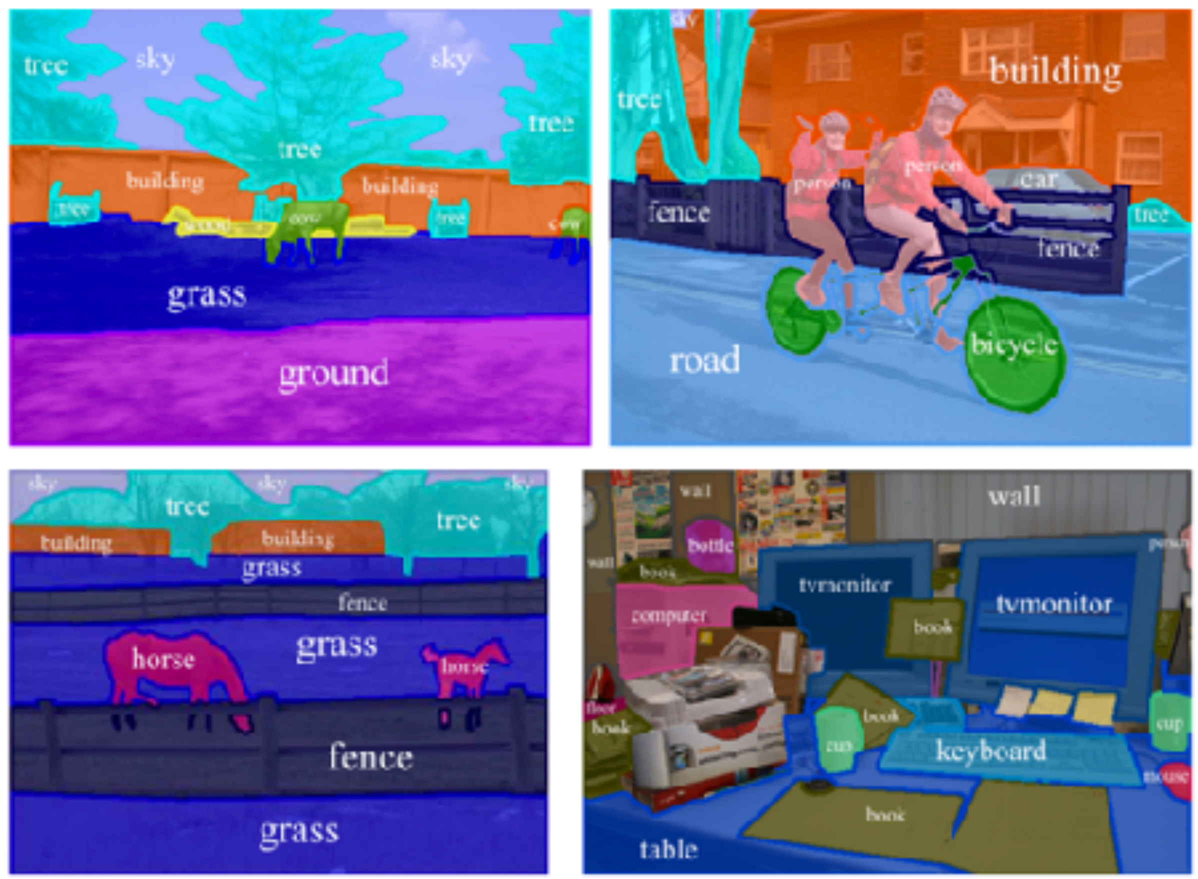
Semantic Segmentation



UCLA:https://goo.gl/images/l0VTi2

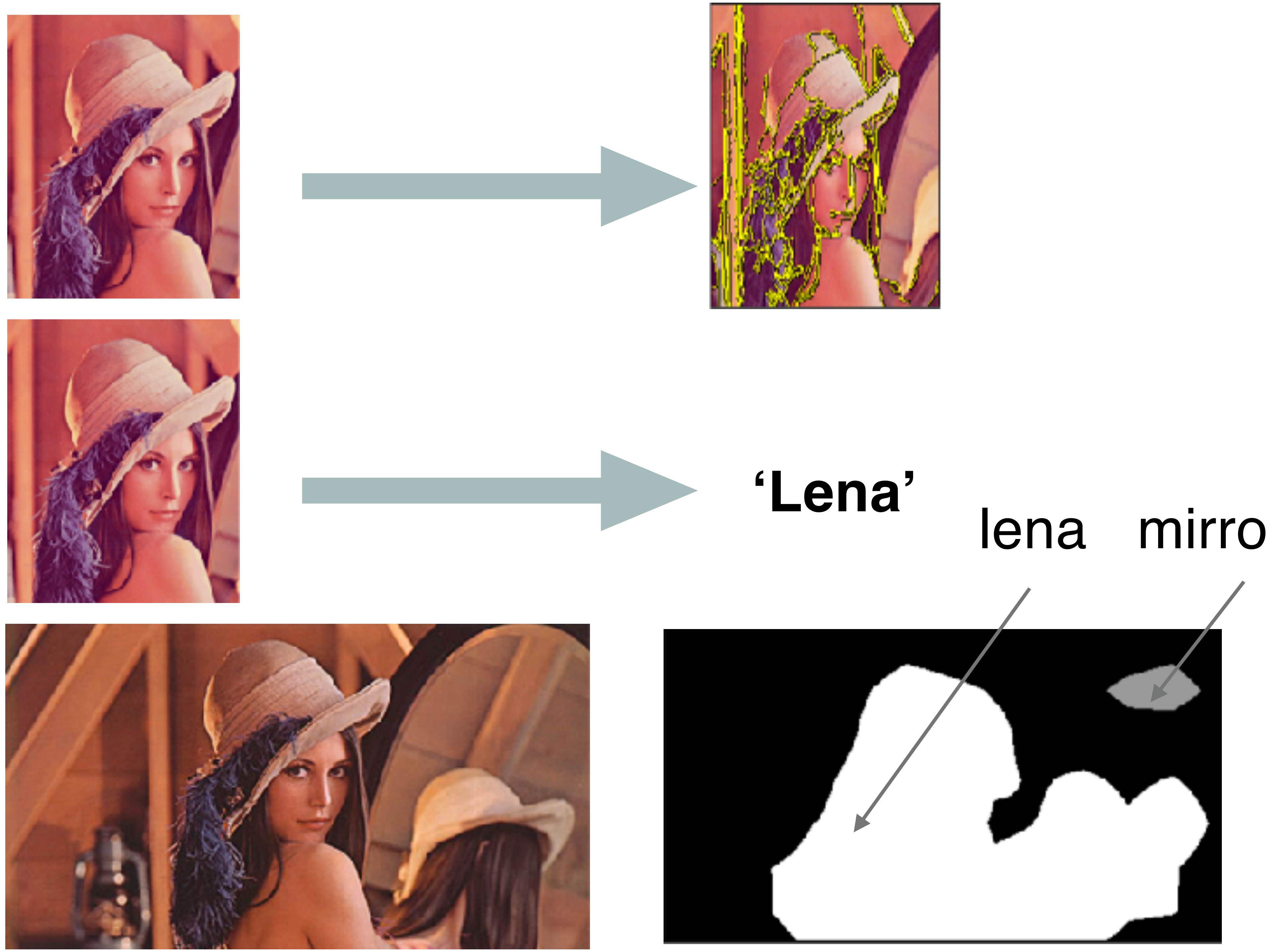
OUTLINE

- **Semantic Segmentation**
- **Why?**
- **Paper to talk about:**

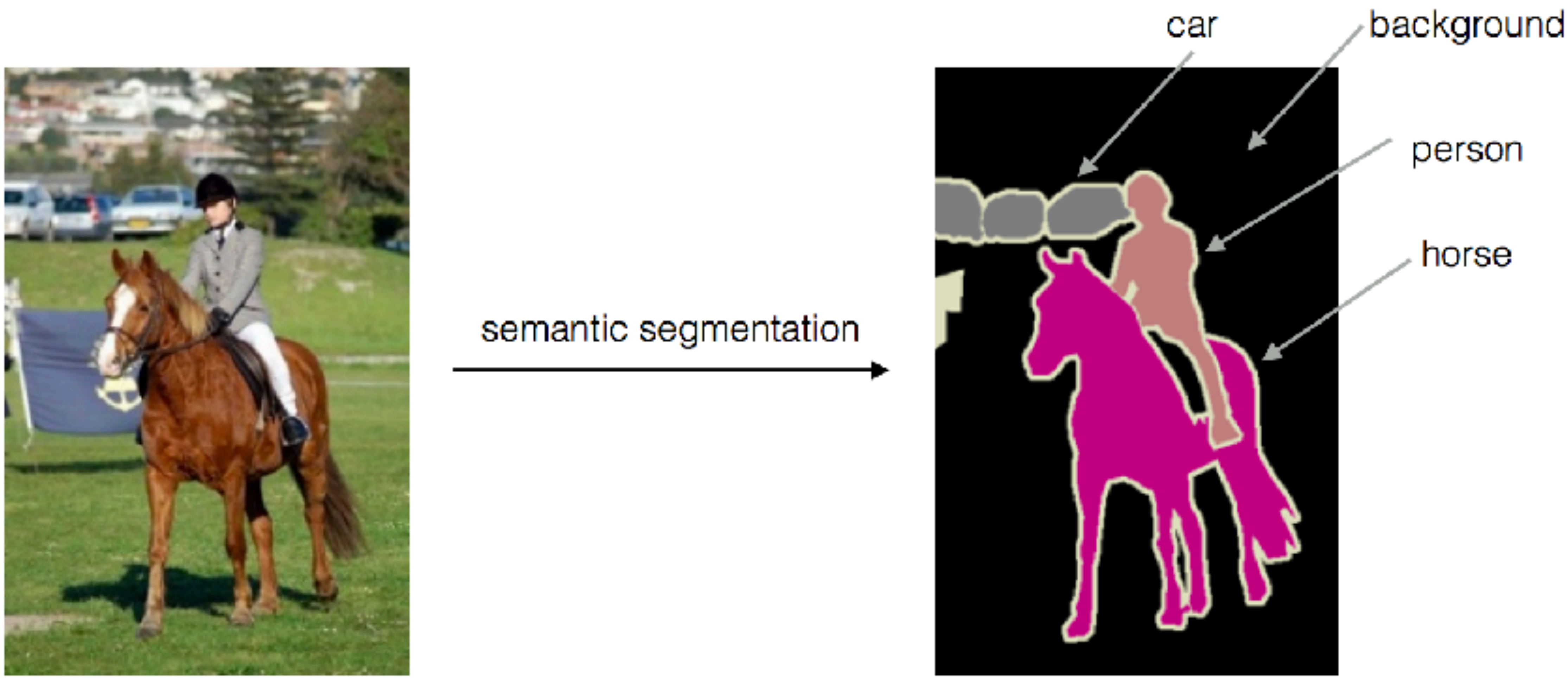
Fully Convolutional Networks for Semantic Segmentation. J. Long, E. Shelhamer, and T. Darrell, CVPR 2015

Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille. ICLR 2015

What is Semantic Segmentation



What is Semantic Segmentation



Goal:

Partition the image into semantically meaningful parts, and classify each part ——>Patch-wise

Recognizing and delineating objects in an image

Classifying each pixel in the image ——>Pixel-wise

Why Semantic Segmentation?

