Further applications of Convnets

Lecture 11

Overview

- Body pose tracking
  - Combine Convnet with graphical model [Thompson et al. NIPS 2014]

- Methods for semantic segmentation of scene
  - Output is now 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
  - Image colorization [Zhang et al. ECCV 2016]

Overview

- Body pose tracking
  - Combine Convnet with graphical model [Thompson et al. NIPS 2014]

- Methods for semantic segmentation of scene
  - Output is now 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
  - Image colorization [Zhang et al. ECCV 2016]

BODY TRACKING

J. Tompson, A. Jain, Y. LeCun, C. Bregler, “Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation”, NIPS 2014

Human Body Pose Inference

Motivation:

- From my hand tracking work: do similar techniques work in RGB?
- How about full-body tracking?

Problem definition:

- Track human body joints on a monocular RGB image
- Arbitrary pose and background

Very long history of prior work

- Analysis-by-synthesis models ~ 1990s - 2000s
- DPM models (graphical models) ~ 2000s - 2014
- Particle Filtering Methods ~ 2000s
- Microsoft Kinect - 2008
- Convolutional Networks - 2013 - 2014

Some Related Work

Andriluka et al. (CVPR 2009)

Pictorial Structures Revisited: People Detection and Articulated Pose Estimation

Shape context descriptors trained using ADABOOST + generative model and belief propagation

Felzenszwalb and Sirshick (PAMI 2010)

Object Detection with Discriminatively Trained Part-Based Models

Deformable Part Model (DPM) → "structure as latent variable"

Sapp and Taskar (CVPR 2013)

Modec: Multimodal decomposable models for human pose estimation

HOG + Graphical Model (and FLIC dataset)

Jain et al. (including me) (ICLR 2014)

Learning Human Pose Estimation Features with Convolutional Networks

Simple ConvNet with simple MRF spatial model

Toshev & Szegedy (CVPR 2014)

DeepPose: Human pose estimation via deep neural networks

Large ConvNet cascade to perform direct regression on UV locations

More in the paper (and more recent)…

(Pishchulin, Andriluka, LeCun, Johnson, Chen, …)

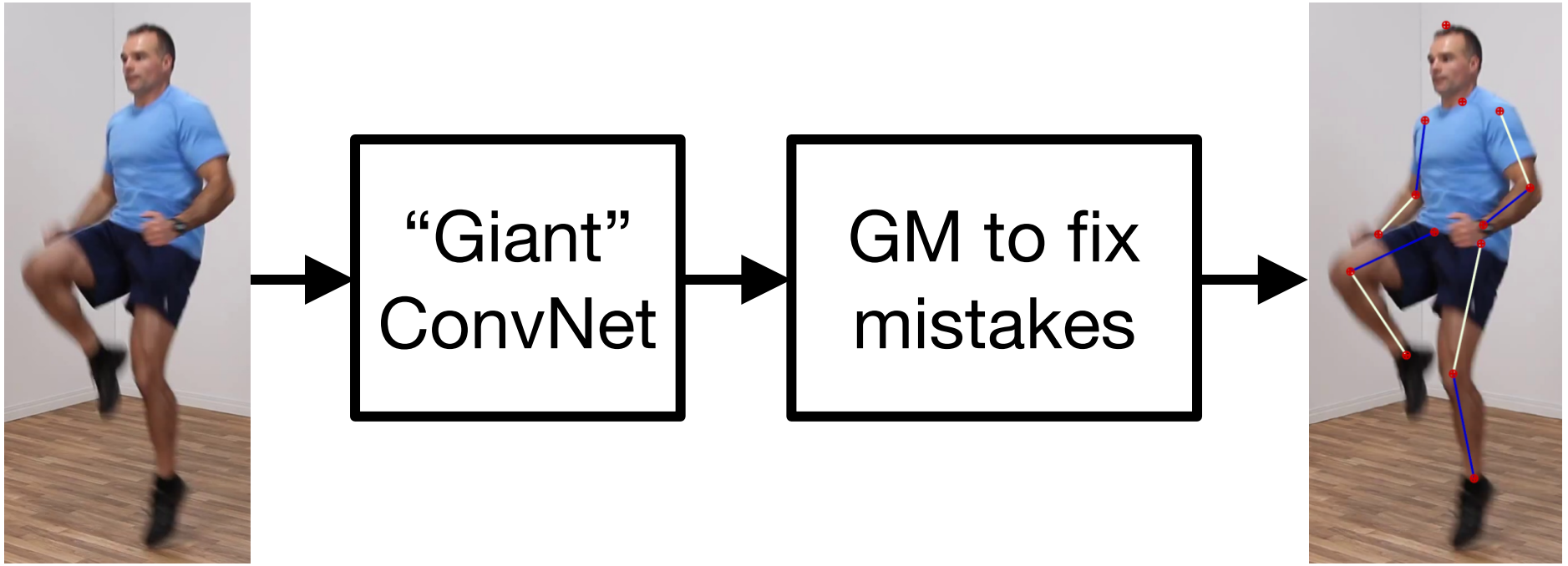
My Basic Idea

Two Parts:

ConvNet to track joints

Graphical Model to stitch them together

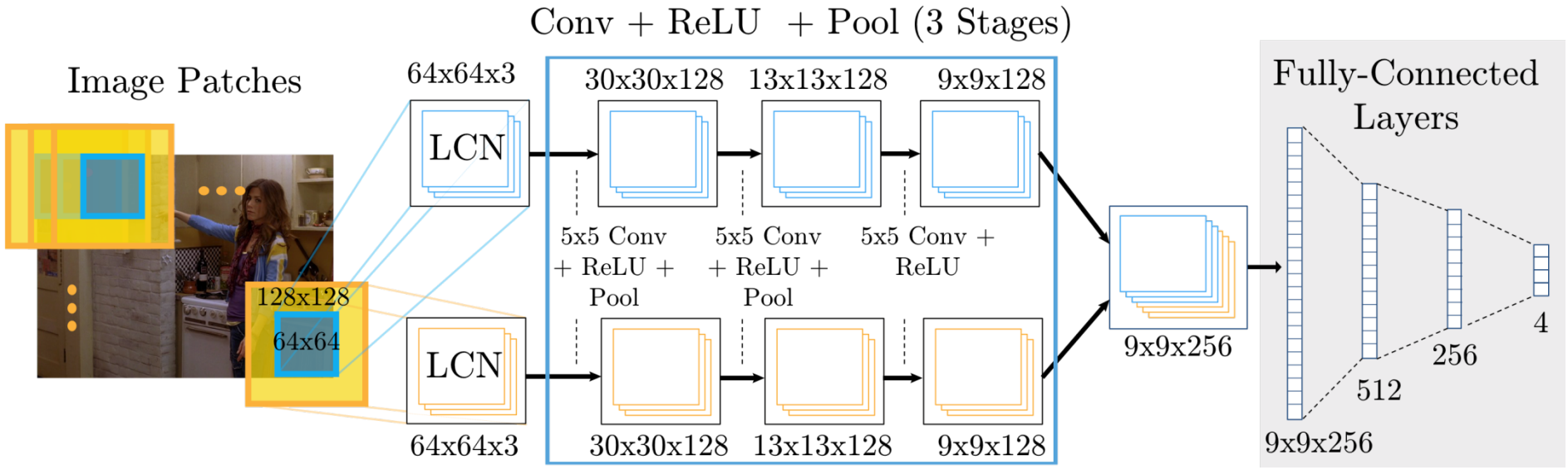
JOINTLY TRAIN THEM!



Part Detector

ConvNet Architecture:

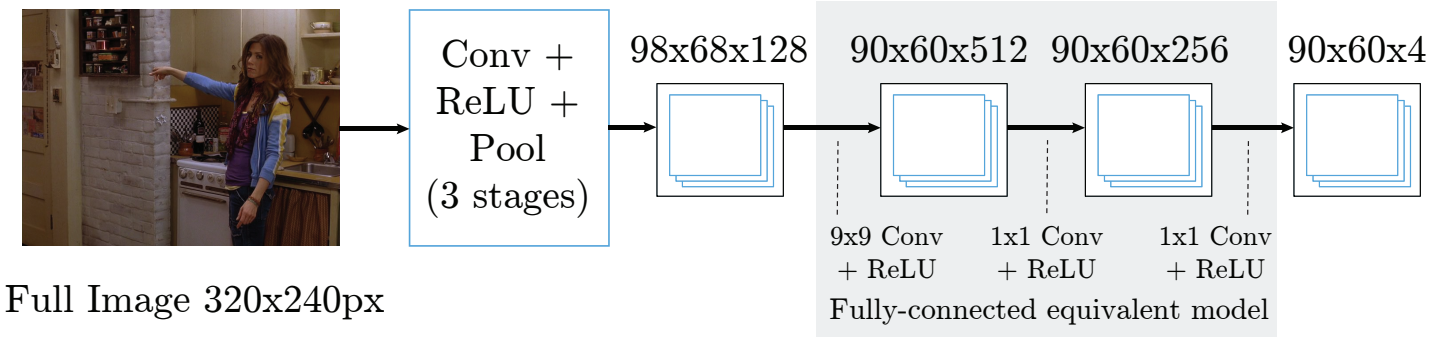
Multi-resolution sliding window with overlapping receptive fields



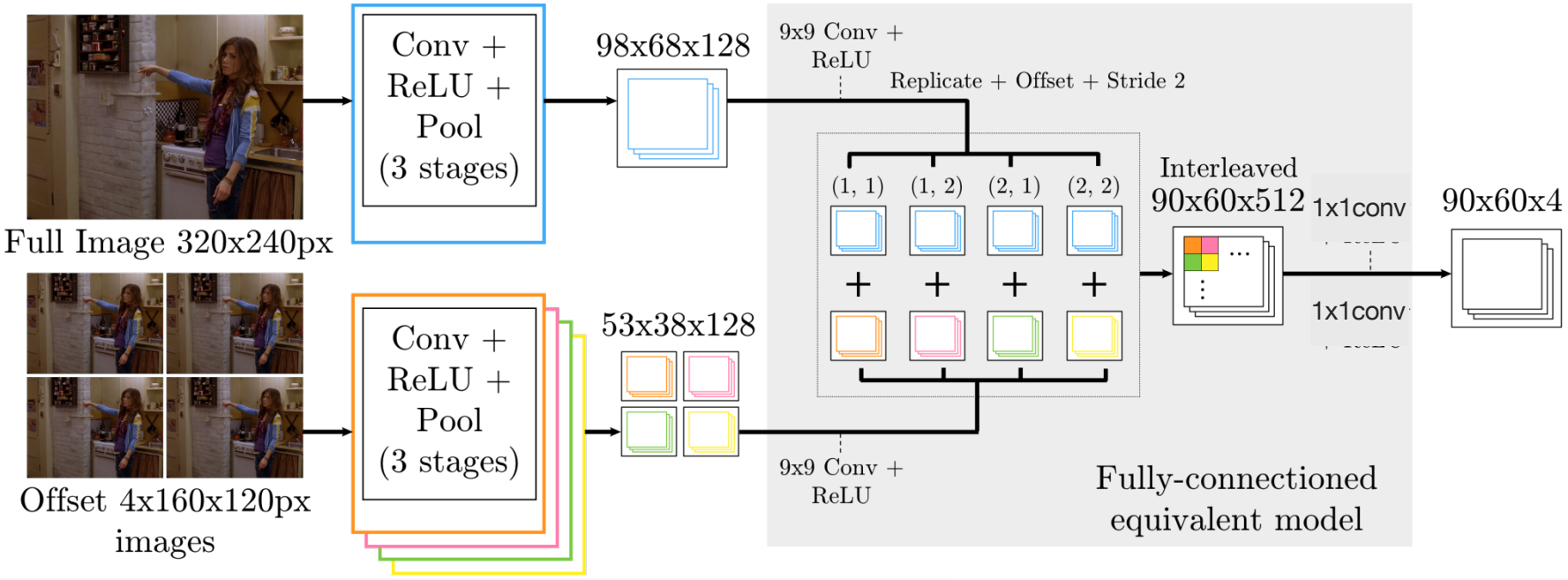
NYU

Part Detector

Efficient model (P. Sermanet and others):



Multi-resolution efficient model (my work):



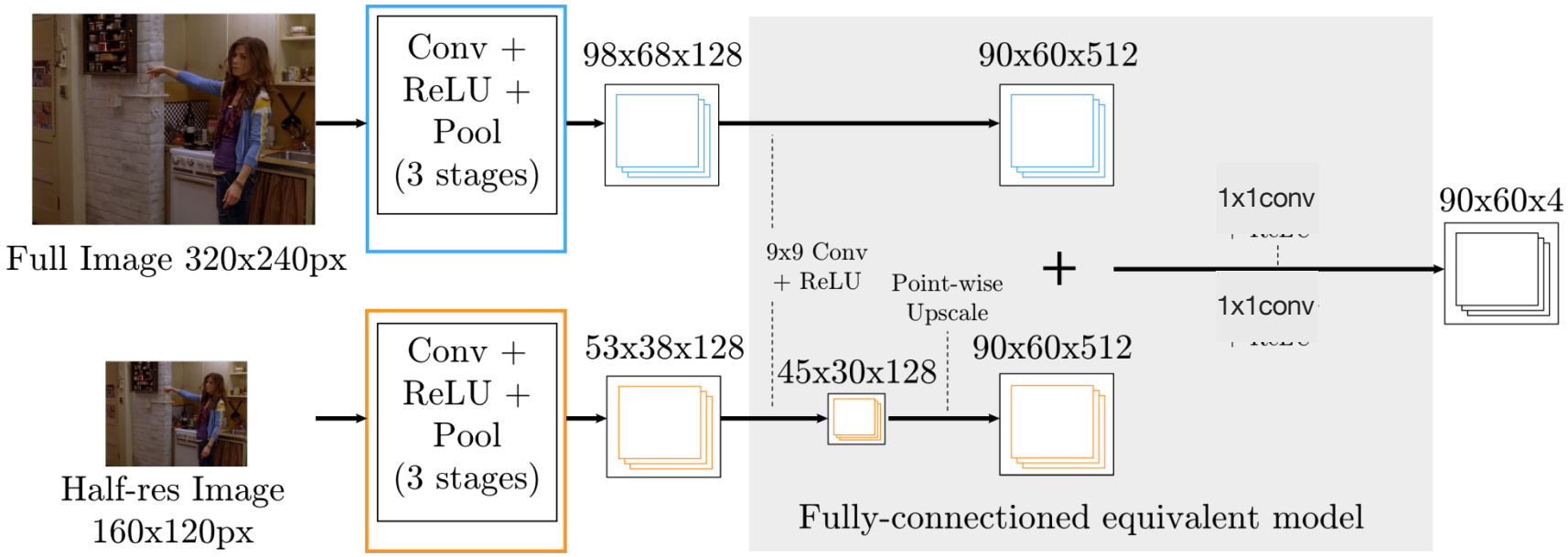
NYU

Part Detector

Simplified model:

Performance is close to the “full” model

Use this for real-time demo



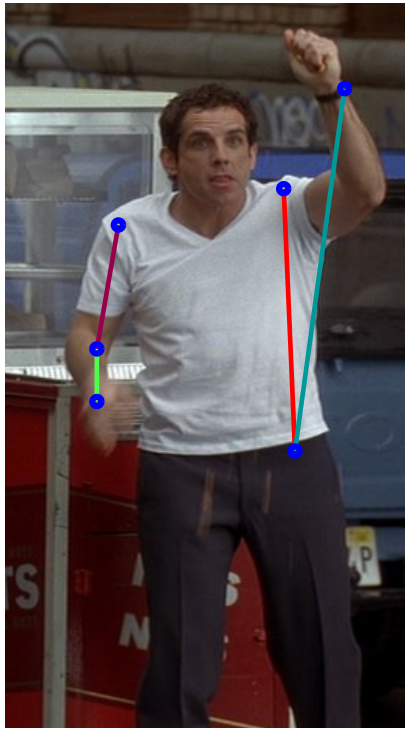
Part Detector Results

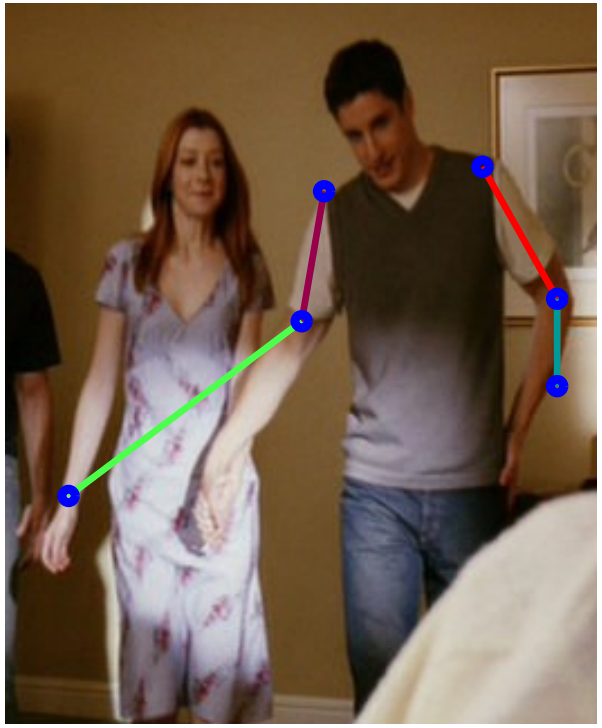
What's wrong so far

Independent joint terms in objective function

We're hoping the network implicitly learns joint consistency

Failure cases are usually stupid:

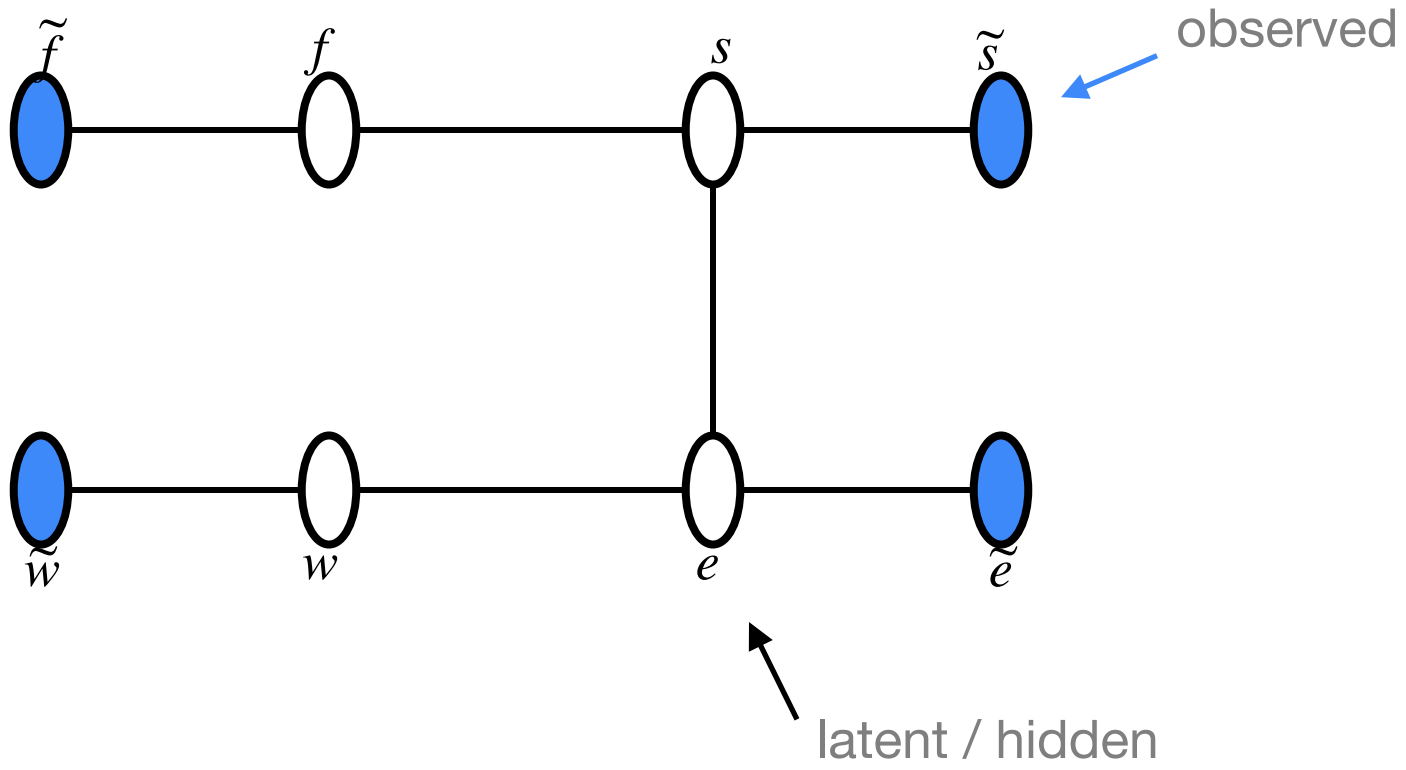




SPATIAL MODEL

Start with my GM from ICLR 2013 paper

Use **MRF** over spatial locations

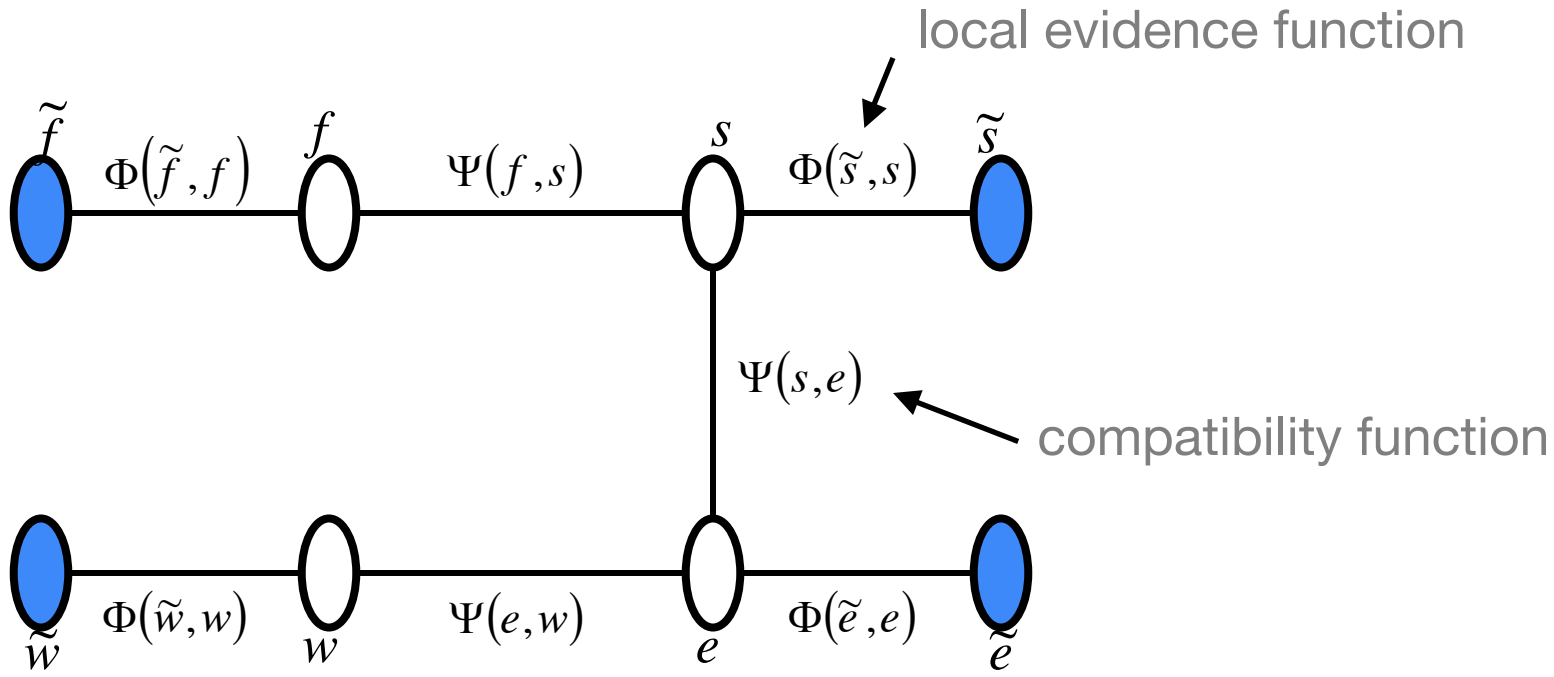


NYU

SPATIAL MODEL

Start with my GM from ICLR 2013 paper

Use **MRF** over spatial locations



Joint Distribution:

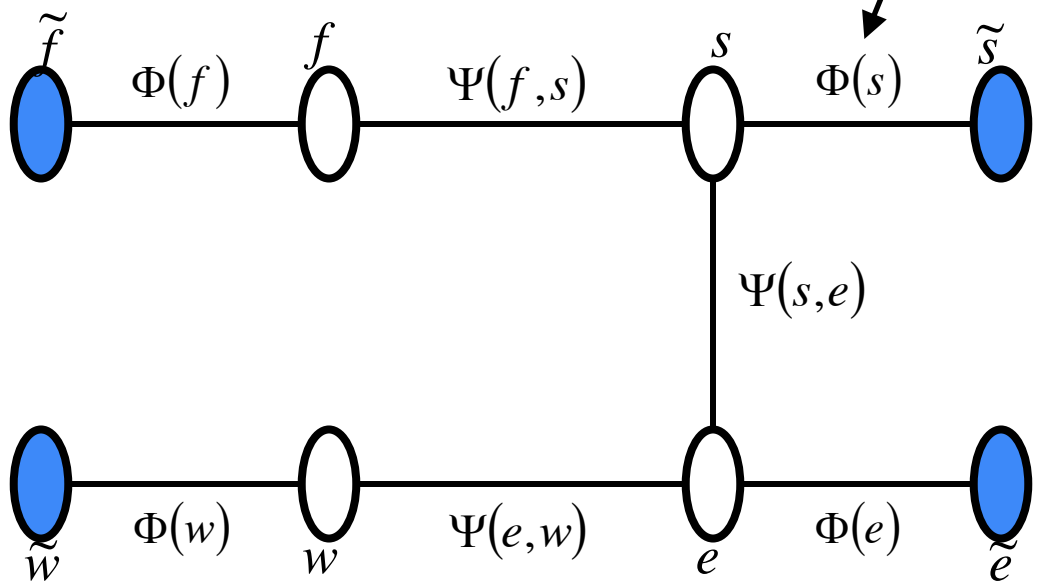
$$P(f,s,e,w)=\frac{1}{Z}\prod_{i,j}\Psi(x_i,x_j)\prod_i\Phi(x_i,\tilde{x}_i)$$

Spatial Model

Start with my GM from ICLR 2013 paper

Use **MRF** over spatial locations

Replace with "unary" factors (just short-hand)



Joint Distribution:

$$P(f,s,e,w) = \frac{1}{Z}\prod_{i,j}\Psi(x_i,x_j)\prod_i \Phi(x_i)$$

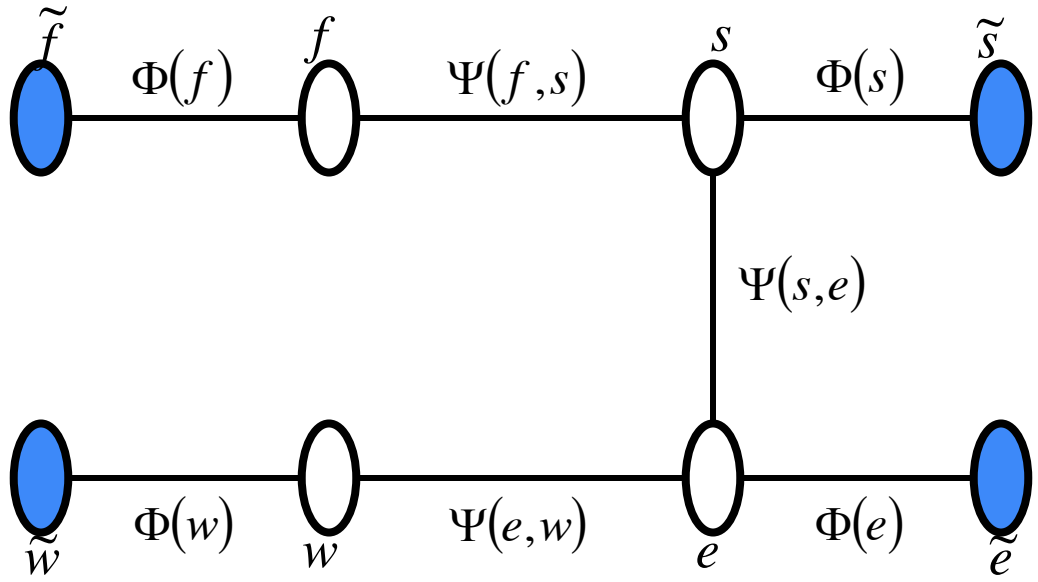
Spatial Model

Start with my GM from ICLR 2013 paper

Use **MRF** over spatial locations

Belief Propagation to calculate marginals

f, s, e, w



classic sum-product BP

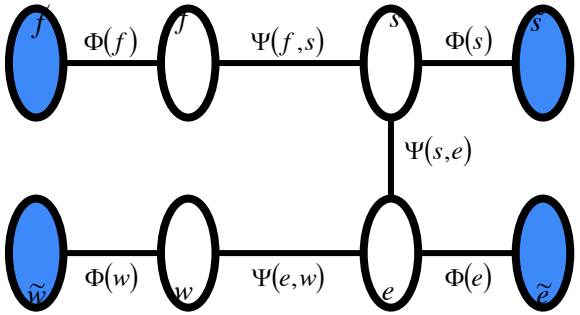
$$b(x_i) = k\,\Phi(x_i)\prod_{j\in N(i)} m_{ji}(x_i) \quad , \quad m_{ji}(x_i) \leftarrow \sum_{x_j}\Phi(x_j)\Psi(x_j,x_i)\prod_{k\in N(j)\setminus i} m_{kj}(x_j)$$

Spatial Model

Start with my GM from ICLR 2013 paper

- Use **MRF** over spatial locations
- Belief Propagation to calculate marginals

$$f, s, e, w$$



$$b(x_i) = k\,\Phi(x_i)\prod_{j\in N(i)} m_{ji}(\mathrm{x_i}), \quad m_{ji}(\mathrm{x_i}) \leftarrow \sum_{x_j}\Phi(x_j)\Psi(x_j,x_i)\prod_{k\in N(j)\setminus i} m_{kj}(x_j)$$

$$b(f) = k\,\Phi(f)\,m_{sf}(\mathrm{f})$$

$$b(f) = k\,\Phi(f)\sum_s \Phi(s)\Psi(s,f)\,m_{es}(\mathrm{s})$$

$$b(f) = k\,\Phi(f)\sum_s \Phi(s)\Psi(s,f)\sum_e \Phi(e)\Psi(e,s)\,m_{we}(e)$$

$$b(f) = k\,\Phi(f)\sum_s \Phi(s)\Psi(s,f)\sum_e \Phi(e)\Psi(e,s)\sum_w \Phi(w)\Psi(w,e)$$

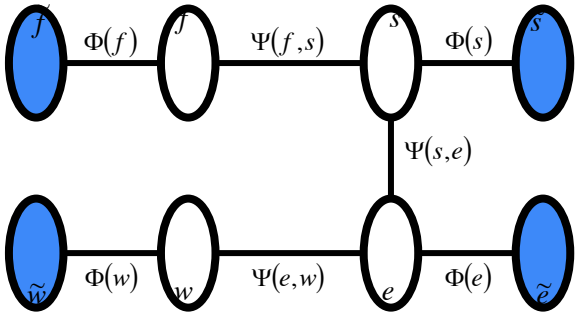
Spatial Model

Start with my GM from ICLR 2013 paper

Use **MRF** over spatial locations

Belief Propagation to calculate marginals

$$f, s, e, w$$



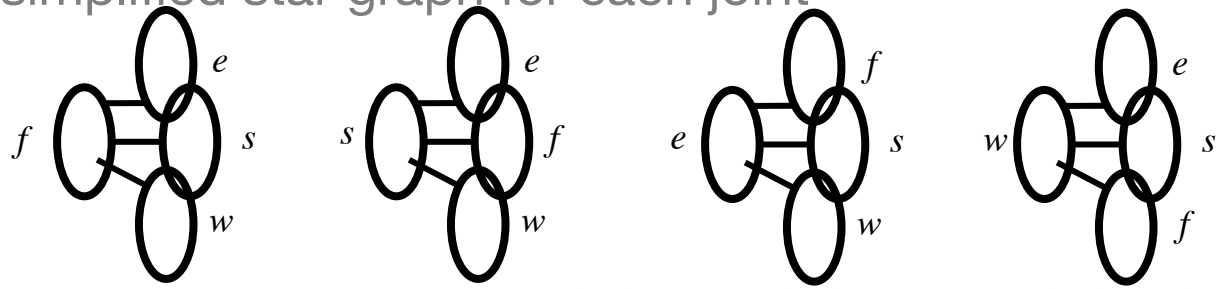
$$b(x_i) = k\,\Phi(x_i)\prod_{j\in N(i)} m_{ji}(x_i), \quad m_{ji}(x_i) \leftarrow \sum_{x_j}\Phi(x_j)\Psi(x_j,x_i)\prod_{k\in N(j)\setminus i} m_{kj}(x_j)$$

Ignore these terms…

$$b(f) = k\,\Phi(f)\sum_s \Phi(s)\Psi(s,f)\boxed{\sum_e \Phi(e)\Psi(e,s)\sum_w \Phi(w)\Psi(w,e)}$$

Assumption of correlation decay

Use simplified star graph for each joint



$$b'(f) = k\,\Phi(f)\prod_i \sum_{x_i}\Phi(x_i)\Psi(x_i,f)$$

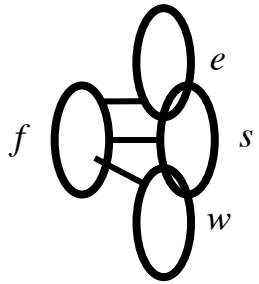
NYU

Spatial Model

Start with my GM from ICLR 2013 paper

- Use **MRF** over spatial locations
- Belief Propagation to calculate marginals
- Use simplified star graph for each joint
- Replace tensor factor product & marginal integral with local convolution and conditional factor (NOT EQUIVALENT)

$$f, s, e, w$$



$$b'(f) = k\,\Phi(f)\prod_{\mathrm{i}}\sum_{x_i}\Phi(x_i)\Psi(x_i, f)$$

$$b''(f) = k\,\Phi(f)\prod_{\mathrm{i}}\Phi(x_i) * \Psi(f \mid x_i)$$

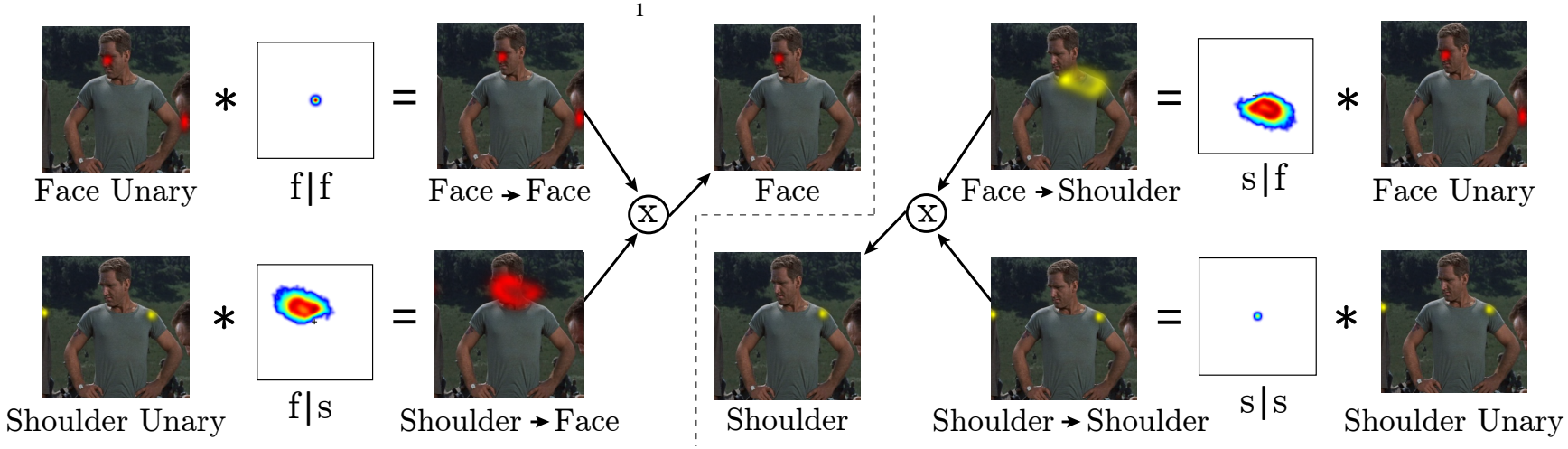
Spatial Model

Start with my GM from ICLR 2013 paper

- Use **MRF** over spatial locations
- Belief Propagation to calculate marginals
- Use simplified star graph for each joint
- Replace tensor factor product & marginal integral with local convolution and conditional factor (NOT EQUIVALENT)
- Add background probability on each "message"

f, s, e, w

$$b''(f) = k\,\Phi(f)\prod_{i}\left(\Phi(x_i) * \Psi(f \mid x_i) + c(f \mid x_i)\right)$$



