Semantic Segmentation
and
Image Processing

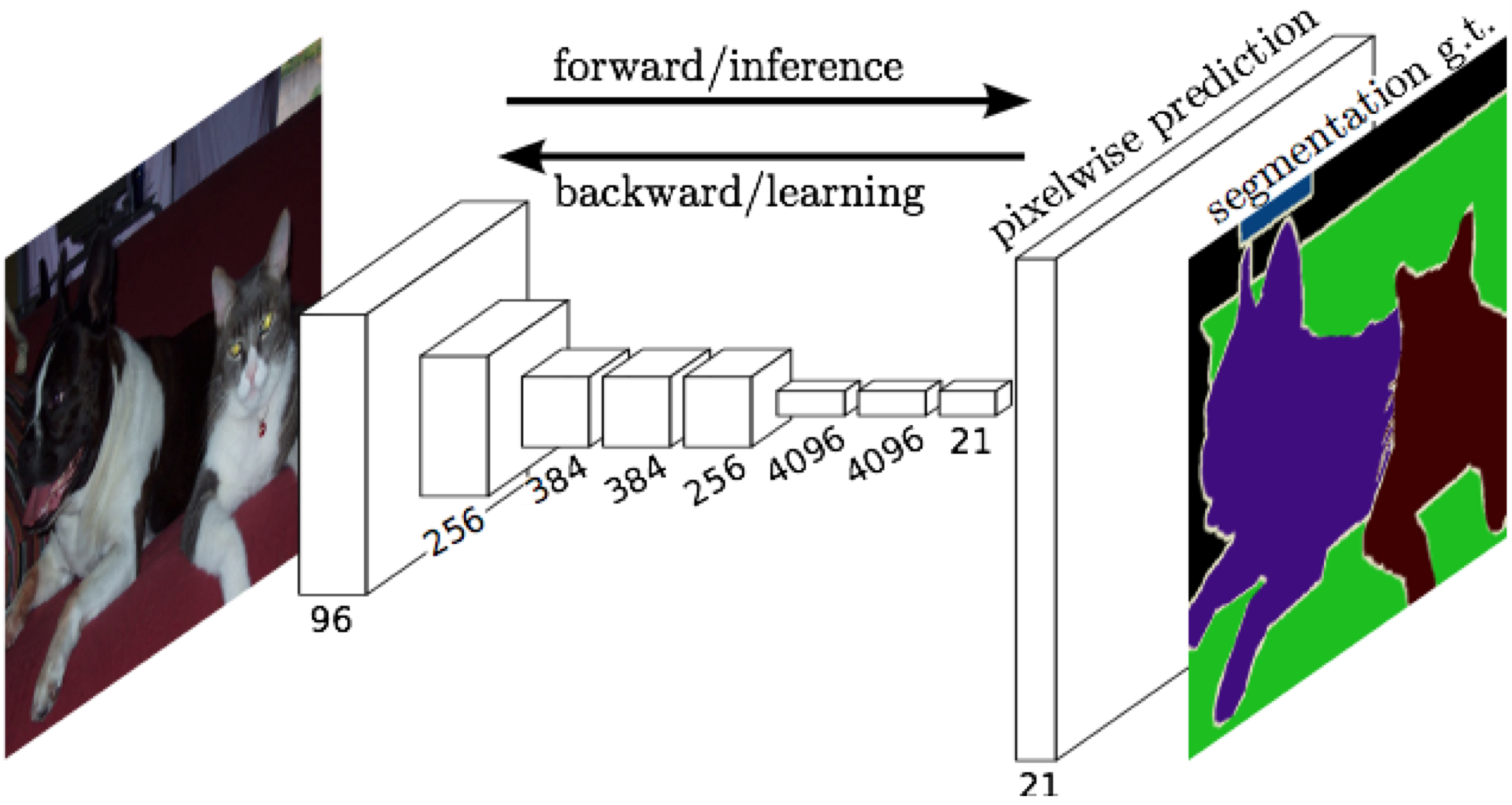
with Convnets

Overview

- Methods where output is also an image
  - Fully Convolutional Nets [Long et al., CVPR 2015]
  - Depth, normals and semantic labels from a single image [Eigen ICCV 2015]

- Image processing with Convnets

A Fuller Understanding of Fully Convolutional Networks



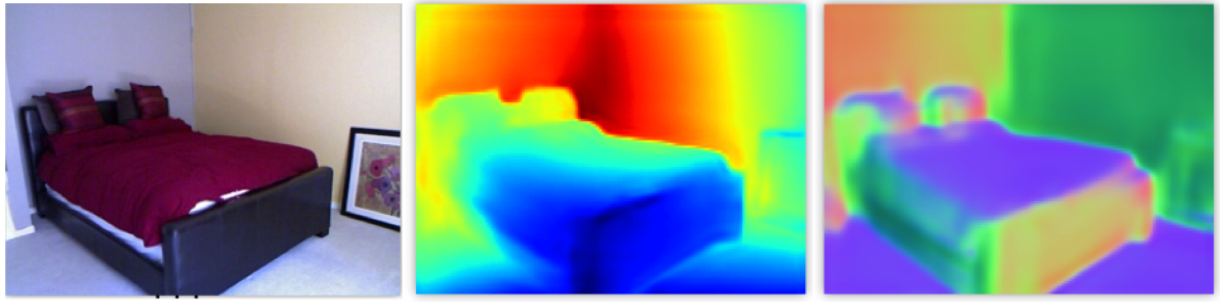
Evan Shelhamer* Jonathan Long* Trevor Darrell

UC Berkeley in CVPR'15, PAMI'16

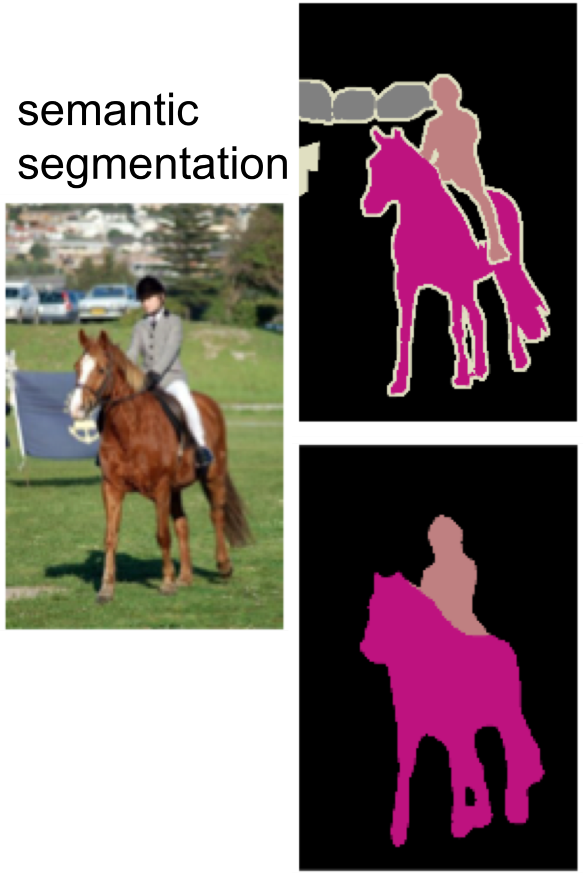
pixels in, pixels out

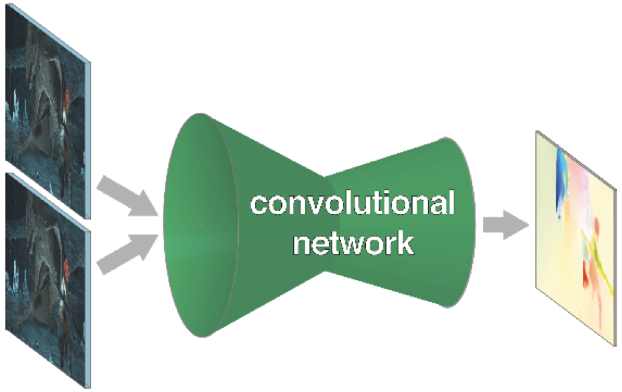
colorization
Zhang et al.2016

monocular depth + normals Eigen & Fergus 2015



semantic
segmentation

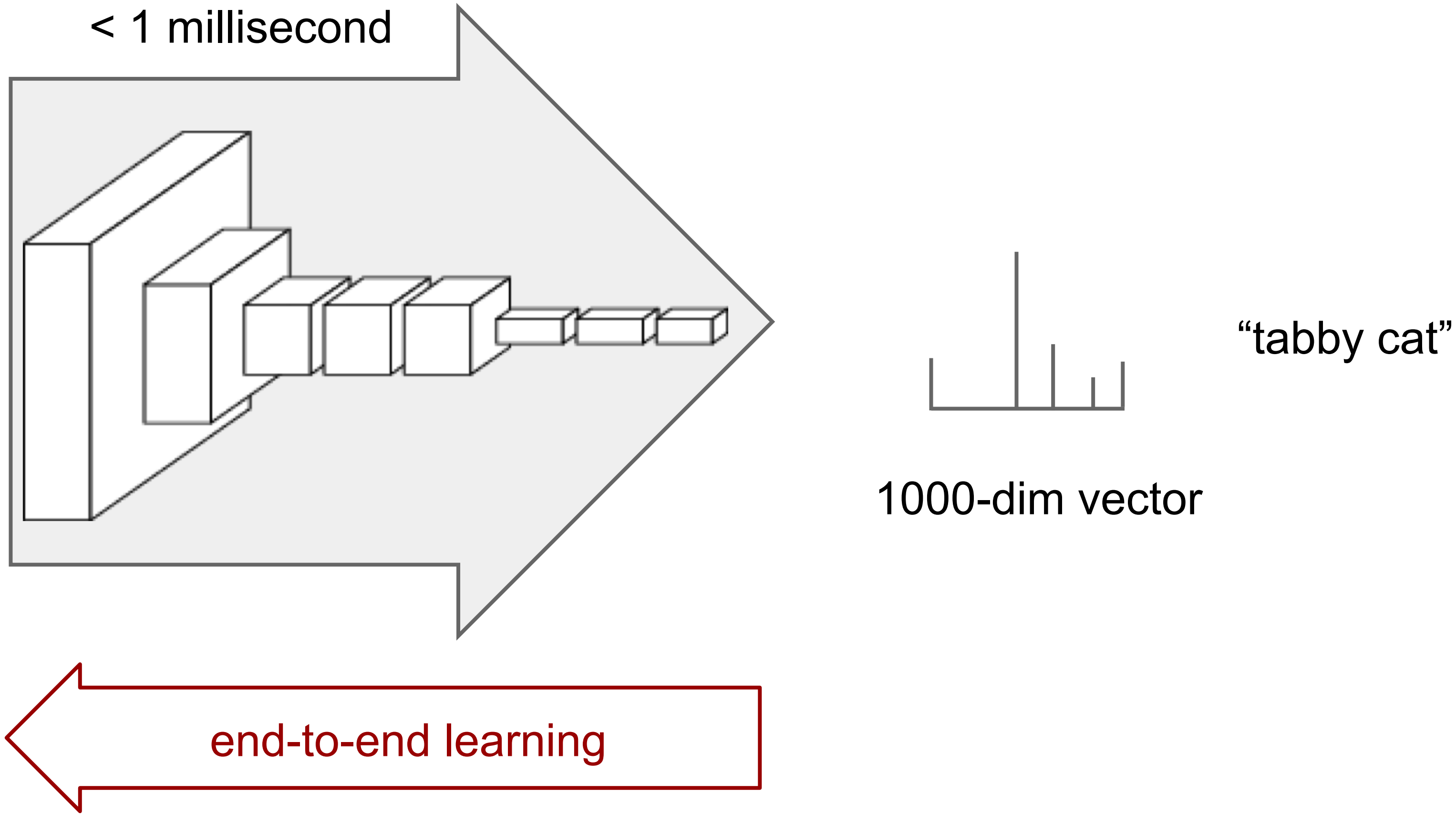




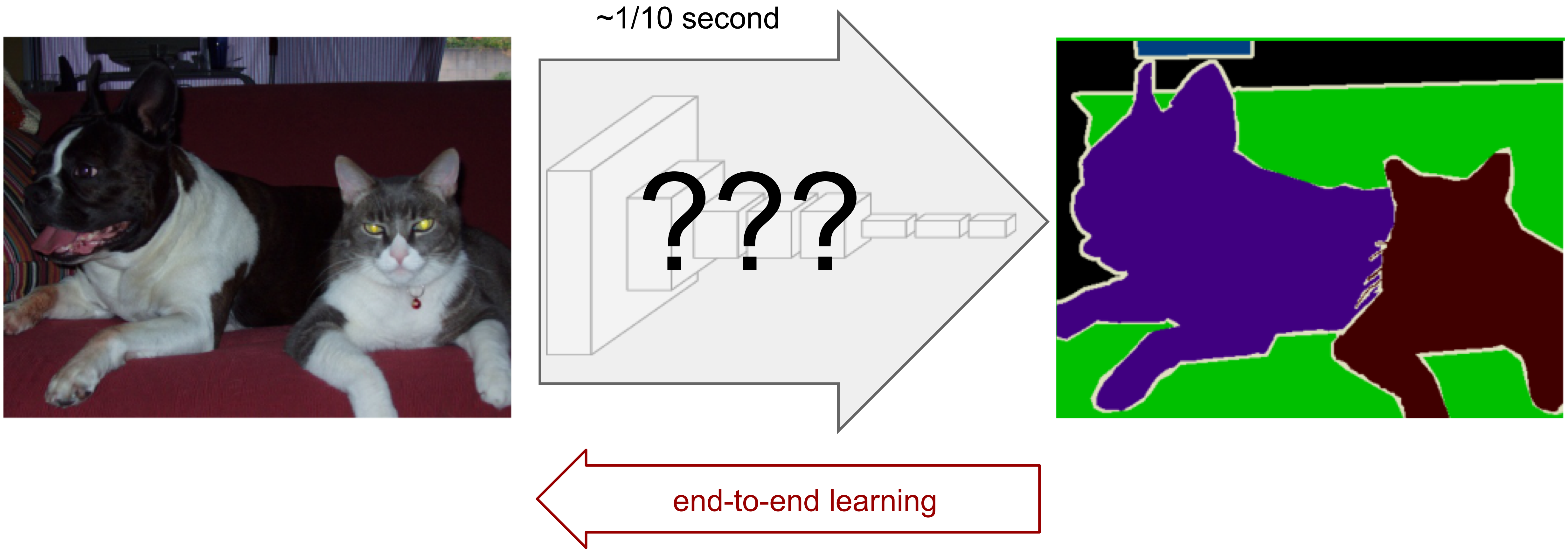
optical flow Fischer et al. 2015

boundary prediction Xie & Tu 2015

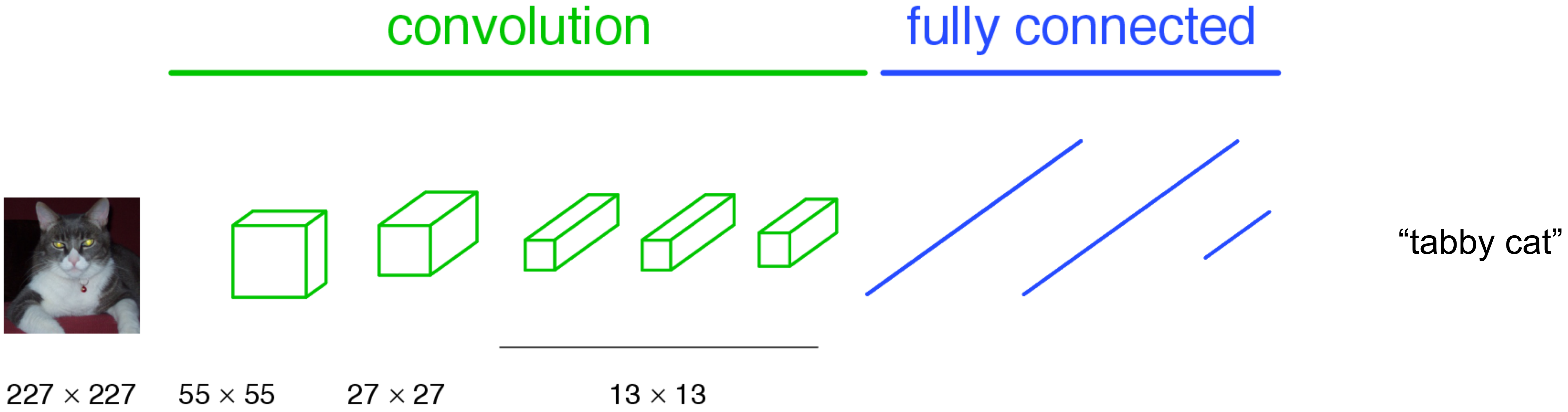
convnets perform classification



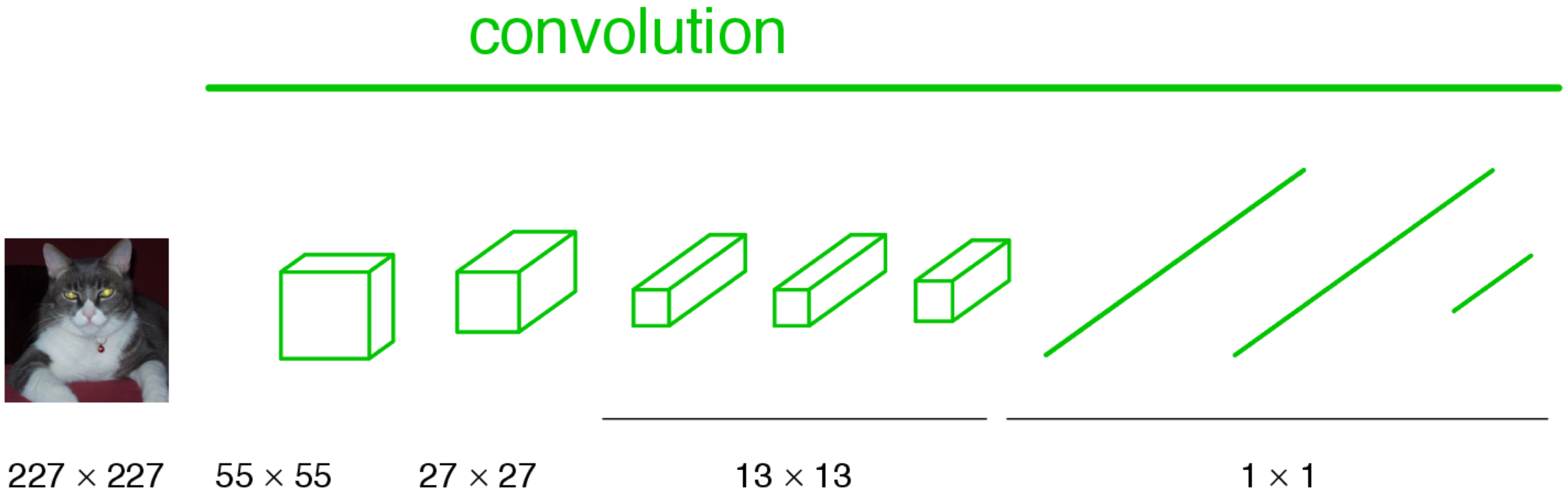
lots of pixels, little time?