- To let robots segment objects so that they can grasp them

https://goo.gl/images/6xAQAM

Why Semantic Segmentation?

- **Useful tool for editing images, visual effects**

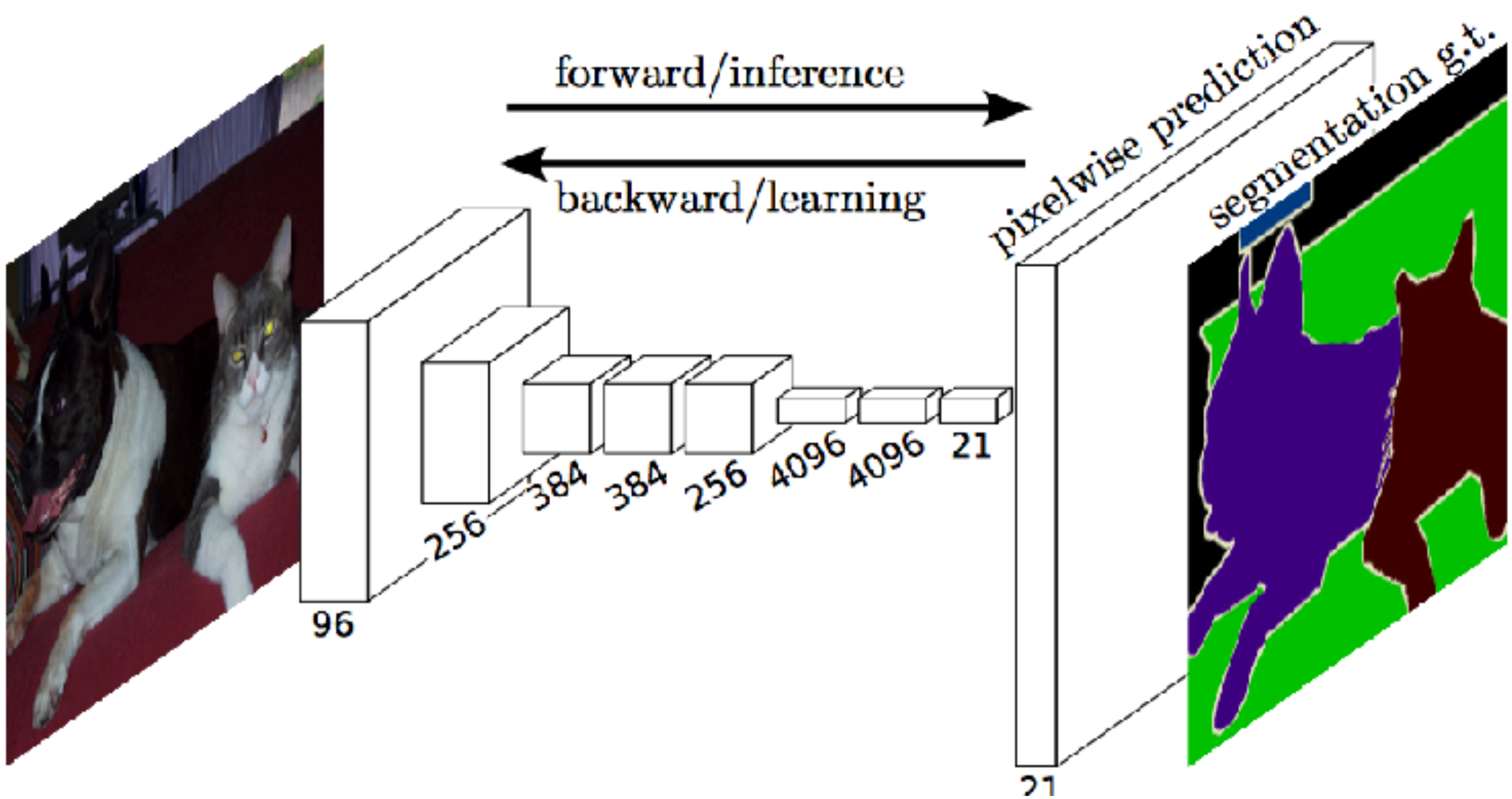
CVFX Lecture1: https://www.youtube.com/watch?v=rE-hVtytT-I

Why Semantic Segmentation?

- **Autonomous Driving, to differentiate pedestrian and background**

Citydataset

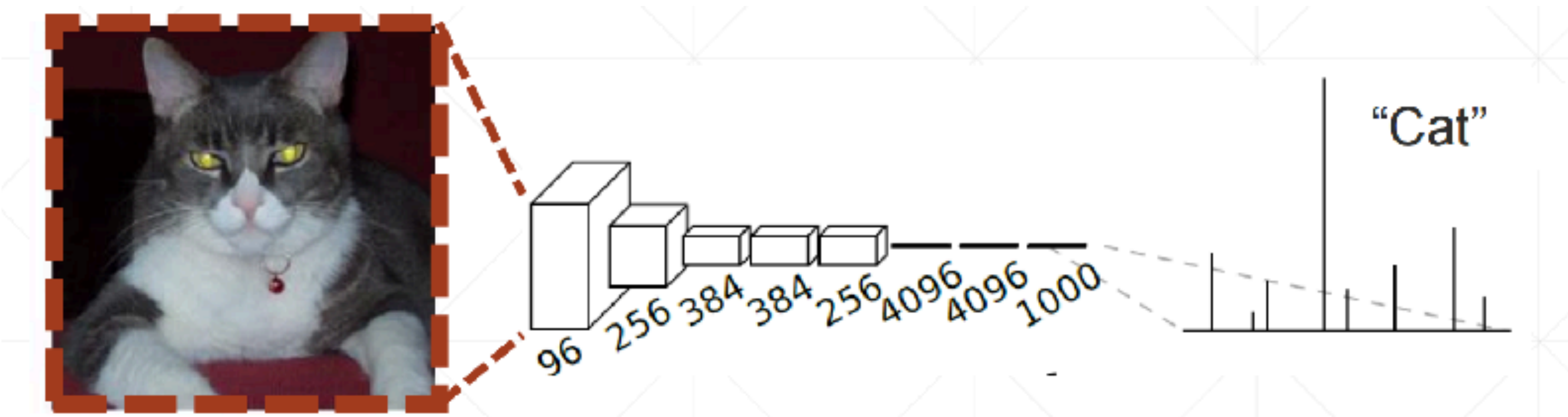
Fully Convolutional Networks for Semantic Segmentation. J. Long, E. Shelhamer, and T. Darrell, CVPR 2015



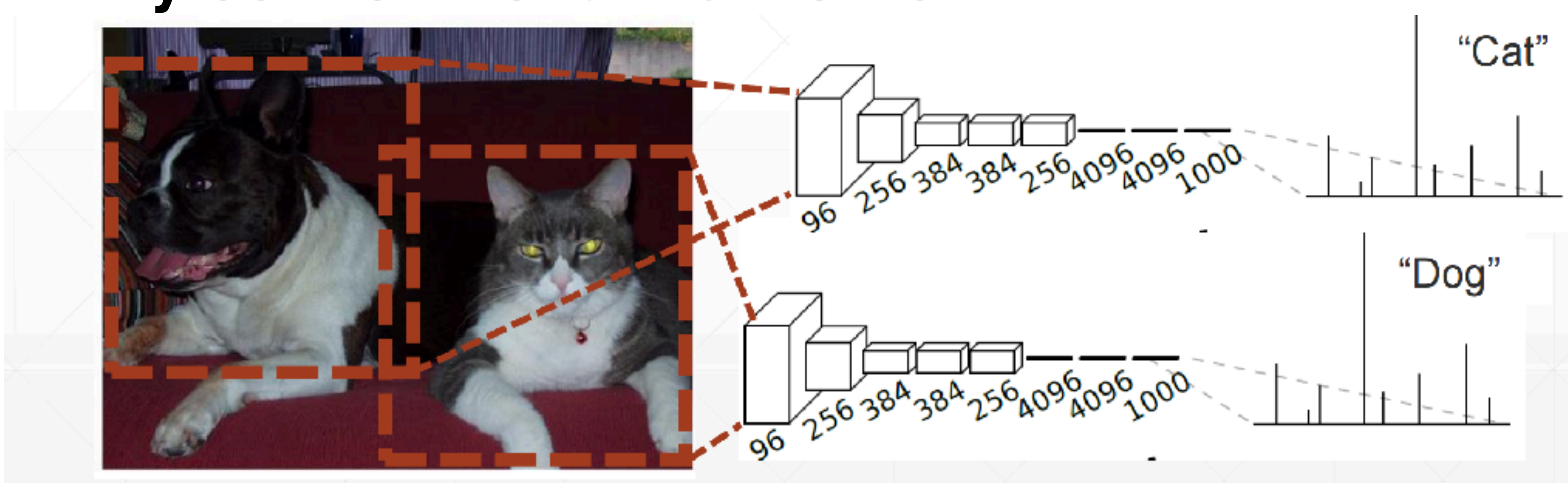
ECE 6554: Topic Presentation
Xiaolong Li

