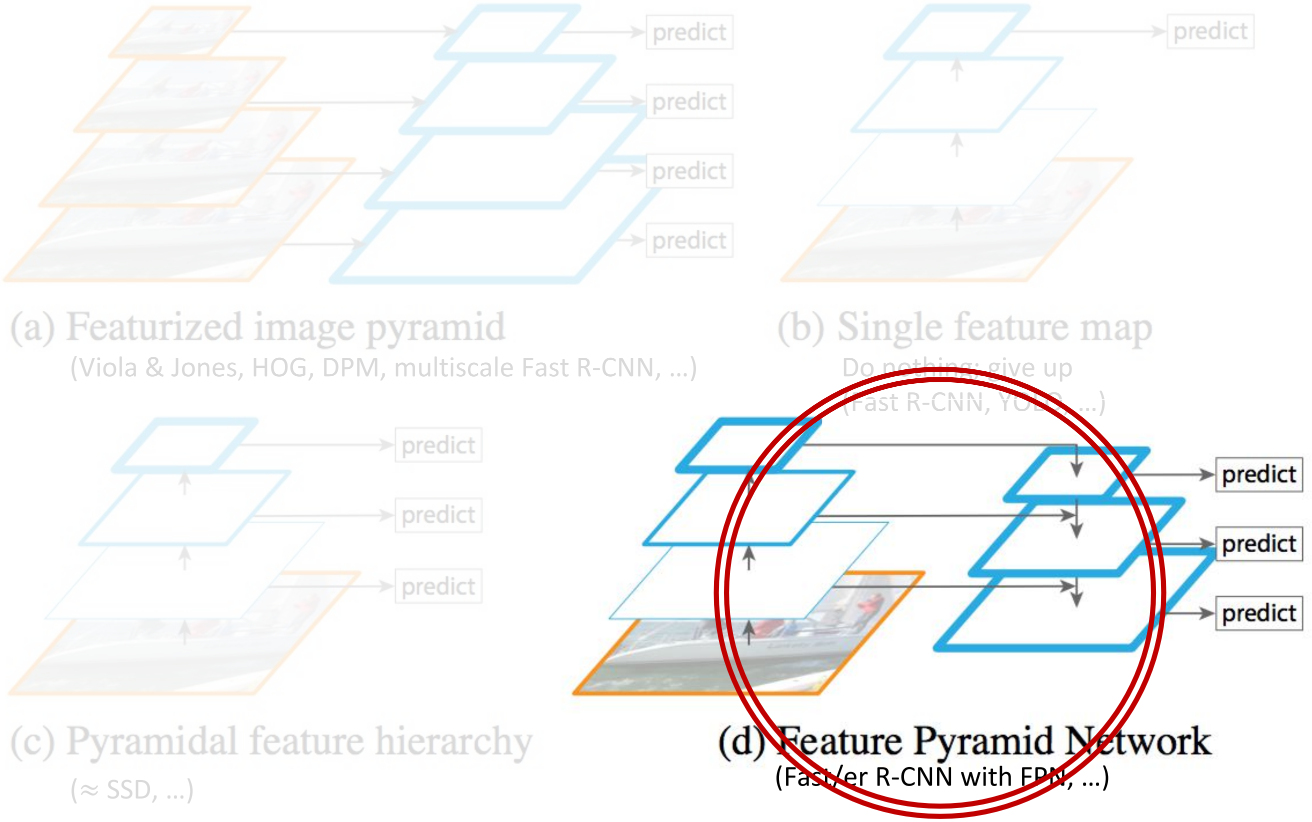


The “native” in-network feature pyramid poses an inherent tradeoff

2. … + Feature Pyramid Network (FPN)

[Optional, but recommended]