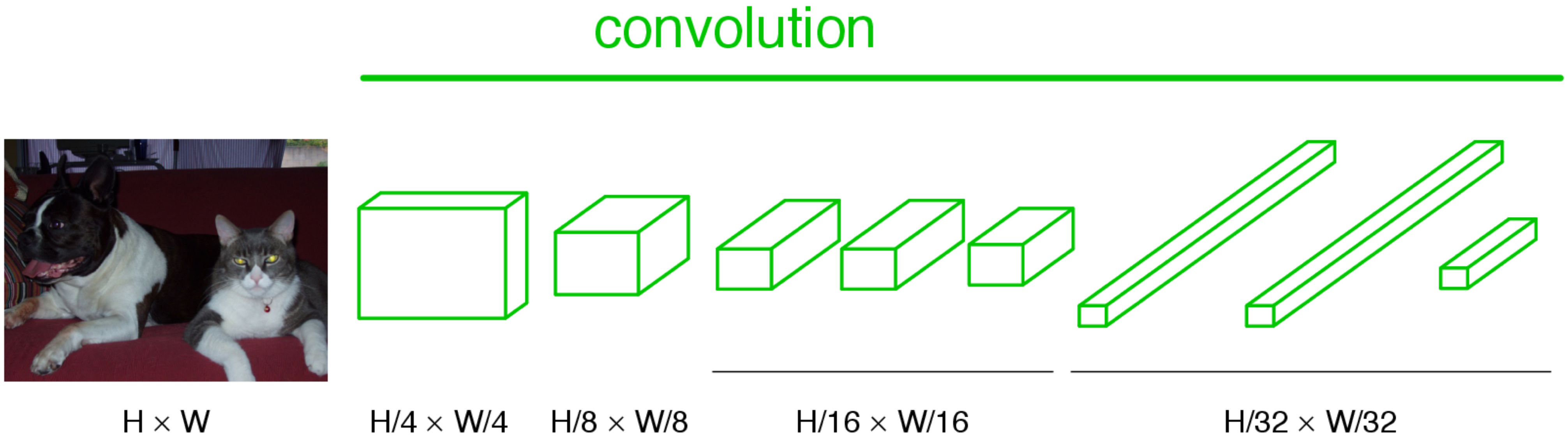
a classification network



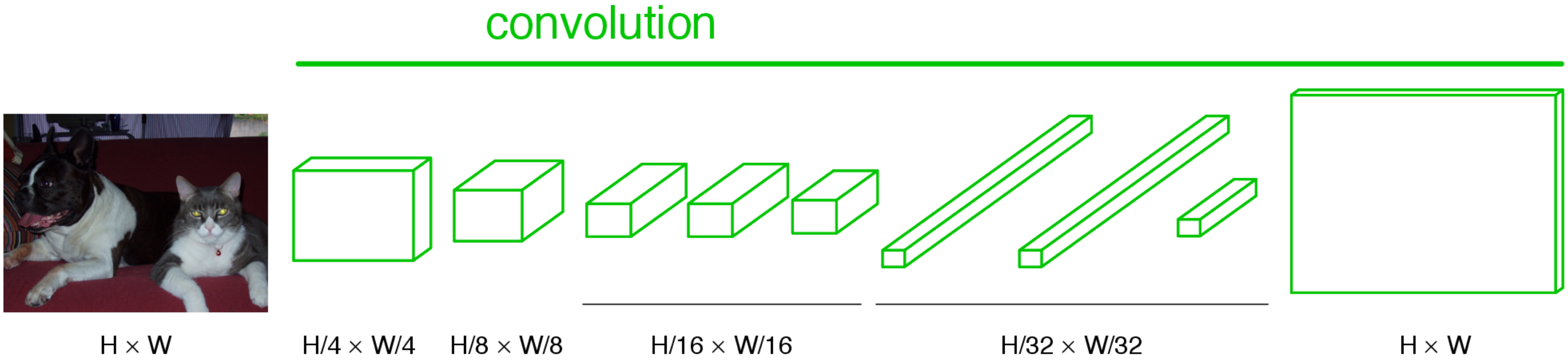
becoming fully convolutional



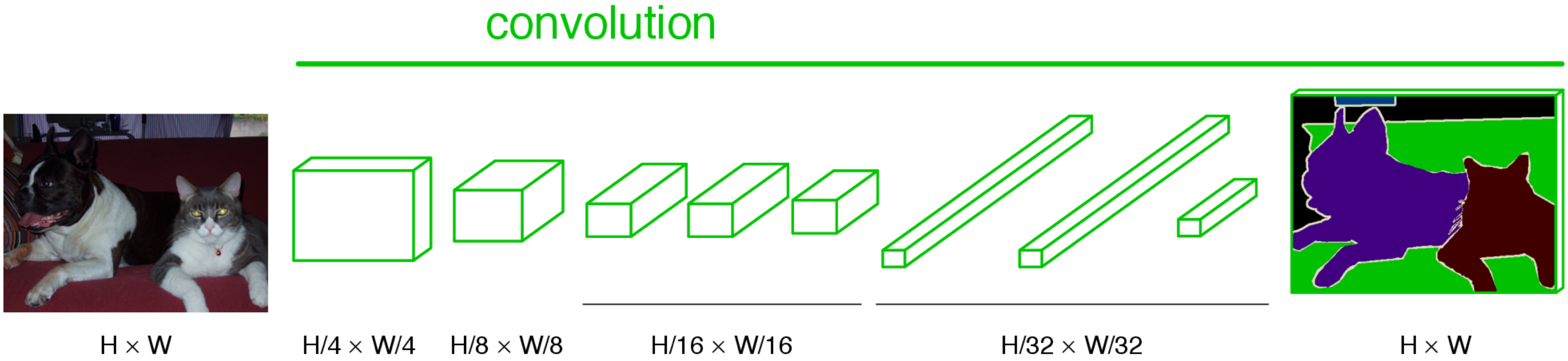
becoming fully convolutional



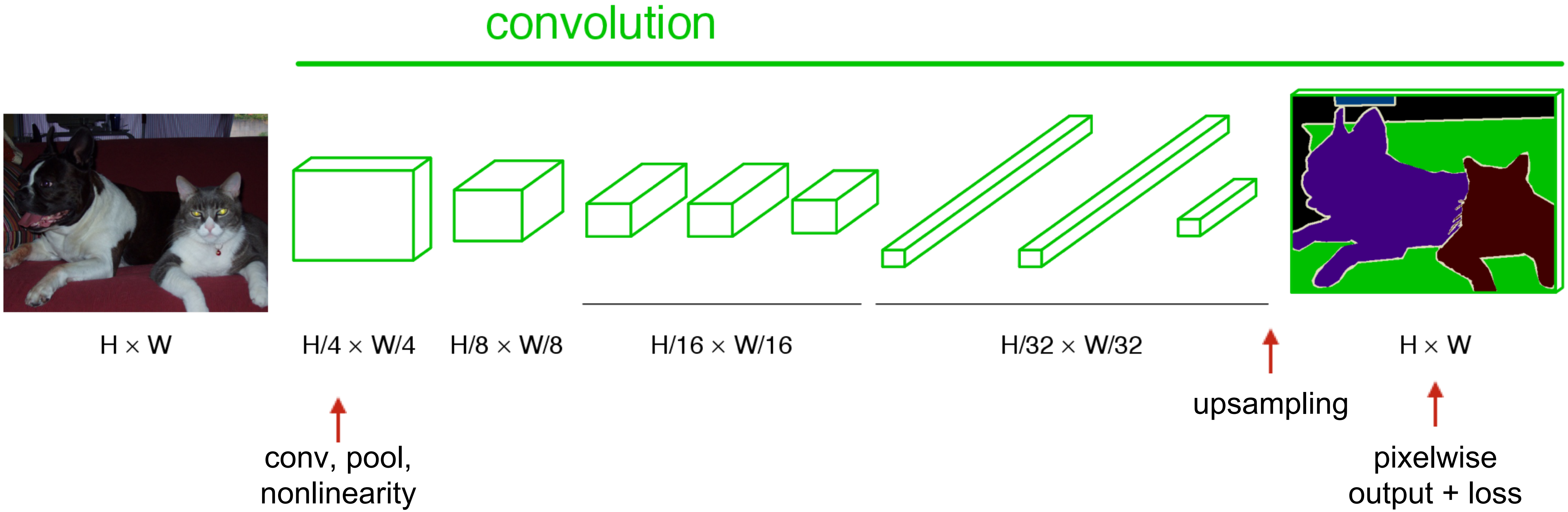
upsampling output



end-to-end, pixels-to-pixels network

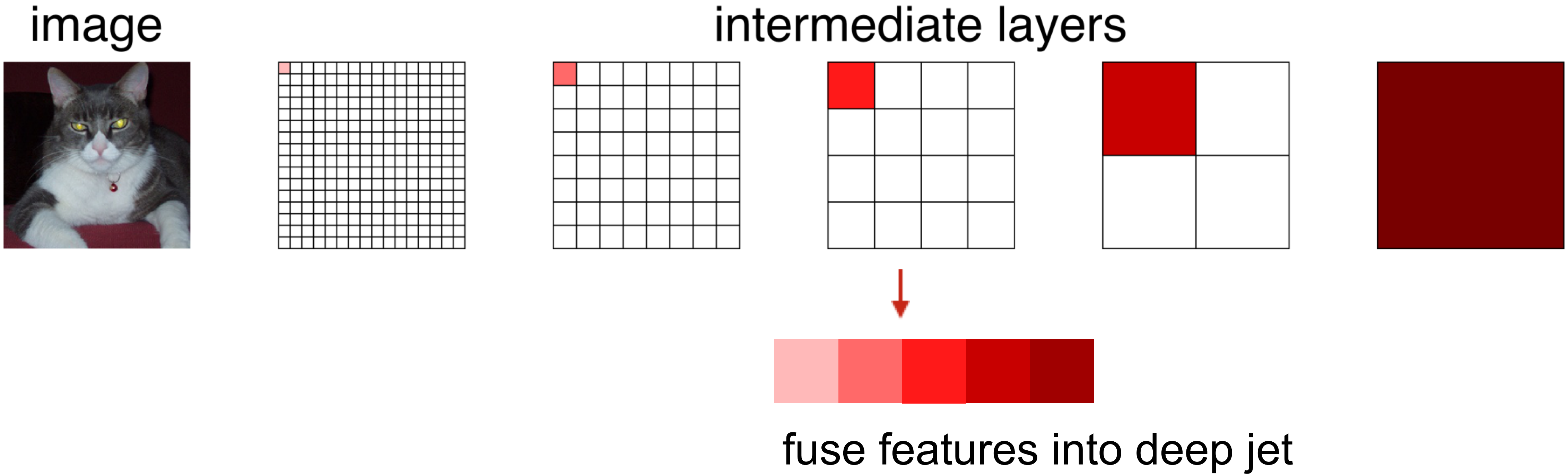


end-to-end, pixels-to-pixels network



spectrum of deep features

combine where (local, shallow) with what (global, deep)



(cf. Hariharan et al. CVPR15 “hypercolumn”)

skip layers

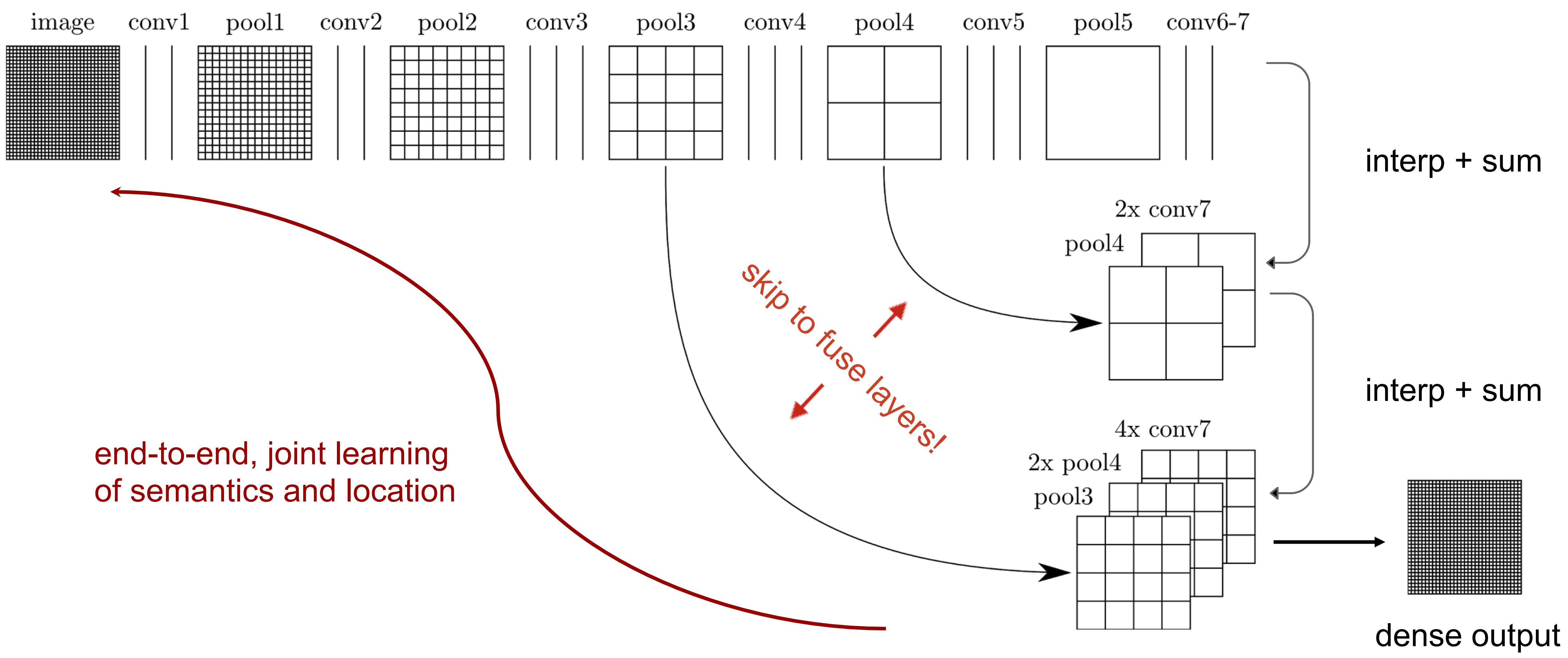
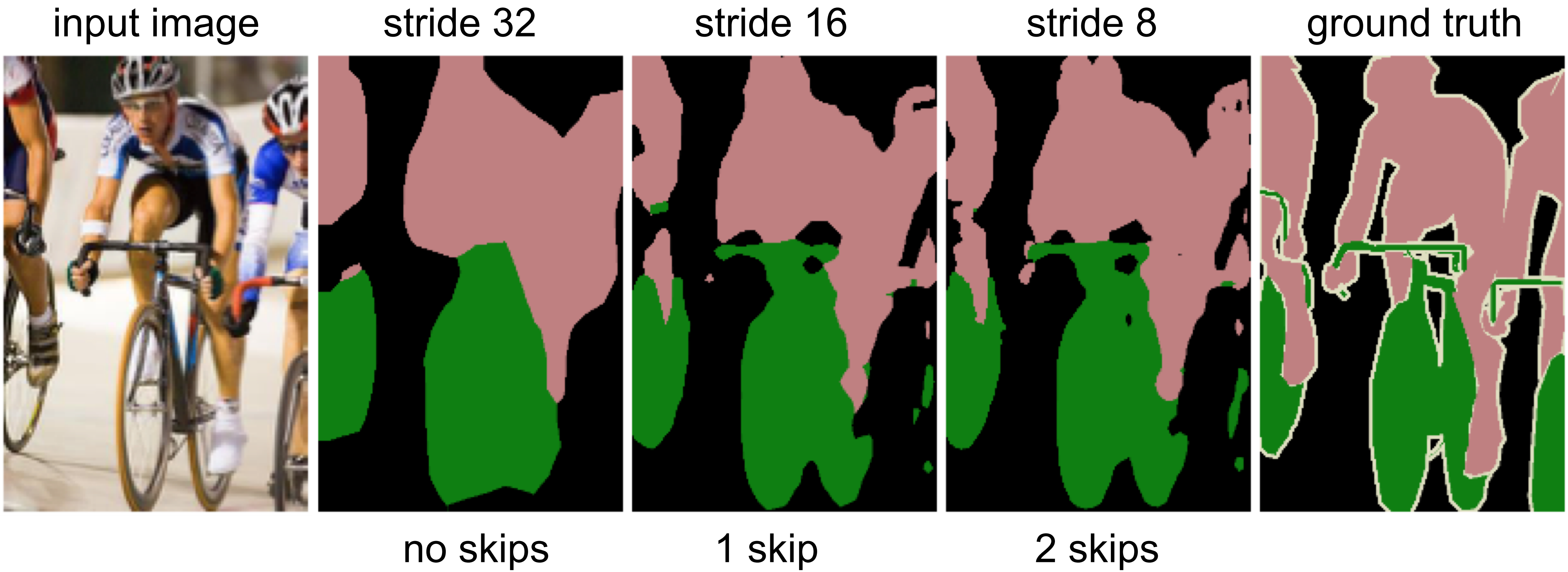
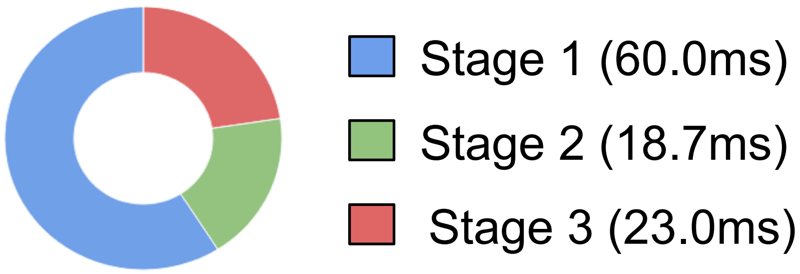