Fully Convolutional Networks for Semantic Segmentation.
J. Long, E. Shelhamer, and T. Darrell, CVPR 2015

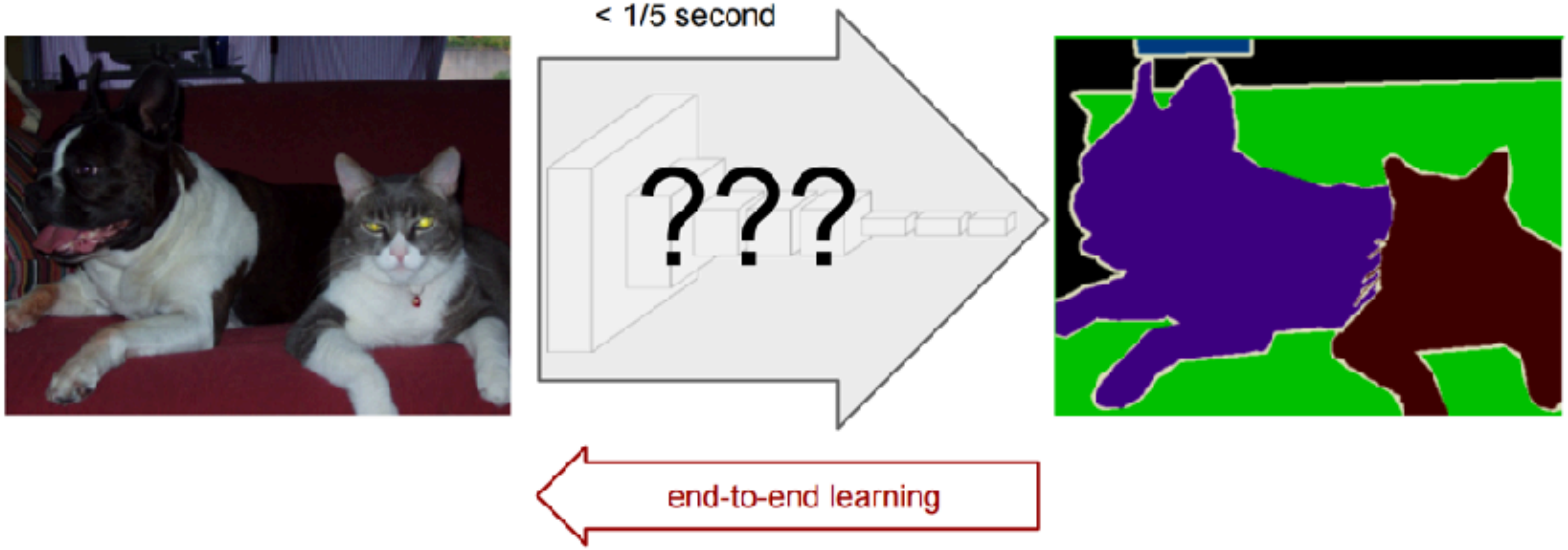
Usual convolutional networks



Fully convolutional networks



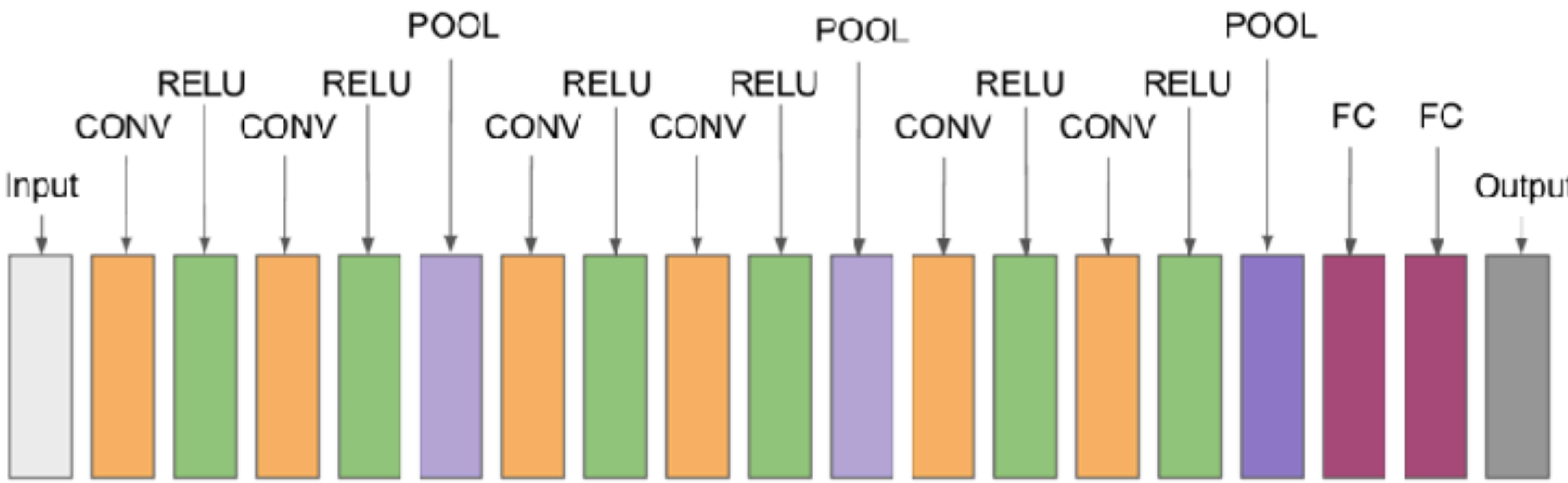
To understand “Fully Convolutional”



2015, Berkeley Vision: http://tutorial.caffe.berkeleyvision.org/

To understand “Fully Convolutional”

A typical CNN

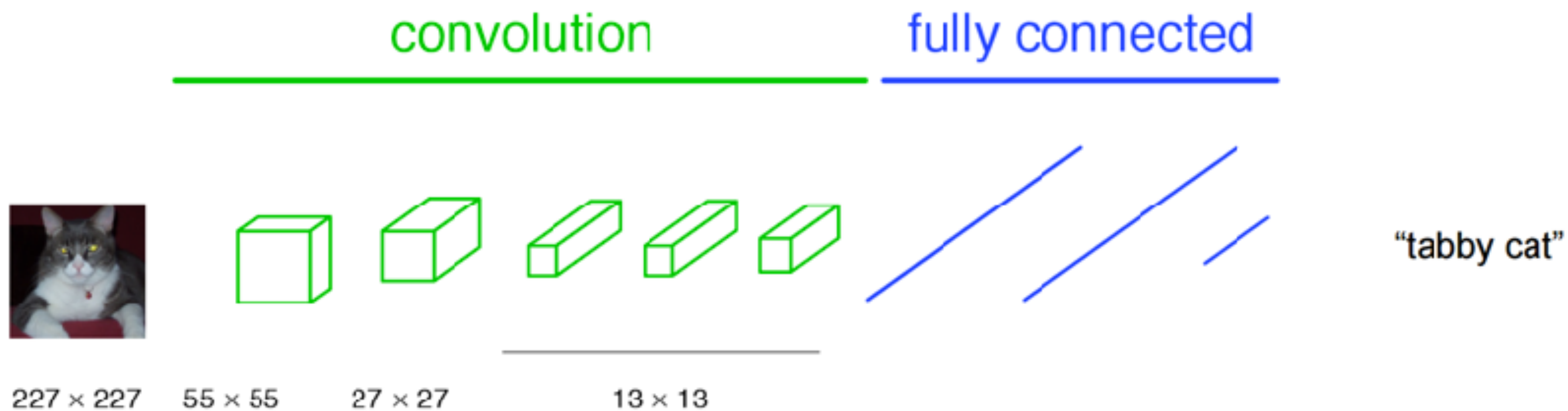


Example of a CNN Architecture

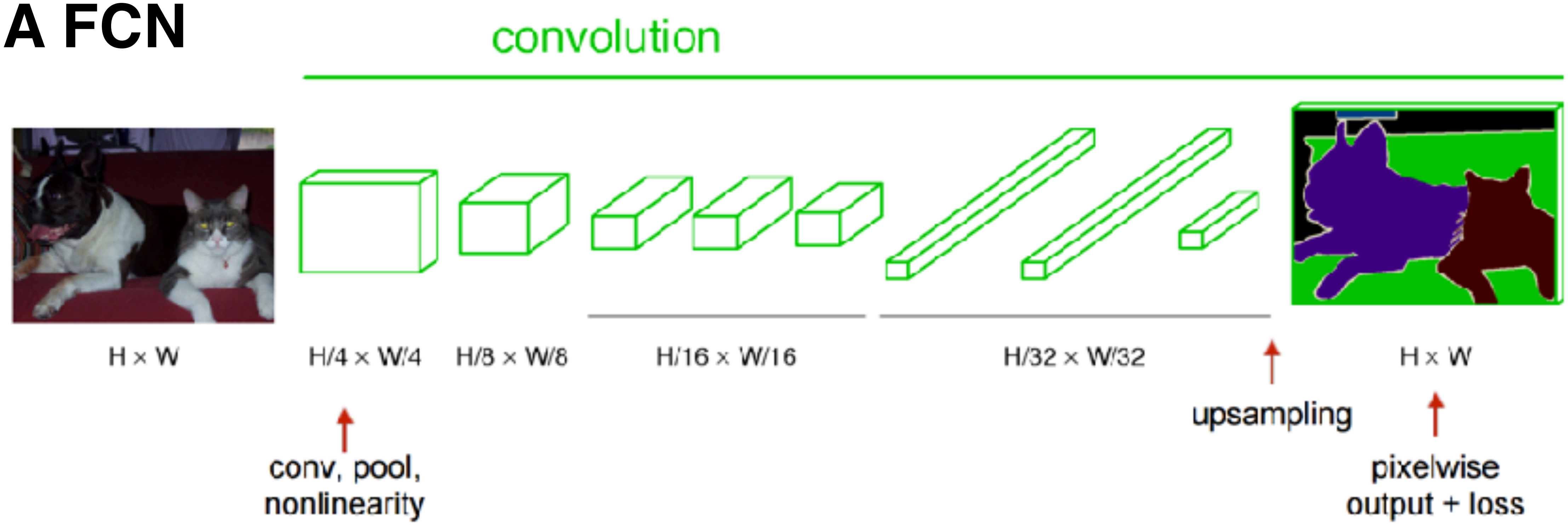
2015, Berkeley Vision: http://tutorial.caffe.berkeleyvision.org/

To understand “Fully Convolutional”

A classification CNN



A FCN



2015, Berkeley Vision: http://tutorial.caffe.berkeleyvision.org/

FCN:

segmentation that combines layers of hierarchy and refines the spatial precision of the output.