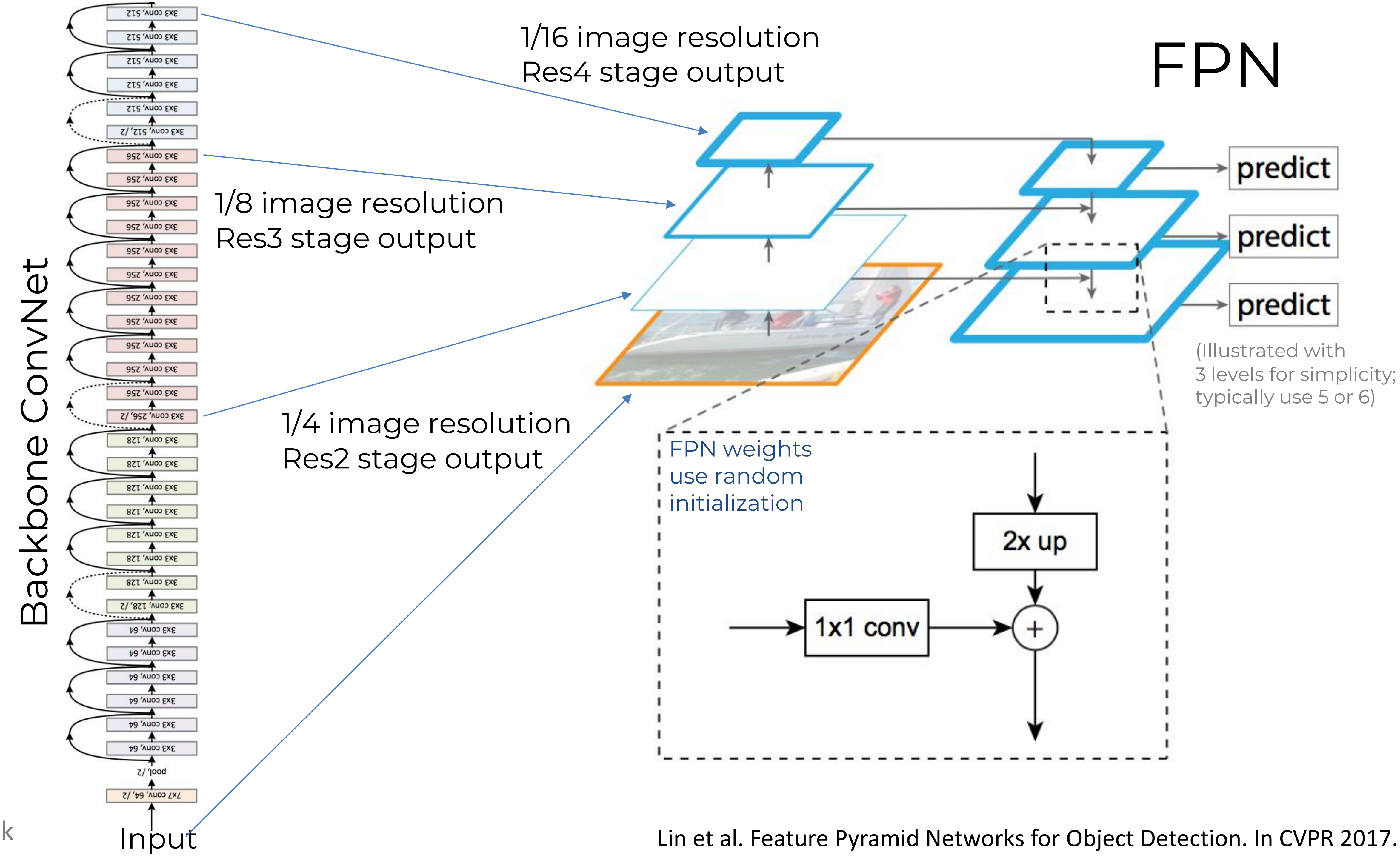




Lin et al. Feature Pyramid Networks for Object Detection. In CVPR 2017.