NYU

SPATIAL MODEL

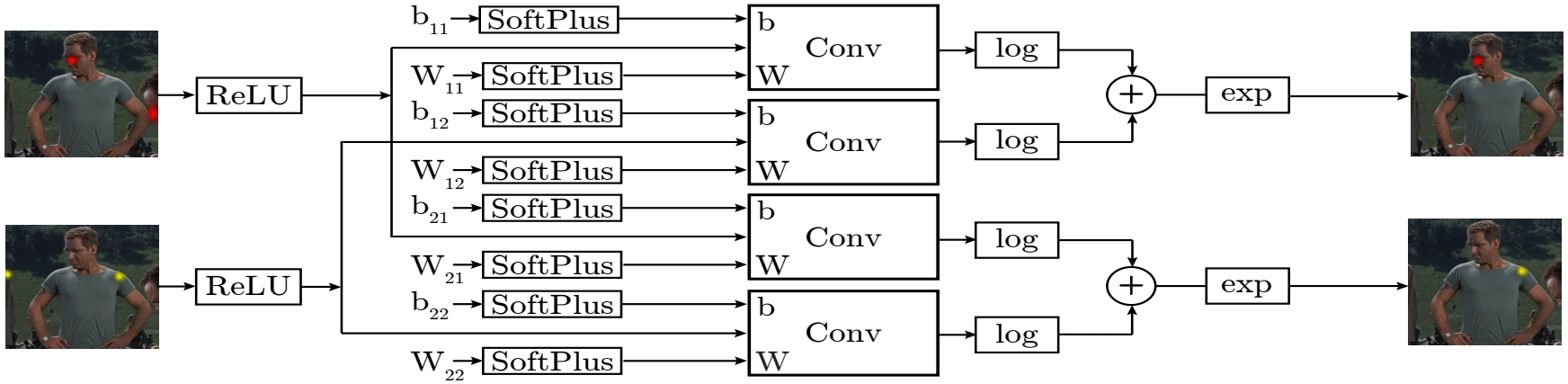
Implement it as a network

$$b''(f) = k\,\Phi(f)\prod_{i}\left(\Phi(x_i) * \Psi(f \mid x_i) + c(f \mid x_i)\right)$$

$$\bar{e}_A = \exp\left(\sum_{v\in V}\left[\log\left(\text{SoftPlus}\left(e_{A|v}\right) * \text{ReLU}\left(e_v\right) + \text{SoftPlus}\left(b_{v\to A}\right)\right)\right]\right)$$

$$\text{where: } \text{SoftPlus}(x) = {}^{1}/_{\beta}\log\left(1+\exp\left(\beta x\right)\right),\ {}^{1}/_{2} \le \beta \le 2$$

$$\text{ReLU}(x) = \max(x, \epsilon),\ 0 < \epsilon \le 0.01$$



NYU

Spatial Model

Wait, where did the partition function go?

$$b'''(\tilde{x}_1) \approx \frac{1}{Z} \Phi(\tilde{x}_1) \prod_i \left(\Phi(\tilde{x}_i) * \Psi(\tilde{x}_i \mid \tilde{x}_1) + b(\tilde{x}_i \mid \tilde{x}_1) \right)$$

One Reason why it's OK

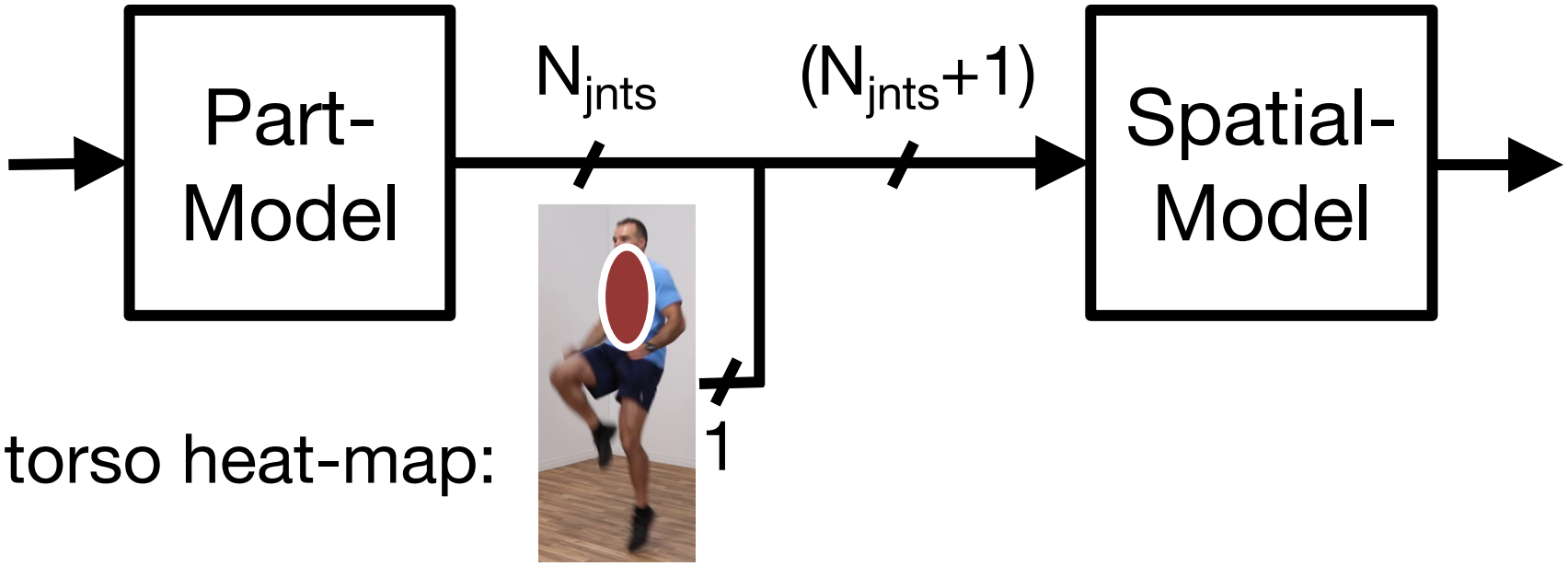
We only care about Maximum Likelihood

→ Distribution Shape not important

SPATIAL MODEL

Two additional details

1. Spatial model kernel size is 128x128! → Have to use FFT[1]
2. For standard datasets → add (noisy) torso location



[1] M. Mathieu, M. Henaff, and Y. LeCun. Fast training of convolutional networks through FFTs. 2013.

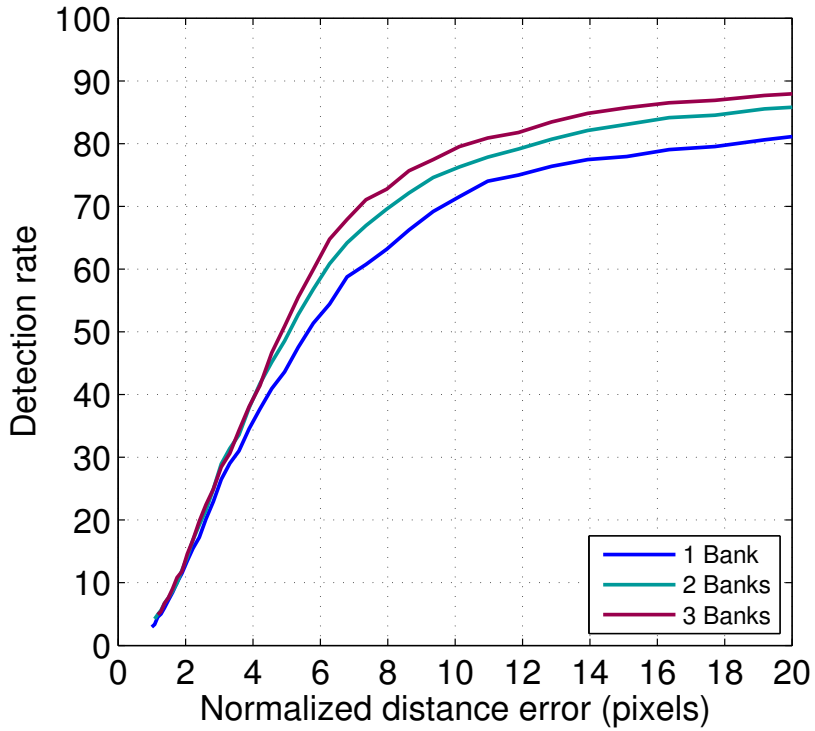
JOINT TRAINING

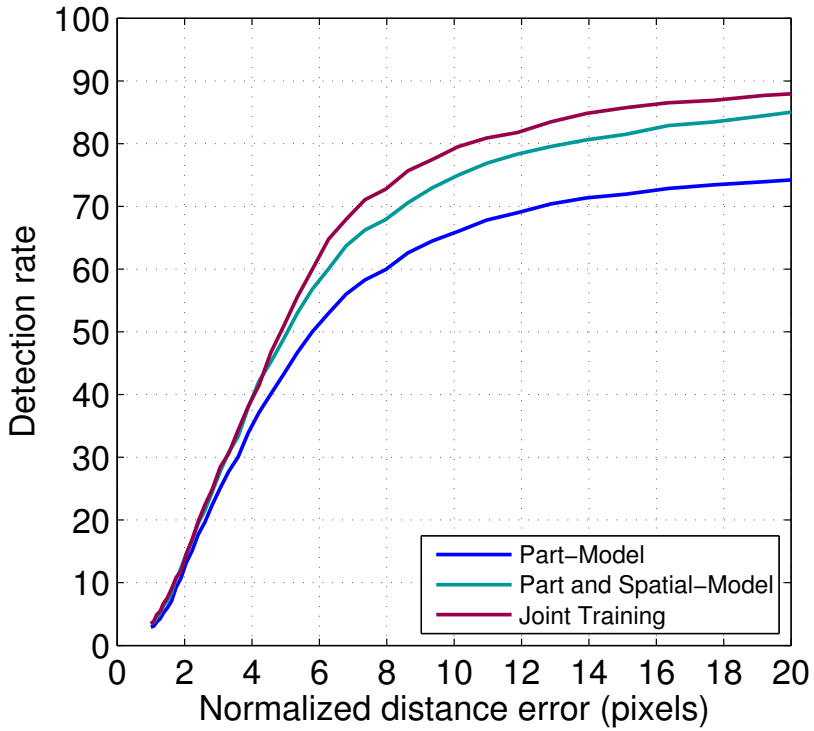
Joint training

- Pre-train both models separately
- Joint train (BPROP) through both models

Wrist Performance:

$$\text{DetectionRate}\,(R) = \frac{100}{N}\sum_{t=1}^{N}\left(\frac{\|x \quad - x^t\|_2}{(\text{torso height t})\,/100} \leq R\right)$$





FLIC-PLUS

FLIC-Extended[1] is not fair!

- 20928 training / 1014 test samples
- 800 out of the 1014 test images (~80%) have an image in the training set that is at most 40 frames away

“FLIC-Plus”:

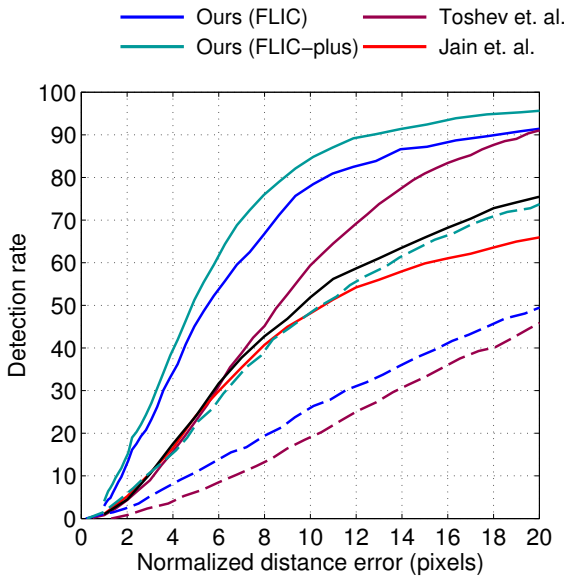
- Use Mechanical Turk to find scenes in training and test sets
- Reject training set images from the same scene

cims.nyu.edu/~tompson/flic_plus.htm

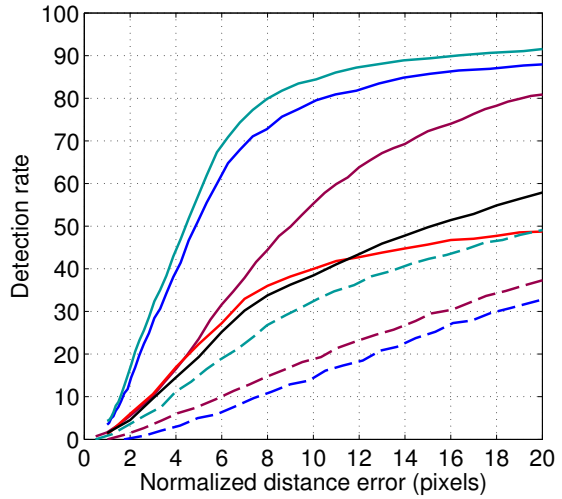
(1) B. Sapp and B. Taskar. MODEC: Multimodel decomposition models for human pose estimation. CVPR’13

RESULTS

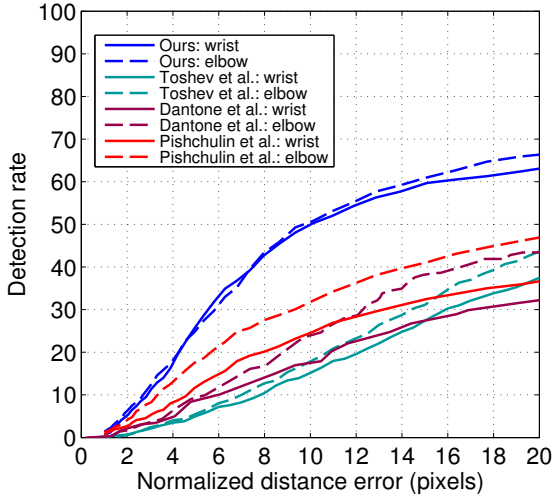
FLIC[1] Elbow



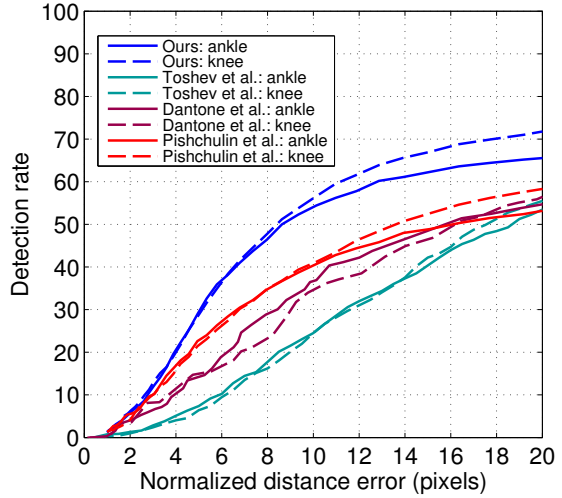
FLIC[1] Wrist



LSP[2] Arms



LSP[1] Legs



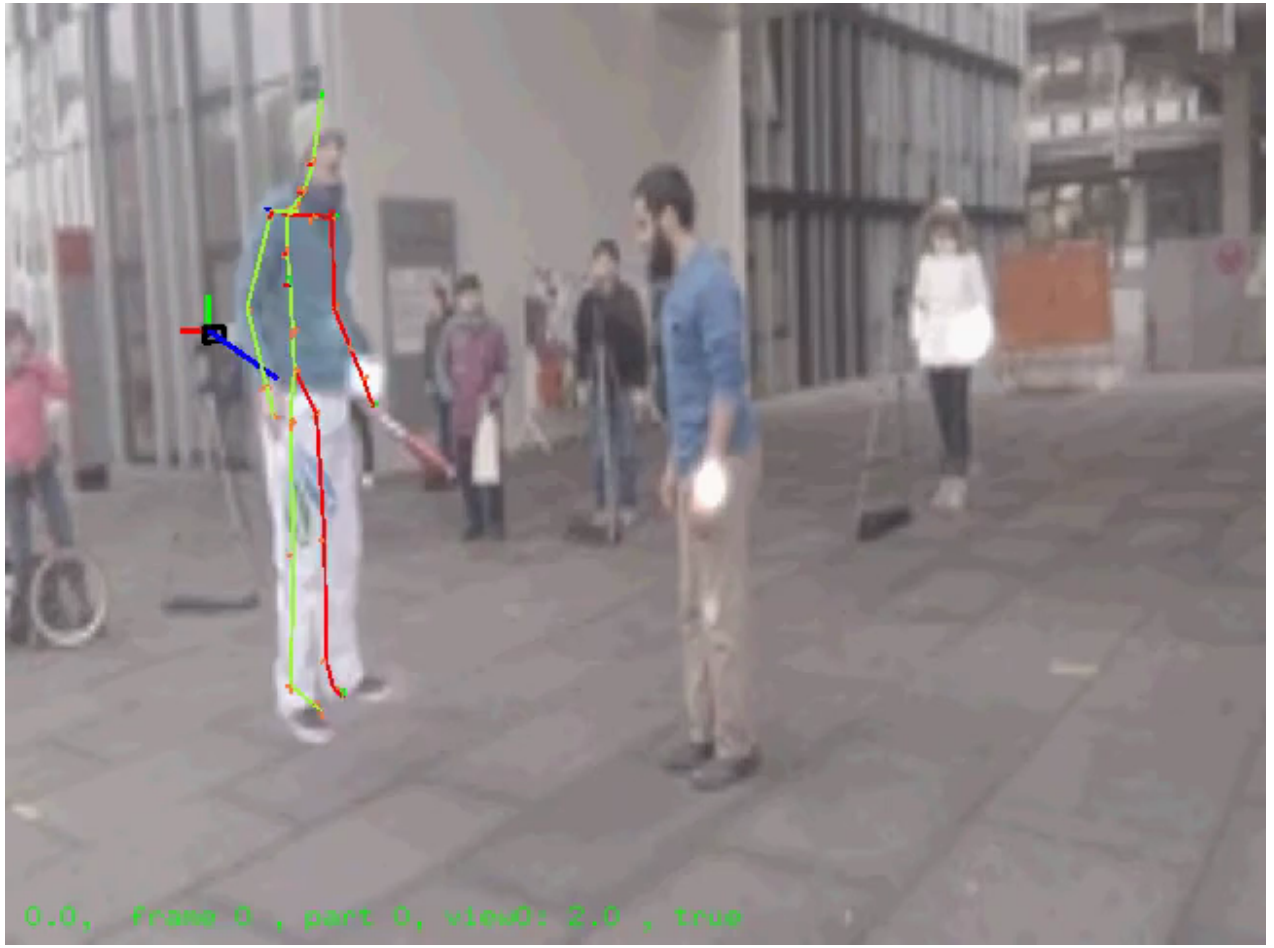
(1) B. Sapp and B. Taskar. MODEC: Multimodel decomposition models for human pose estimation. CVPR'13
(2) S. Johnson and M. Everingham. Learning Effective Human Pose Estimation for Inaccurate Annotation. CVPR'11