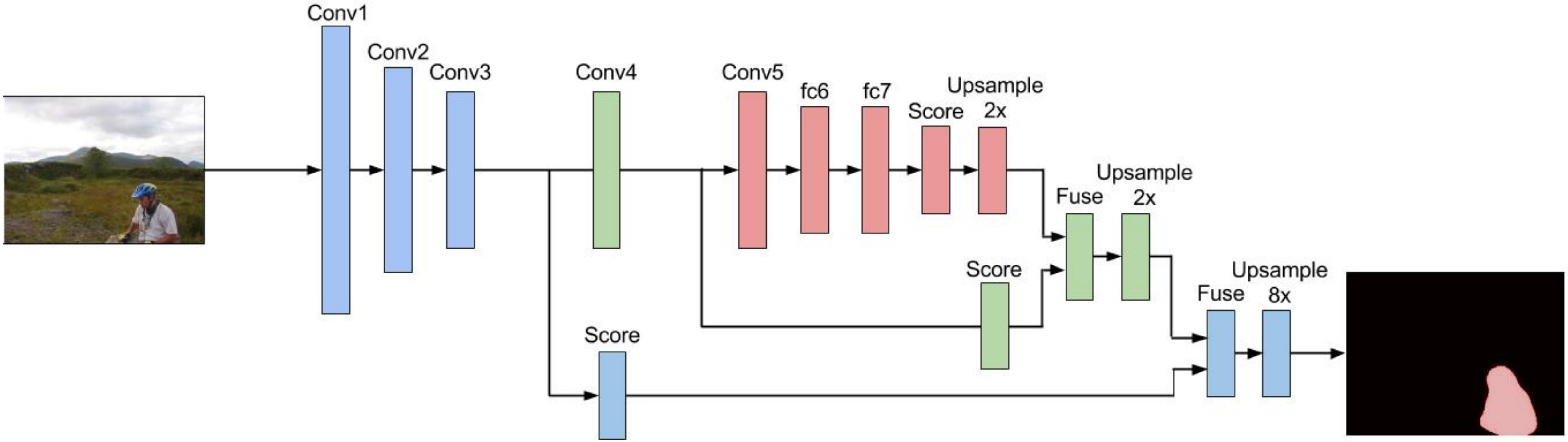
image
conv1
pool1
conv2
pool2
conv3
pool3
conv4
pool4
conv5
pool5
conv6-7
interp + sum
2x conv7
pool4
skip to fuse layers!
interp + sum
4x conv7
2x pool4
pool3
end-to-end, joint learning
of semantics and location
dense output

skip layer refinement

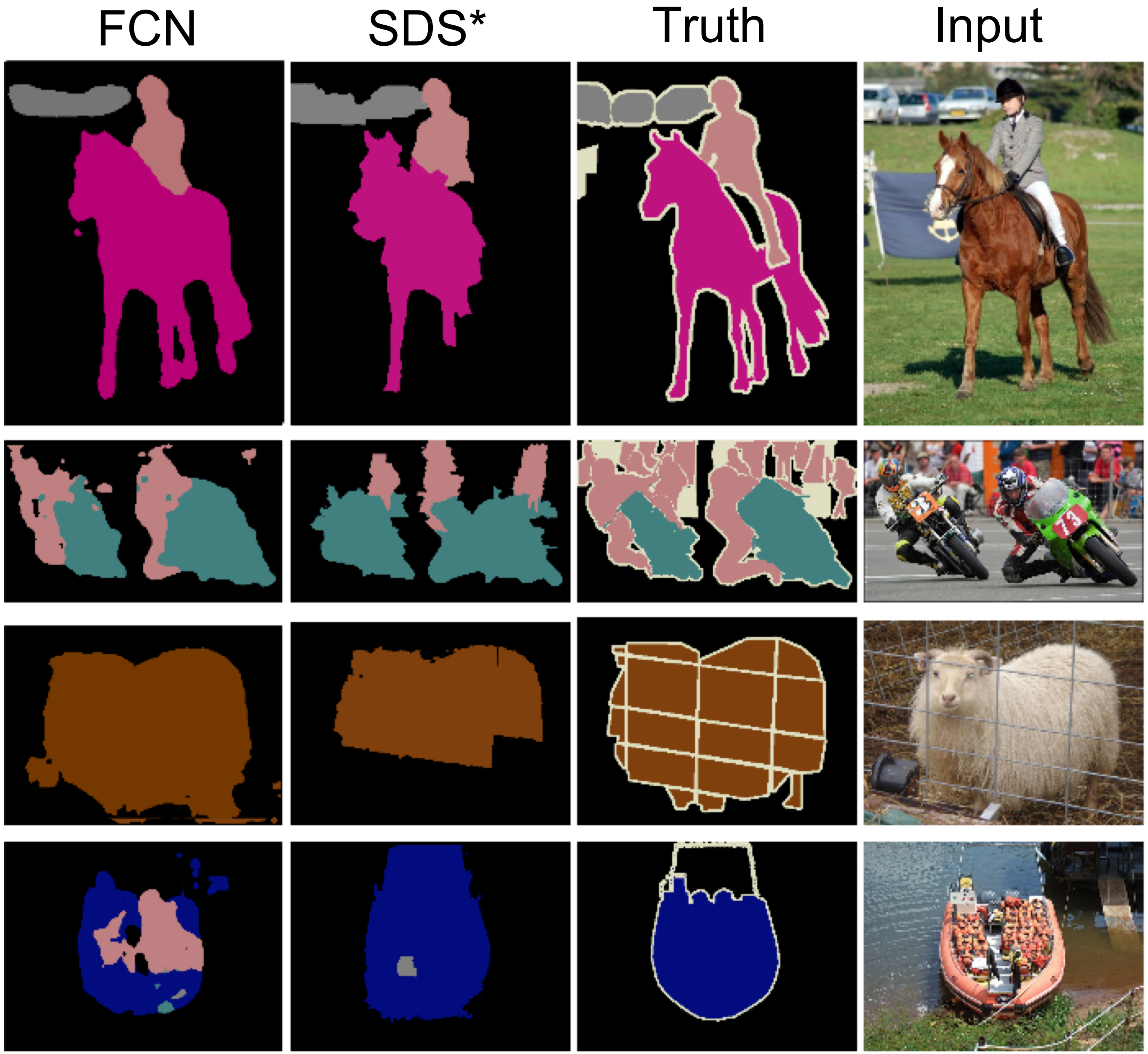


skip FCN computation





A multi-stream network that fuses features/predictions across layers

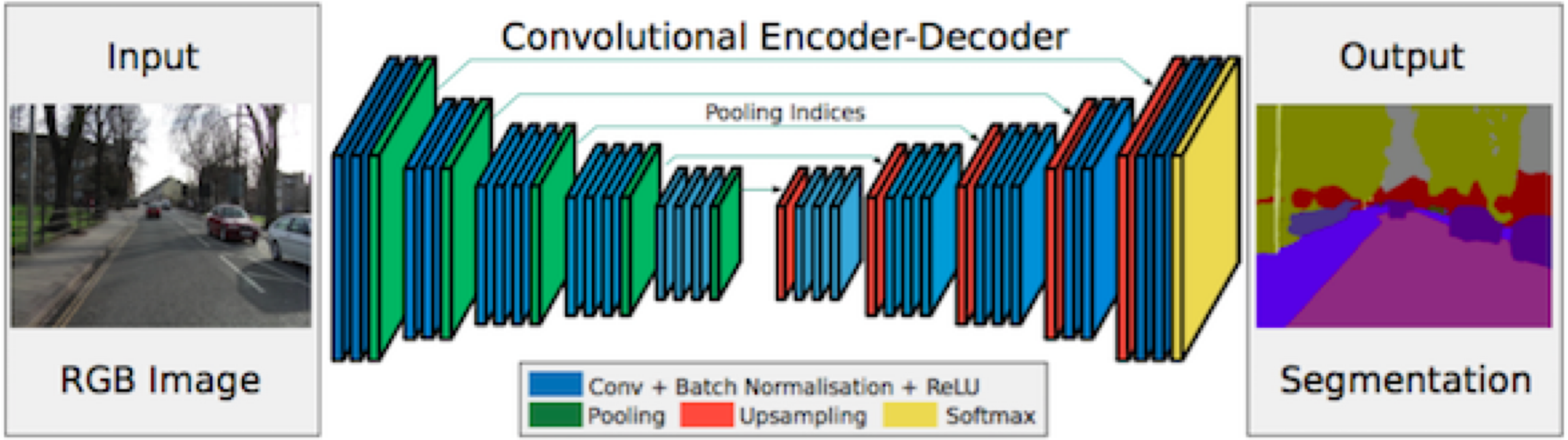


Relative to prior state-of-the-art SDS:

- 30% relative improvement for mean IoU
- 286× faster

*Simultaneous Detection and Segmentation
Hariharan et al. ECCV14

SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation

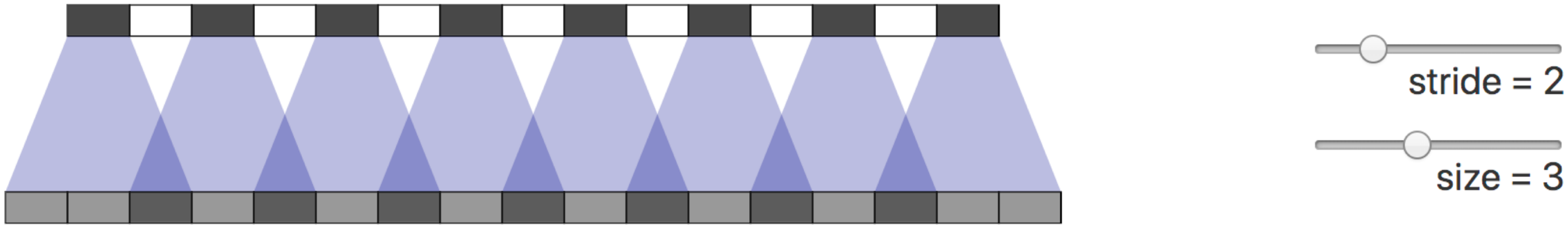


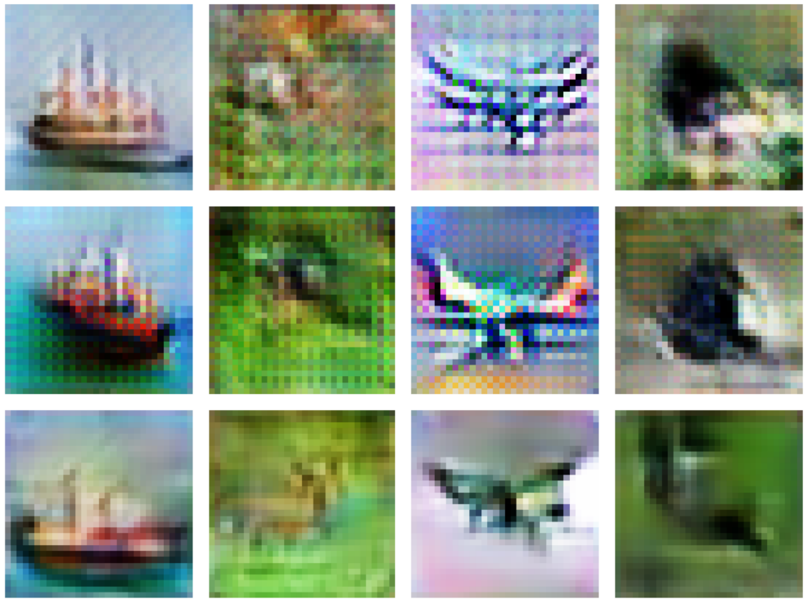
Max pooling indices transferred to decoder to improve output resolution

https://arxiv.org/abs/1511.00561

How to do the Upsampling?

Also known as Deconvolution
See https://distill.pub/2016/deconv-checkerboard/





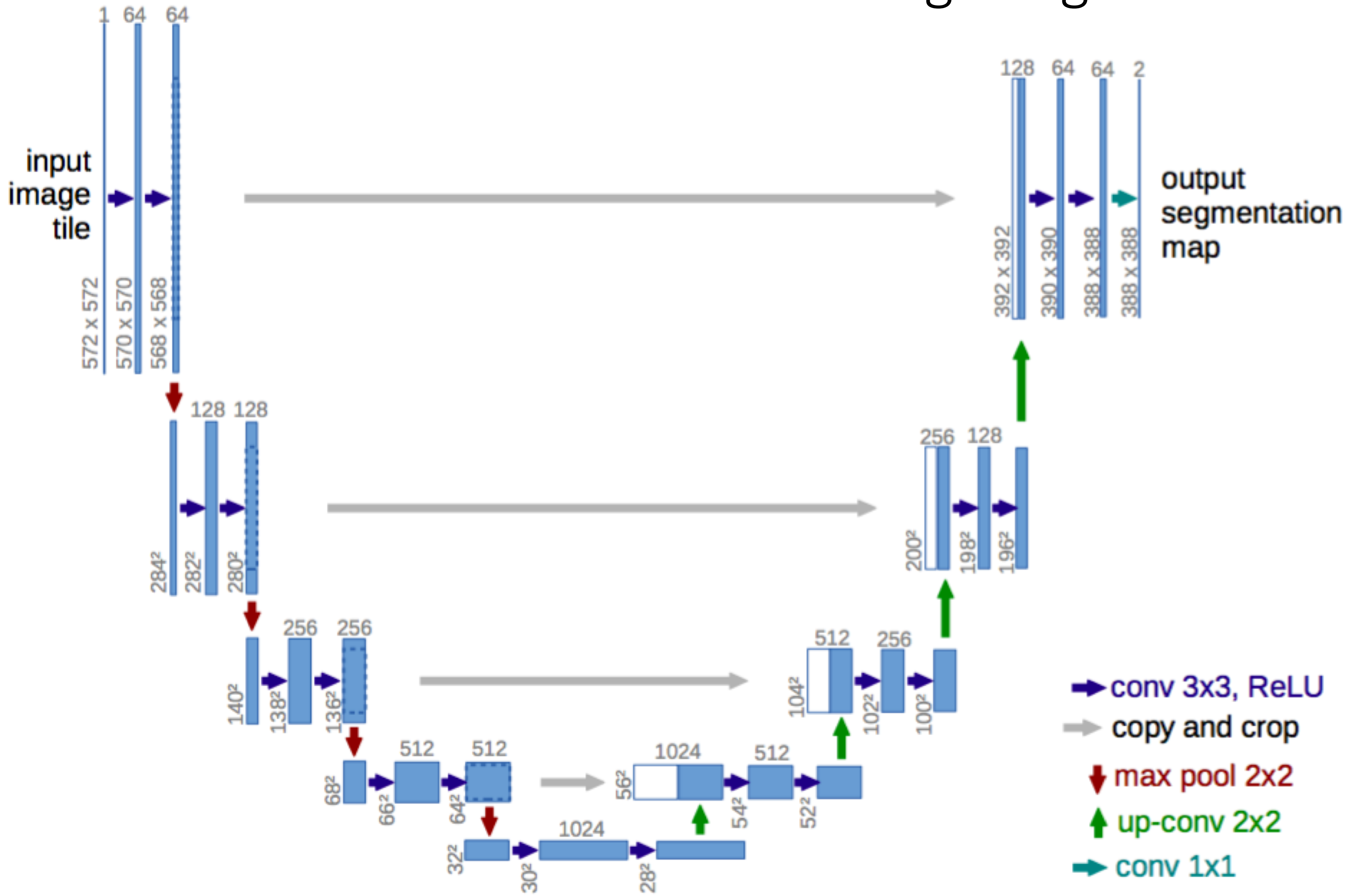
Deconv in last two layers.
Other layers use resize-convolution.
Artifacts of frequency 2 and 4.

Deconv only in last layer.
Other layers use resize-convolution.
Artifacts of frequency 2.

All layers use resize-convolution.
No artifacts.

Avoid artifacts by doing bilinear interpolation

UNet: Convolutional Networks for Biomedical Image Segmentation



https://arxiv.org/abs/1505.04597

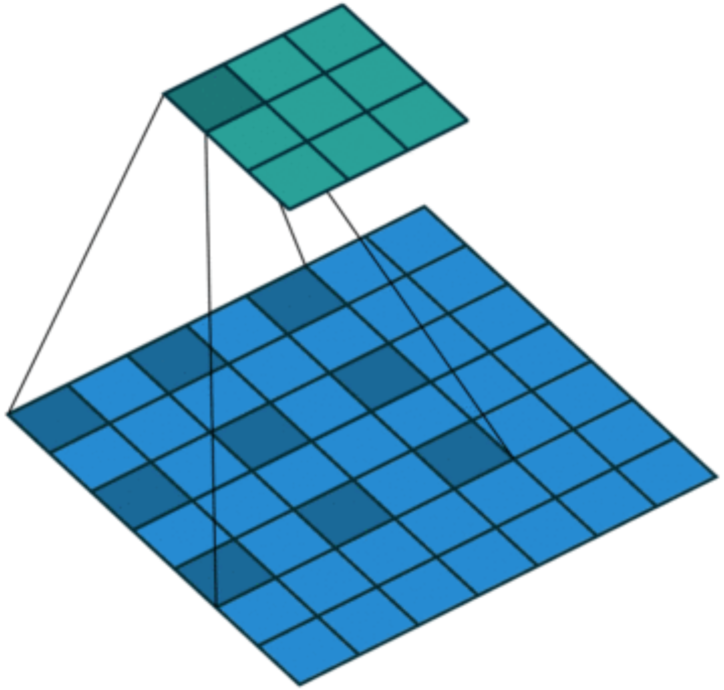
Segmentation of a 512x512 image takes less than a second on a recent GPU

Dilated / Atrous Convolutions

[Multi-Scale Context Aggregation by Dilated Convolutions, Yu and Koltun, 2015]

- No pooling operations
- Constant resolution feature maps
- Integrate increasing spatial context by special kind of dilated convolution

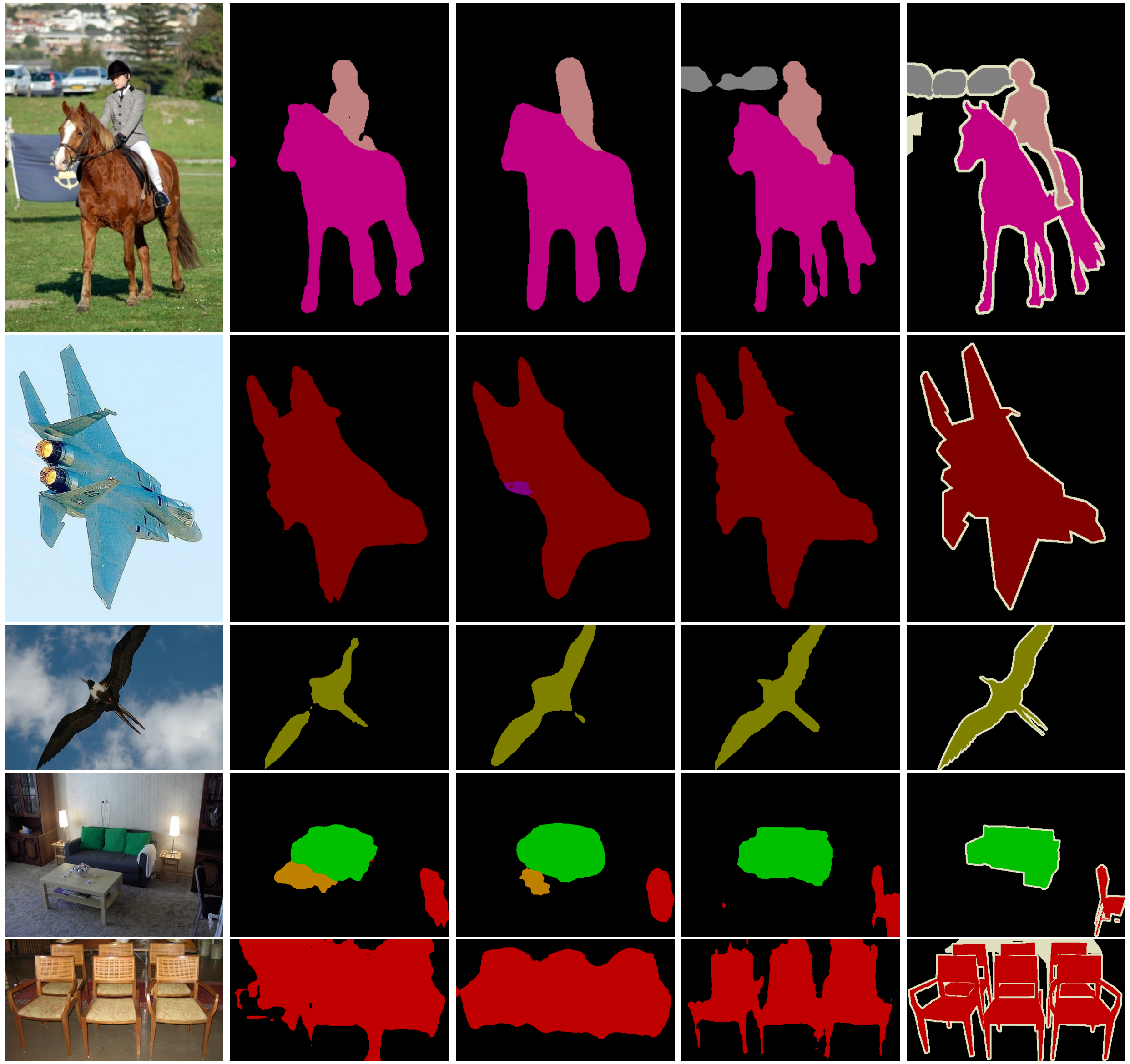
Layer	1	2	3	4	5	6	7	8
Convolution	3×3	3×3	3×3	3×3	3×3	3×3	3×3	1×1
Dilation	1	1	2	4	8	16	1	1
Truncation	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Receptive field	3×3	5×5	9×9	17×17	33×33	65×65	67×67	67×67
Output channels								
Basic	C	C	C	C	C	C	C	C
Large	$2C$	$2C$	$4C$	$8C$	$16C$	$32C$	$32C$	C