R. Girshick

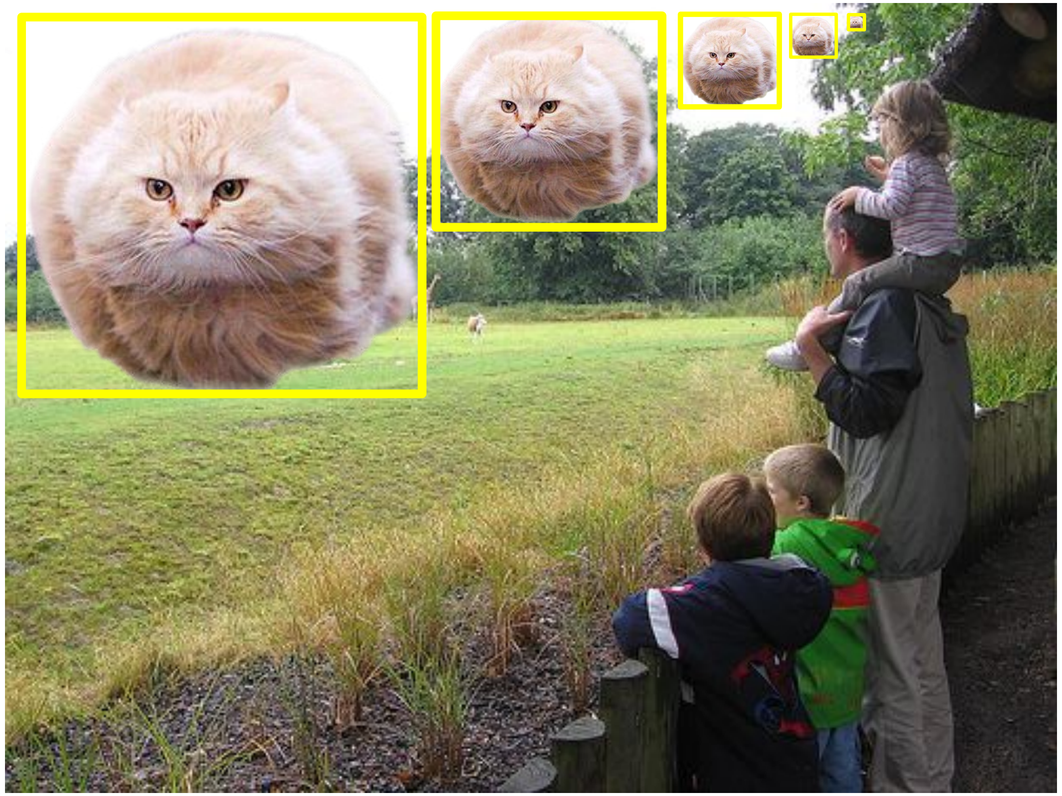
FPN

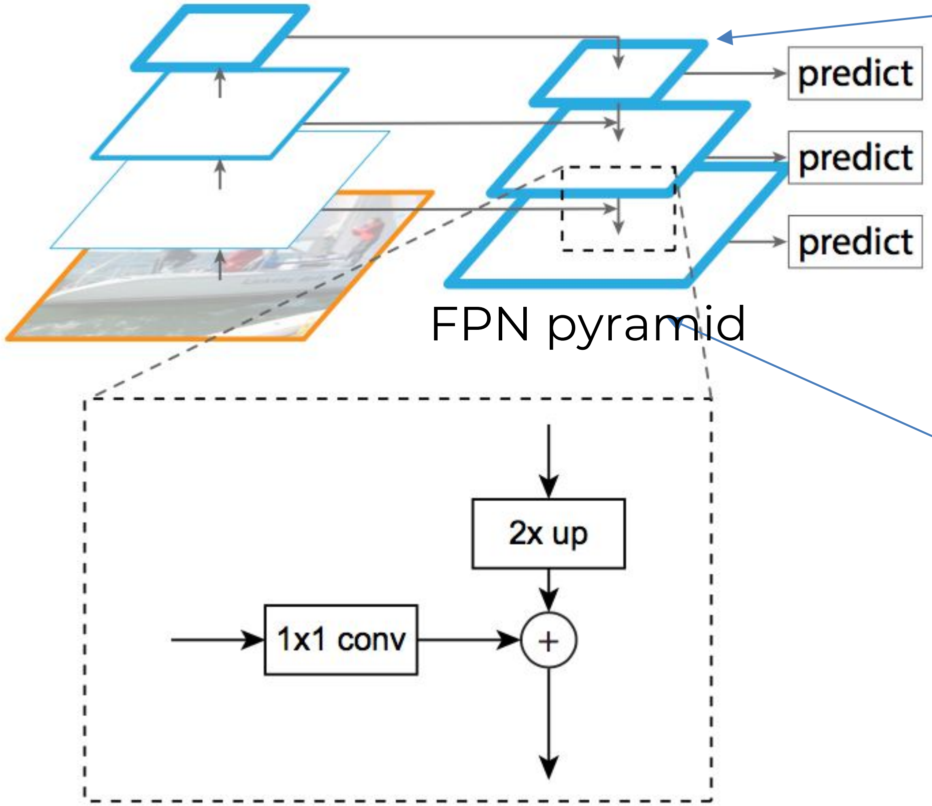


R. Girshick

Lin et al. Feature Pyramid Networks for Object Detection. In CVPR 2017.

No Compromise on Feature Quality, still Fast





Low resolution,
Strong features

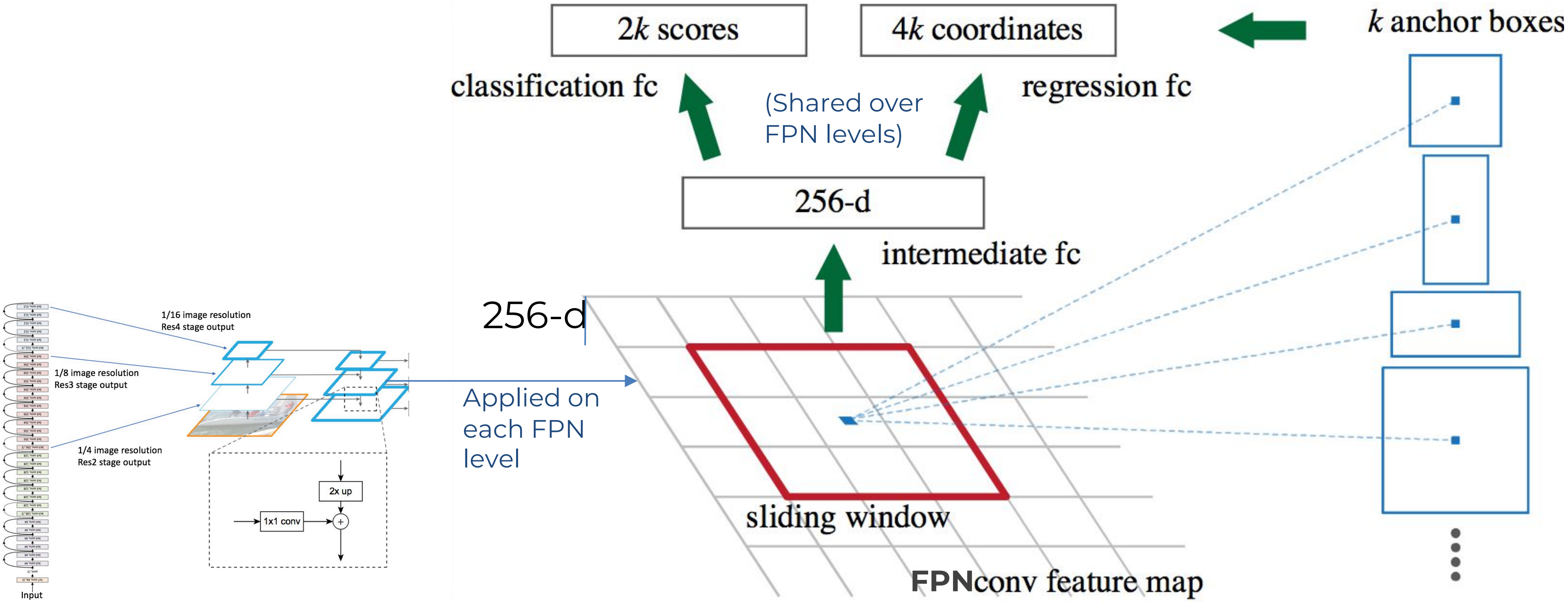
High resolution,
Strong features

Lin et al. Feature Pyramid Networks for Object Detection. In CVPR 2017.