NYU

RESULTS

Joint work with MPII

Use our detector + analysis by synthesis technique



NYU

RESULTS

Joint work with MPII

Use our detector + analysis by synthesis technique

NYU

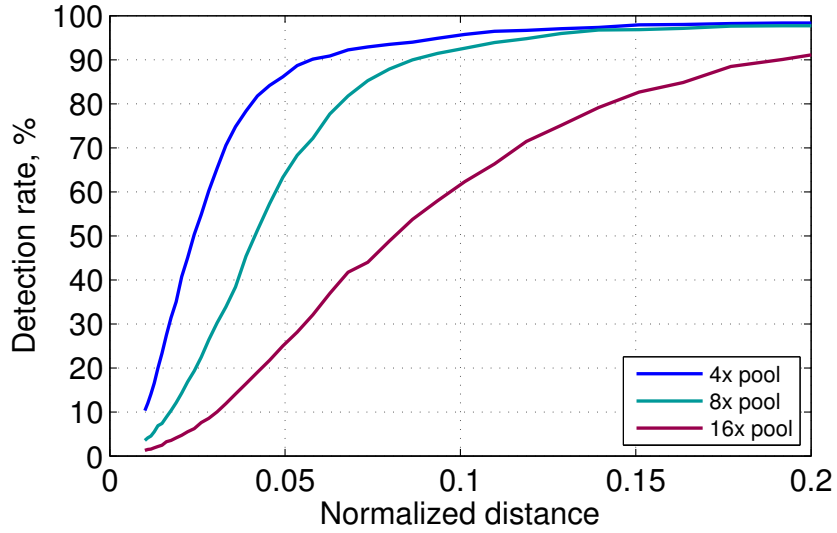
BODY TRACKING

J. Tompson, R. Goroshin, A. Jain, Y. LeCun, C. Bregler, “Efficient Object Localization Using Convolutional Networks”, CVPR 2015

MOTIVATION

We use pooling

Too much pooling: bad spatial accuracy



Too little pooling: Over-trains, **slow**, not enough context

Idea:

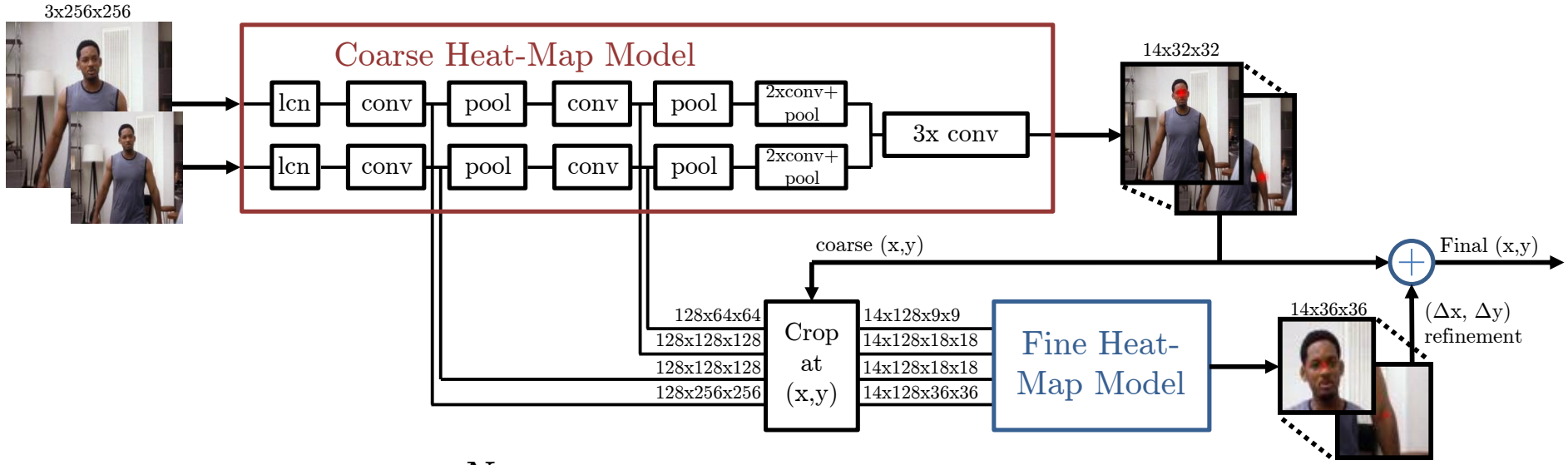
Use lots of pooling but recover spatial accuracy

A “smart” cascade (highly engineered)

High Level Architecture

Highly “engineered”

Coarse to Fine architecture with shared features



$$E_1 = \frac{1}{N}\sum_{j=1}^{N}\sum_{xy}\left\|H'_j(x,y) - H_j(x,y)\right\|^2$$

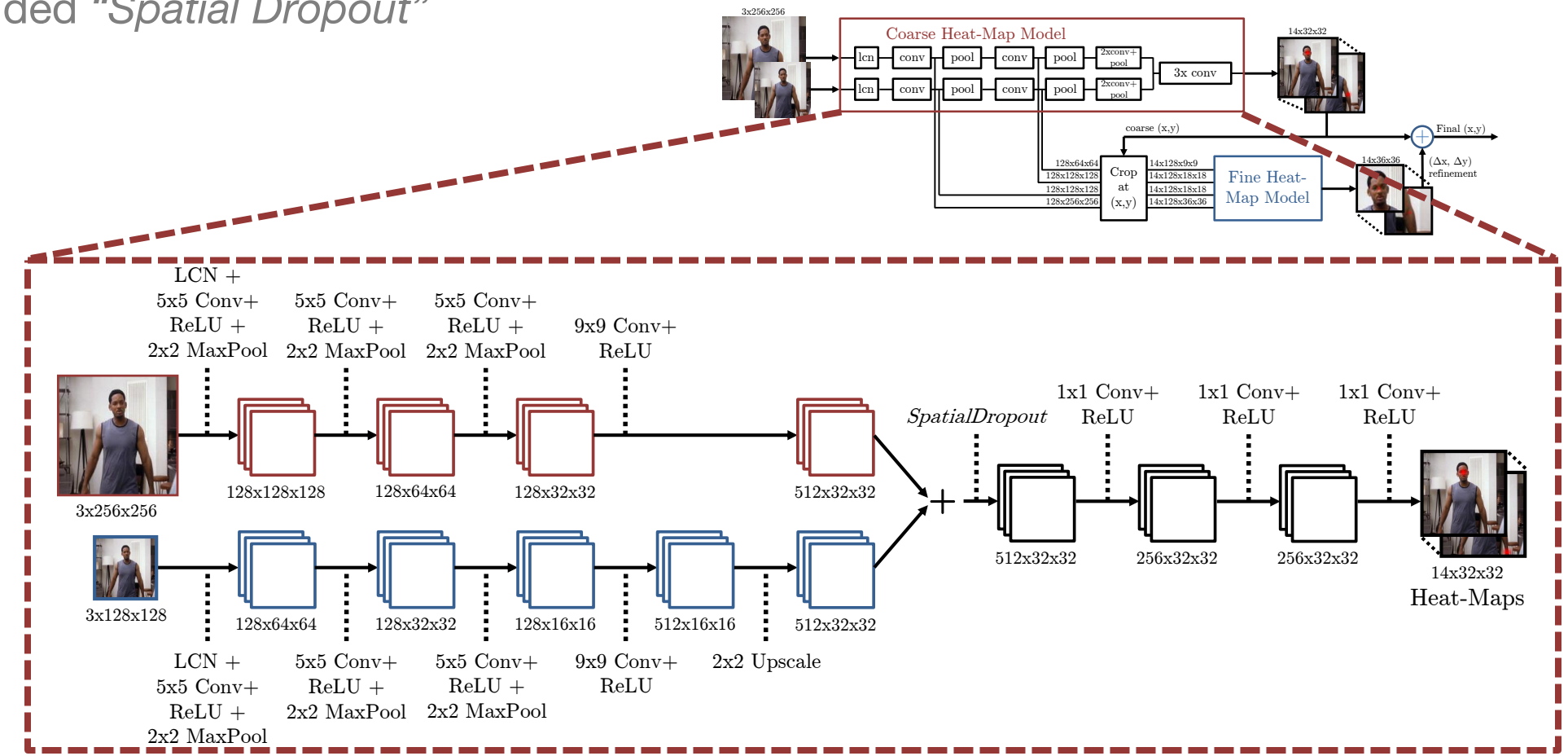
$$E_2 = E_1 + \lambda\frac{1}{N}\sum_{j=1}^{N}\sum_{x,y}\left\|G'_j(x,y) - G_j(x,y)\right\|^2$$

NYU

COARSE MODEL

Straight from the NIPS paper

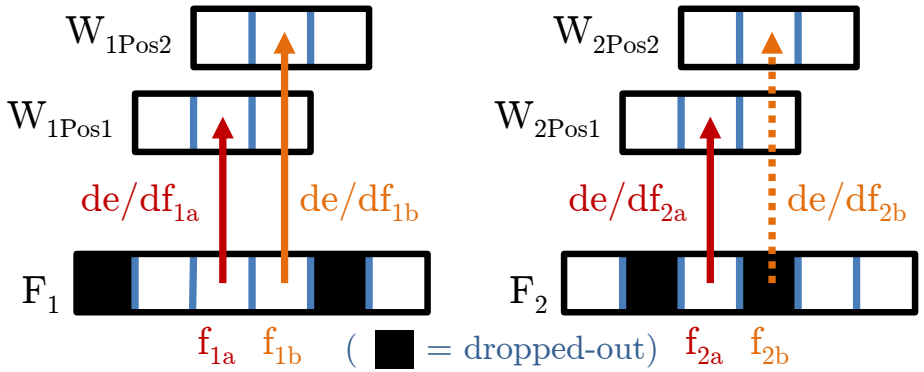
- Many more features (more engineering)
- Added *"Spatial Dropout"*



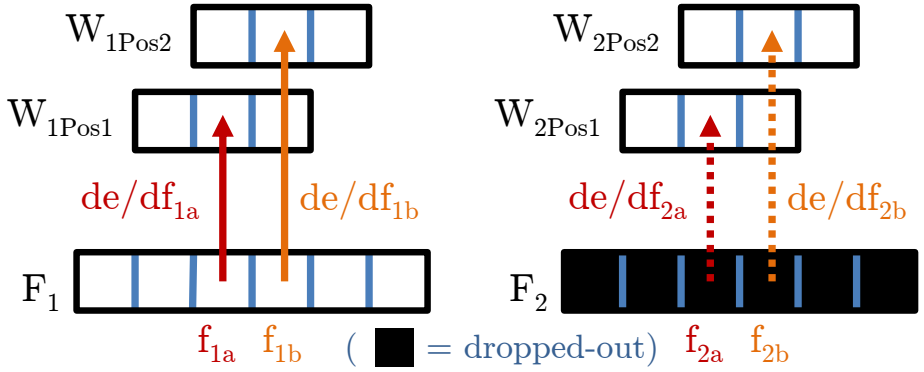
Spatial Dropout

Dropout fails for fully-convolutional networks

- Gradients are correlated because pixels are highly correlated
- Dropout just scales the learning rate



Instead Dropout the entire feature



NYU

CASCADE CROP

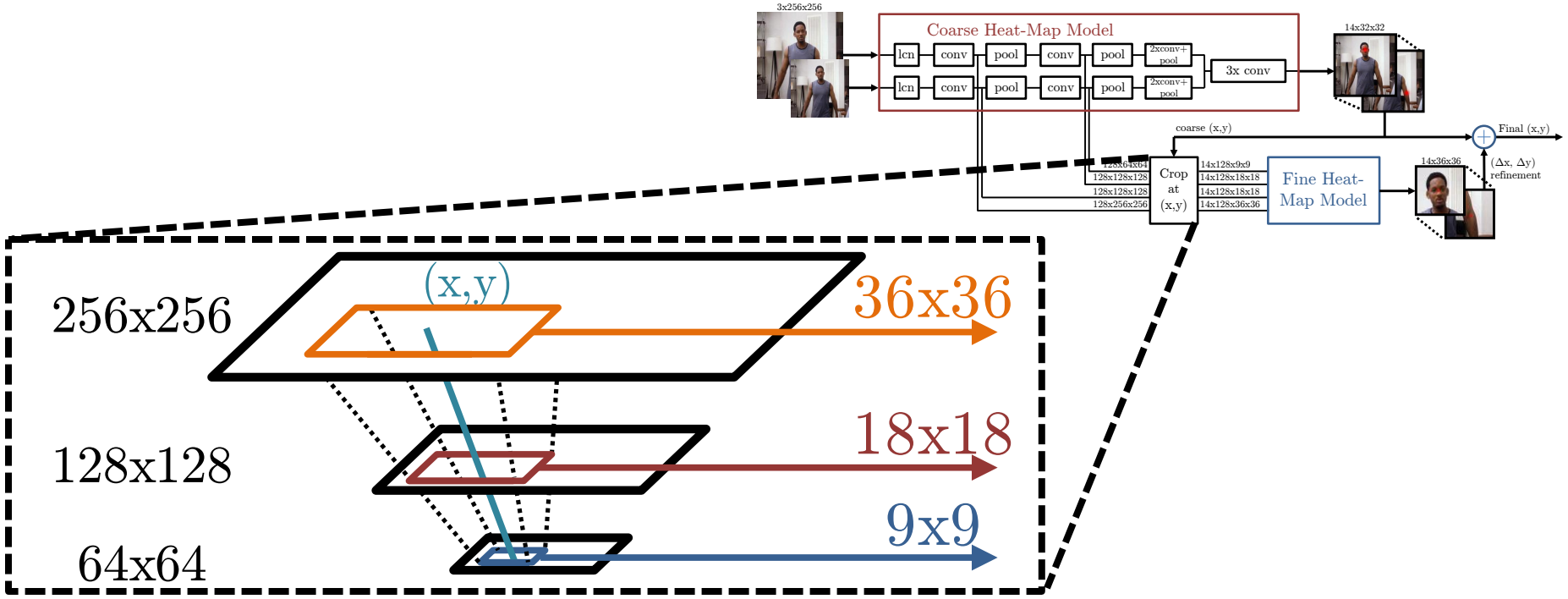
What do we want?

To refine the coarse heat-map by cropping around the approx. UV

What features do we need?