- Constant 64x64 spatial resolution throughout

Dilated / Atrous Convolutions

[Multi-Scale Context Aggregation by Dilated Convolutions, Yu and Koltun, 2015]



(a) Image (b) FCN-8s (c) DeepLab (d) Our front end (e) Ground truth

Further Resources

http://blog.qure.ai/notes/semantic-segmentation-deep-learning-review

Overview

- Methods where output is now an image
  - Fully Convolutional Nets [Long et al., CVPR 2015]
  - Depth, normals and semantic labels from a single image [Eigen ICCV 2015]

- Image processing with Convnets
  - Image colorization [Zhang et al. ECCV 2016]

Beyond Object Classification with Convolutional Networks

David Eigen (NYU -> Clarifai)
Rob Fergus (Facebook / NYU)

Motivation

Input Image

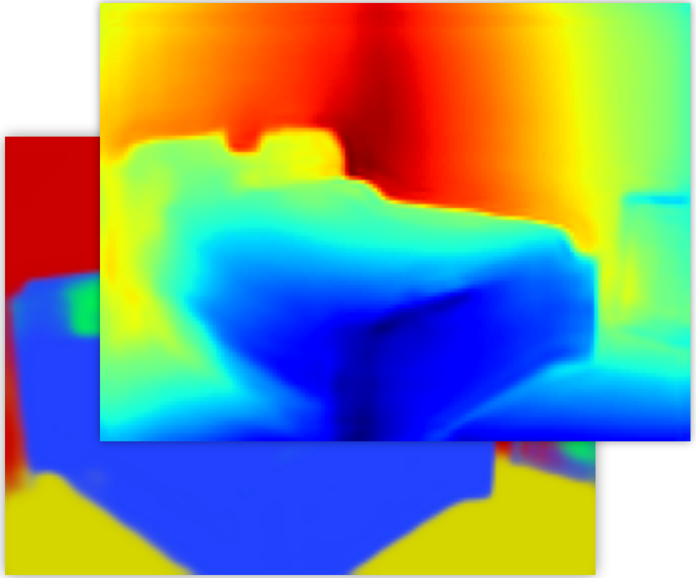
Semantic Map

- Understand input scene
  - Semantic
  - Geometric

Motivation

Input Image

Depth

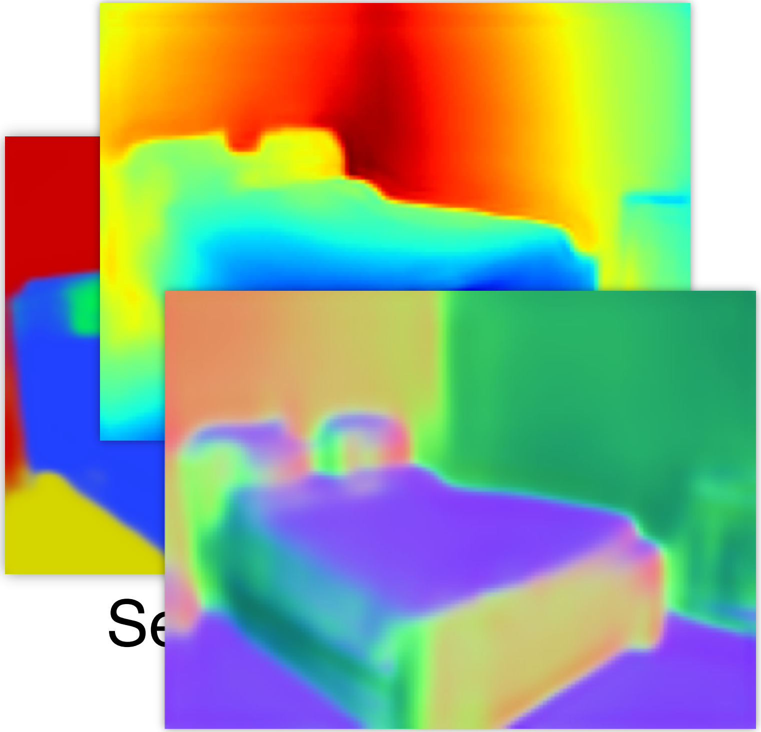


Semantic Map

- Understand input scene
  - Semantic
  - Geometric

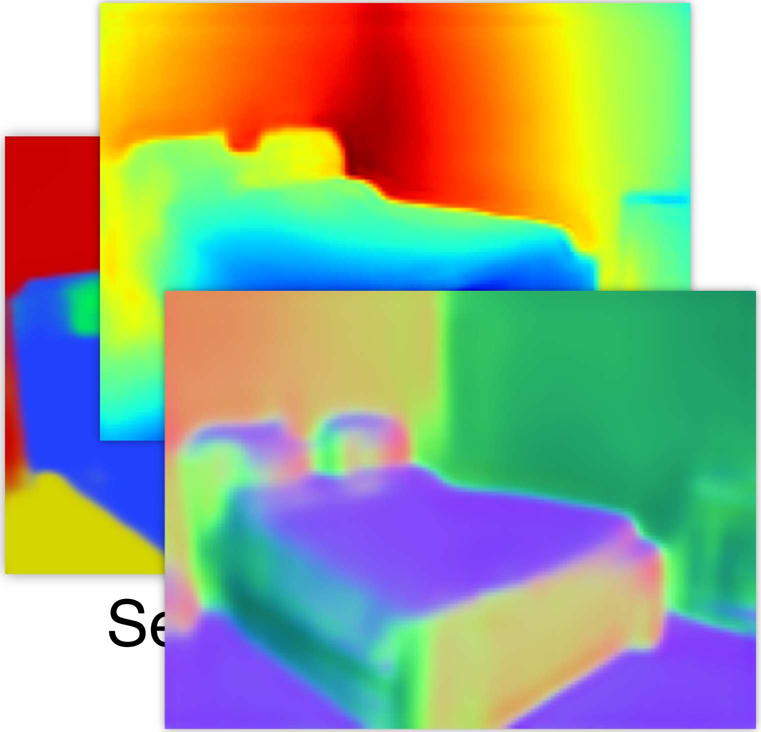
Motivation

Input Image



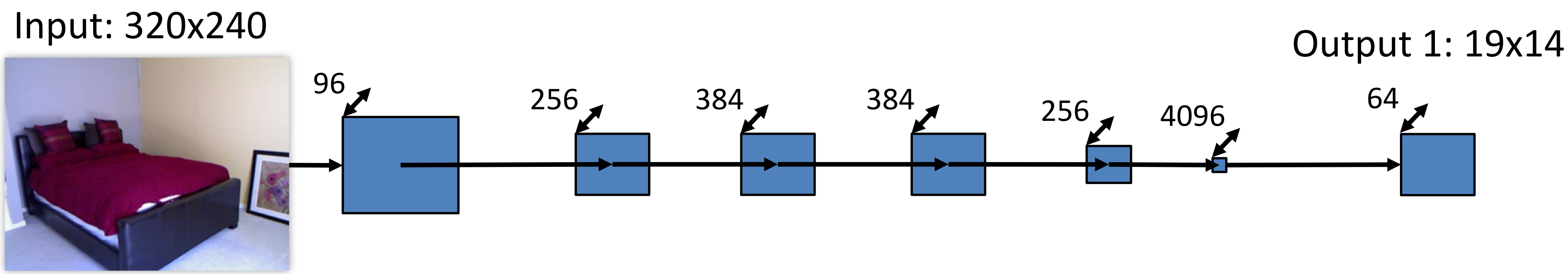
- Understand input scene
  - Semantic
  - Geometric