Segmentation Architecture

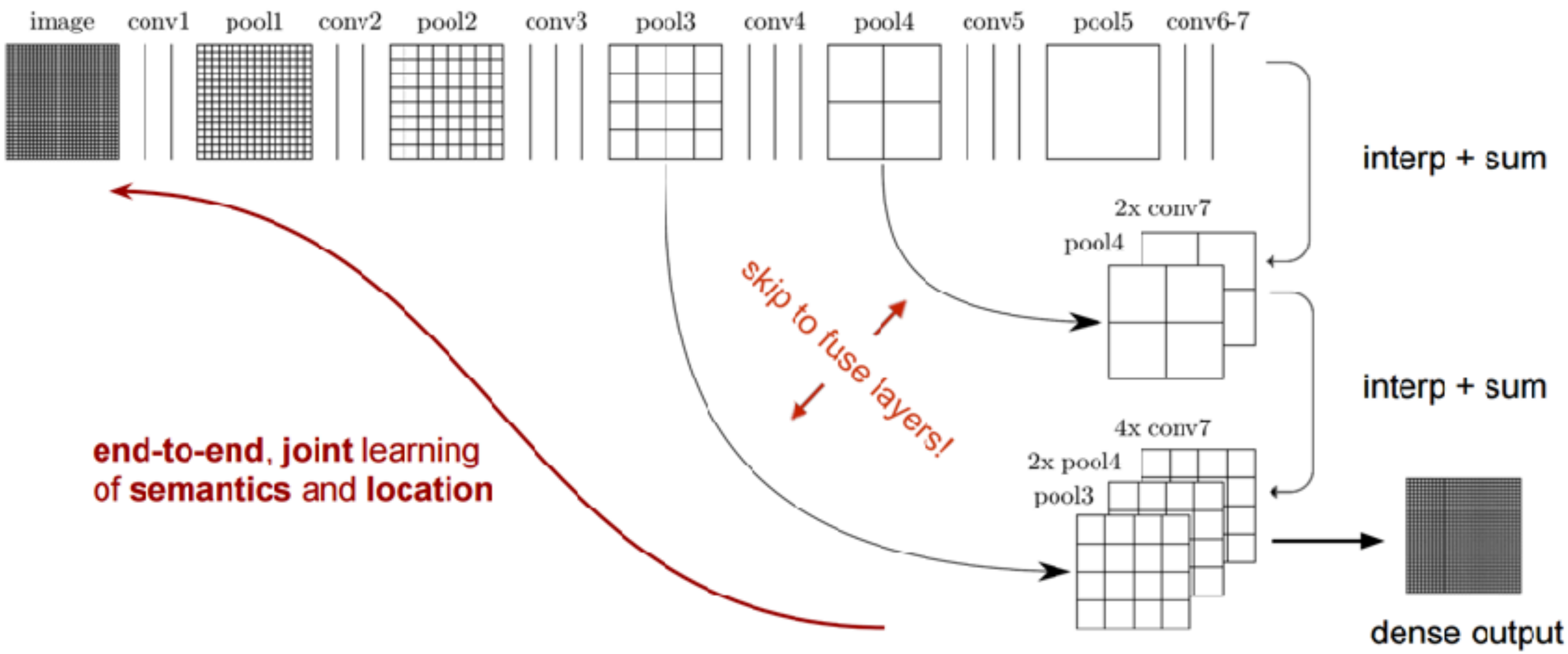
1. ILSVRC classifiers, in-network up sampling and a pixel-wise loss.

2. Add skips between layers to fuse coarse, semantic and local, appearance

3. Dense predictions, pixel-wise prediction

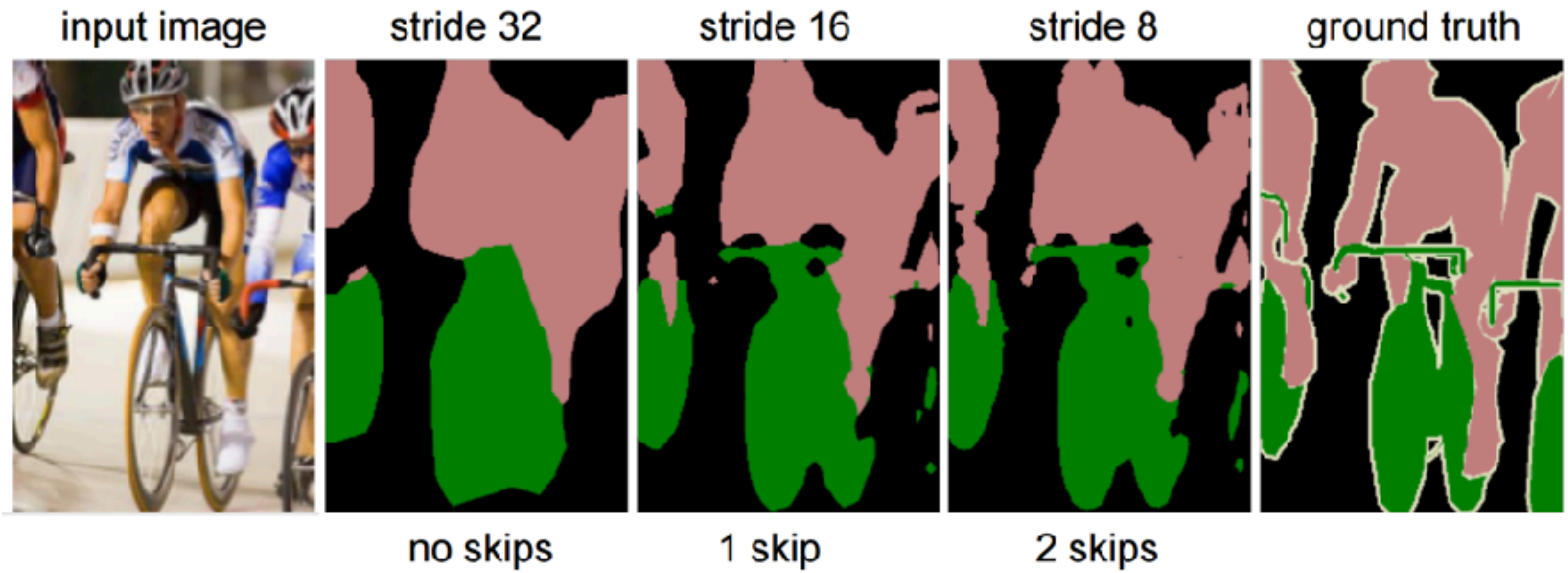
Some Tricks

skip layers



Some Tricks

skip layers refinement



Some Tricks

Interpolation

1. Up-sampling is performed in-network for end-to-end learning by back-propagation from the pixel wise loss.
2. The deconvolution filter in such a layer can be learned.

Some results:

PASCAL VOC

	mean IU VOC2011 test	mean IU VOC2012 test	inference time
R-CNN [12]	47.9	-	-
SDS [17]	52.6	51.6	$\sim$ 50 s
FCN-8s	**62.7**	**62.2**	$\sim$ **175 ms**

NYUDv2

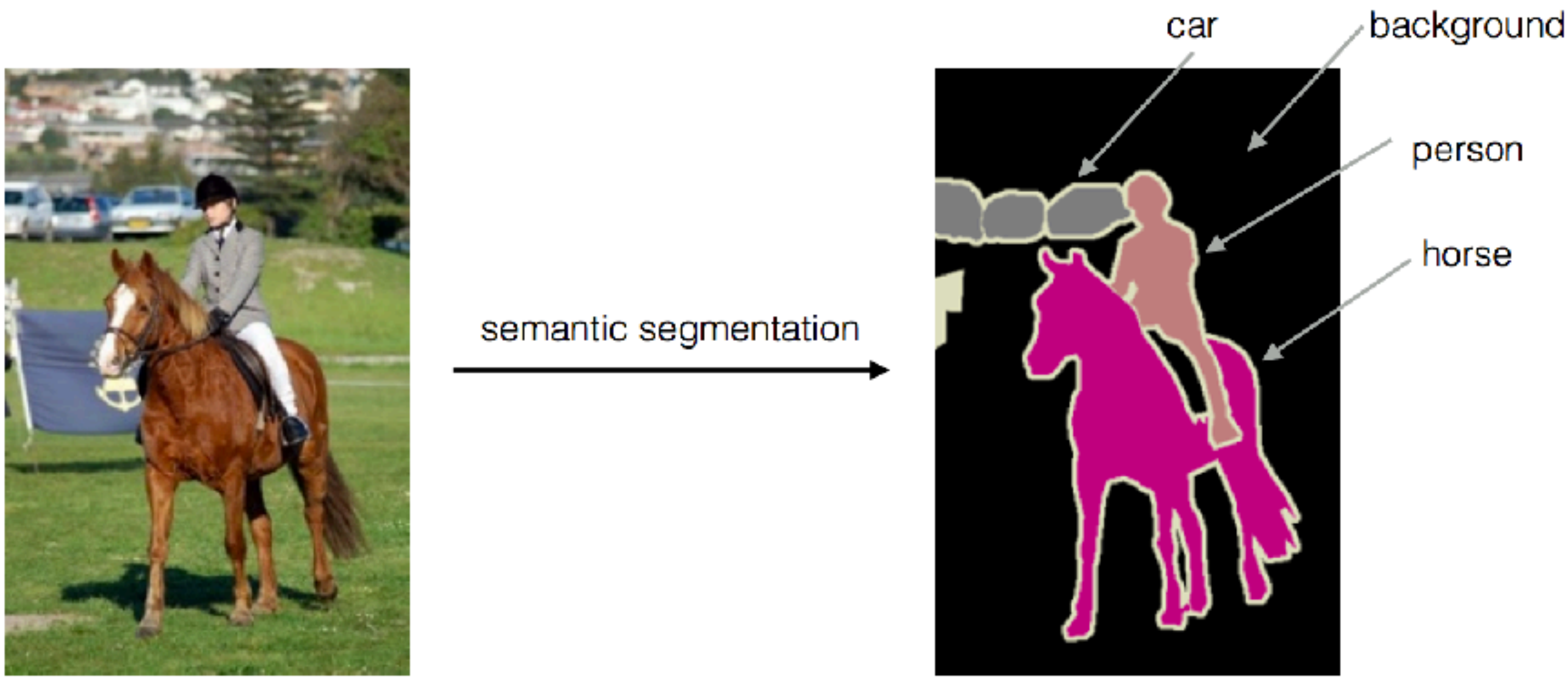
	pixel acc.	mean acc.	mean IU	f.w. IU
Gupta *et al.* [15]	60.3	-	28.6	47.0
FCN-32s RGB	60.0	42.2	29.2	43.9
FCN-32s RGBD	61.5	42.4	30.5	45.5
FCN-32s HHA	57.1	35.2	24.2	40.4
FCN-32s RGB-HHA	64.3	44.9	32.8	48.0
FCN-16s RGB-HHA	**65.4**	**46.1**	**34.0**	**49.5**