Early layers that haven't specialized

Consistent Spatial Context from all layers

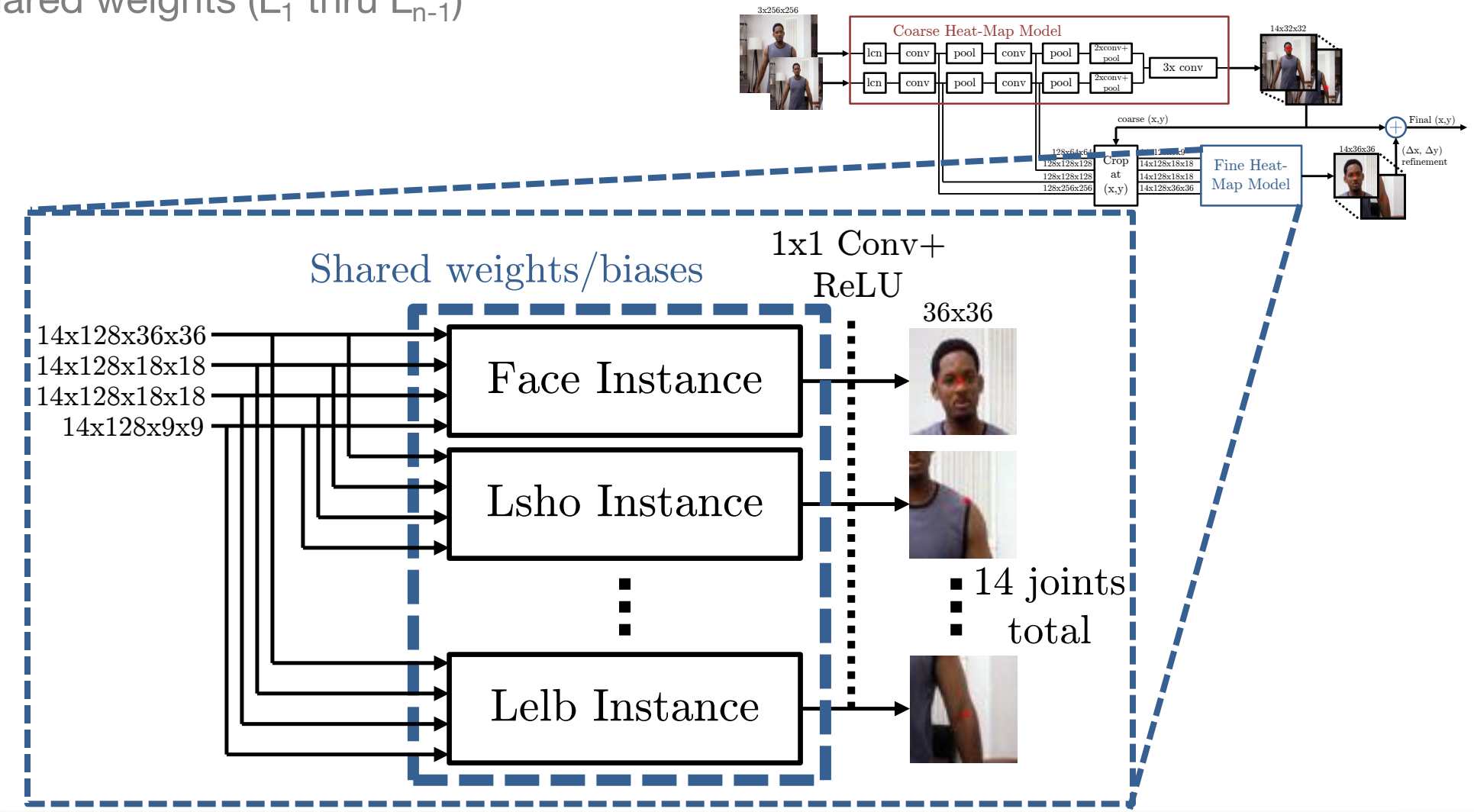


NYU

FINE MODEL

Inputs sampled from multiple locations → Need separate networks (Siamese)

Shared weights (L_1 thru L_{n-1})

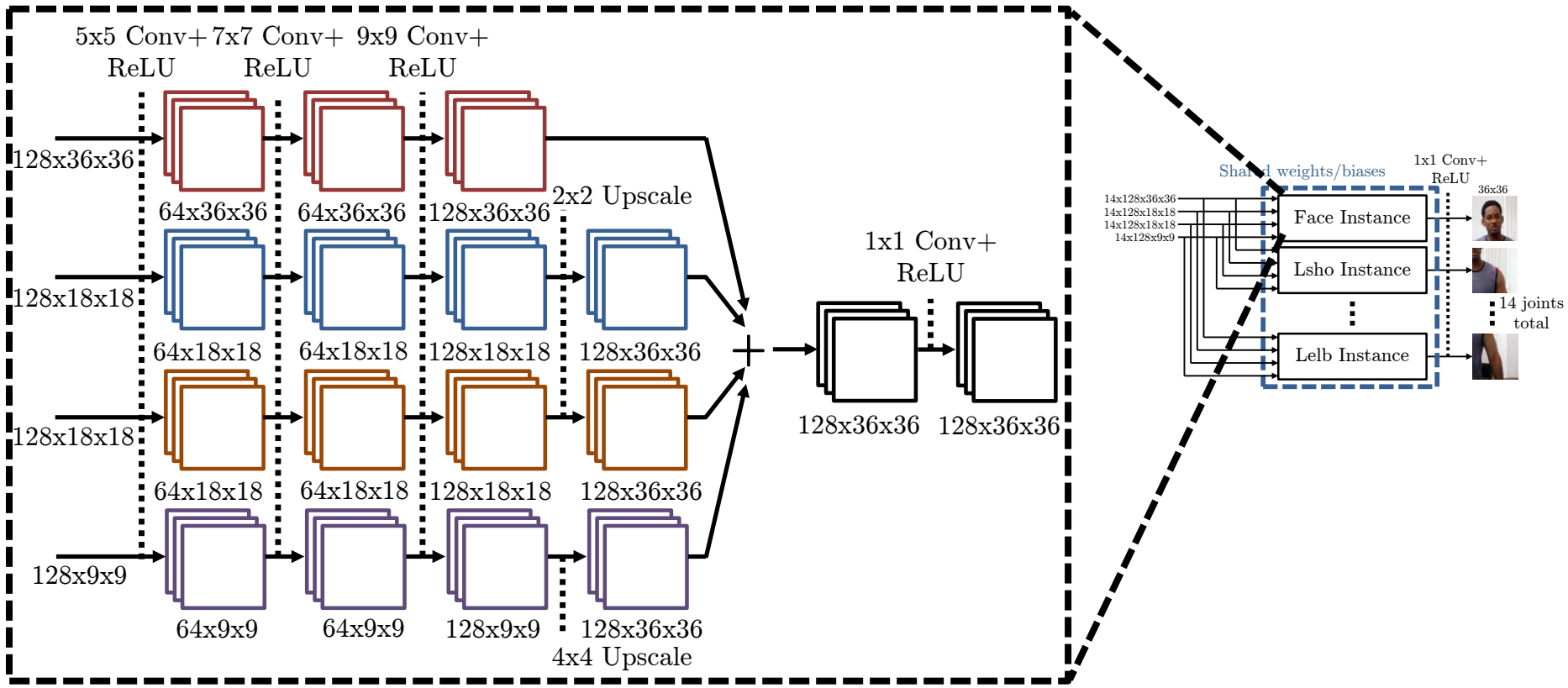


NYU

Fine Model – Replicated Instance

Use same strategy as the Coarse Model

- Fully Convolutional (with 1x1 conv layers)
- Up-sample to bring features into canonical resolution

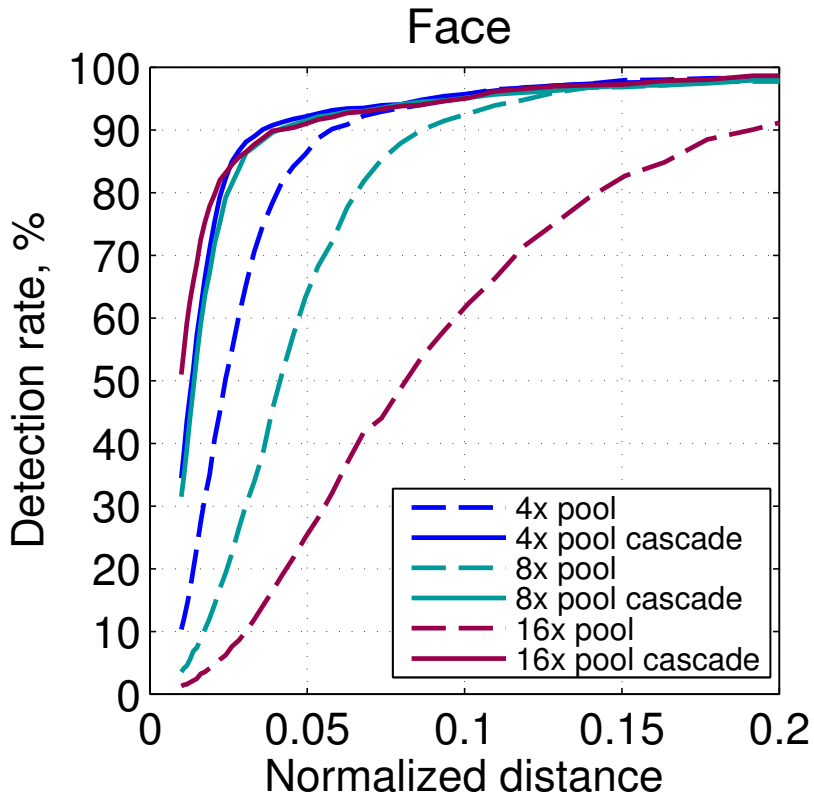


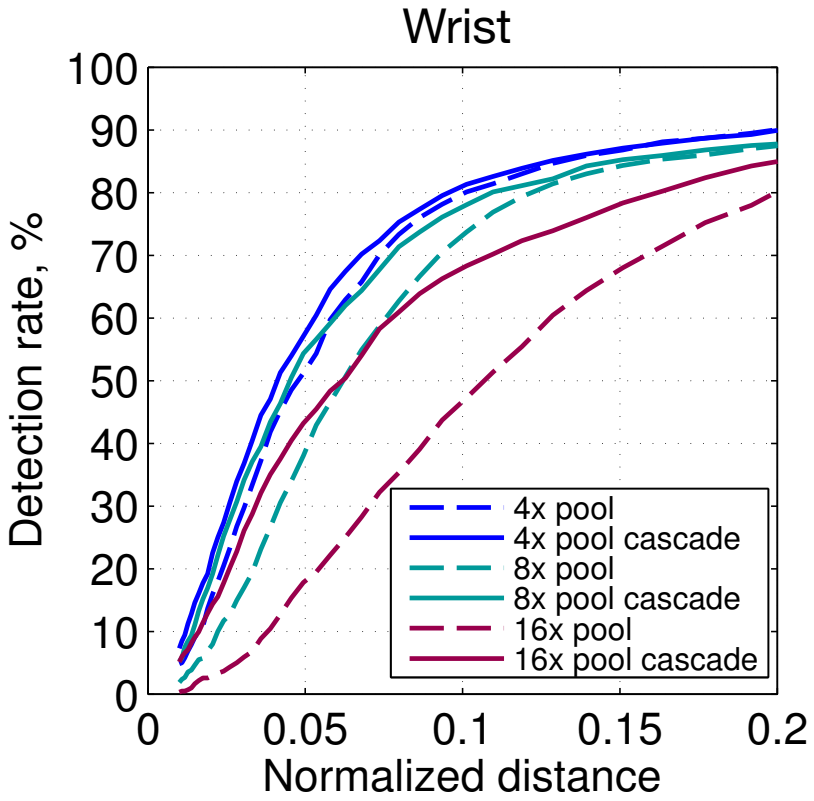
RESULTS

RESULTS

FLIC Dataset again

- Small improvement for wrist:
  - Not enough context in fine model
  - Too many mistakes in the coarse model



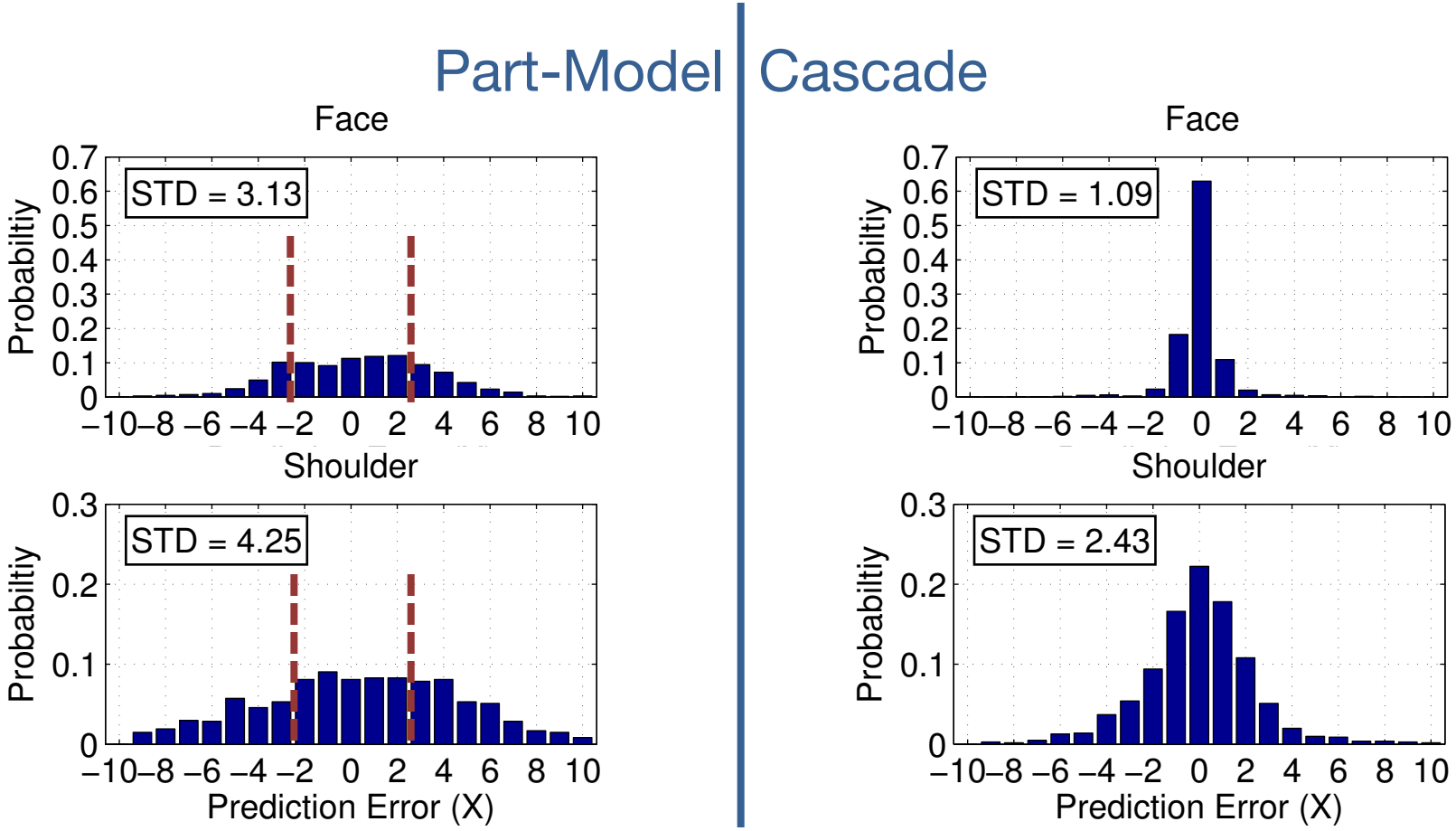


NYU

How Well Are We Doing?

Evaluation of our model

Here we use a part model with 4x pooling (+/- 2 pixel uncertainty)



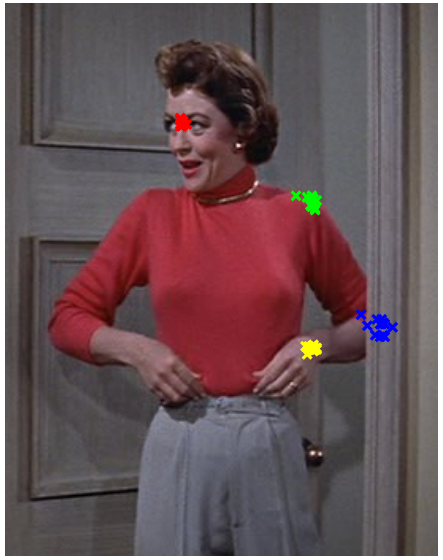
NYU

How Well Are We Doing?

Informal study to evaluate human performance

1. Show users examples and explain desired joint location
2. Ask users to label 10 images from FLIC
3. Compare this to our performance






How Well Are We Doing?

Comparing our ConvNet to Human Variance

The 16x pooling module actually does quite well despite pooling.

	Face	Shoulder	Elbow	Wrist
Label Noise (10 images)	0.65	2.46	2.14	1.57
This work 4x (test-set)	1.09	2.43	2.59	2.82
This work 8x (test-set)	1.46	2.72	2.49	3.41
This work 16x (test-set)	1.45	2.78	3.78	4.16

Time (ms) for FPROP

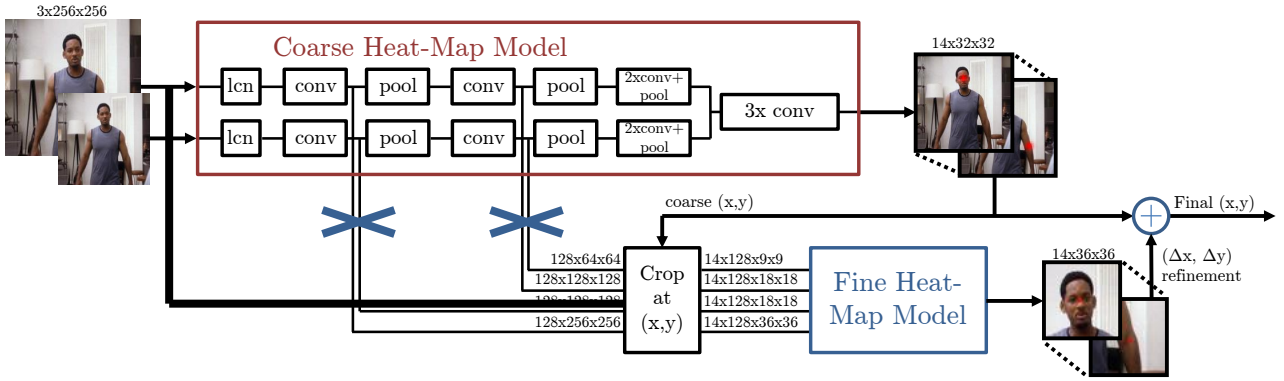
For 16x pooling: first 2 layers dominates runtime

	4x pool	8x pool	16x pool
Coarse-Model	140.0	74.9	54.7
Fine-Model	17.2	19.3	15.9
Cascade	157.2	94.2	70.6

Well, What About a Standard cascade?