Motivation

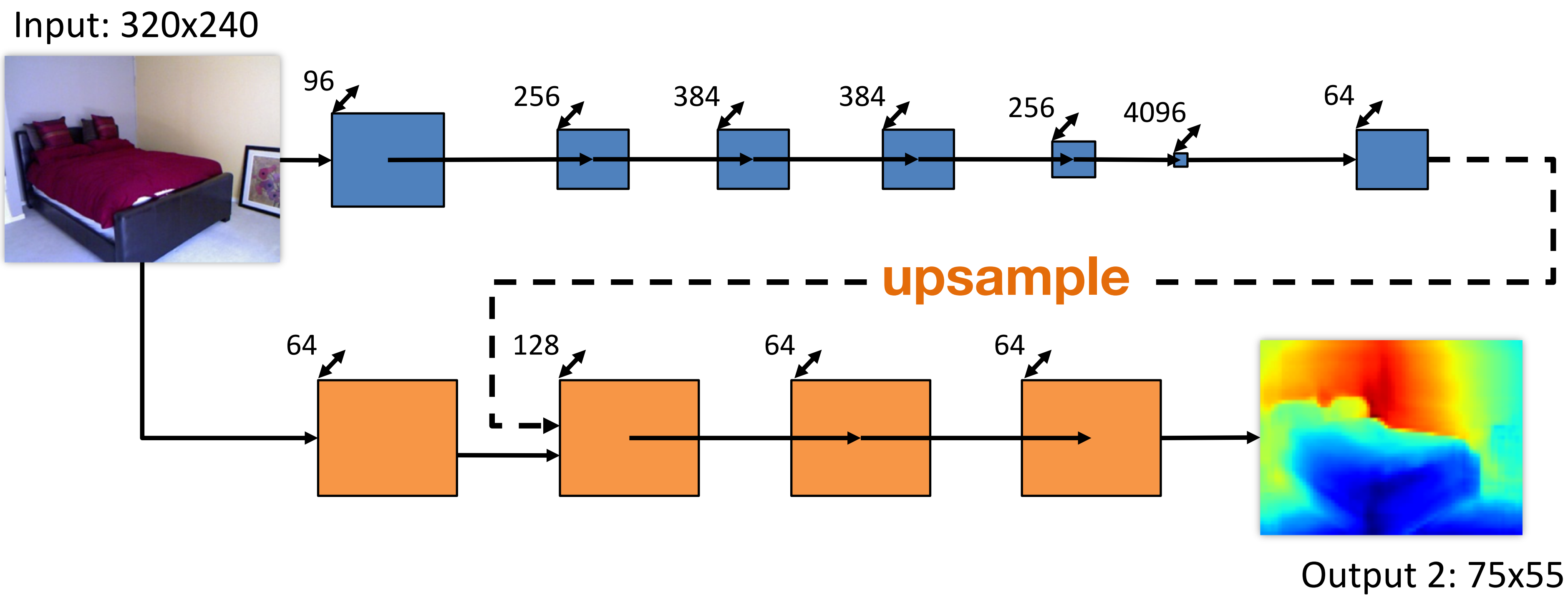


- **<u>Predict Pixel Maps from a Single Image</u>**

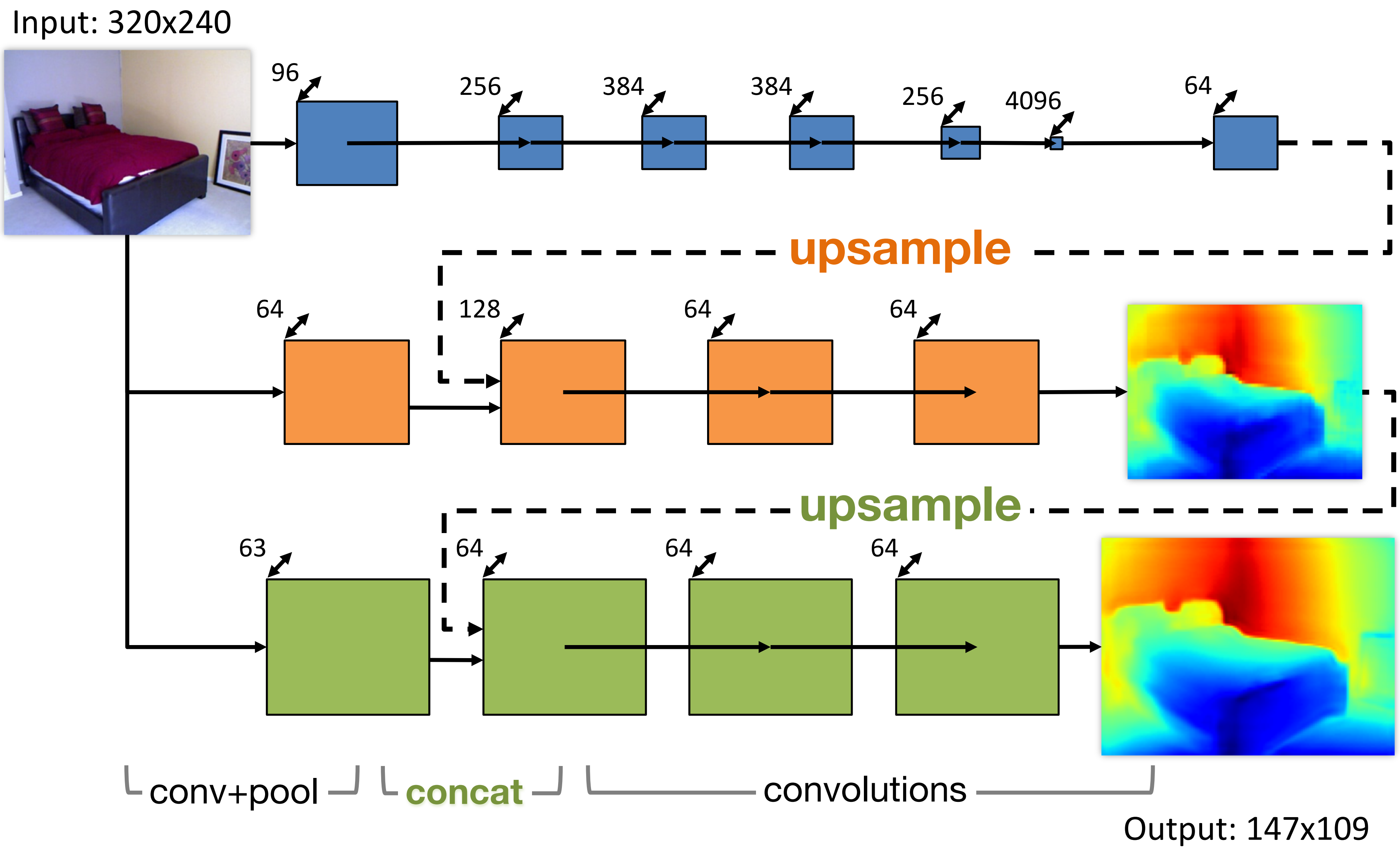
Architecture



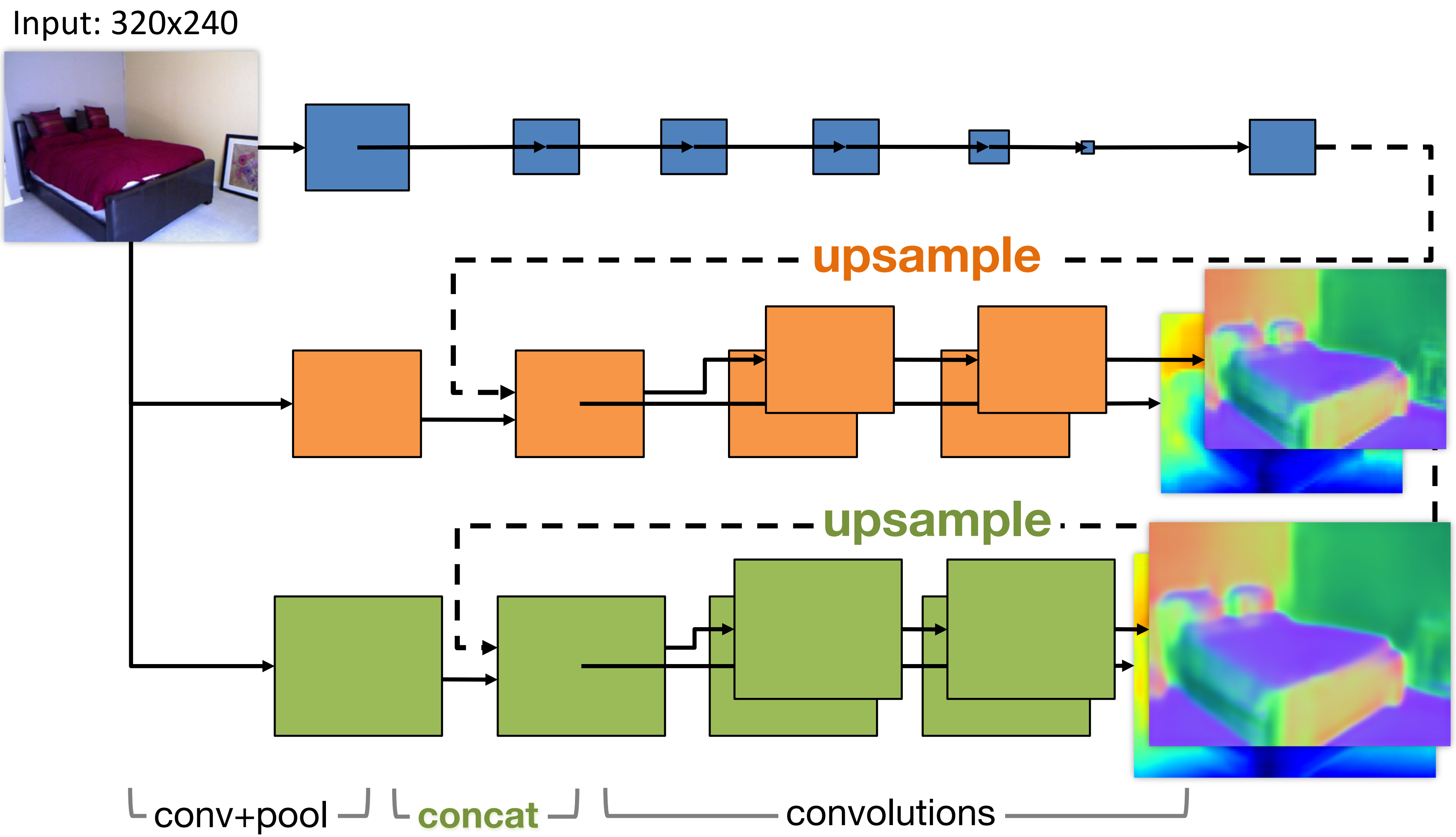
Architecture



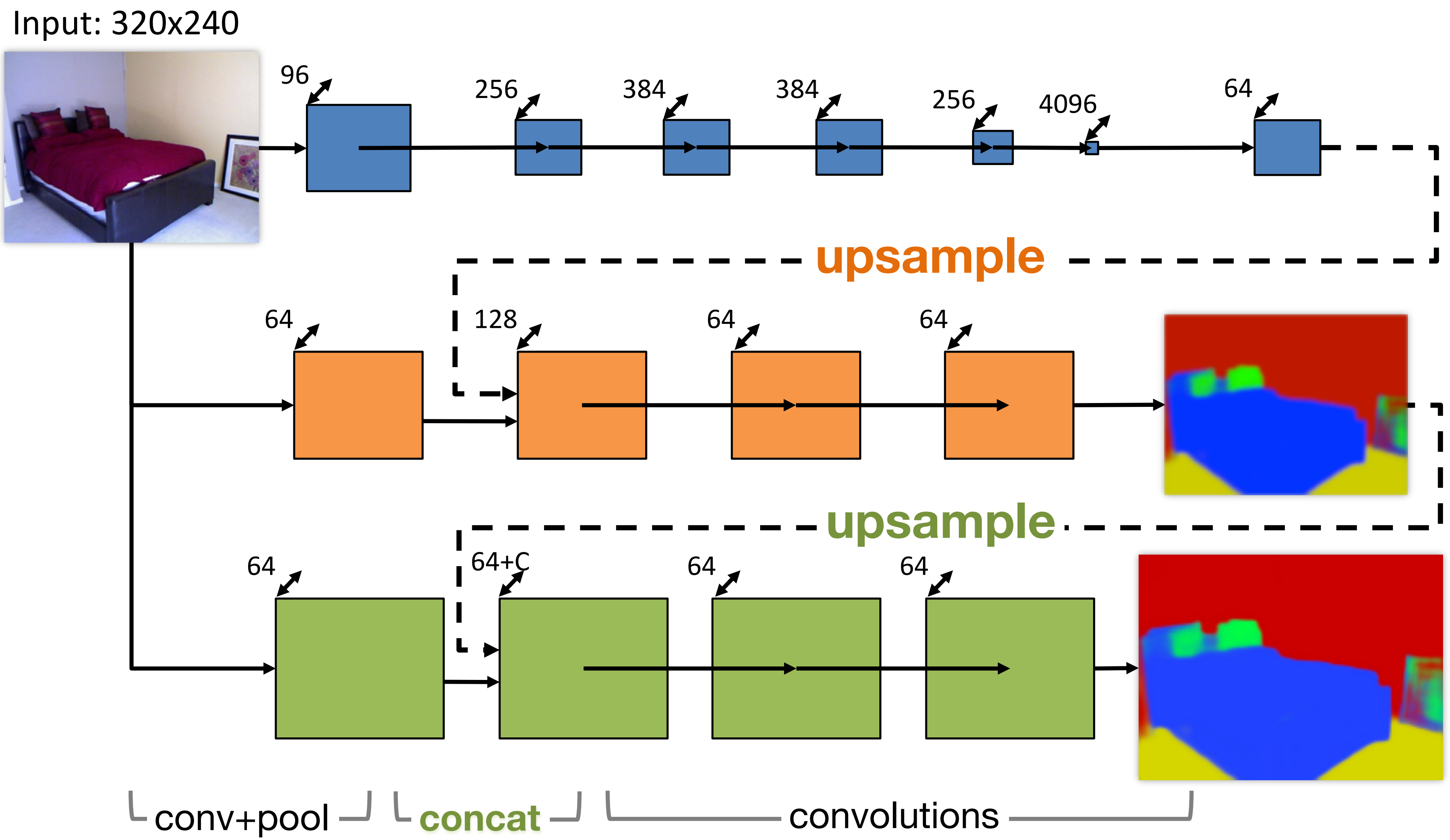
Architecture



Architecture



Architecture



Losses

Depth: $d = D - D^*$ D = log predicted depth, D* = log true depth

$$L_{depth}(D, D^*) = \frac{1}{n}\sum_i d_i^2 - \frac{1}{2n^2}\left(\sum_i d_i\right)^2 + \frac{1}{n}\sum_i[(\nabla_x d_i)^2 + (\nabla_y d_i)^2]$$

Norm

Labels

Training

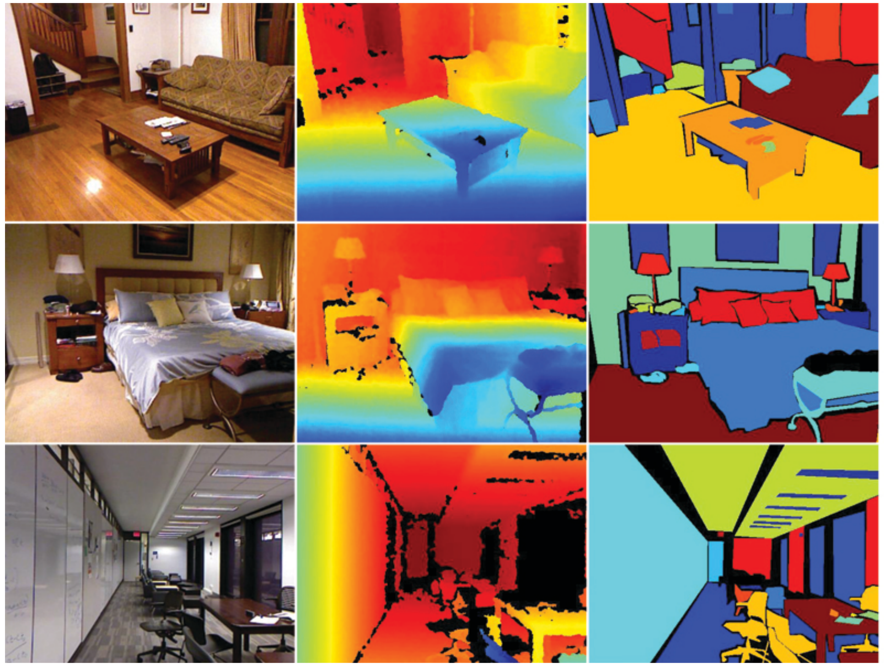
- Pre-train Alexnet/VGGnet scale 1 with Imagenet
- Scale 2 & 3 random initialization
- Joint train layers 1 & 2 for each task
  - Loss on output of layer 2

- Fix layers 1 & 2, train layer 3

- For depth & normals task, share scale 1
  - But separate scale 2 & 3's
  - 1.6x speedup

Evaluation

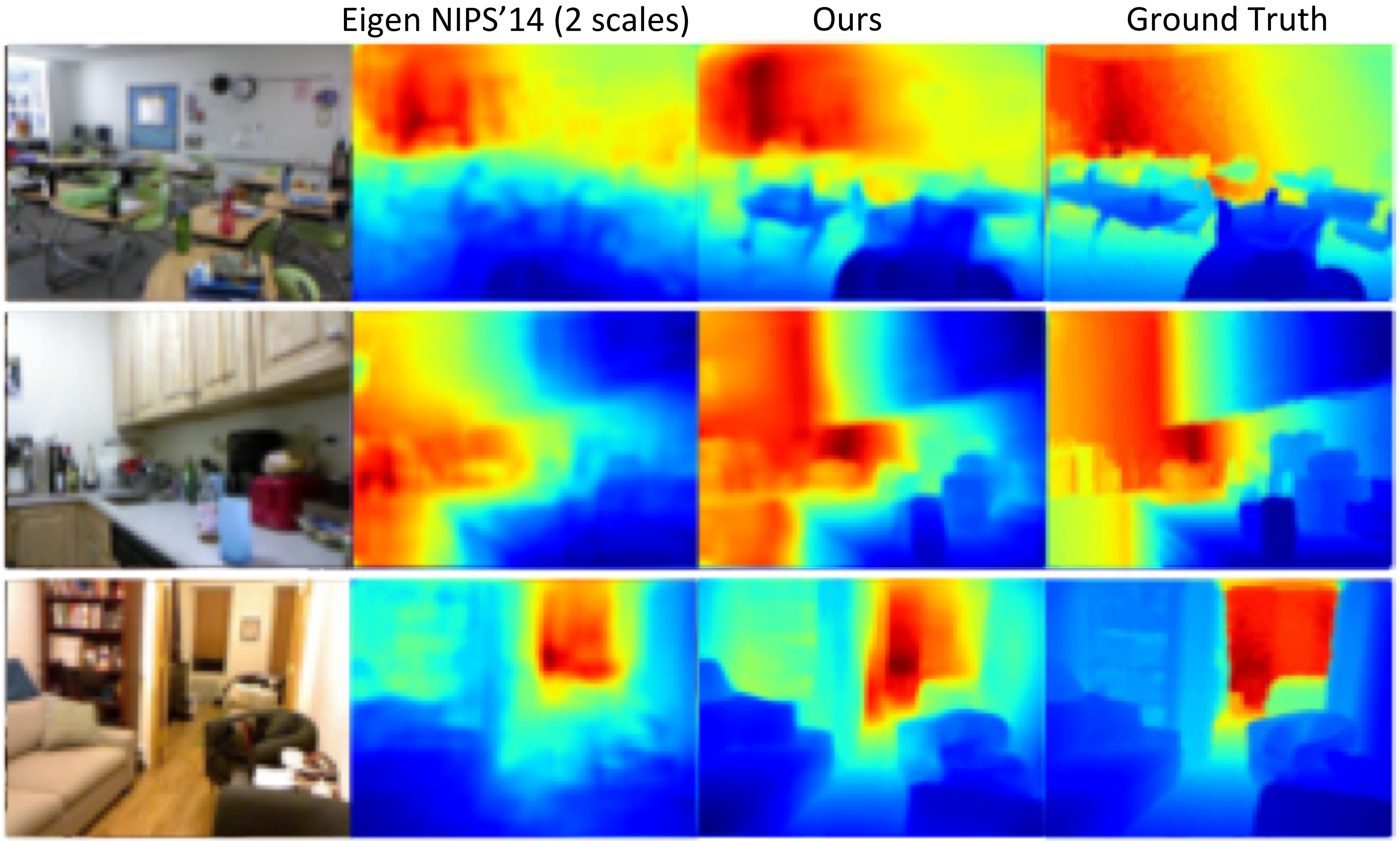
- NYU Depth dataset
  – RGB, Depth
    and per-pixel labels
  – Indoor scenes



- Supervised training of models

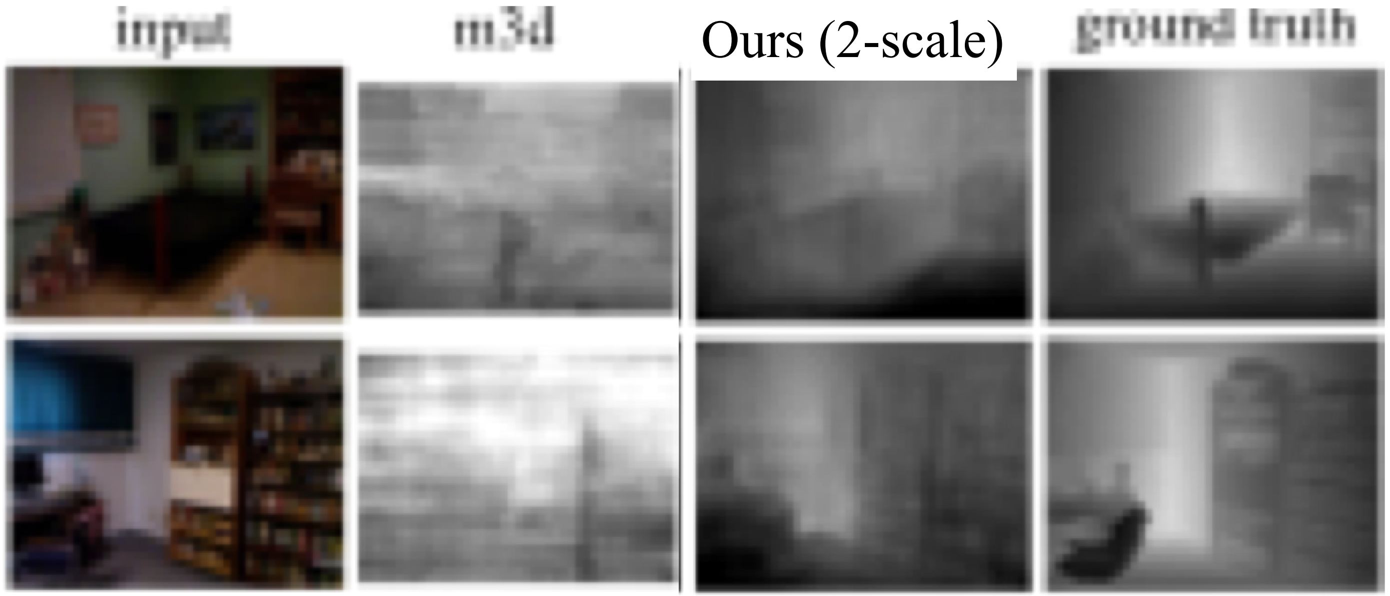
- Compare to range of other methods
  – Also on SIFTFlow and PASCAL VOC'11

Depths Comparison



Depth Comparison

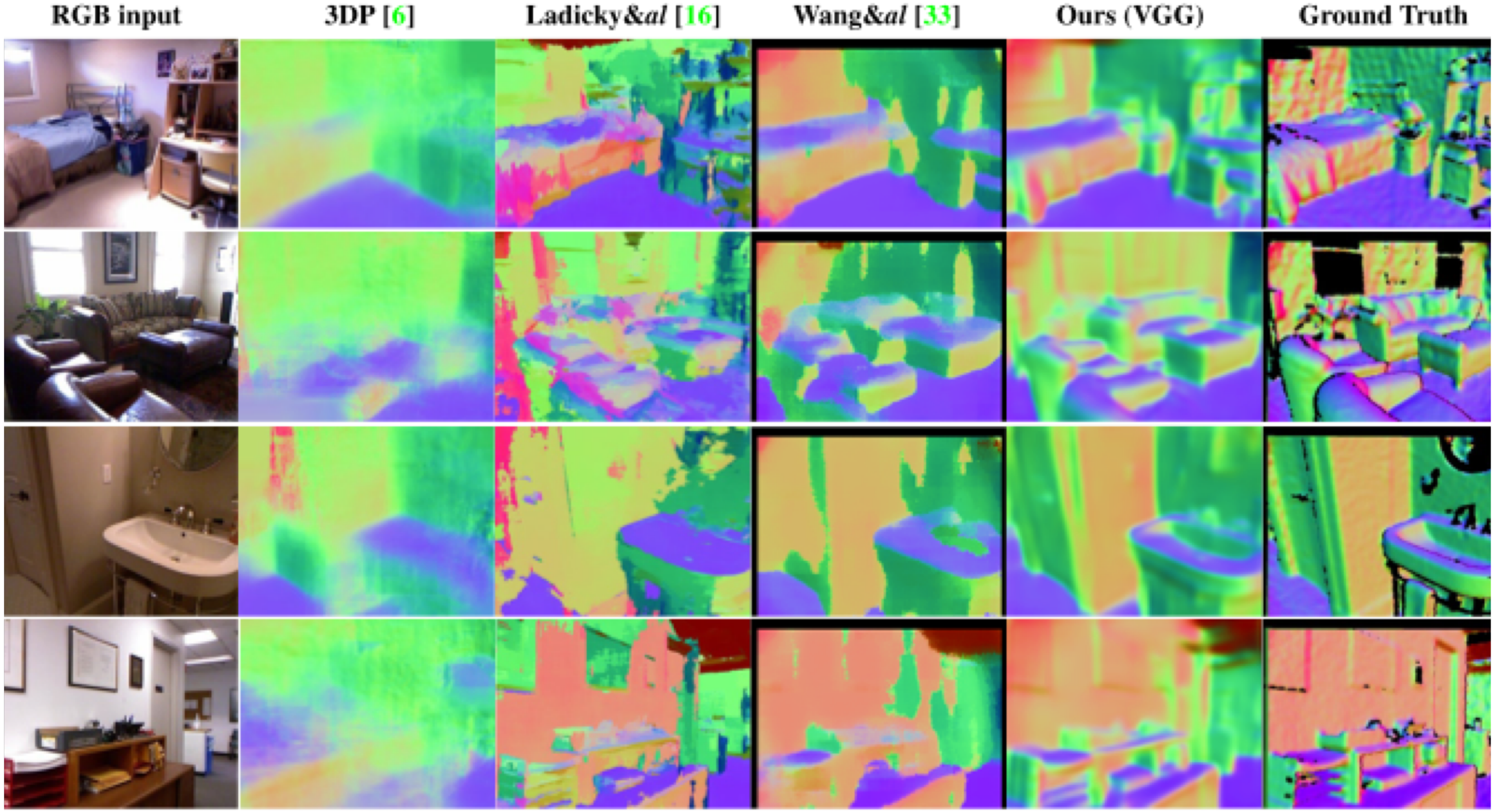
- m3d = Make3D [Saxena & Ng 2006]



Depth Prediction

	Ladicky[20]	Karsch[14]	Baig [1]	Liu [18]	Eigen[4]	Ours(A)	Ours(VGG)
$\delta < 1.25$	0.542	–	0.597	0.614	0.614	0.697	**0.769**
$\delta < 1.25^2$	0.829	–	–	0.883	0.888	0.912	**0.950**
$\delta < 1.25^3$	0.940	–	–	0.971	0.972	0.977	**0.988**
abs rel	–	0.350	0.259	0.230	0.214	0.198	**0.158**
sqr rel	–	–	–	–	0.204	0.180	**0.121**
RMS(lin)	–	1.2	0.839	0.824	0.877	0.753	**0.641**
RMS(log)	–	–	–	–	0.283	0.255	**0.214**
sc-inv.	–	–	0.242	–	0.219	0.202	**0.171**