Conclusion

1. Fine-tuning from classification to segmentation gives reasonable predictions for each net.

2. Learning through up-sampling combined with the skip layer fusion to be more effective and efficient

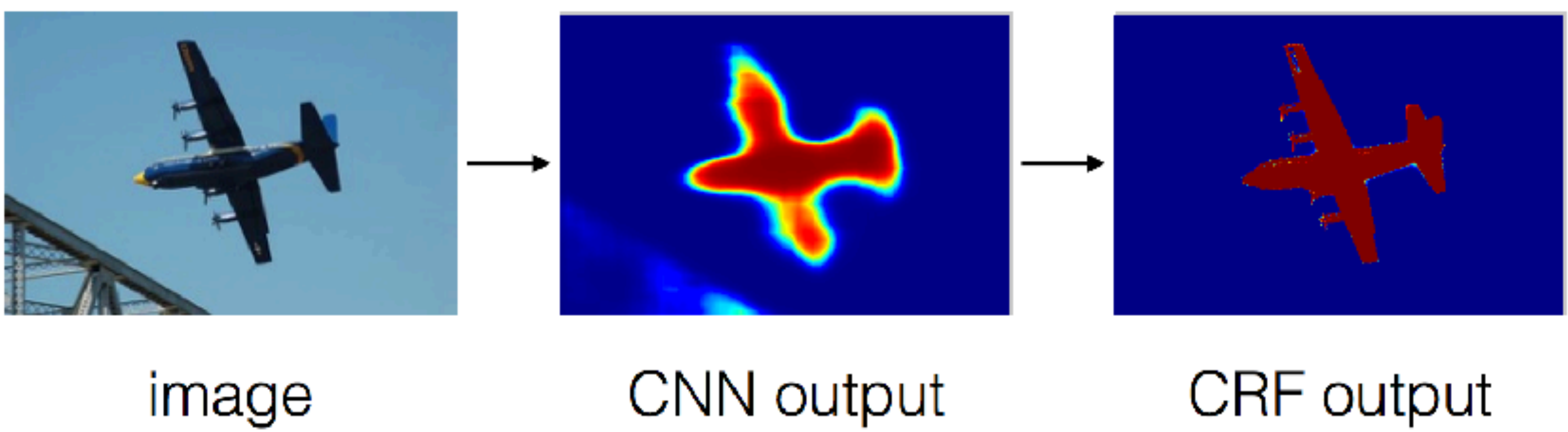
Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille. ICLR 2015



Paper’s main idea

1. Use CNN to generate a rough prediction of segmentation (smooth, blurry heat map)

2. Refine this prediction with a conditional random field (CRF)



Why are CNNs insufficient?

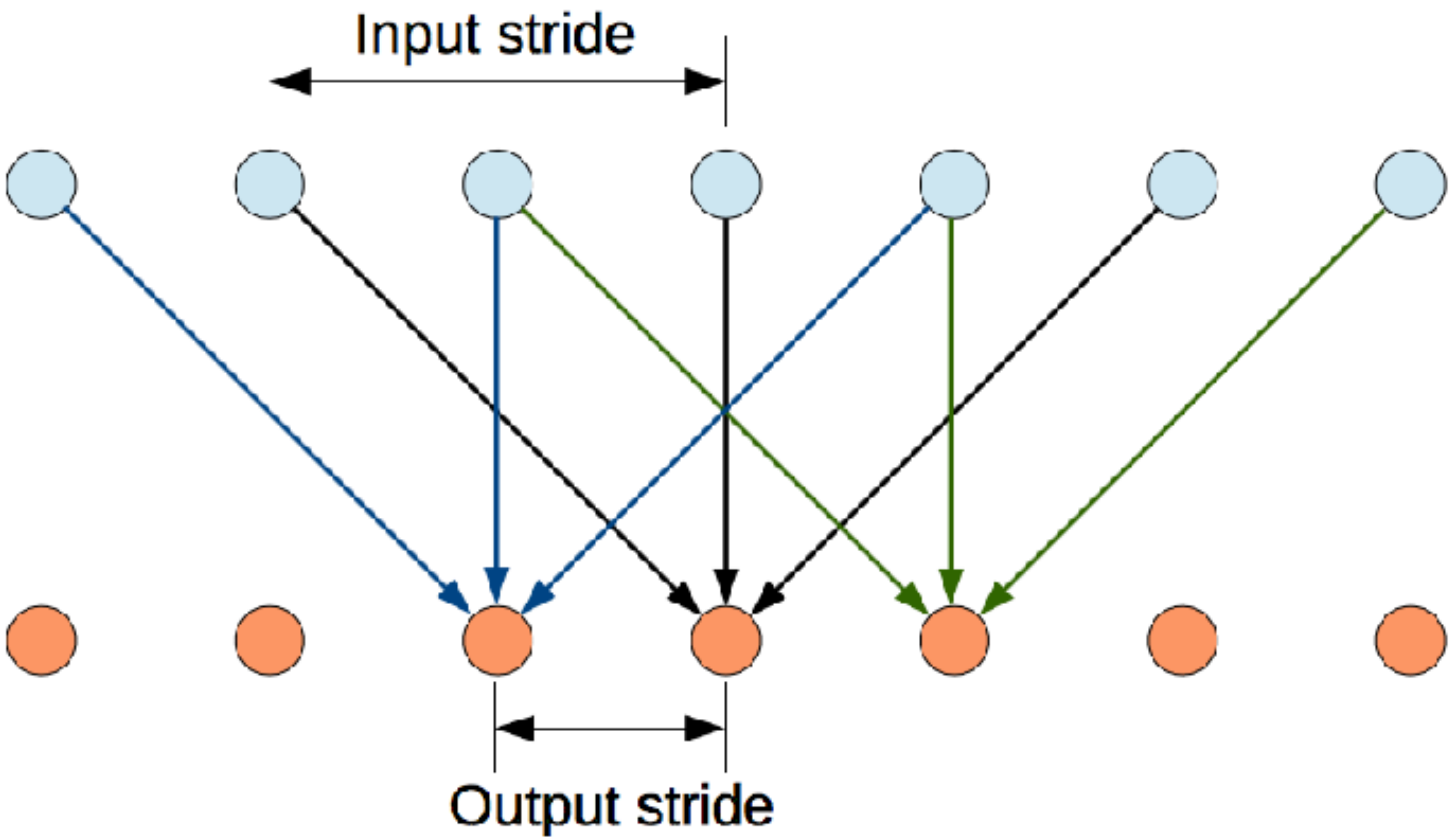
Good for high-level vision tasks like classification, bad for low level tasks like segmentation.

- **Problem: subsampling**
- **Problem: spatial invariance (shared kernel weights)**