R. Girshick

3. … + Region Proposal Network (RPN)

Proposals = sliding window object/not-object classifier
+ box regression *inside the same network*



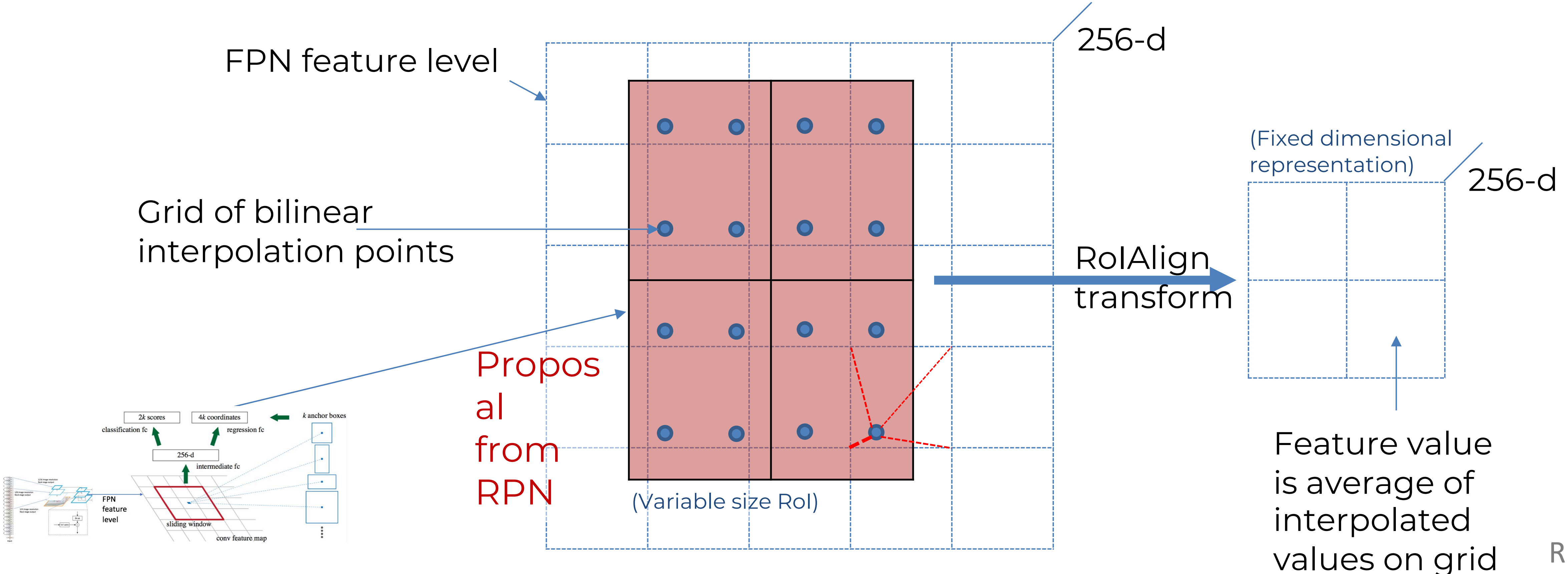
Anchors are
prototypical
object boxes

Ren et al. Faster R-CNN: Region Proposal Networks for Real-Time Object Detection. In NIPS 2015

R. Girshick

4. … + RoIAlign Transform (on each Proposal)

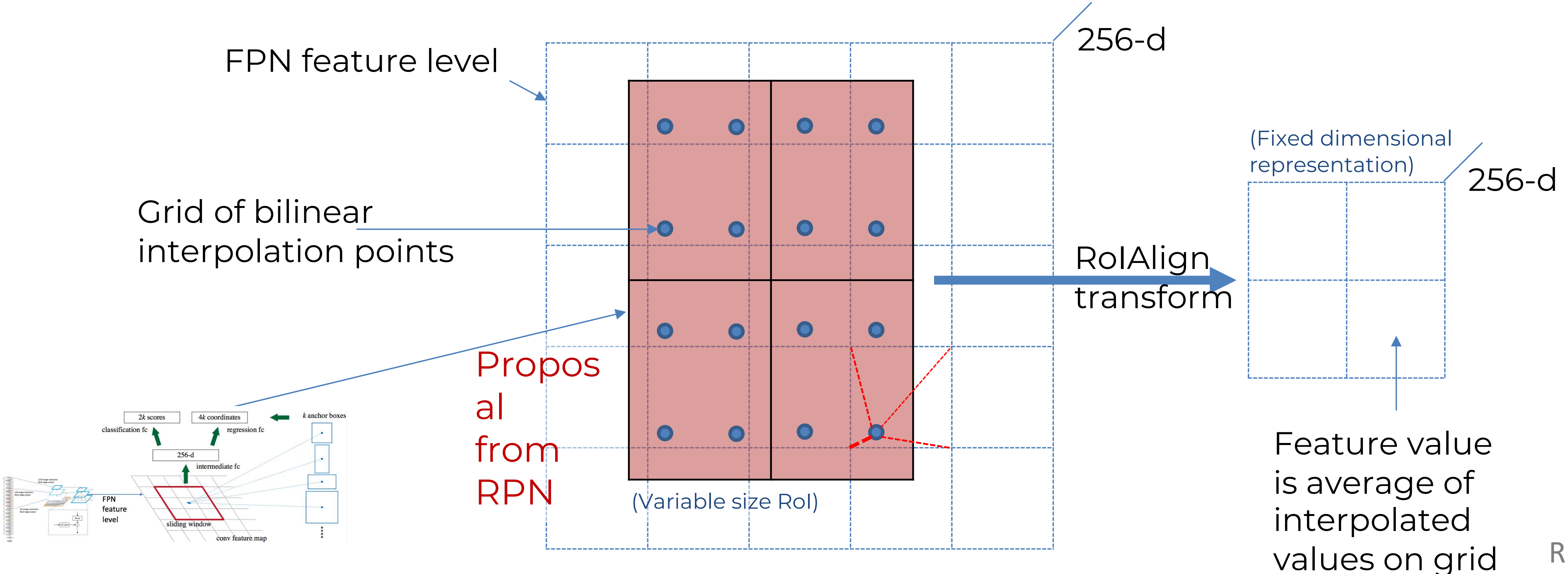
Smoothly normalize features and predictions into coordinate frame
free of scale and aspect ratio