Baseline experiment

Construct a ConvNet that ONLY samples from the RGB

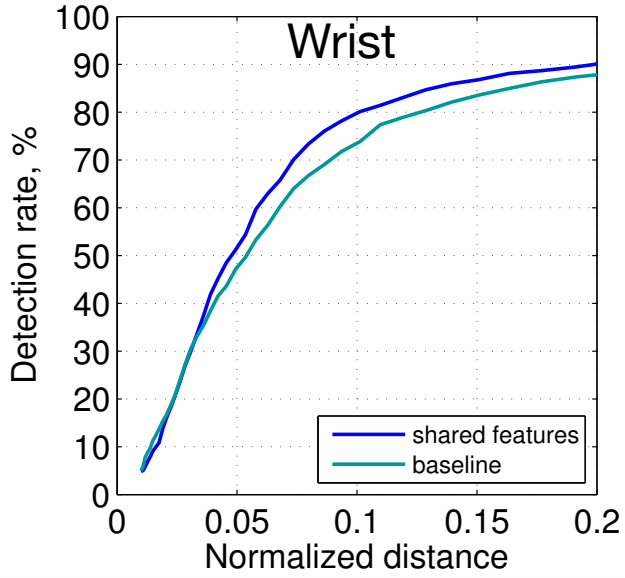


Shared features help

Regularization

Shared capacity

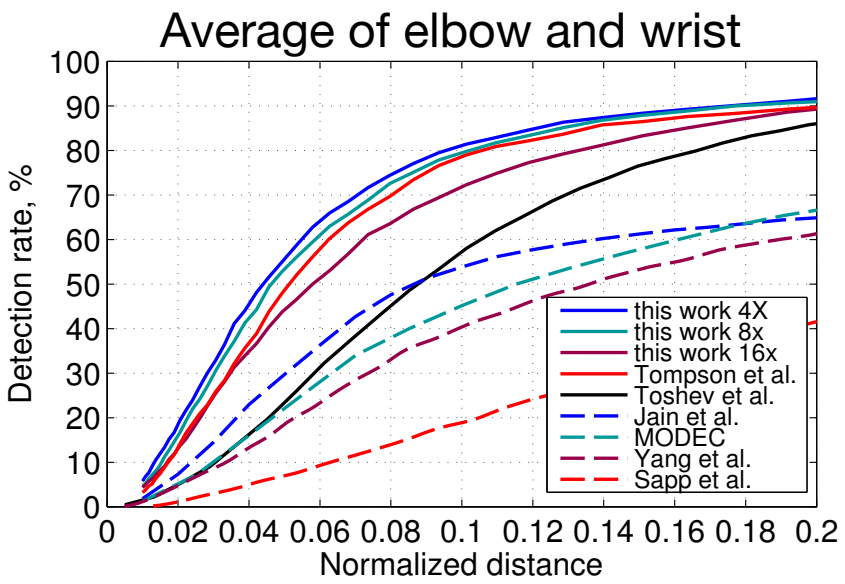
More pooling → Higher delta



NYU

COMPARISON WITH OTHER MODELS

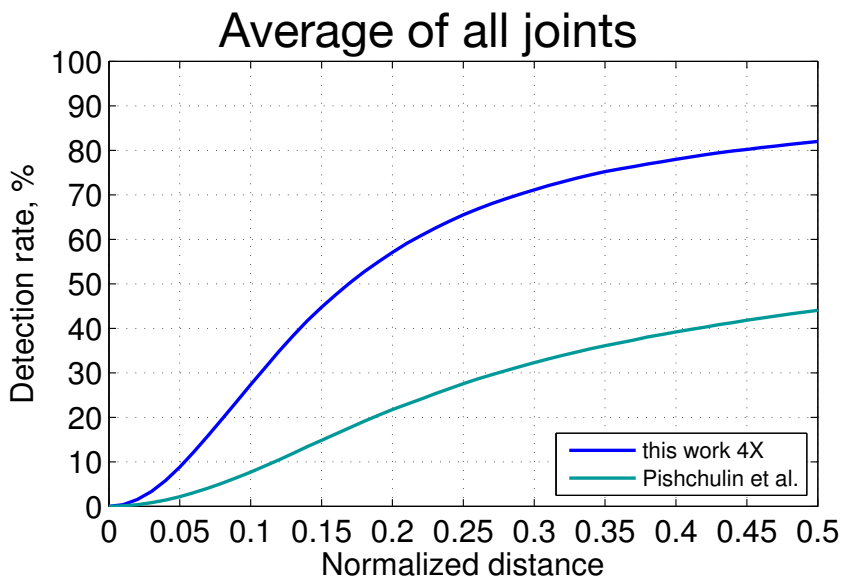
FLIC



PCK @ 0.05

	Head	Shoulder	Elbow	Wrist
Yang et al.	-	-	22.6	15.3
Sapp et al.	-	-	6.4	7.9
Eichner et al.	-	-	11.1	5.2
MODEC et al.	-	-	28.0	22.3
Toshev et al.	-	-	25.2	26.4
Jain et al.	-	42.6	24.1	22.3
Tompson et al.	90.7	70.4	50.2	55.4
This work 4x	**92.6**	73.0	**57.1**	**60.4**
This work 8x	92.1	**75.8**	55.6	56.6
This work 16x	91.6	73.0	47.7	45.5

MPII



PCK @ 0.2

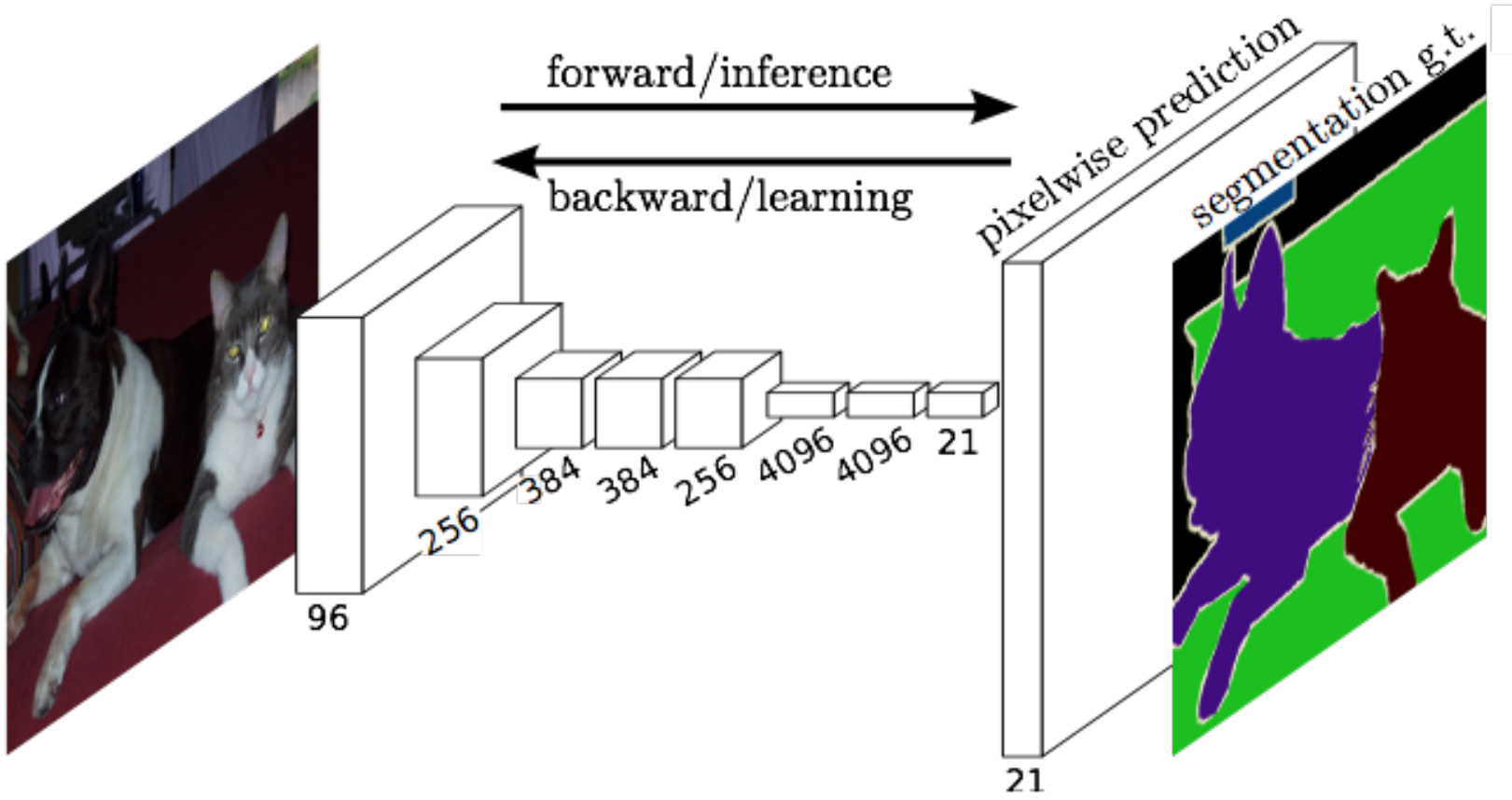
	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Upper Body	Full Body
Gkioxari et al.	-	36.3	26.1	15.3	-	-	-	25.9	-
Sapp & Taskar	-	38.0	26.3	19.3	-	-	-	27.9	-
Yang & Ramanan	73.2	56.2	41.3	32.1	36.2	33.2	34.5	43.2	44.5
Pishchulin et al.	74.2	49.0	40.8	34.1	36.5	34.4	35.1	41.3	44.0
This work 4x	**96.0**	**91.9**	**83.9**	**77.7**	**80.9**	**72.2**	**64.8**	**84.5**	**82.0**

NYU

Overview

- Body pose tracking
  - Combine Convnet with graphical model [Thompson et al. NIPS 2014]
- Methods for semantic segmentation of scene
  - Output is now 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
  - Image colorization [Zhang et al. ECCV 2016]