Surface Normals

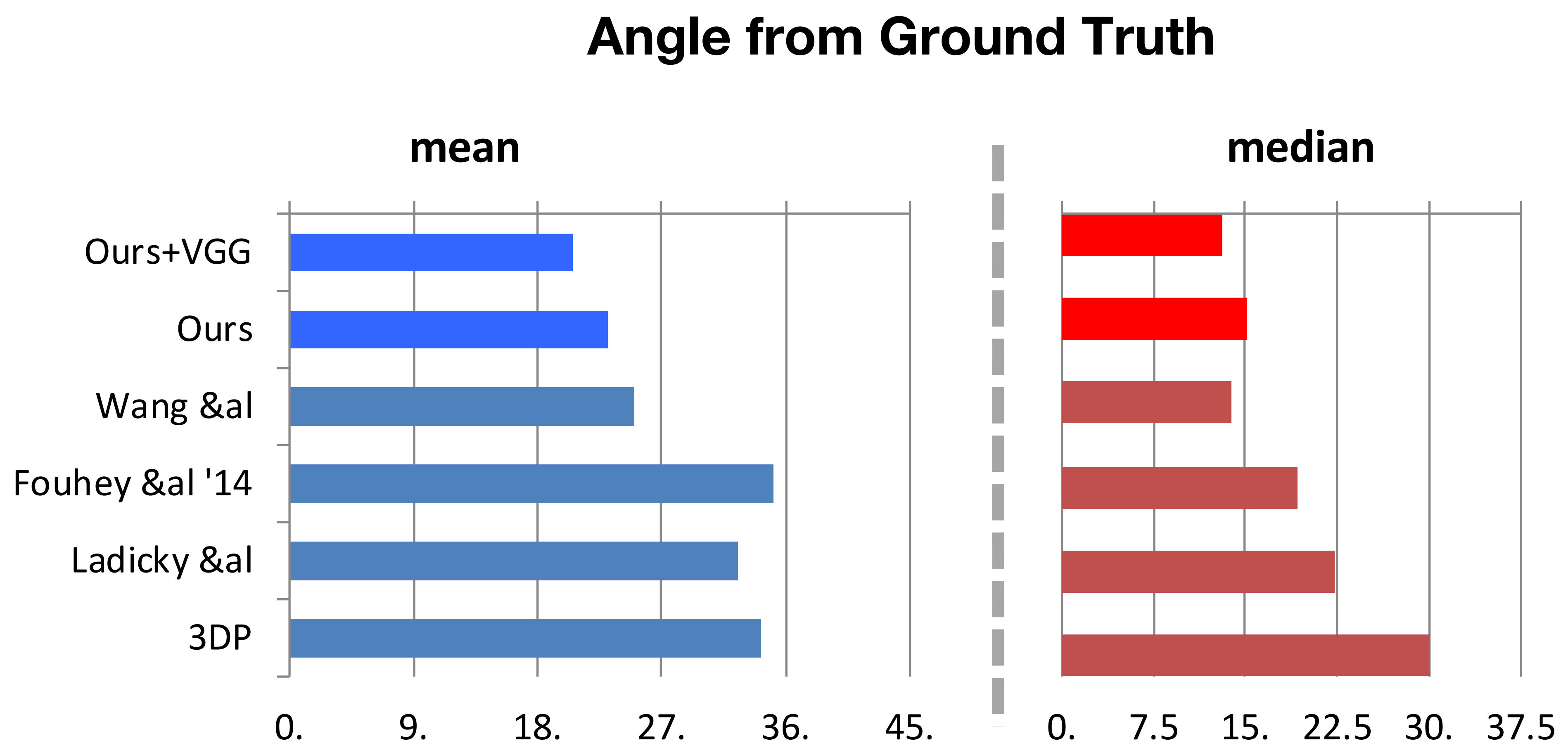


Surface Normals

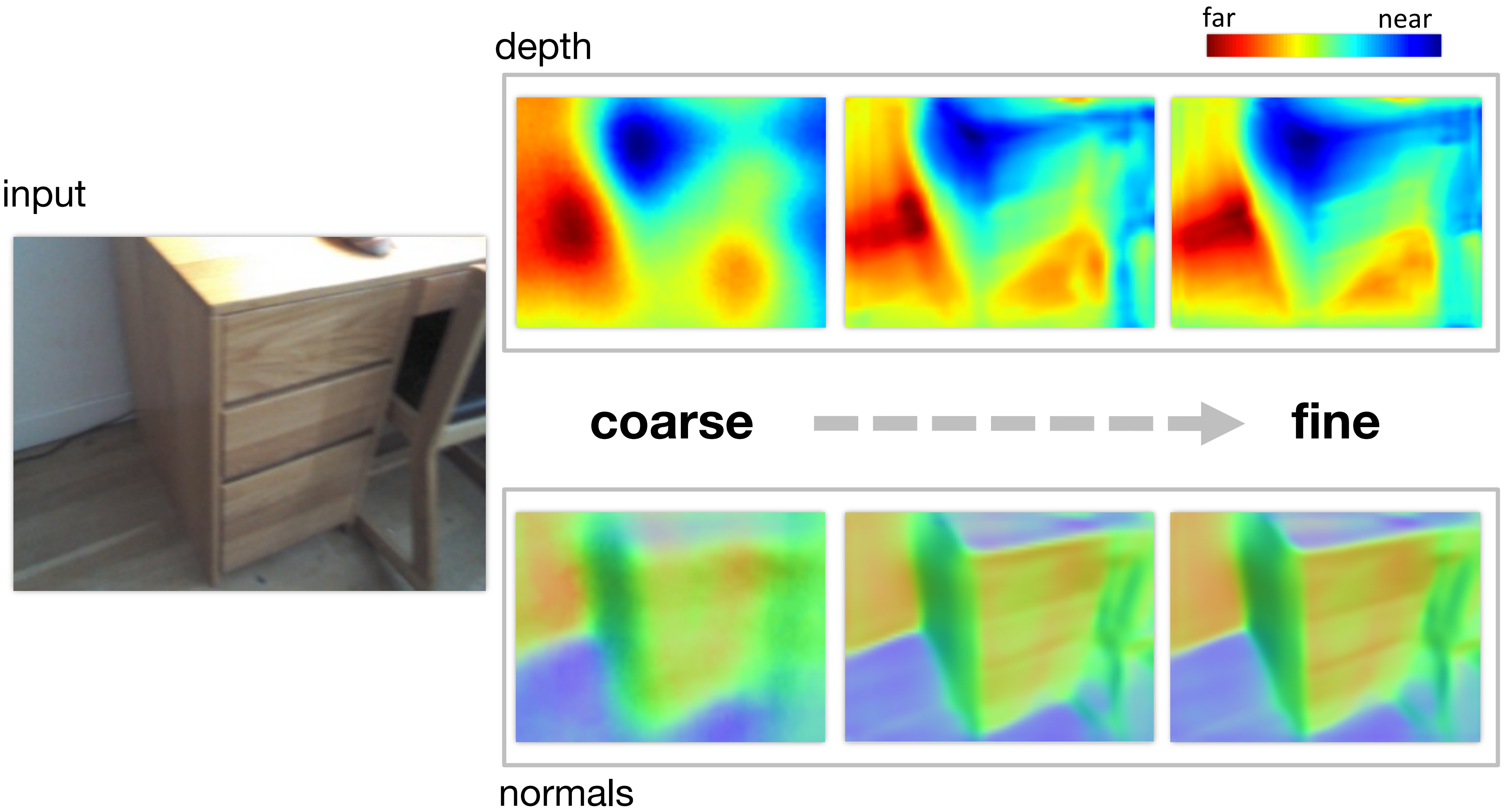
Surface Normal Estimation (GT [6])					
	Angle Distance		Within t° Deg.		
	Mean	**Median**	11.25°	22.5°	30°
3DP [6]	34.2	30.0	18.5	38.6	50.0
Ladicky *&al* [16]	32.5	22.3	27.4	50.2	60.1
Fouhey *&al* [7]	35.1	19.2	37.6	53.3	58.9
Wang *&al* [33]	26.6	15.3	40.1	61.4	69.0
Ours (AlexNet)	23.1	15.1	39.4	63.6	72.7
Ours (VGG)	**20.5**	**13.2**	**44.0**	**68.5**	**77.2**

Surface Normal Estimation (GT [27])					
	Angle Distance		Within t° Deg.		
	Mean	**Median**	11.25°	22.5°	30°
3DP [6]	37.7	34.1	14.0	32.7	44.1
Ladicky *&al* [16]	35.5	25.5	24.0	45.6	55.9
Wang *&al* [33]	28.8	17.9	35.2	57.1	65.5
Ours (AlexNet)	25.9	18.2	33.2	57.5	67.7
Ours (VGG)	**22.2**	**15.3**	**38.6**	**64.0**	**73.9**

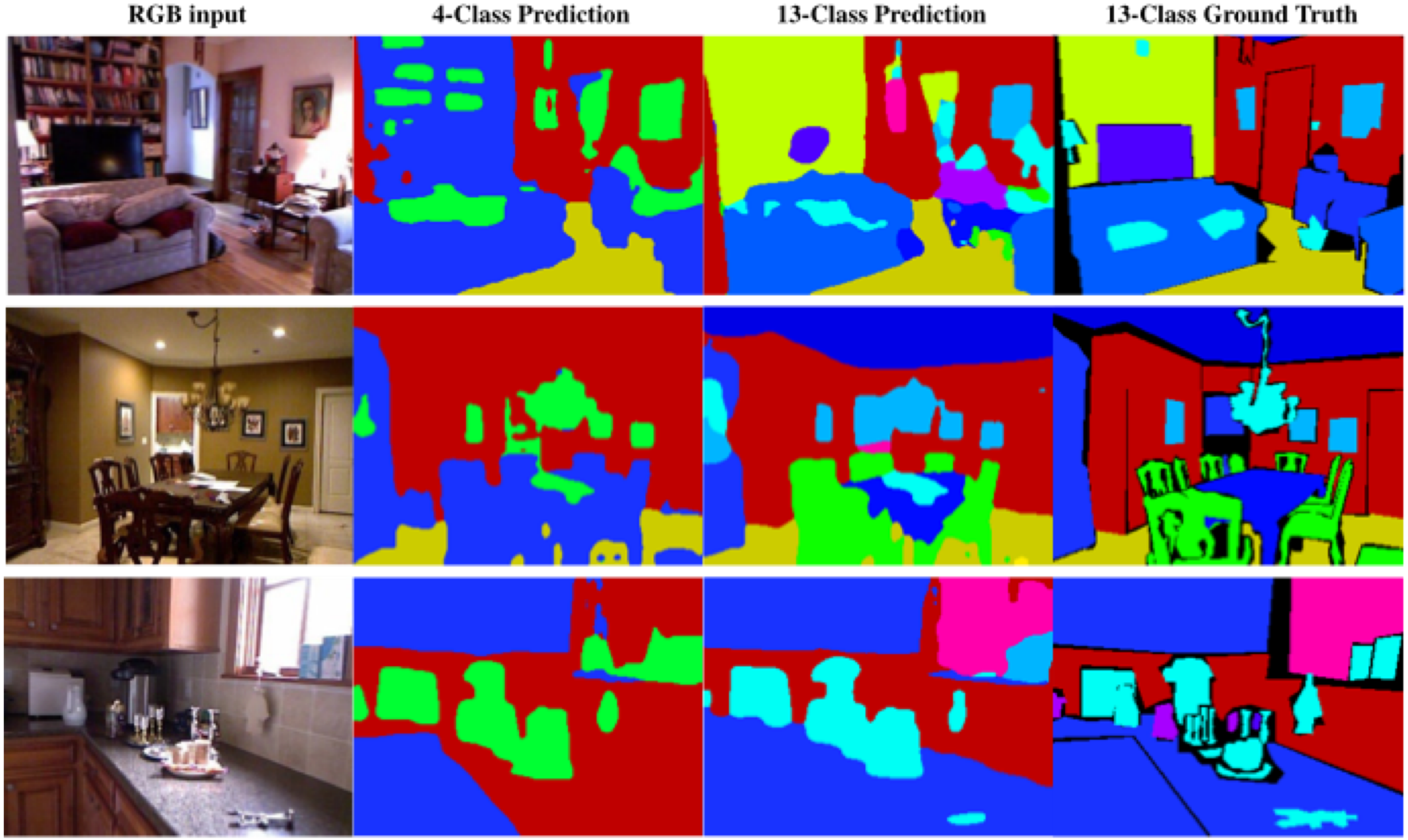
Results: Normals



Output from each scale

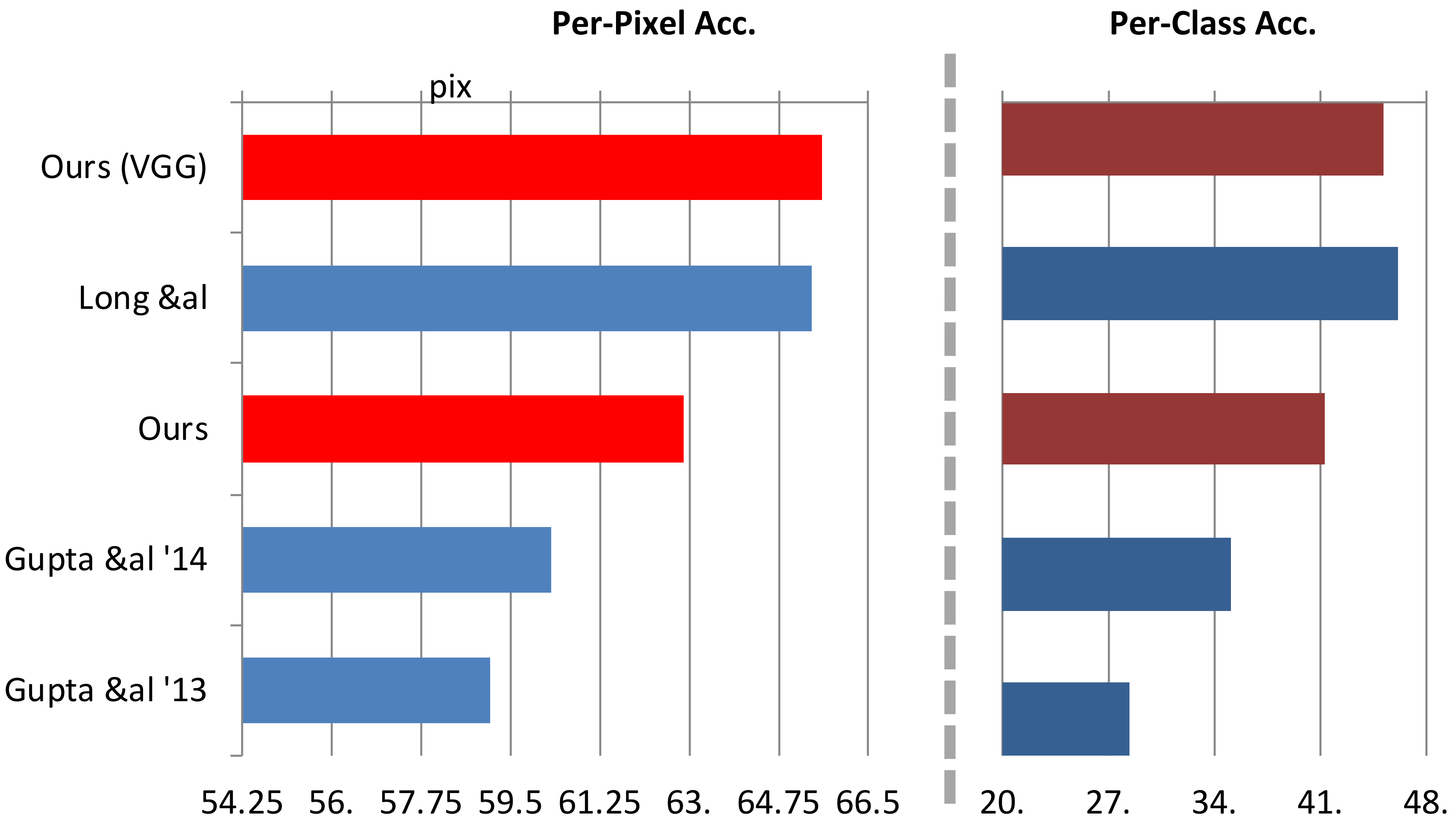


Semantic Labels: NYUD



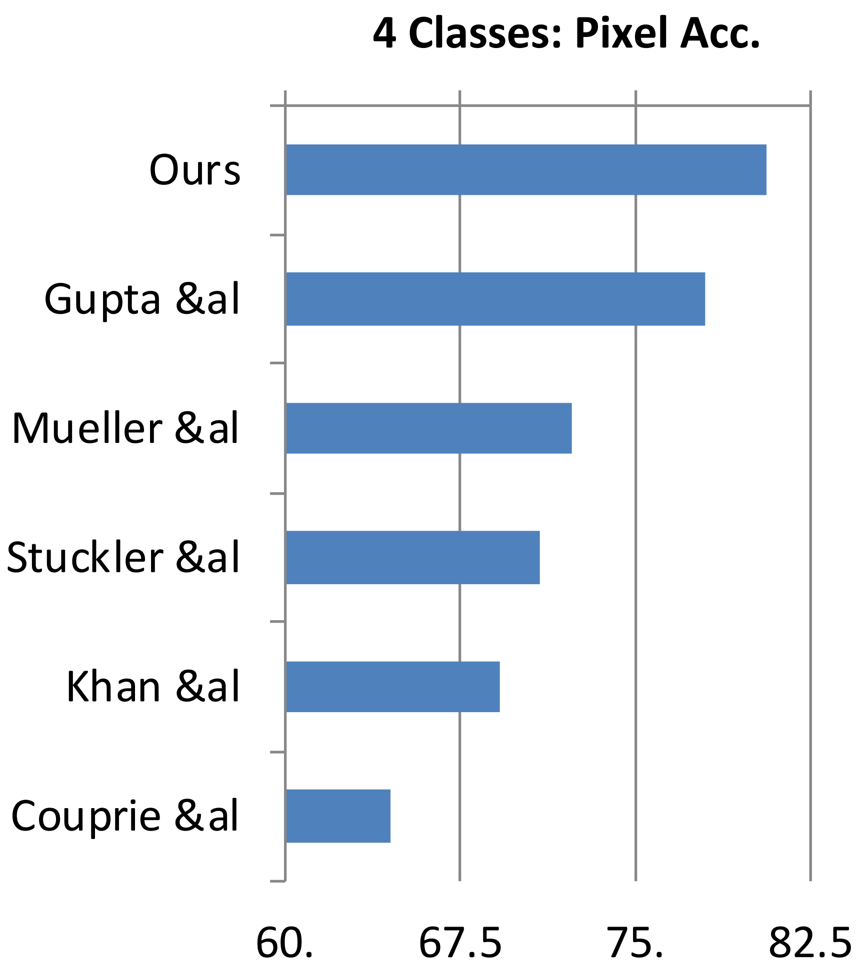
Results: NYUD 40 Classes

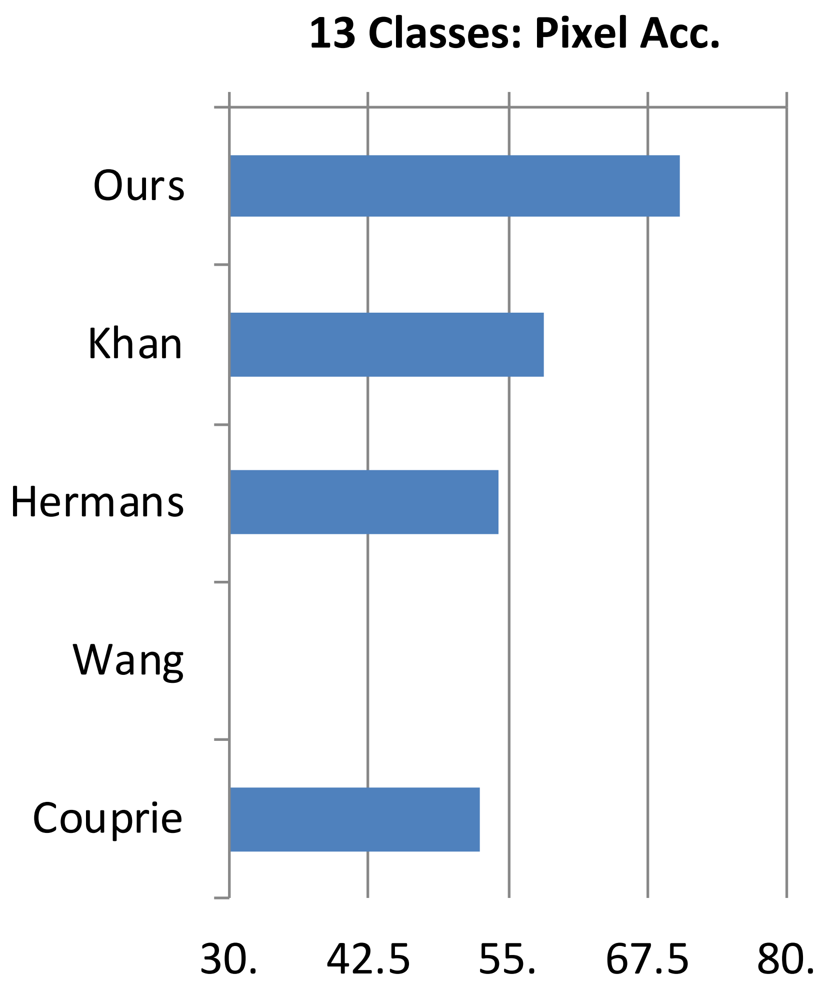
- Use RGB + ground truth depth & normals as inputs