R. Girshick

4. ... + RoIAlign Transform (on each Proposal)

Key: No coordinate quantization
(cf. RoIPool in Fast R-CNN, etc.)

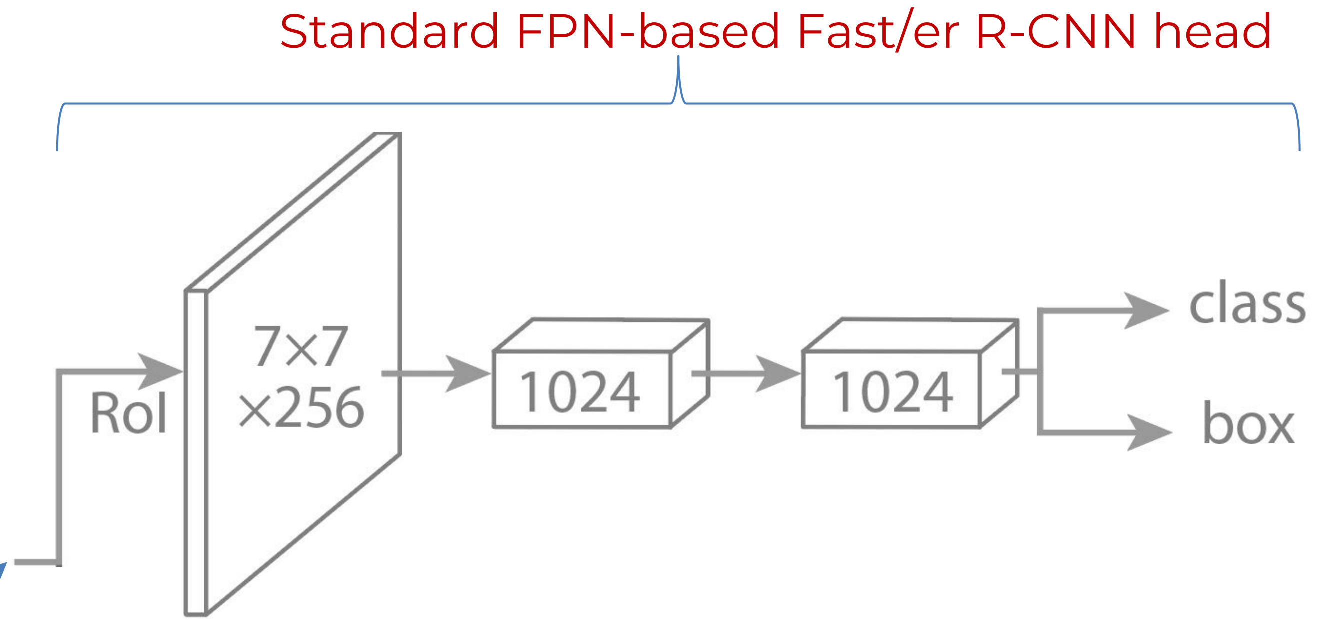


R. Girshick

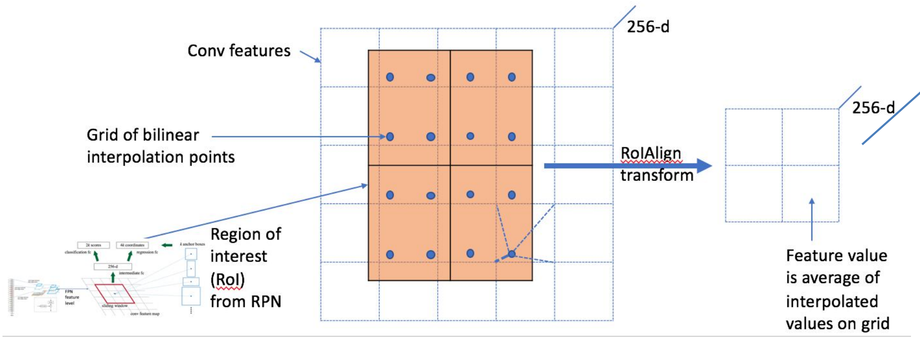
5. … + Task-specific Heads (on each Proposal)