Solution: fully connected CRF

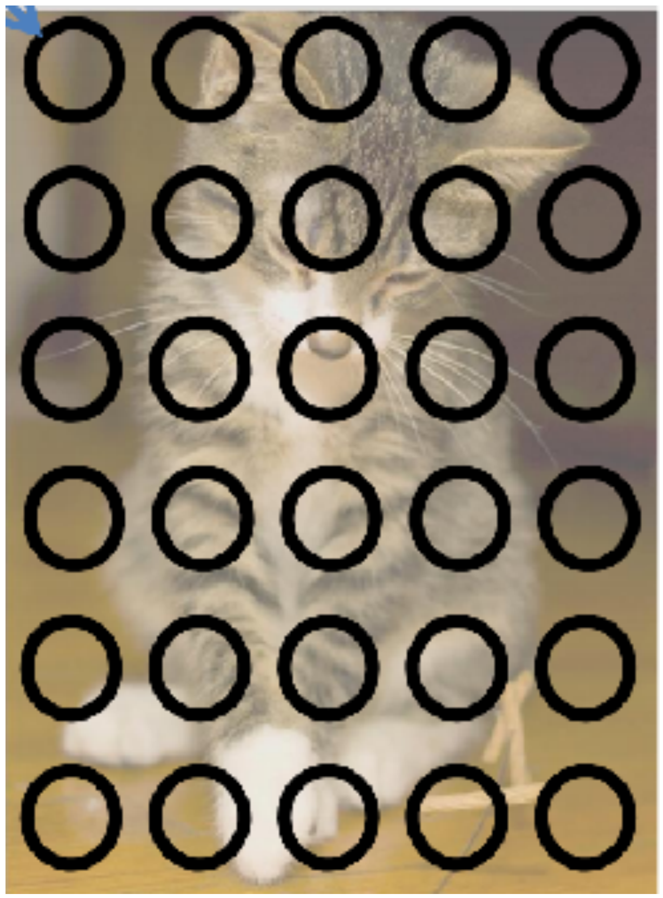
Solution: fully connected CRF

Holes' algorithms



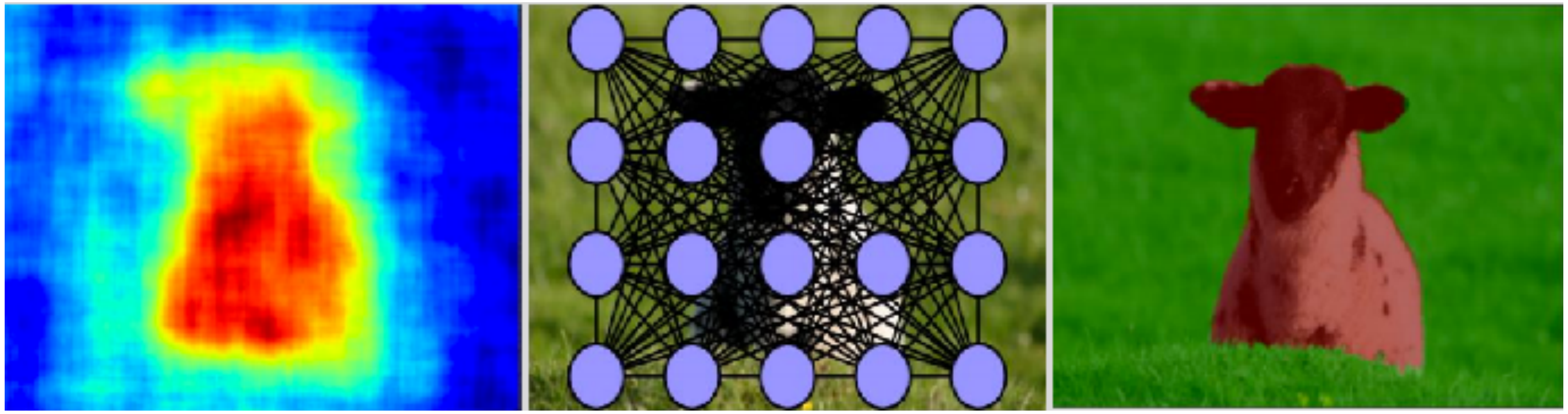
Solution: fully connected CRF

CRF



Randomly choose points and give initial label

CRF Energy Function



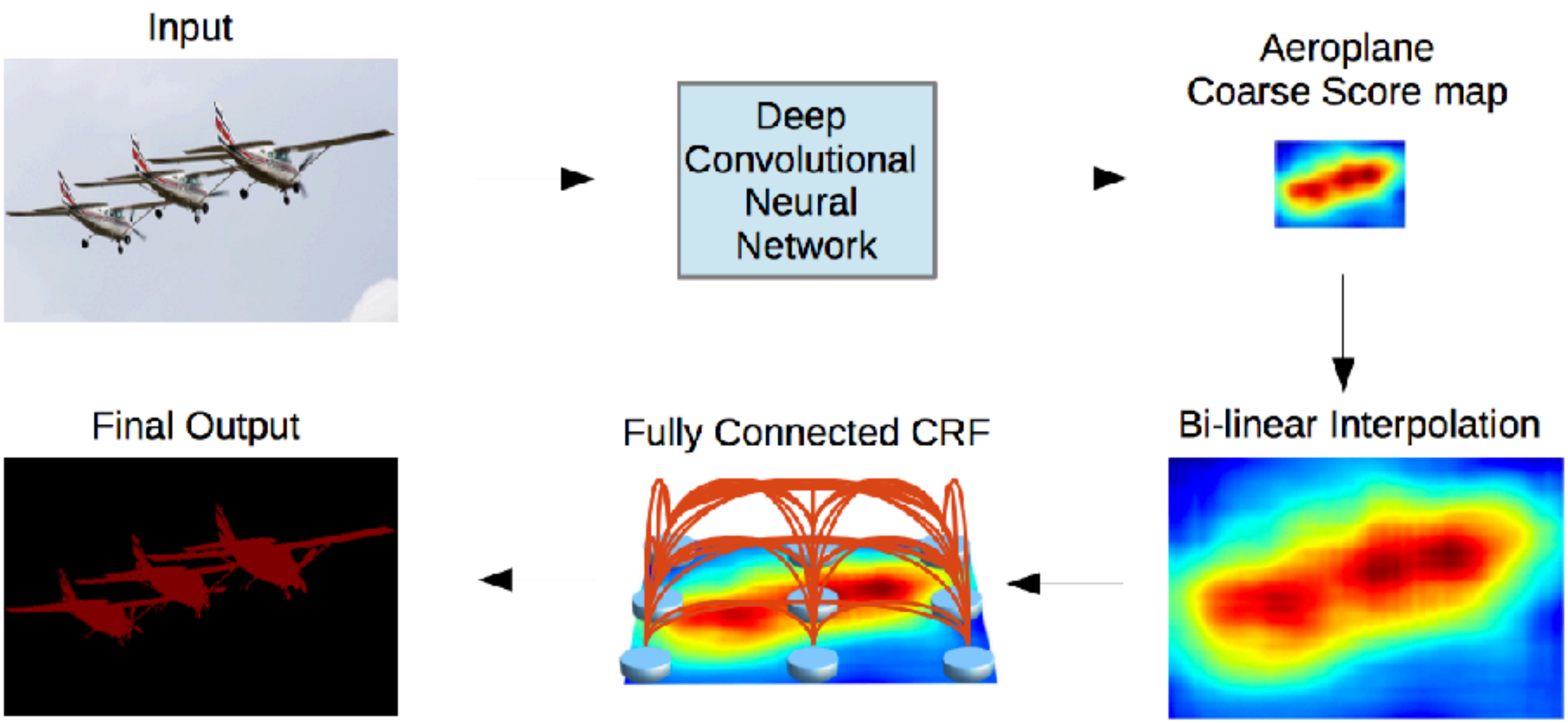
$$E(\boldsymbol{x}) = \sum_i \theta_i(x_i) + \sum_{ij} \theta_{ij}(x_i, x_j)$$

where $\boldsymbol{x_i}$ is assignment of pixel i

$$\theta_i(x_i) = -\log P(x_i)$$

$$P(x_i) = \text{ label assignment probability computed by CNN}$$

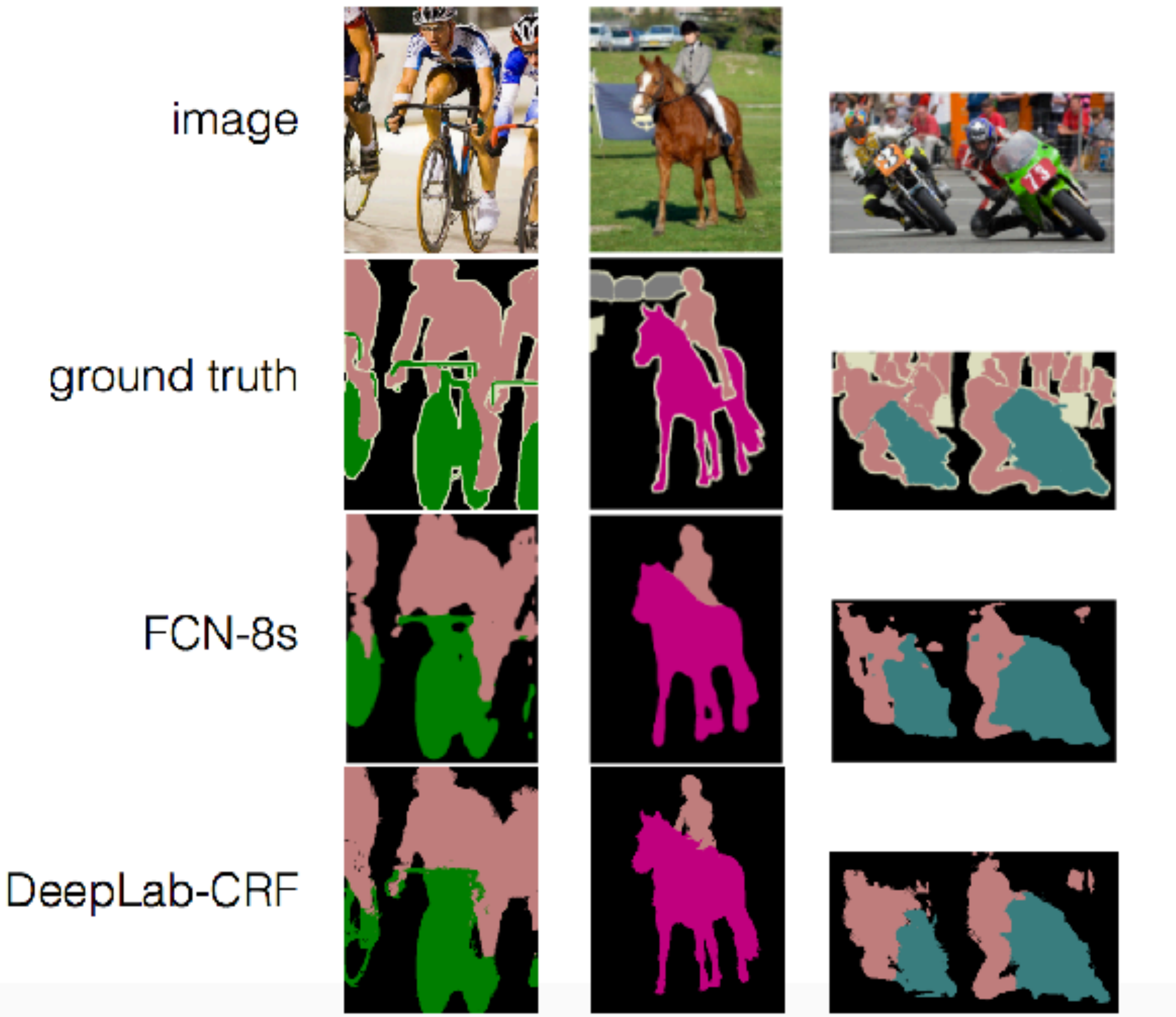
Global Map


Input
Deep
Convolutional
Neural
Network
Aeroplane
Coarse Score map
Bi-linear Interpolation
Fully Connected CRF
Final Output