A Fuller Understanding of Fully Convolutional Networks

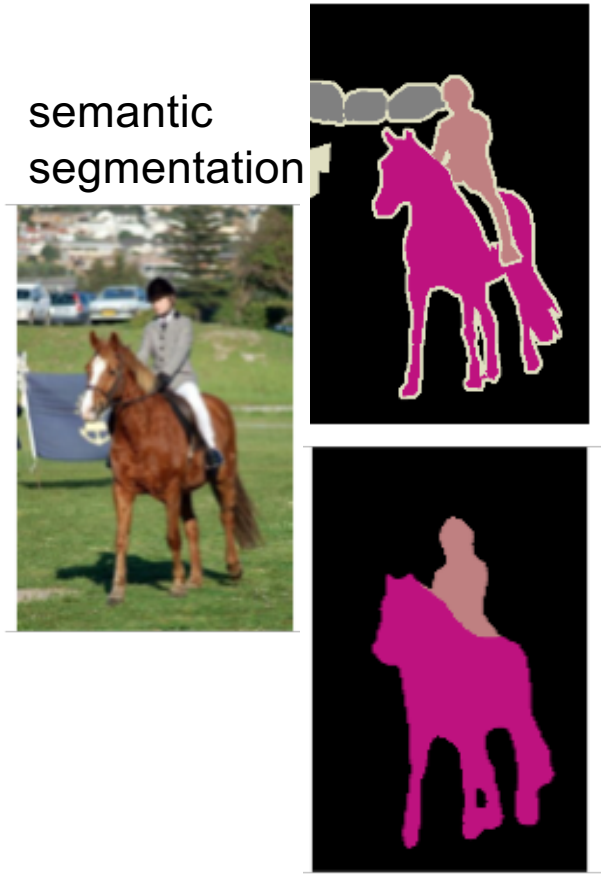


Evan Shelhamer* Jonathan Long* Trevor Darrell

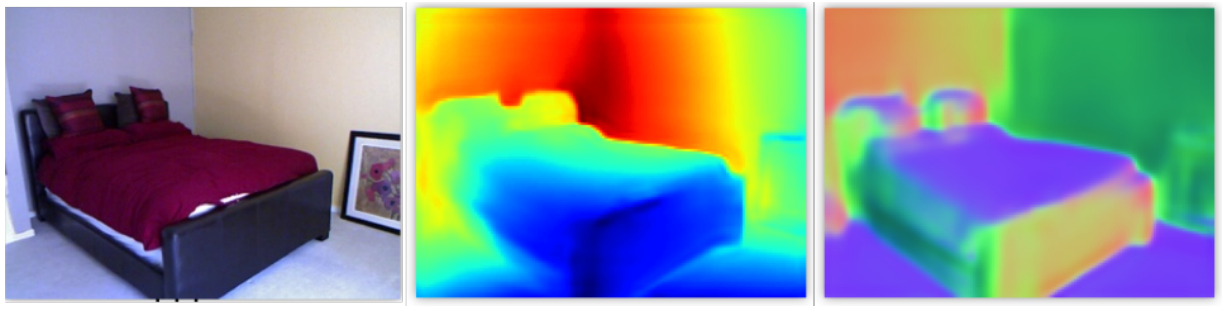
UC Berkeley in CVPR'15, PAMI'16

pixels in, pixels out

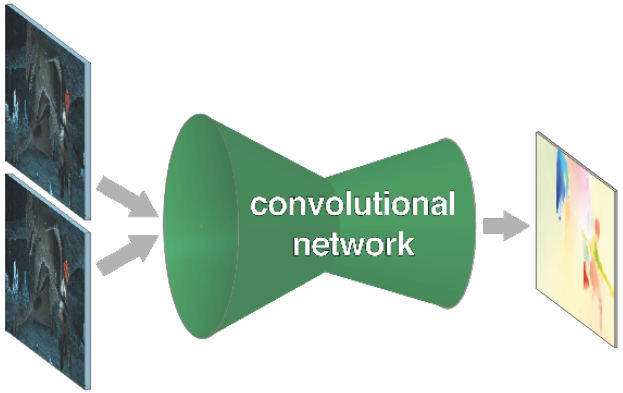
semantic segmentation



monocular depth + normals Eigen & Fergus 2015



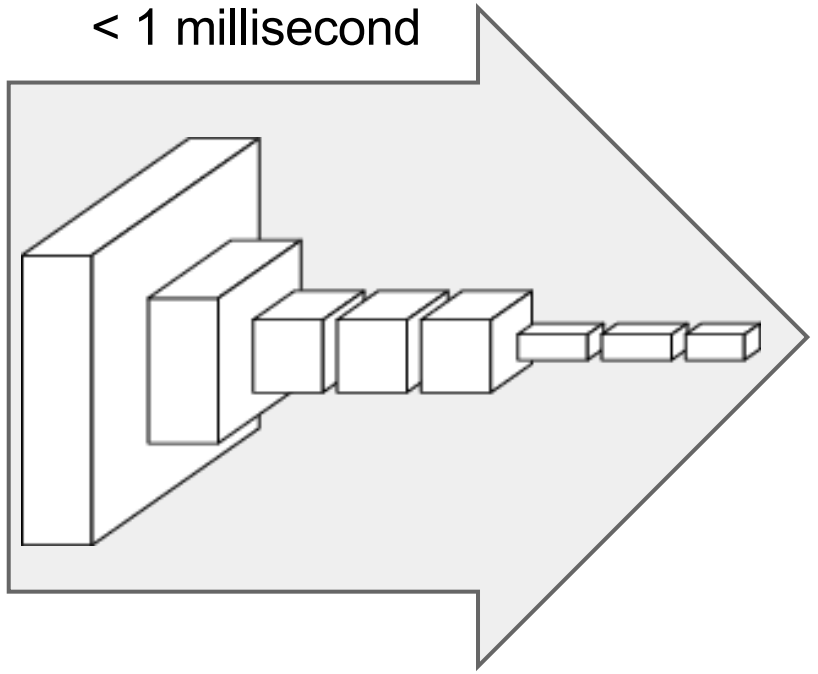
colorization
Zhang et al.2016

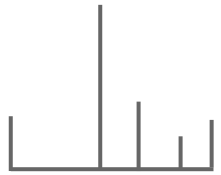


optical flow Fischer et al. 2015

boundary prediction Xie & Tu 2015

convnets perform classification



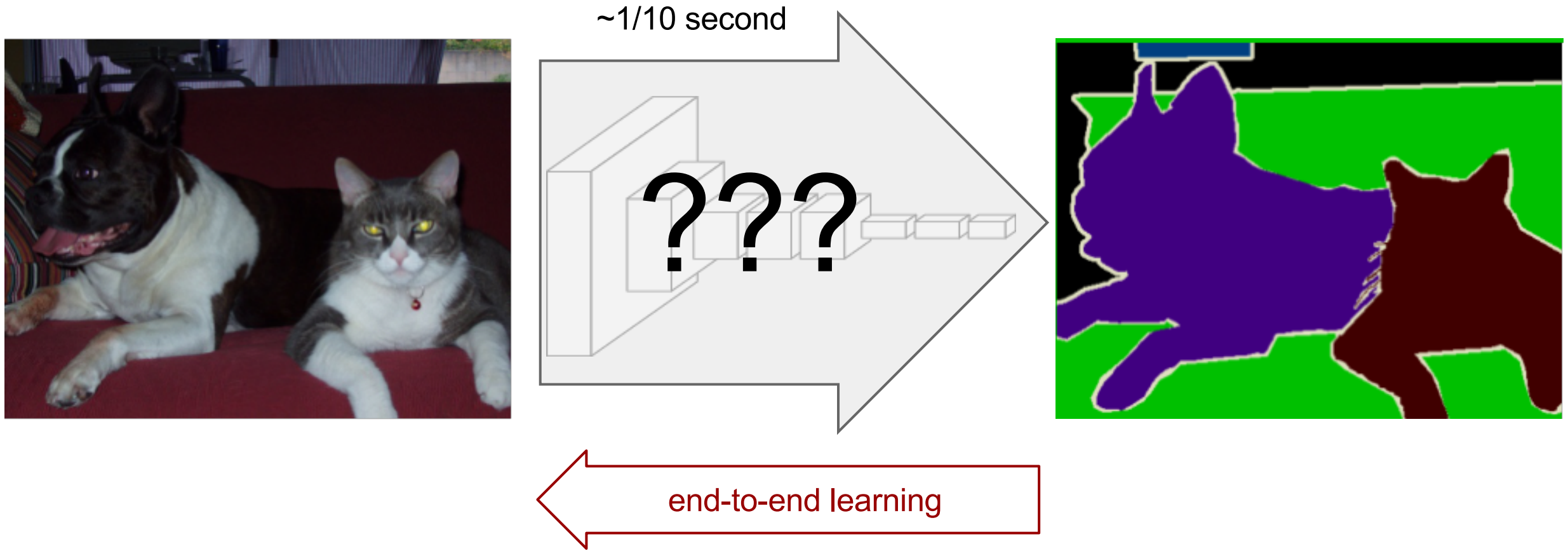


1000-dim vector

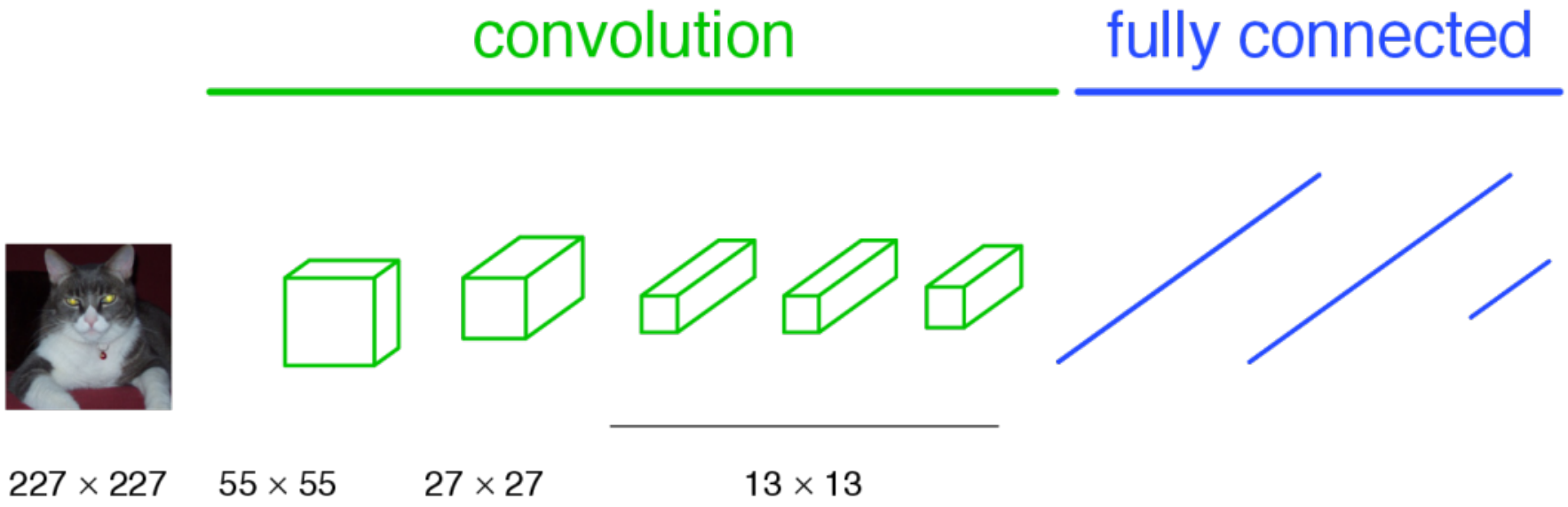
“tabby cat”

end-to-end learning

lots of pixels, little time?

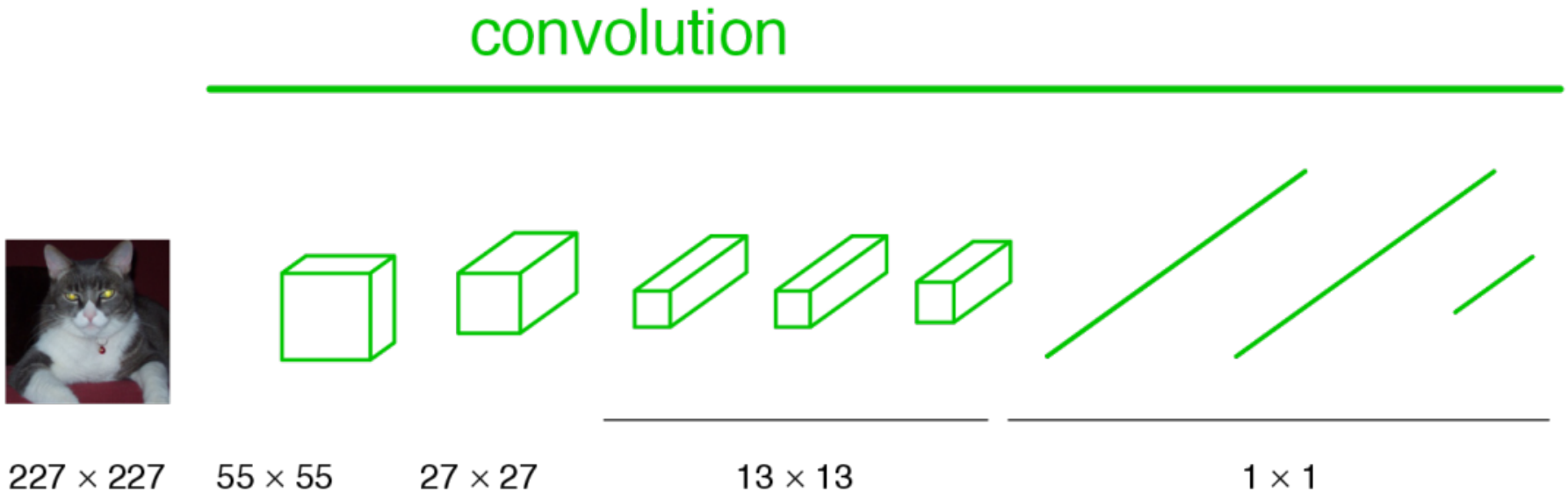


a classification network

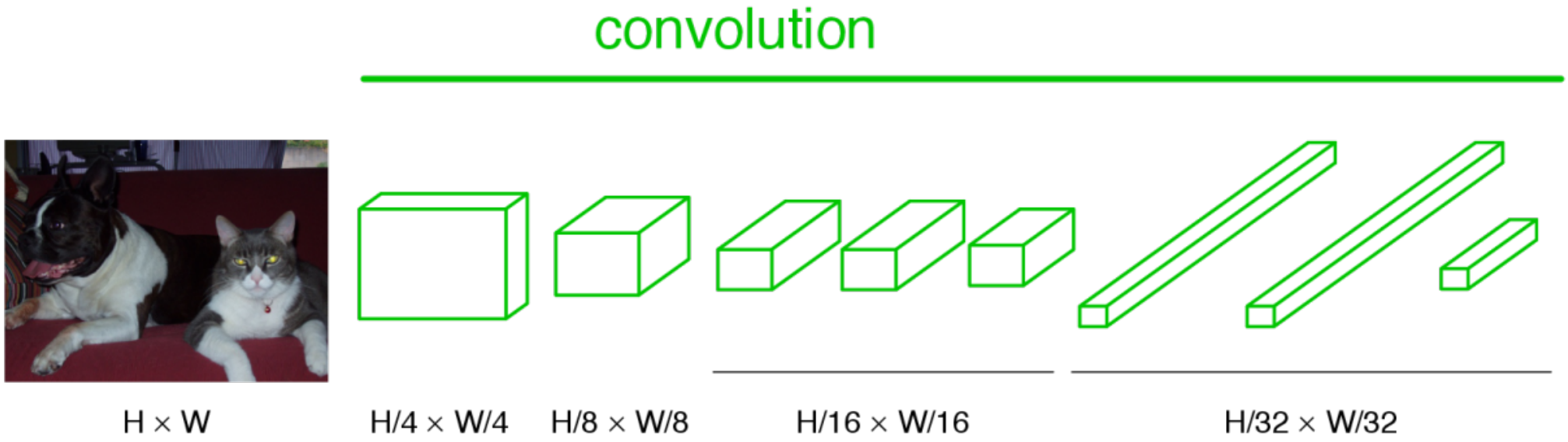


“tabby cat”