Task specific heads for …

- Bounding box regression
- Object classification
- Instance mask prediction
- Human keypoint prediction



RoIAlign transformed features



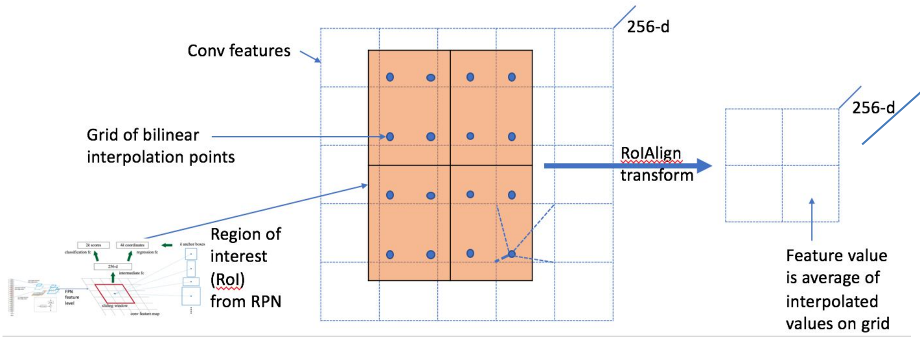
R. Girshick

5. ... + Task-specific Heads (on each Proposal)

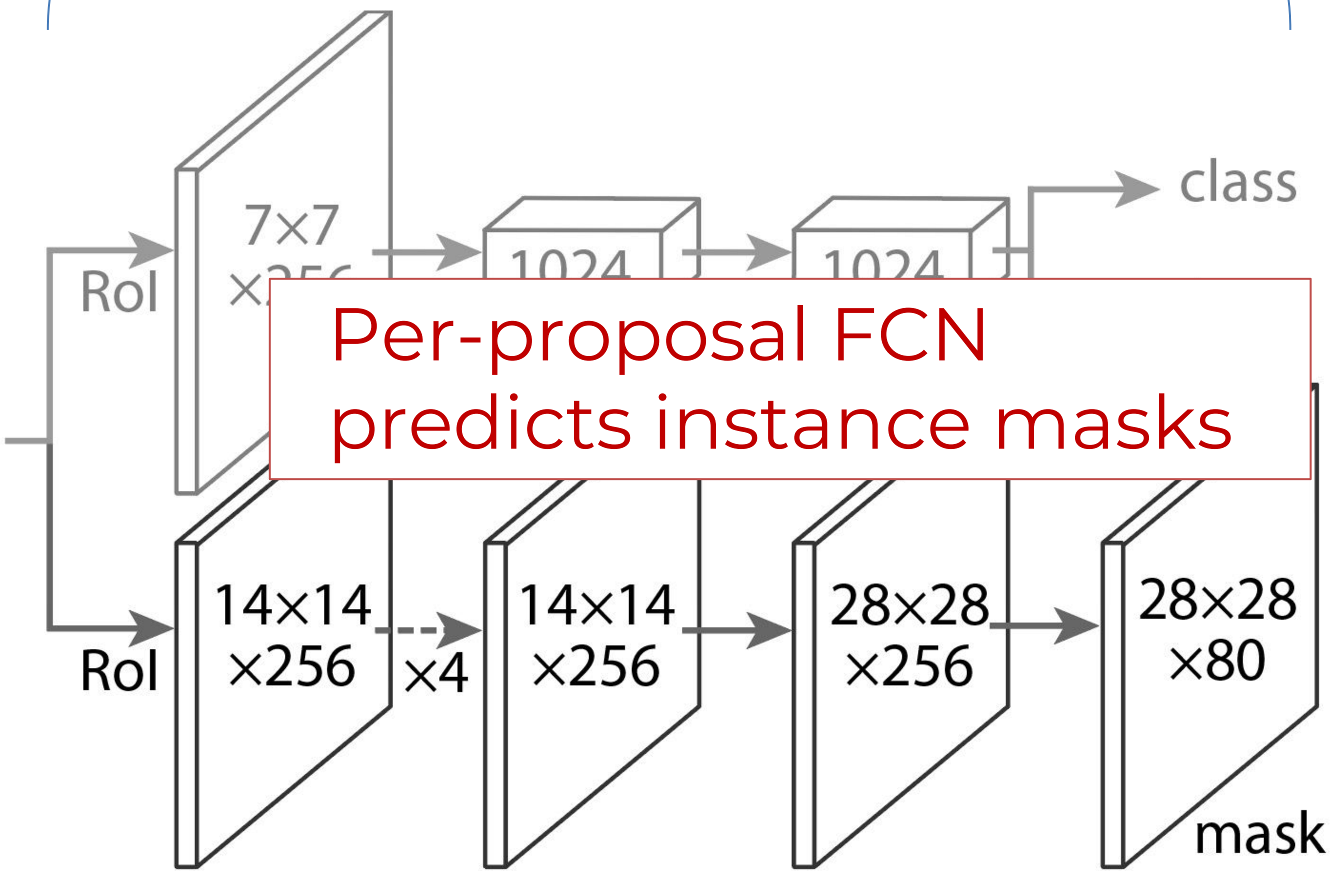
Task specific heads for ...