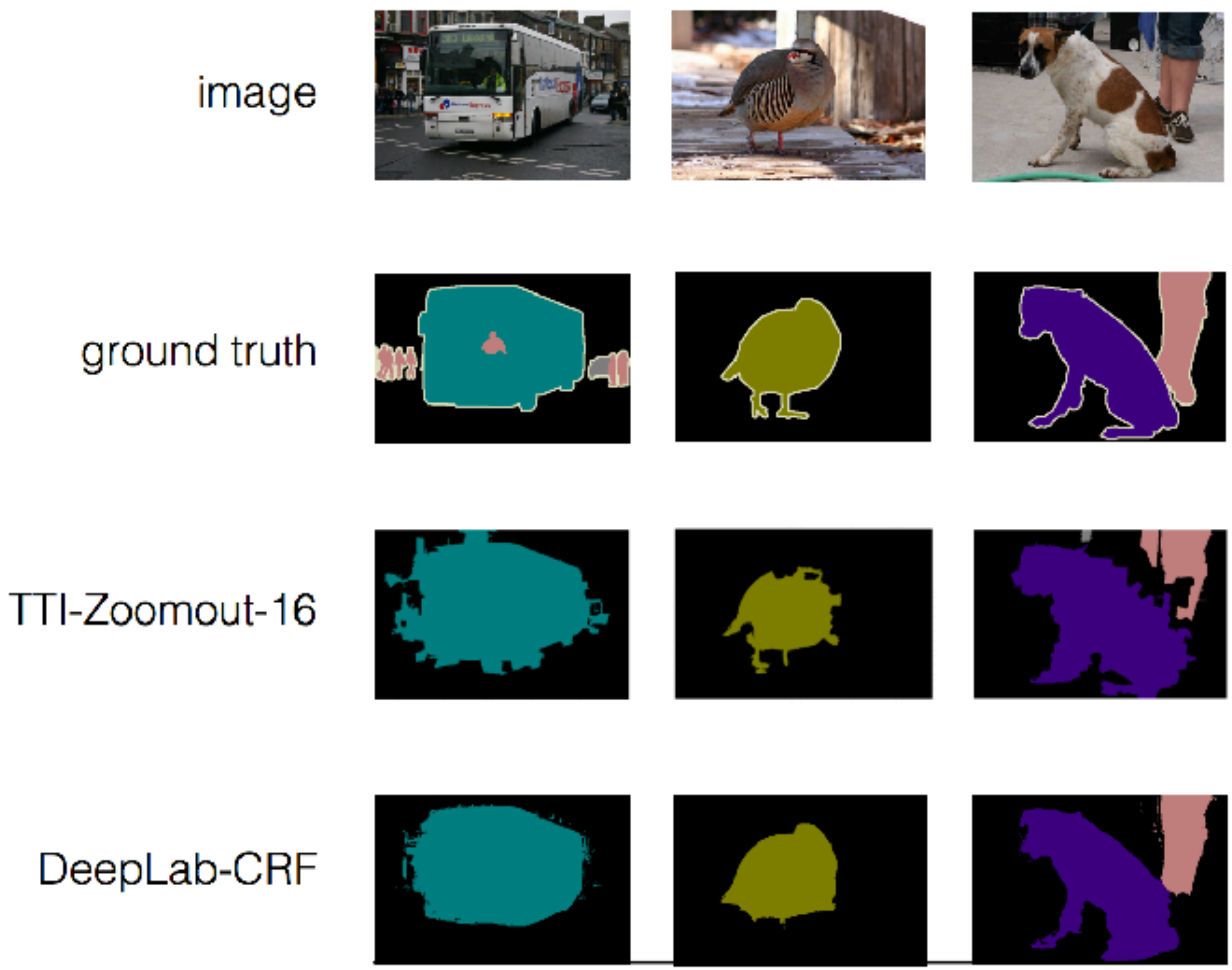
Comparison to state-of-the-art

Method	mean IOU (%)
MSRA-CFM	61.8
FCN-8s	62.2
TTI-Zoomout-16	64.4
DeepLab-CRF	66.4
DeepLab-MSc-CRF	67.1
DeepLab-MSc-CRF-LargeFOV	71.6

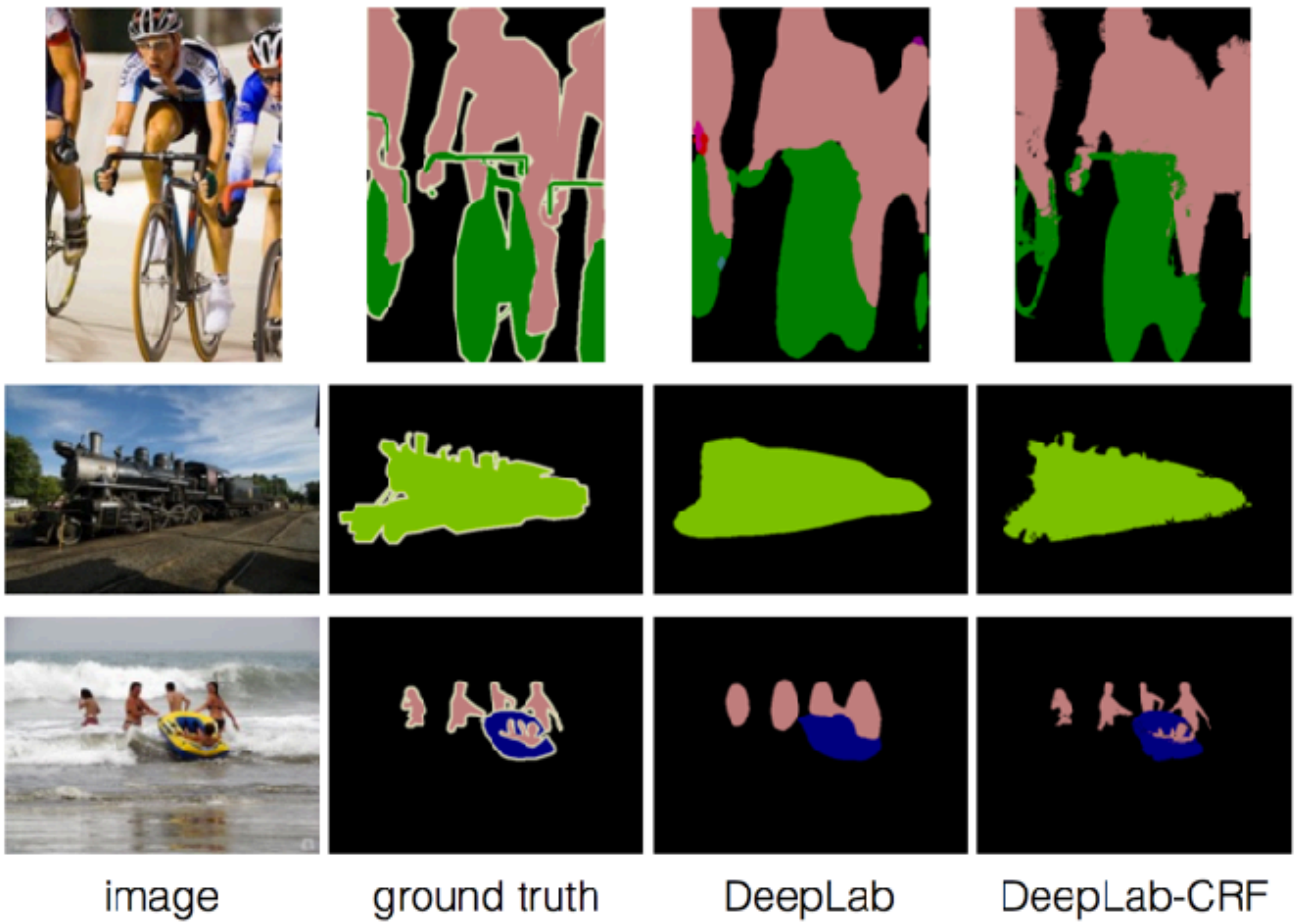
Comparison to state-of-the-art



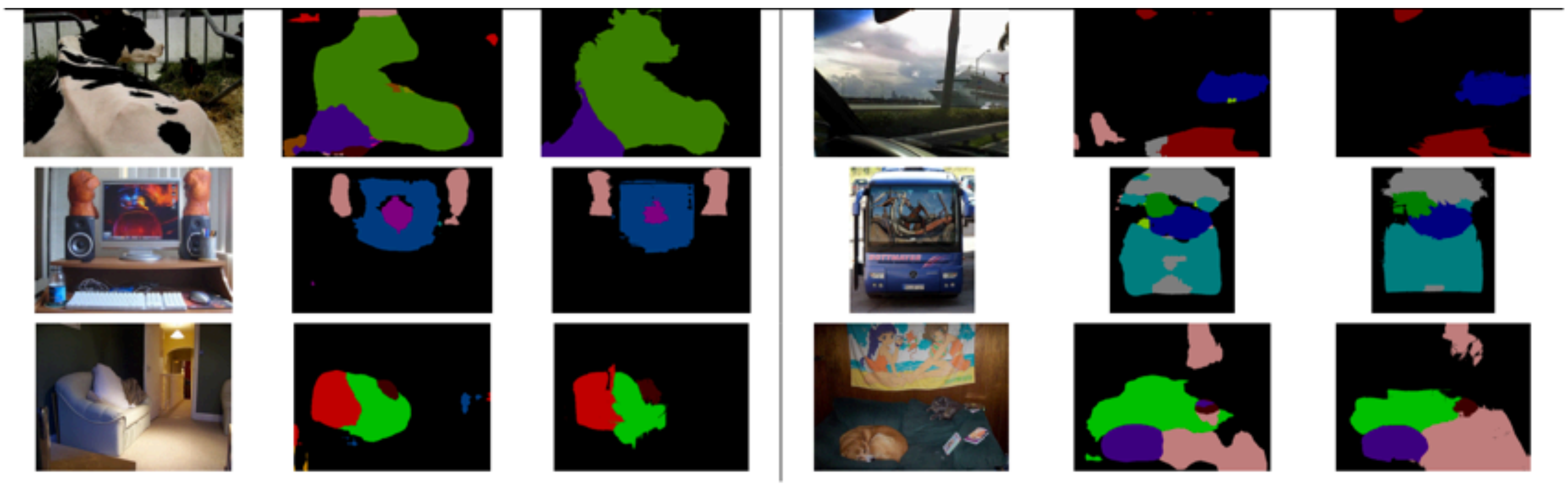
Comparison to state-of-the-art



Successful Cases



Failure Cases



Conclusion

- Modify the CNN architecture to become less spatially invariant.