Results: NYUD Labels

- Use RGB + ground truth depth & normals as inputs





Semantic Labels: Pascal VOC'11

Pascal VOC Semantic Segmentation				
	Pix. Acc.	Per-Cls Acc.	Freq. Jaccard	Av. Jaccard
Long&al [19]	**90.3**	**75.9**	**83.2**	**62.7**
Ours (VGG)	**90.3**	72.4	82.9	62.2

Contribution from different scales

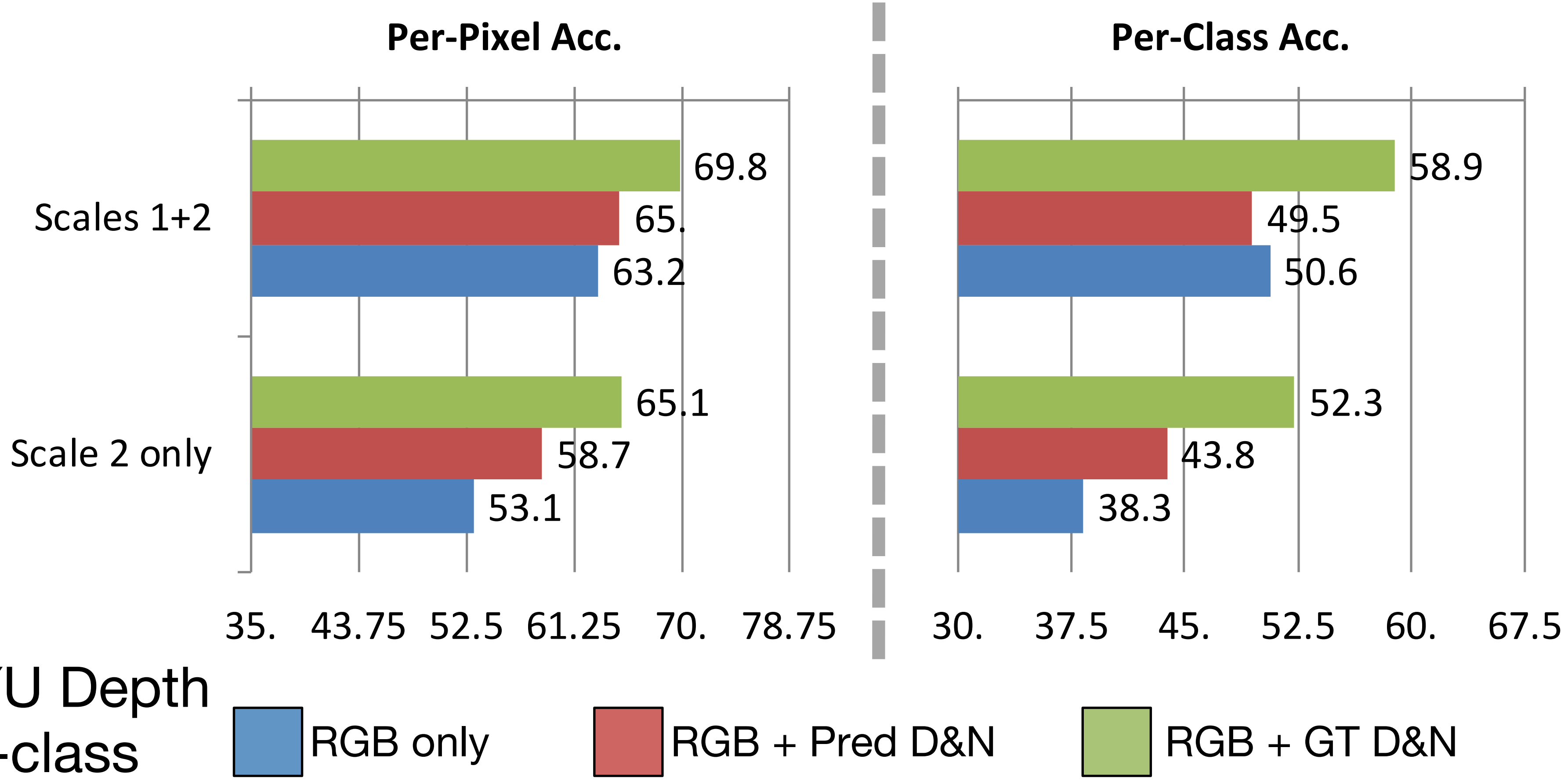
- On NYU Depth

Contributions of Scales						
	Depth	Normals	4-Class		13-Class	
			RGB+D+N	RGB	RGB+D+N	RGB
	Pixelwise Error lower is better		Pixelwise Accuracy higher is better			
Scale 1 only	0.218	29.7	71.5	71.5	58.1	58.1
Scale 2 only	0.290	31.8	77.4	67.2	65.1	53.1
Scales 1 + 2	0.216	26.1	80.1	74.4	69.8	63.2
Scales 1 + 2 + 3	0.198	25.9	80.6	75.3	70.5	64.0

- Depth & normals: scale 1 most important
- Semantic labels: scale 2 most important (if D & N are available)

Using Predicted Depths

- Use predicted depth/normals as input?



- NYU Depth 13-class

Overview

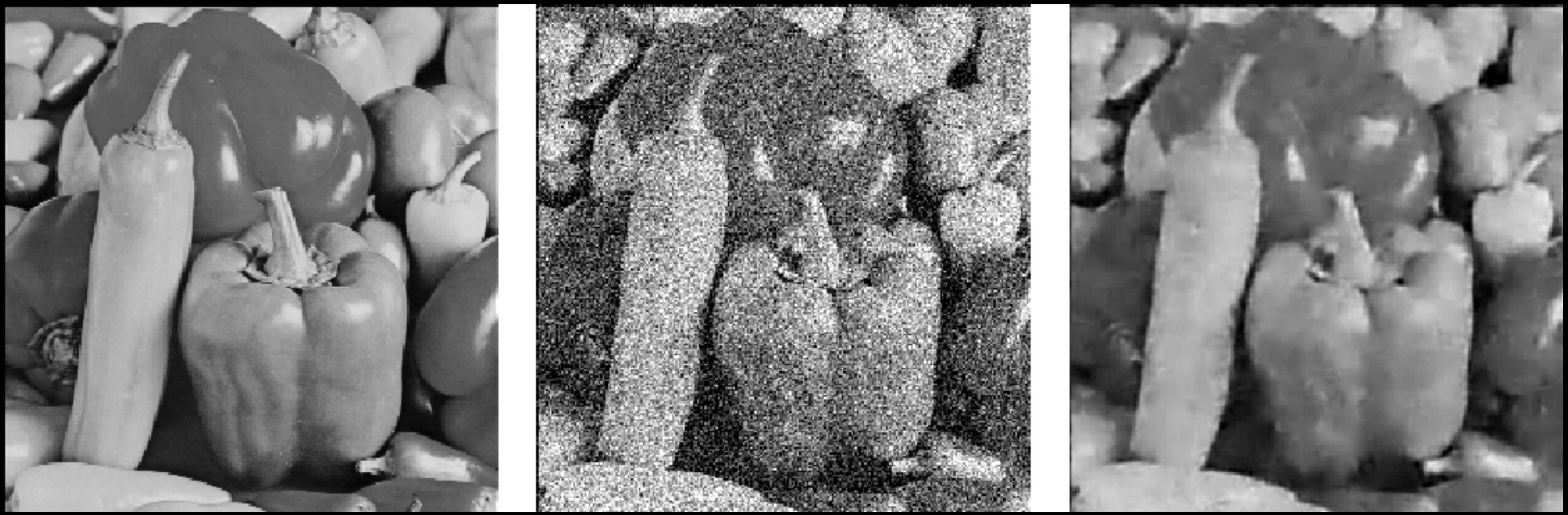
- Methods where output is also an image
  - Fully Convolutional Nets [Long et al., CVPR 2015]
  - Depth, normals and semantic labels from a single image [Eigen ICCV 2015]

- Image processing with Convnets

Denoising with ConvNets

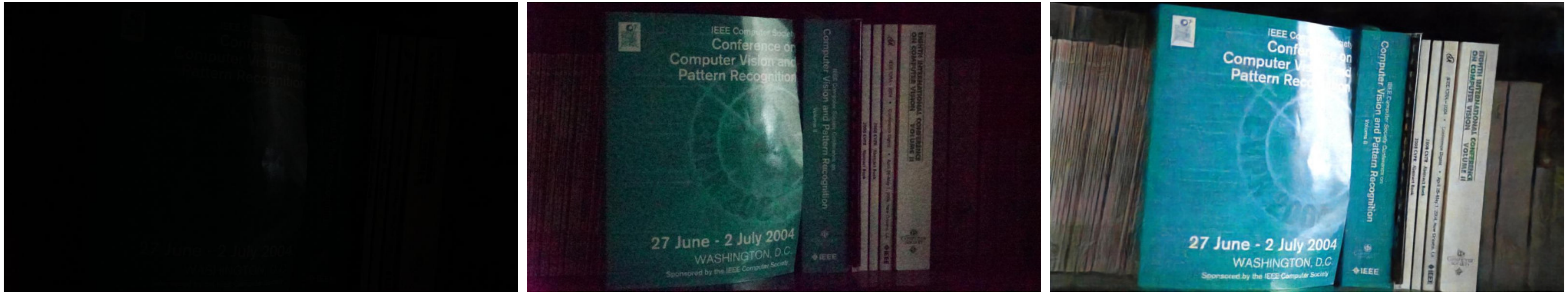
- Burger et al. “Can plain NNs compete with BM3D?” CVPR 2012

Original | Noised | Denoised

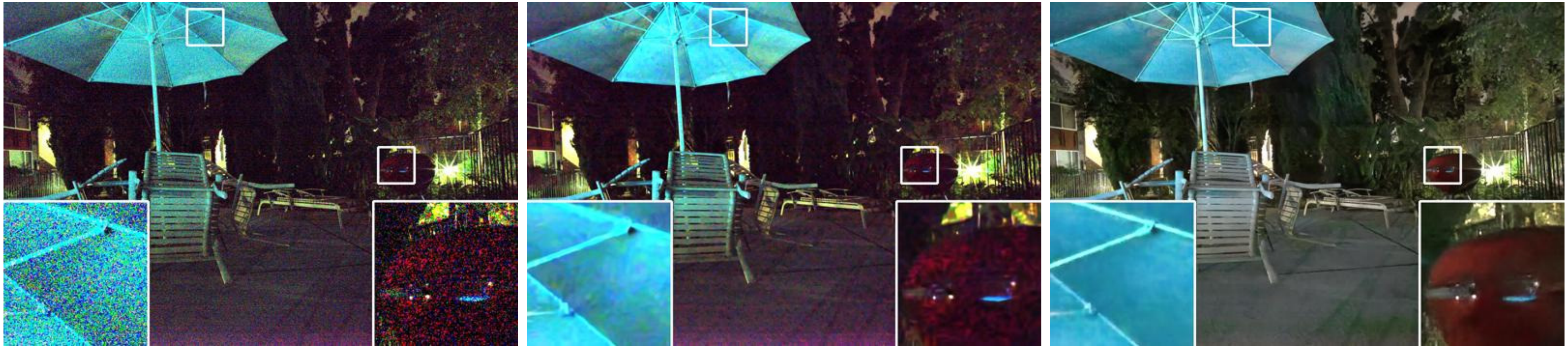


Learning to See in the Dark

[Chen et al., arXiv 1805.01934]



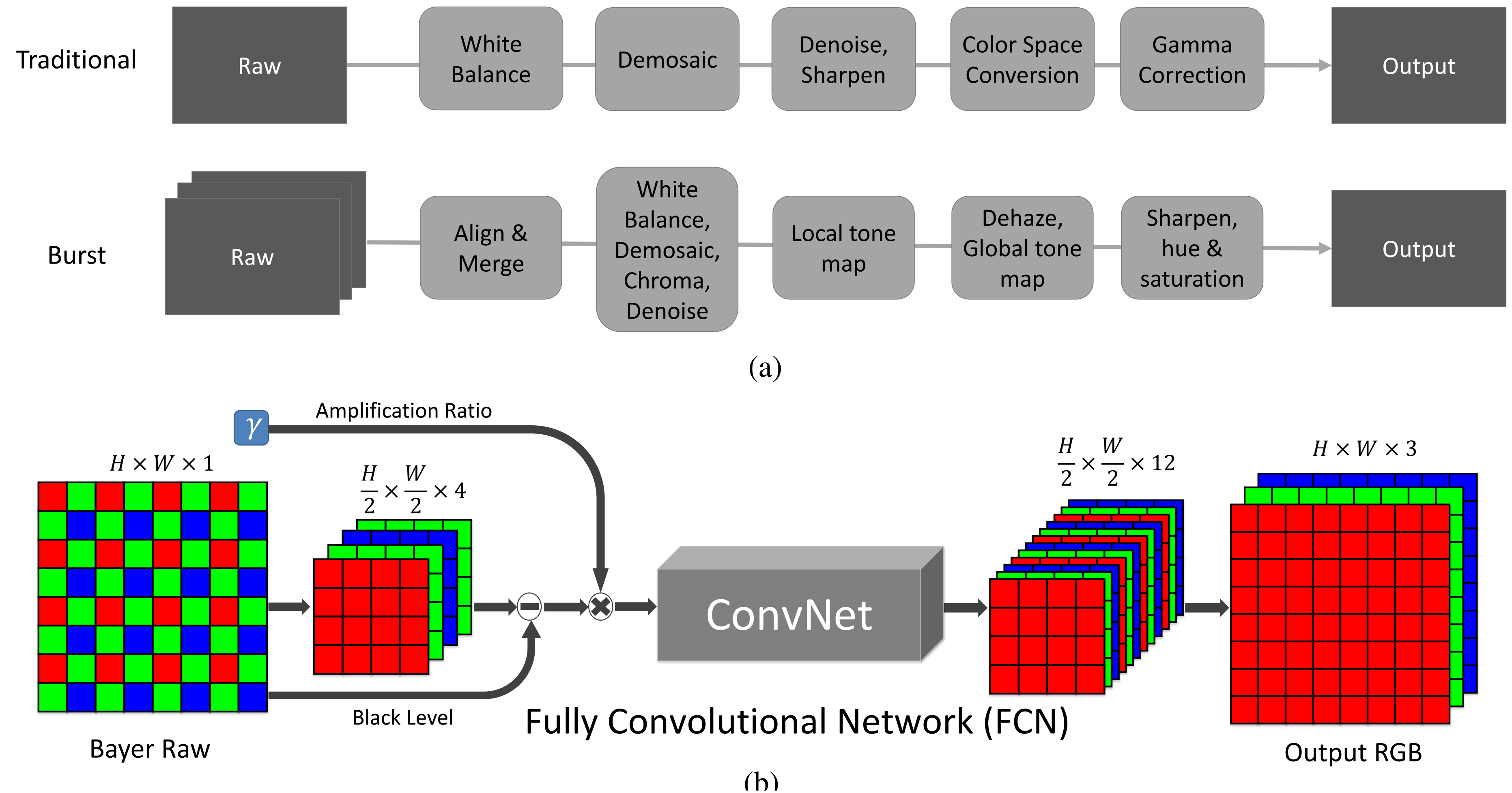
(a) Camera output with ISO 8,000 (b) Camera output with ISO 409,600 (c) Our result from the raw data of (a)



(a) Traditional pipeline (b) ... followed by BM3D denoising (c) Our result

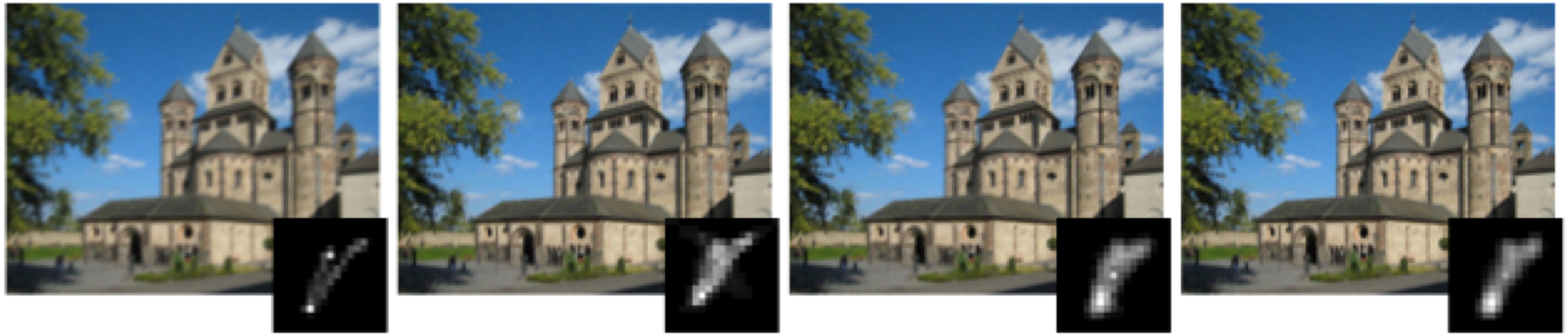
Learning to See in the Dark

[Chen et al., arXiv 1805.01934]



Deblurring with Convnets

- Blind deconvolution
  - Learning to Deblur, Schuler et al., arXiv 1406.7444, 2014



Blurry image with ground truth kernel	Result of [Zho+13] PSNR 23.17	Deblurring result w. noise *agnostic* training PSNR 23.29	Deblurring result w. noise *specific* training **PSNR 23.41**

Inpainting with Convnets

- Image Denoising and Inpainting with Deep Neural Networks, Xie et al. NIPS 2012.
- Mask-specific inpainting with deep neural networks, Köhler et al., Pattern Recognition 2014

Original Schmid CVPR'10 Köhler et al. '14

Removing Local Corruption

- Restoring An Image Taken Through a Window Covered with Dirt or Rain, Eigen et al., ICCV 2013.