- Bounding box regression
- Object classification
- Instance mask prediction
- Human keypoint prediction

RoIAlign transformed features



Standard FPN-based Fast/er R-CNN head



Per-proposal FCN predicts instance masks

Mask R-CNN: Training

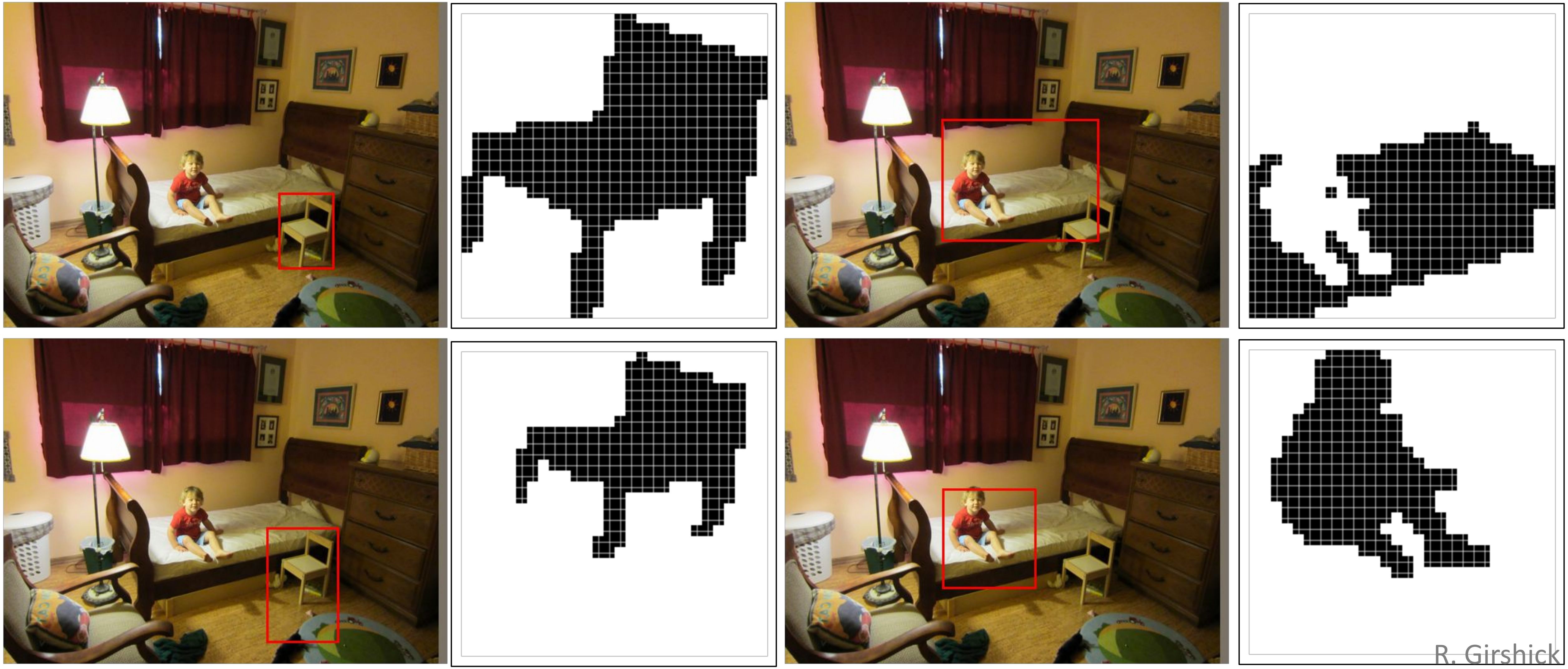
Same as “image centric” Fast/er R-CNN training

- Use precomputed proposals for faster experimentation
- Use joint / end-to-end training for sharing features

But with training targets for masks

Example Mask Training Targets

Image with training proposal | 28x28 mask target | Image with training proposal | 28x28 mask target