- Use the CNN to compute a rough score map.

- Use a fully connected CRF to sharpen the score

Experiments

Intel Xeon E5-2670

NVIDIA GPU

Caffe

VOC_FCN_32s

Python

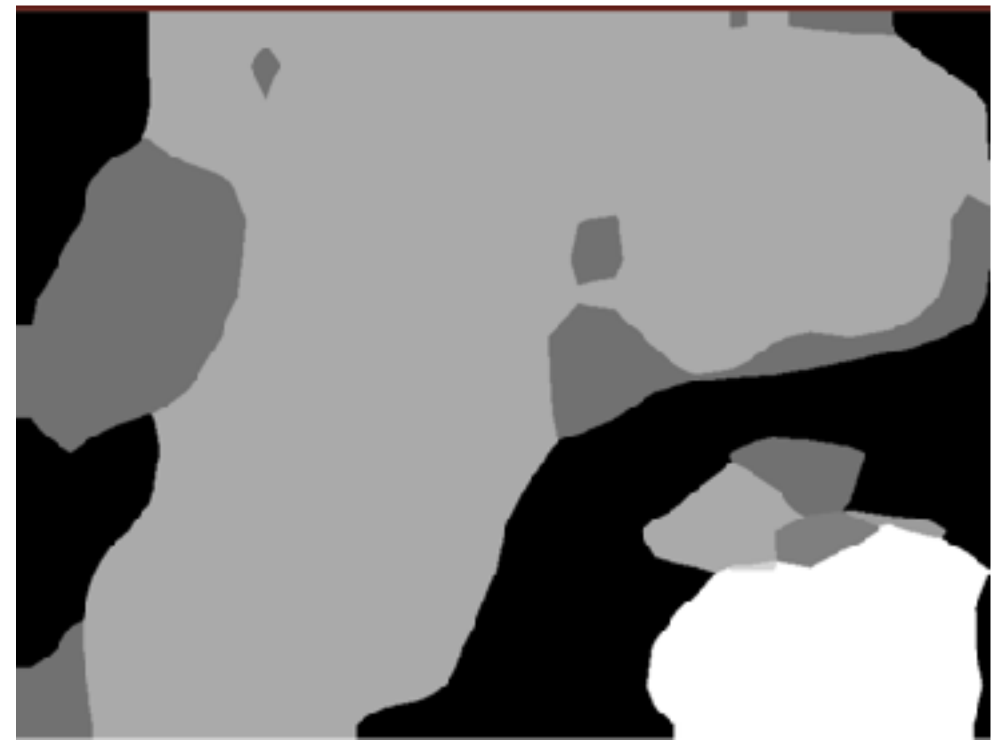
Cuda8.0

Data_preparation

- 2_dog.jpg
- aircraft.jpg
- car.JPG
- cow.jpeg
- dog.jpeg
- dog.jpg
- horse.jpg
- horse_people.jpg
- lena.jpg
- machine.JPG
- n_dog.jpg
- vt.jpg
- vt1.jpg

load image, switch to BGR, subtract mean, and make dims C x H x W for Caffe

Experiment



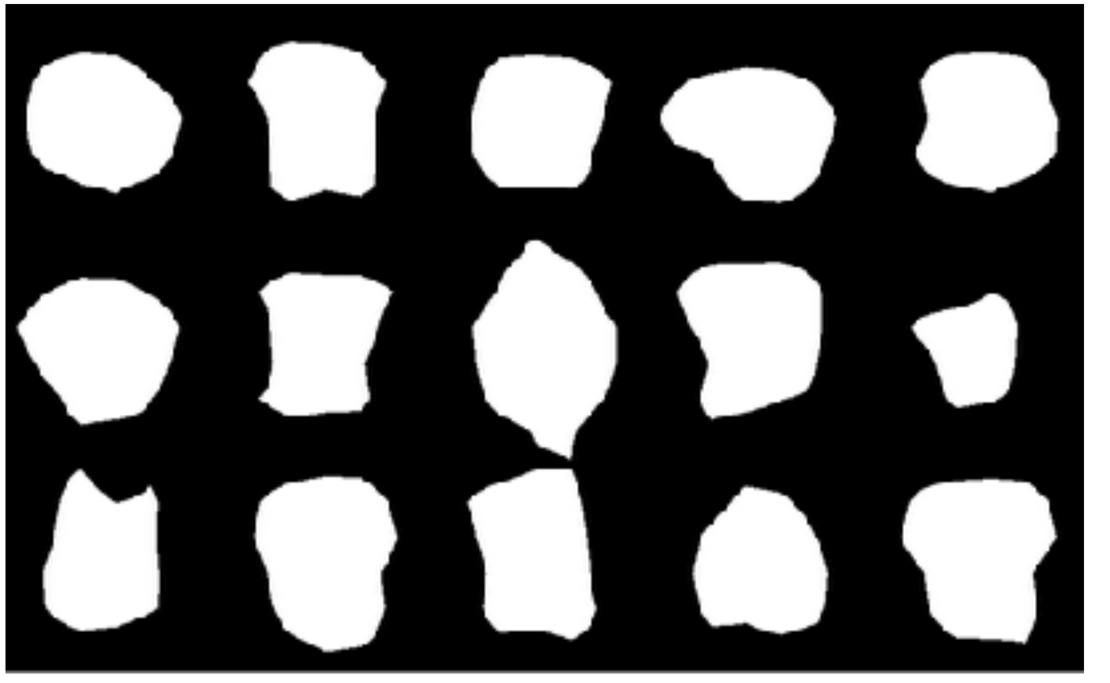
26.862607 **1.238836**

Experiment

39.570141

1.738234

Experiment



32.238836

1.238836

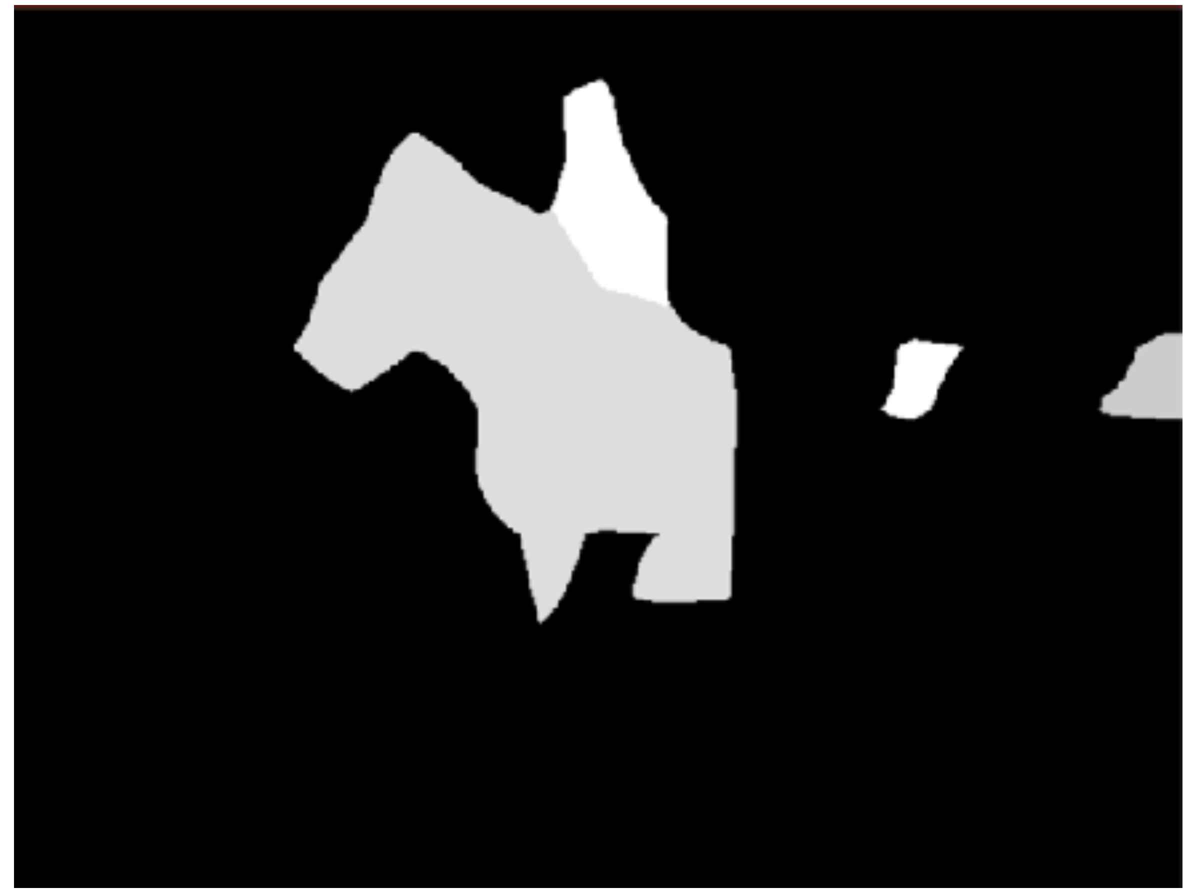
Experiment

39.570141

1.5334832

Experiment

27.895173



1.239234

Conclusion

1.Their network is very fast even when dealing with high resolution image, and GPU is at least 20 times faster than CPU.

2. The algorithms show good performance towards images when the objects are either well-separated or overlapped with each other

3. The background of image like sky, grass has a big influence on the segmentation.

Better performance could be expected with their FCN_8s, and detailed performance on validation dataset needs to be checked.

Thanks