Removing Local Corruption

Restoring An Image Taken Through a Window Covered with Dirt or Rain

Rain Sequence

Each frame processed independently

David Eigen, Dilip Krishnan and Rob Fergus

ICCV 2013

Enhanced Deep Residual Networks for Single Image Super-Resolution, Bee Lim Sanghyun Son Heewon Kim Seungjun Nah Kyoung Mu Le, CVPR 2017 workshop

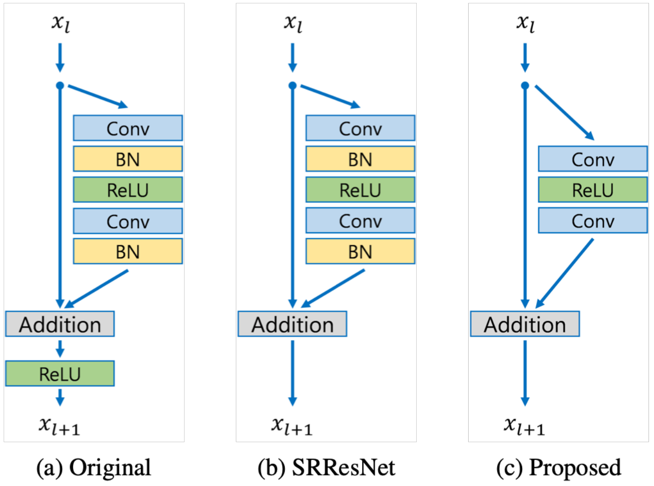


Figure 2: Comparison of residual blocks in original ResNet, SRResNet, and ours.

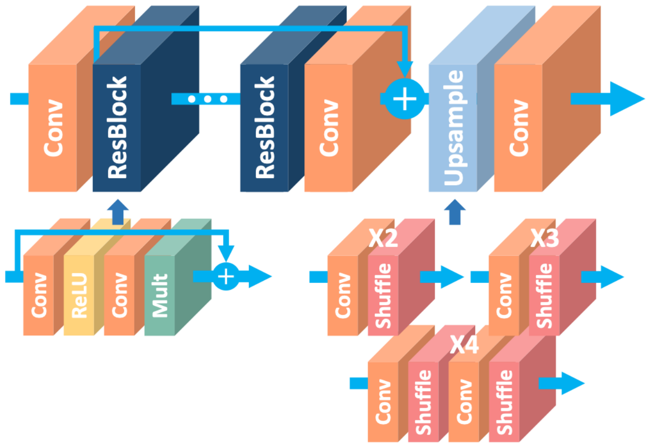
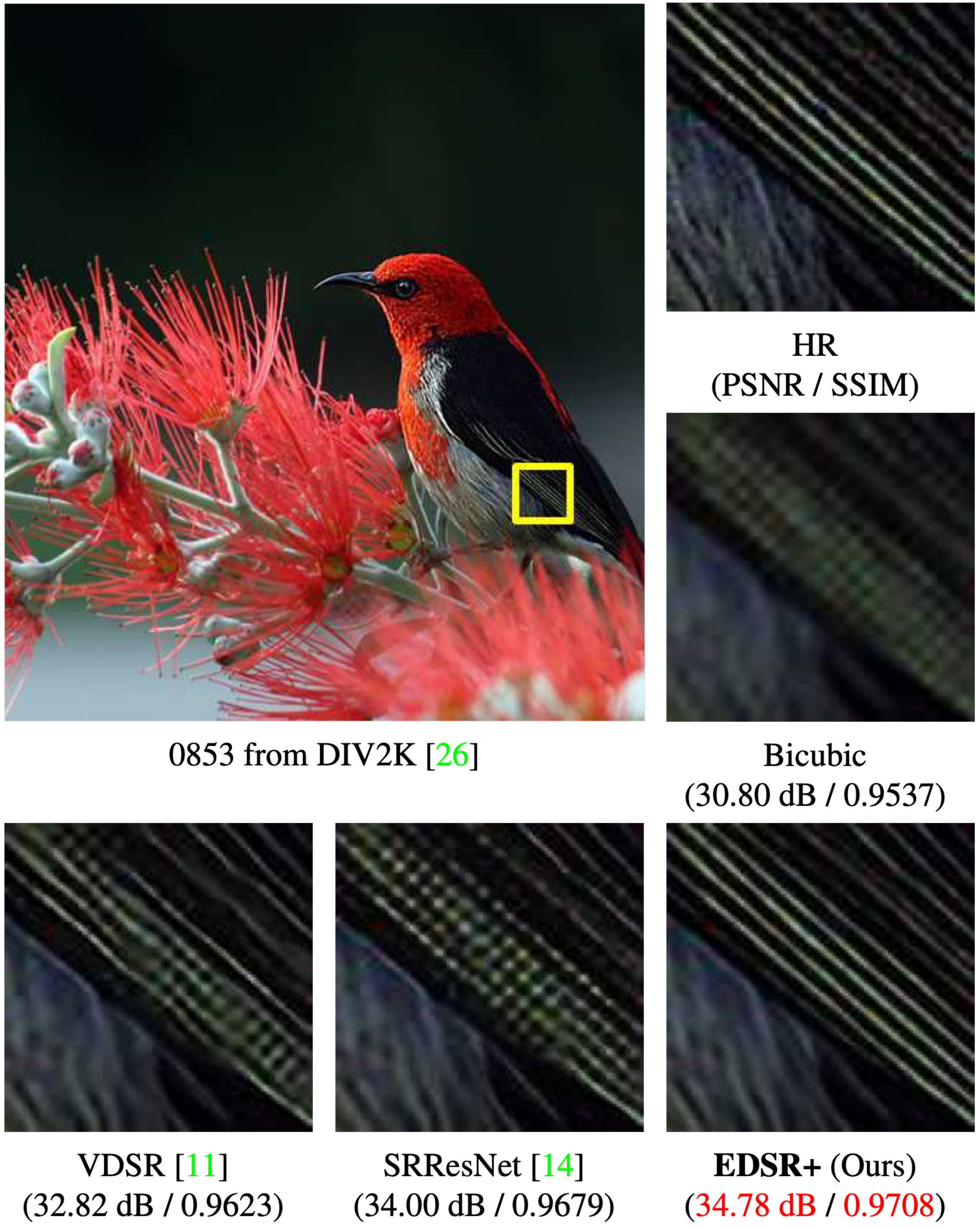


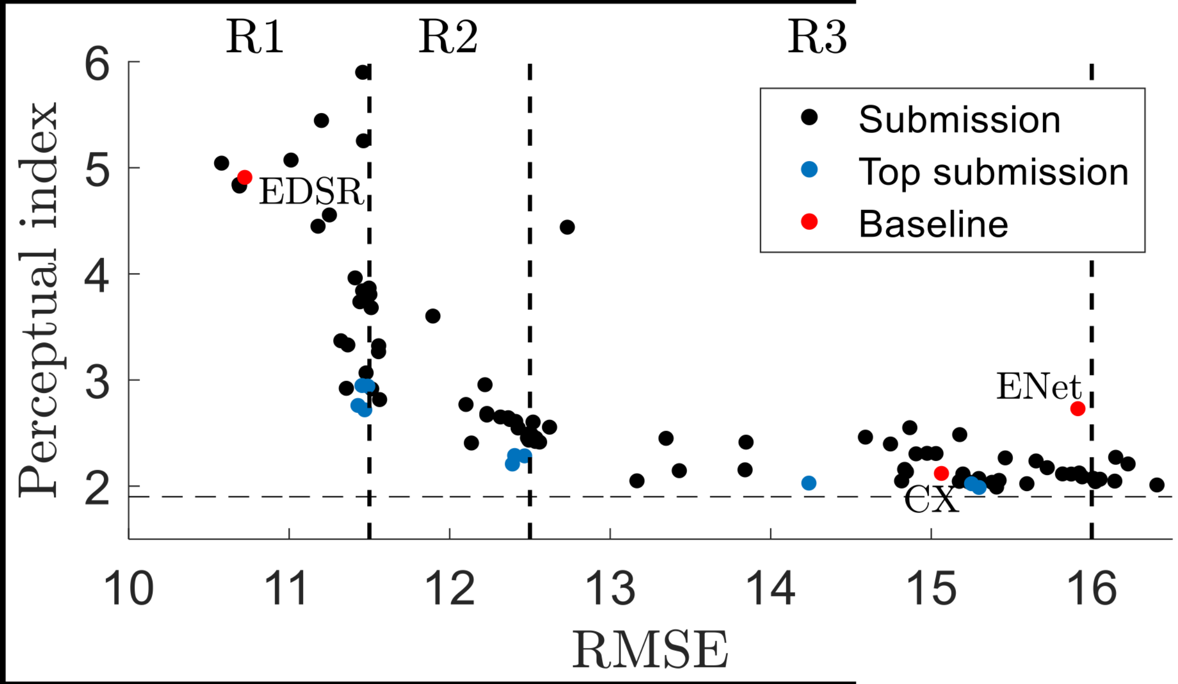
Figure 3: The architecture of the proposed single-scale SR network (EDSR).

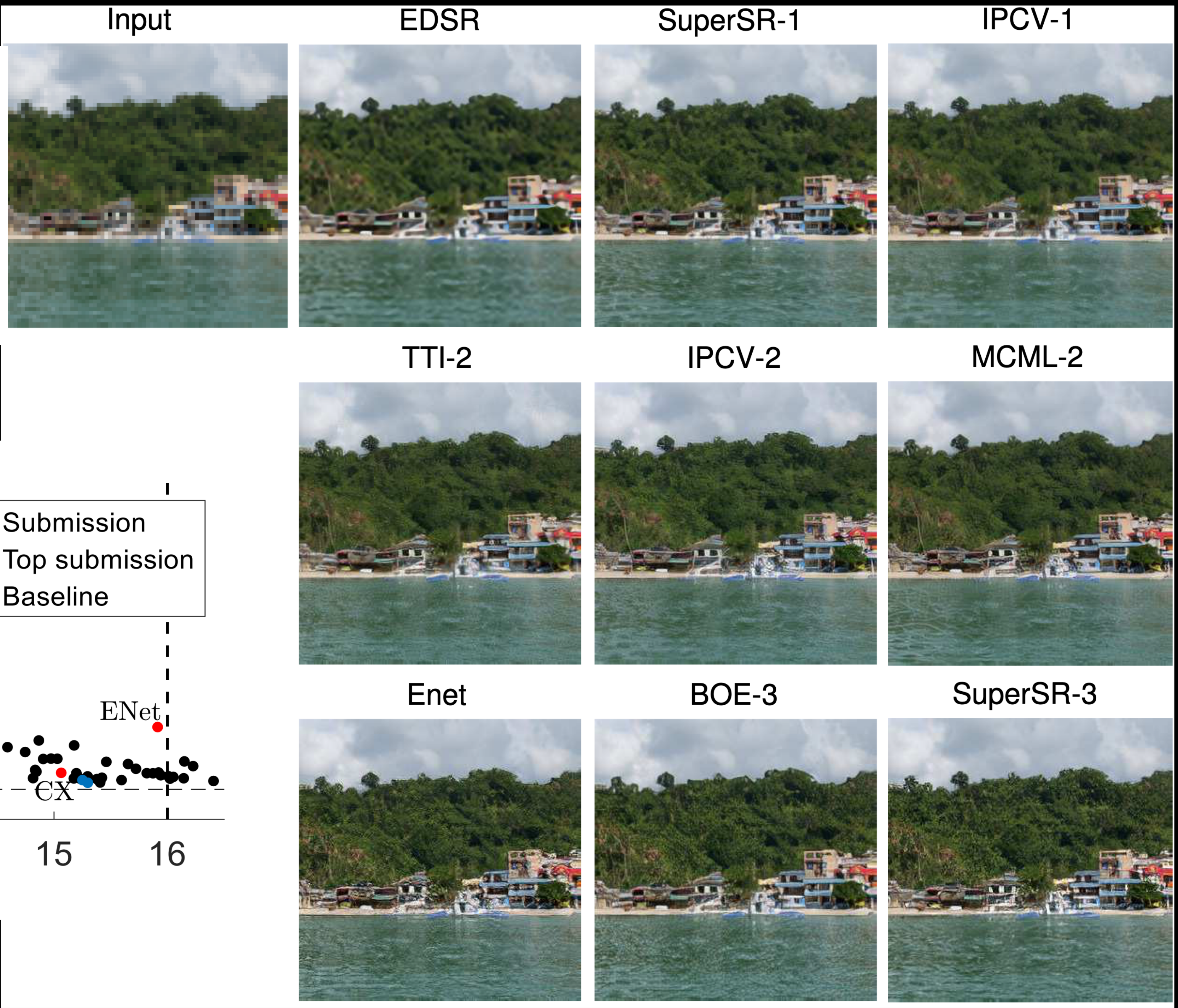


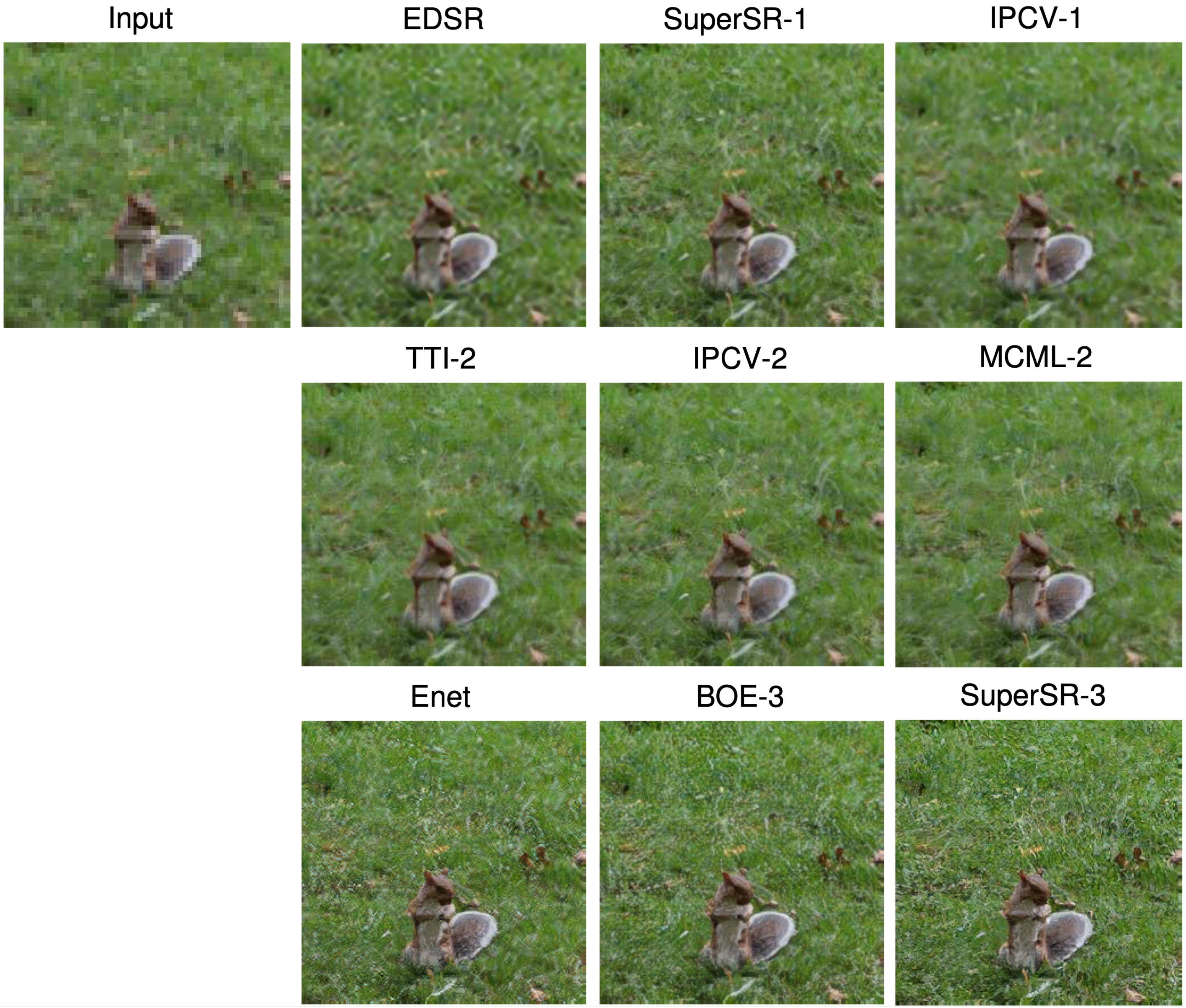
The 2018 PIRM Challenge on Perceptual Image Super-resolution

Yochai Blau[1⋆], Roey Mechrez[1⋆], Radu Timofte[2], Tomer Michaeli[1], and Lihi Zelnik-Manor[1]

[1] Technion–Israel Institute of Technology, Haifa, Israel
[2] ETH Zurich, Switzerland
{yochai,roey}@campus.technion.ac.il






Input
EDSR
SuperSR-1
IPCV-1
TTI-2
IPCV-2
MCML-2
Enet
BOE-3
SuperSR-3

Class Project Admin

- Presentations
- Report
- Deadline is Friday Dec 20th midnight
    - Feel free to turn in earlier
    - Will *try* to grade them and compute final grades by Christmas
- Will post all of this to Piazza

Presentation session

- Thursday, December 19th at 7:00-9:00 pm (405 Silver).

- 2 slides presentation on your project
  – Submit slides beforehand
  – Strict timing (to fit in 2hrs!)
  – Will be part of grading

- Pizza & drinks will be served!

Project Expectations

- Grading (45% of total grade for class)
  - Novelty / Technical difficulty of problem [15%]
  - Quality of Results [15%]
  - Quality of implementation [5%]
  - Quality of writeup [5%]
  - Presentation [5%]
  - How many people in your group

Project Expectations

- Report
  - 4-8 page conference paper style report on your project
  - Intro (with refs to related work)
  - Method (be sure to cite any code/pre-trained models)
  - Experiments (must have plots/results figures; also should have baselines; ideally some kind of ablation experiments too)
  - Discuss (brief)
  - See examples: http://openaccess.thecvf.com/CVPR2018.py
- Zip of source code or link to Github (please ensure you give access to robfergus)
- For presentations:
  - 2 (two) PPT slides only. Will not show more slides.

Project Expectations

- Generalities
  - Please make sure you have *something* working, even if you don’t achieve overall goal
  - Even a small part of an ambitious project can be OK
  - So please have a safe plan B option in mind
  - Expect all projects to train something, i.e. must use b-prop at some point
  - Just evaluating existing models is NOT OK.
  - Cluster gets busy -- please don’t leave it all to last moment.