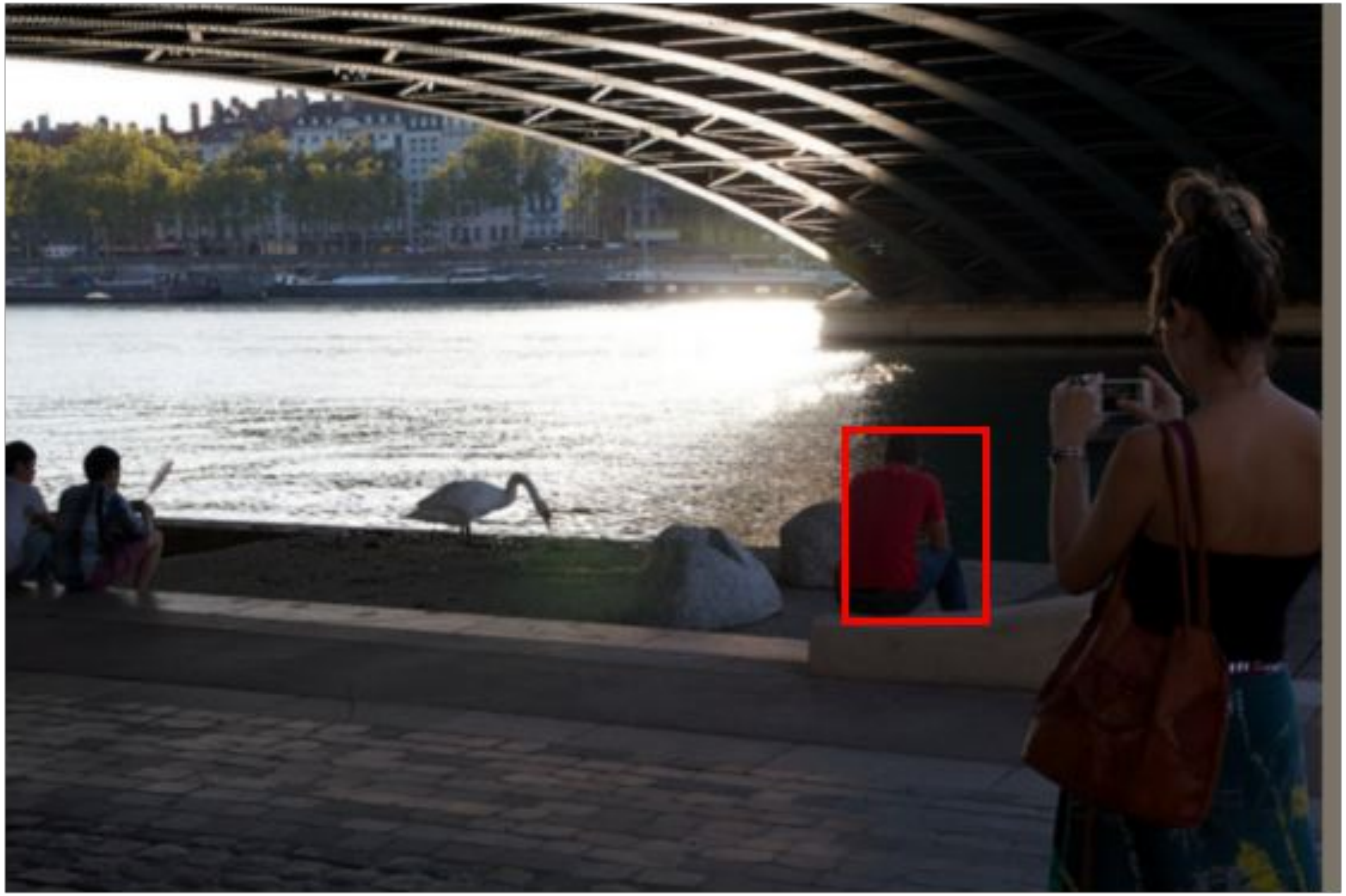
R. Girshick

Mask R-CNN: Inference

1. Perform Faster R-CNN inference
    - Generate proposals (RPN)
    - Score the proposals
    - Regress from proposals to refined detection boxes
    - Apply NMS and take the top *K* (= 100, e.g.)

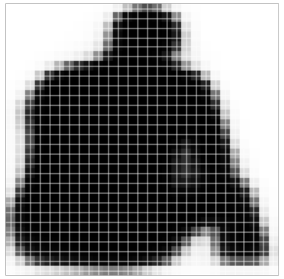
2. Run RoIAlign and mask head on top-*K* refined, post-NMS boxes
    - Fast (only compute masks for top-*K* detections)
    - Improves accuracy (uses *refined* detection boxes, not proposals)

Mask Prediction



Validation image with box detection shown in red

28x28 soft prediction from Mask R-CNN
(enlarged)



Soft prediction resampled to image coordinates
(bilinear and bicubic interpolation work equally well)

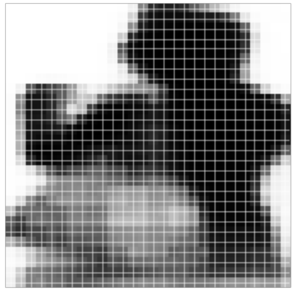
Final prediction (threshold at 0.5)

R. Girshick

Mask Prediction

Validation image with box detection shown in red

28x28 soft prediction



Resized soft prediction

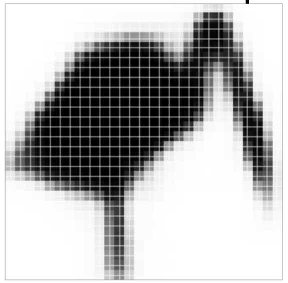
Final mask

R. Girshick

Mask Prediction

Validation image with box detection shown in red

28x28 soft prediction



Resized Soft prediction

Final mask

R. Girshick


person1.00
person1.00
person1.00
person.98
surfboard1.00
person1.00
person.91
surfboard1.00
surfboard.98
surfboard1.00
person.74
surfboard1.00

R. Girshick


person1.00
person.88
tv.98
tv.84
person1.00
person1.00
bottle.97
wine glass.99
dining table.95
wine glass1.00
wine glass1.00

R. Girshick

Segmentation Overview

- Semantic segmentation
  - Classify all pixels
  - Fully convolutional models, downsample then upsample
  - Learnable upsampling: fractionally strided convolution
  - Skip connections can help

- Instance Segmentation
  - Detect instance, generate mask
  - Similar pipelines to object detection