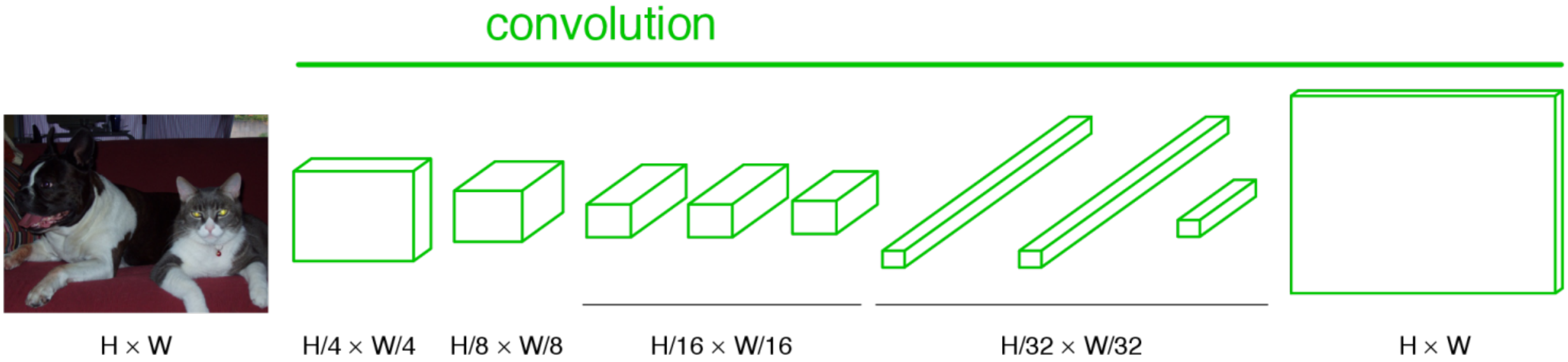
becoming fully convolutional



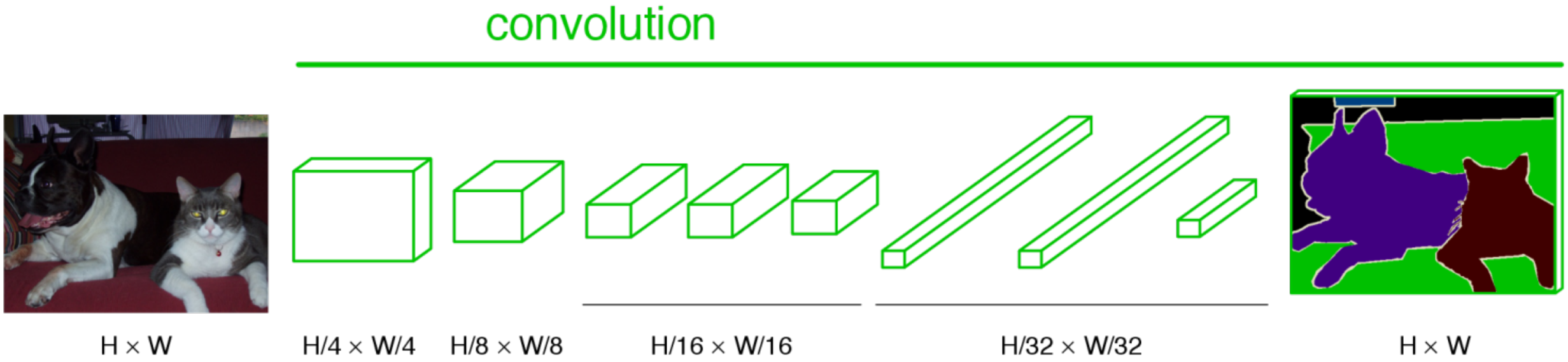
becoming fully convolutional



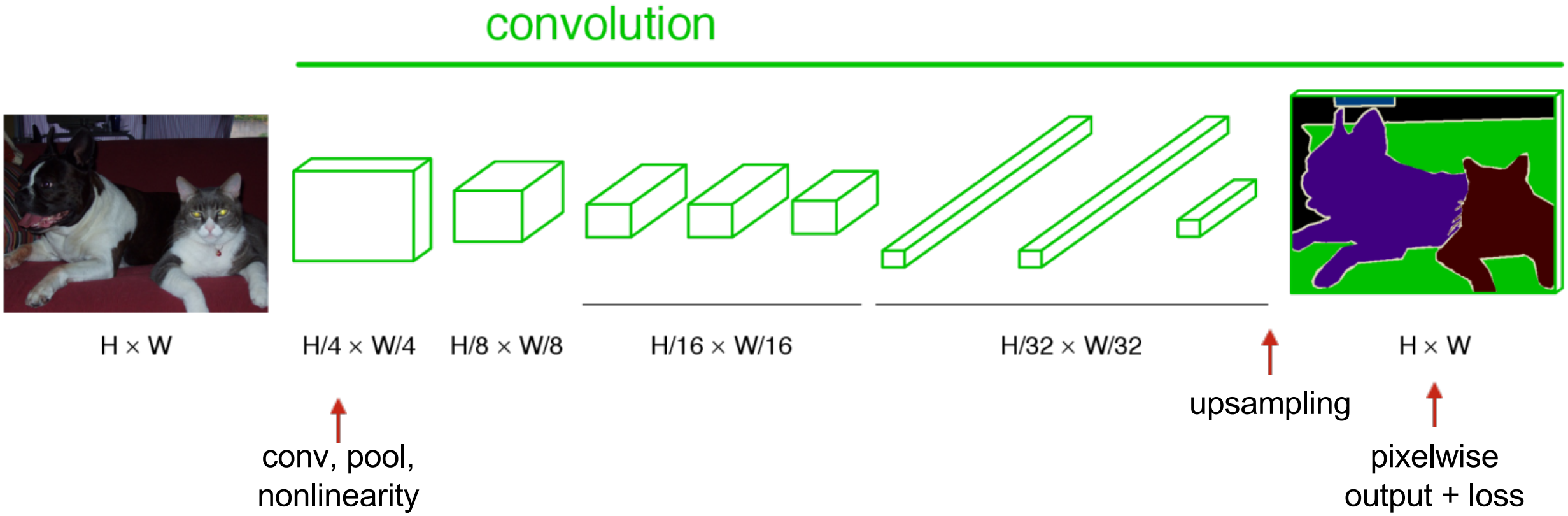
upsampling output



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

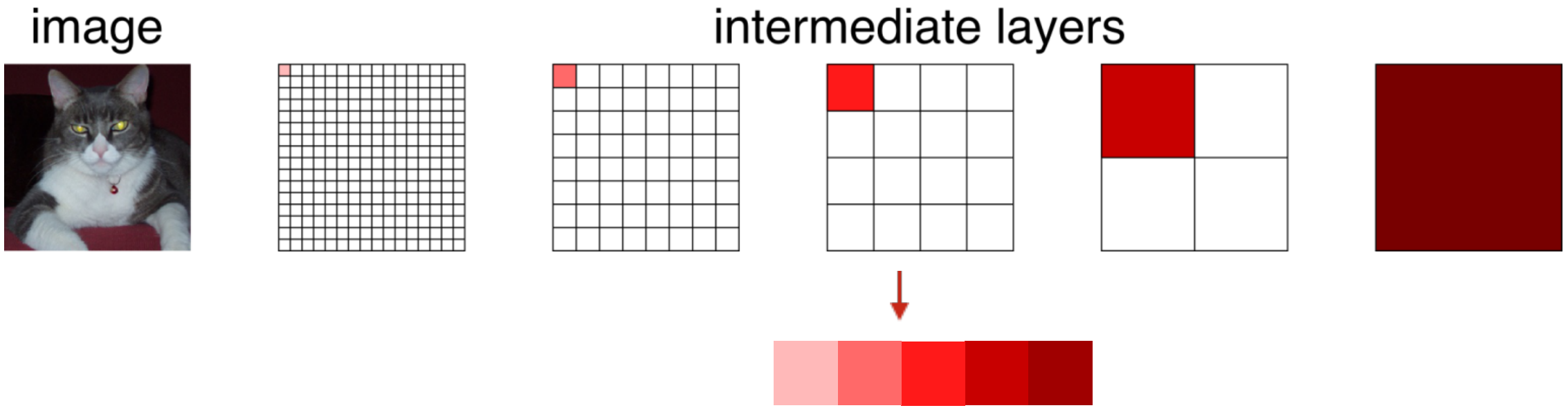


end-to-end, pixels-to-pixels network



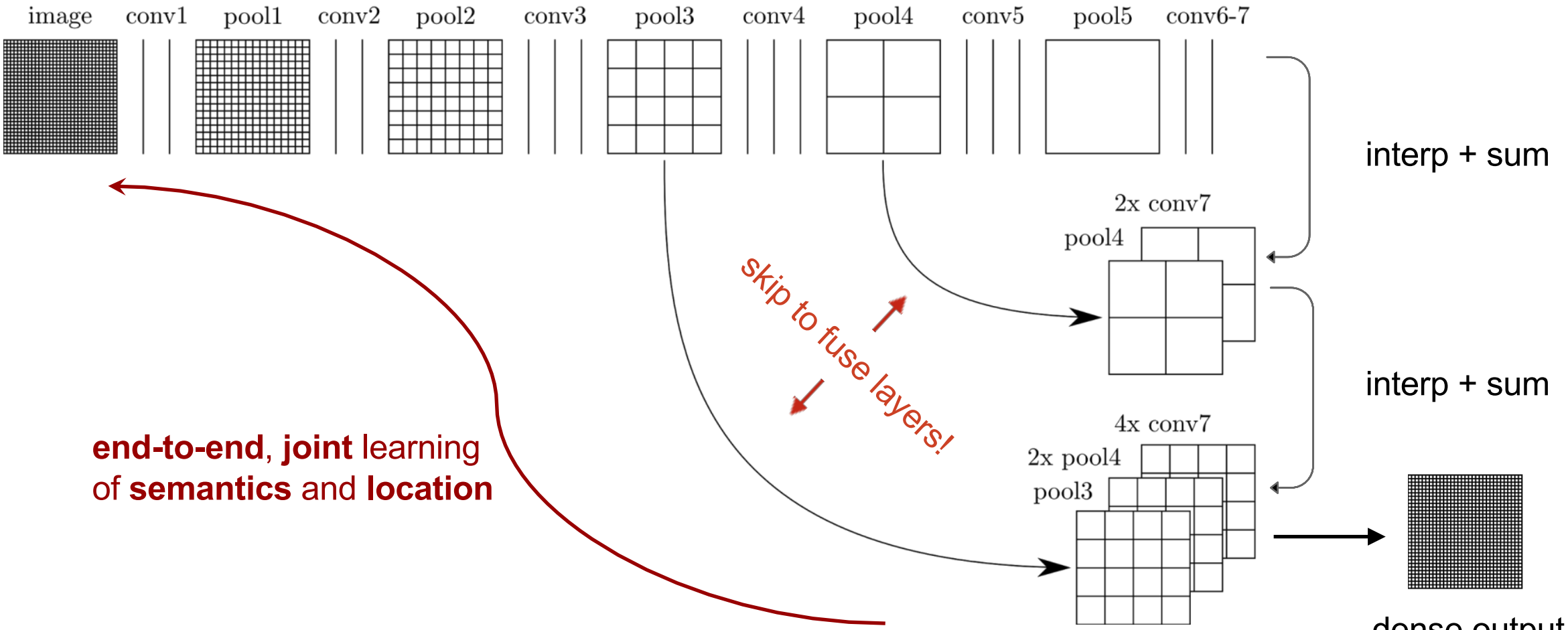
spectrum of deep features

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



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

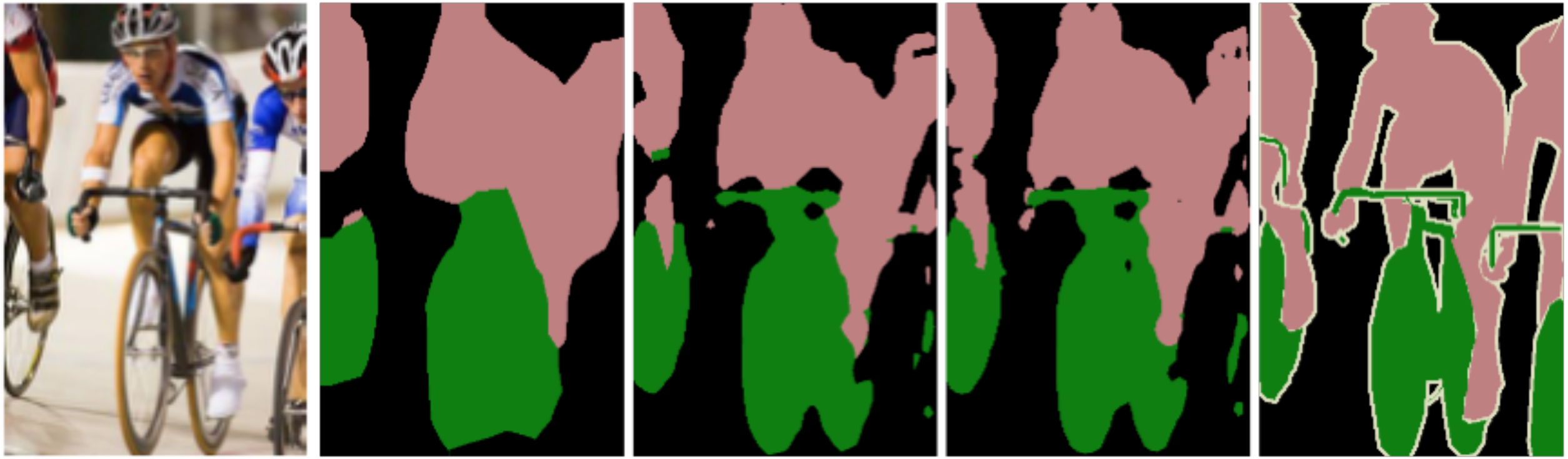
skip layers



end-to-end, **joint** learning
of **semantics** and **location**

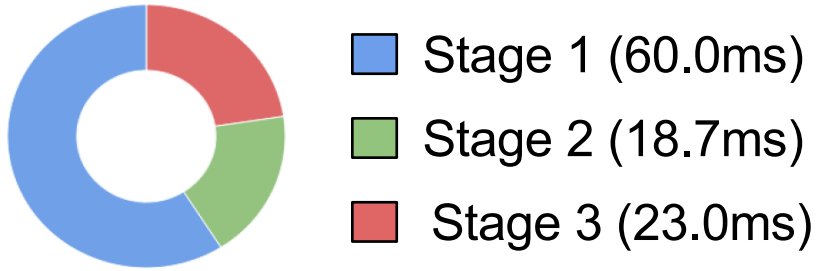
skip layer refinement

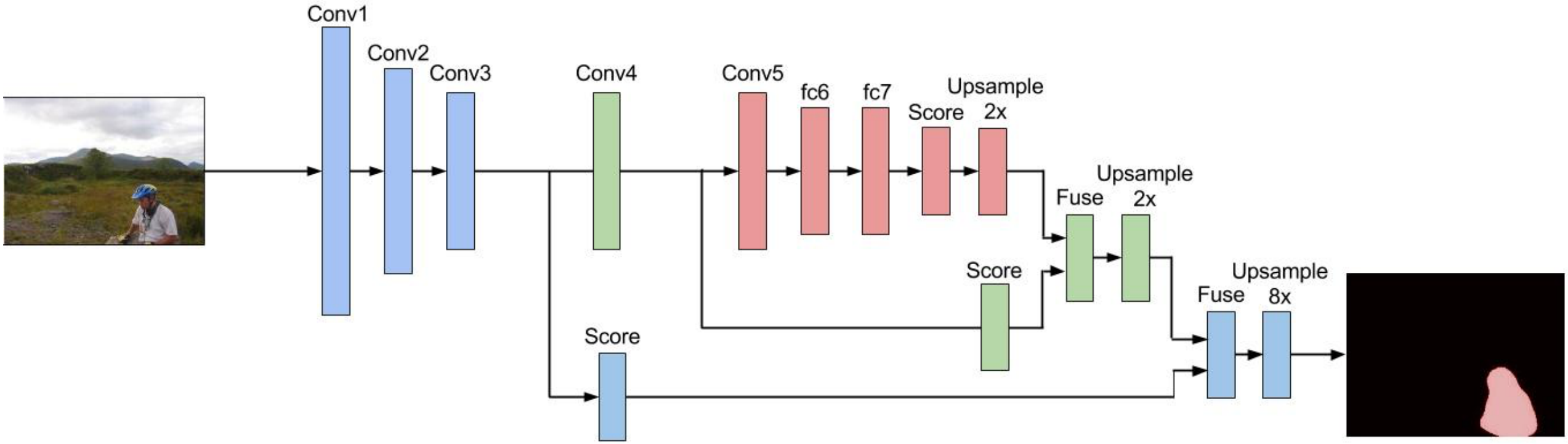
input image | stride 32 | stride 16 | stride 8 | ground truth



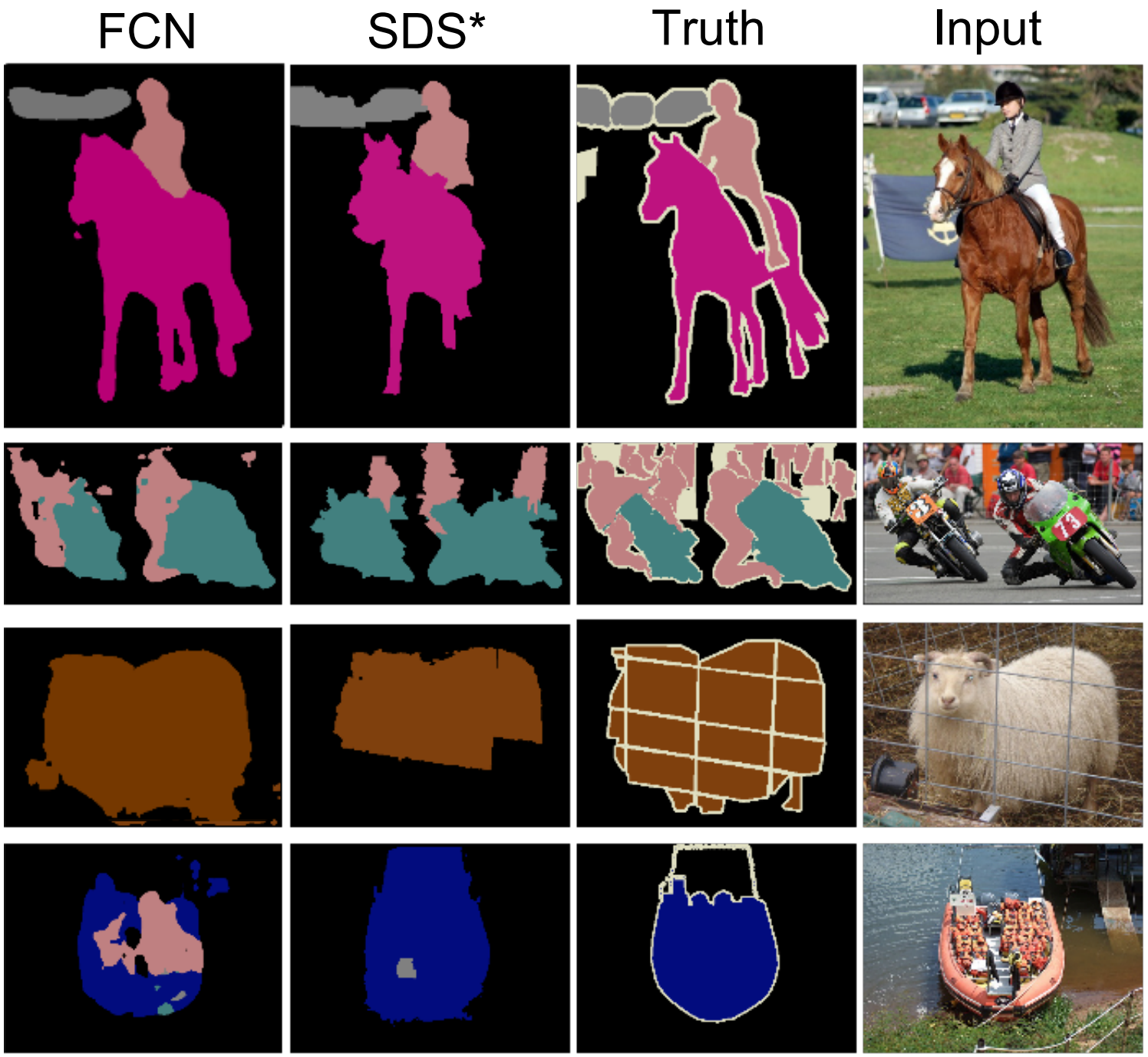
no skips | 1 skip | 2 skips

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

Overview

- Body pose tracking
  - Combine Convnet with graphical model [Thompson et al. NIPS 2014]
- Methods for semantic segmentation of scene
  - Output is now 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
  - Image colorization [Zhang et al. ECCV 2016]

Admin Interlude

- Assignment 1 scores should have been emailed out
- Assignment 2 scores by Dec 1st
- Next lectures given by my PhD students:
  - Dec 1st: Unsupervised Learning (Sainbayar Sukhbaatar)
  - Dec 8th: Images + Text (Elman Mansimov) [WILL BE AWAY]
    - Need to reschedule office hours.
  - Dec 15th: Generative models of images (Emily Denton)
- Project showcase: Mon Dec 19th.
  - Attendance not mandatory, but you must still make poster.

Beyond Object Classification with Convolutional Networks

David Eigen (NYU -> Clarifai)

Rob Fergus (Facebook / NYU)

Motivation

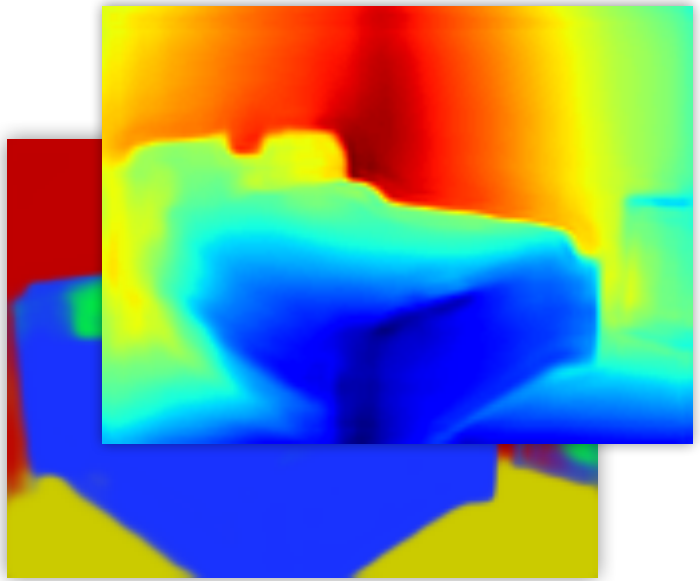
Input Image

Semantic Map

- Understand input scene
  - Semantic
  - Geometric

Motivation

Input Image

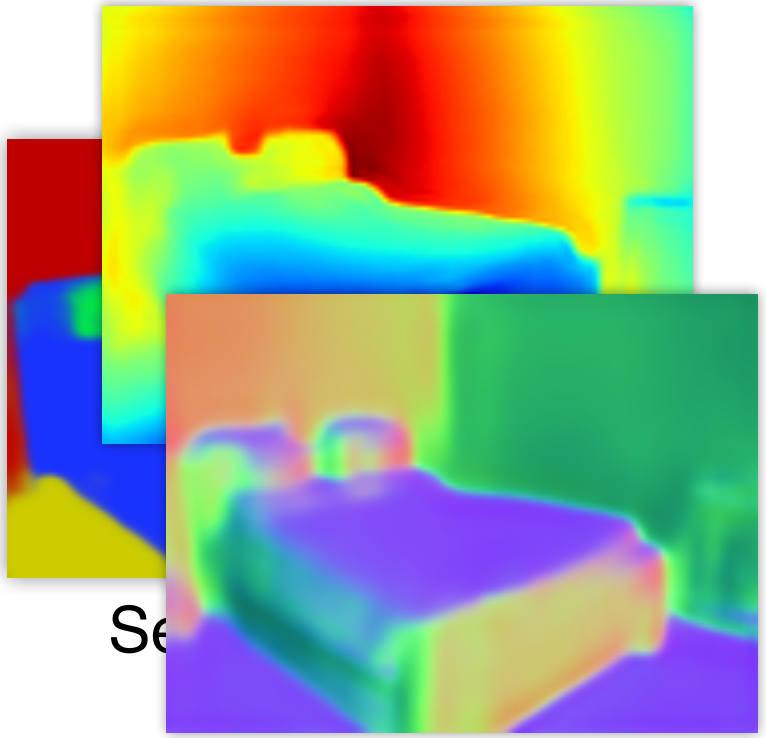


Semantic Map

- Understand input scene
  - Semantic
  - Geometric

Motivation

Input Image



Depth

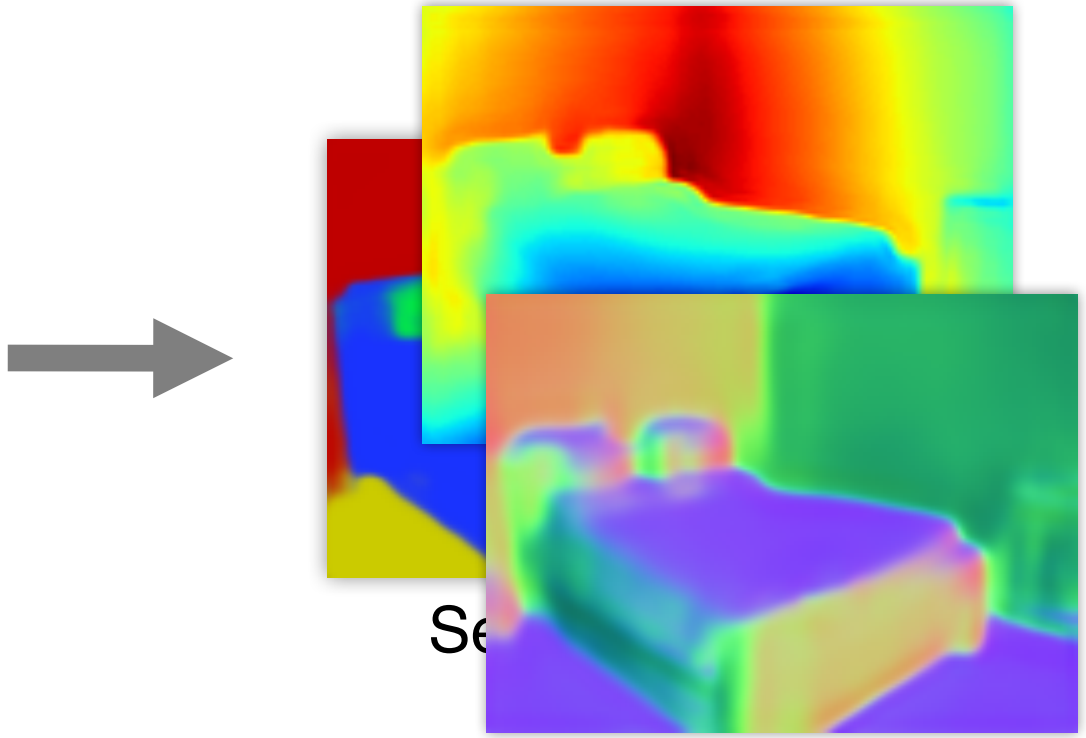
Se

Normals

- Understand input scene
  - Semantic
  - Geometric

Motivation

Input Image



Depth

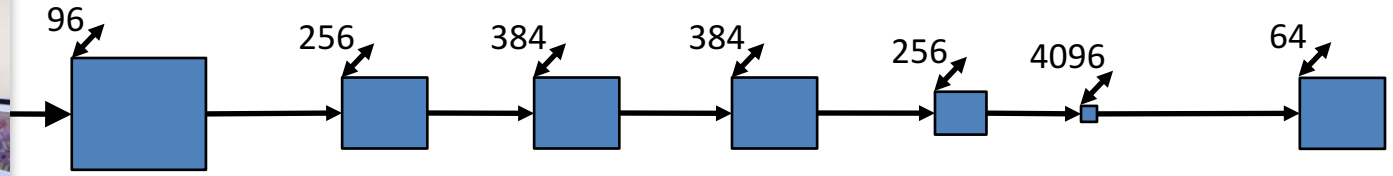
Se

Normals

- **Predict Pixel Maps from a Single Image**

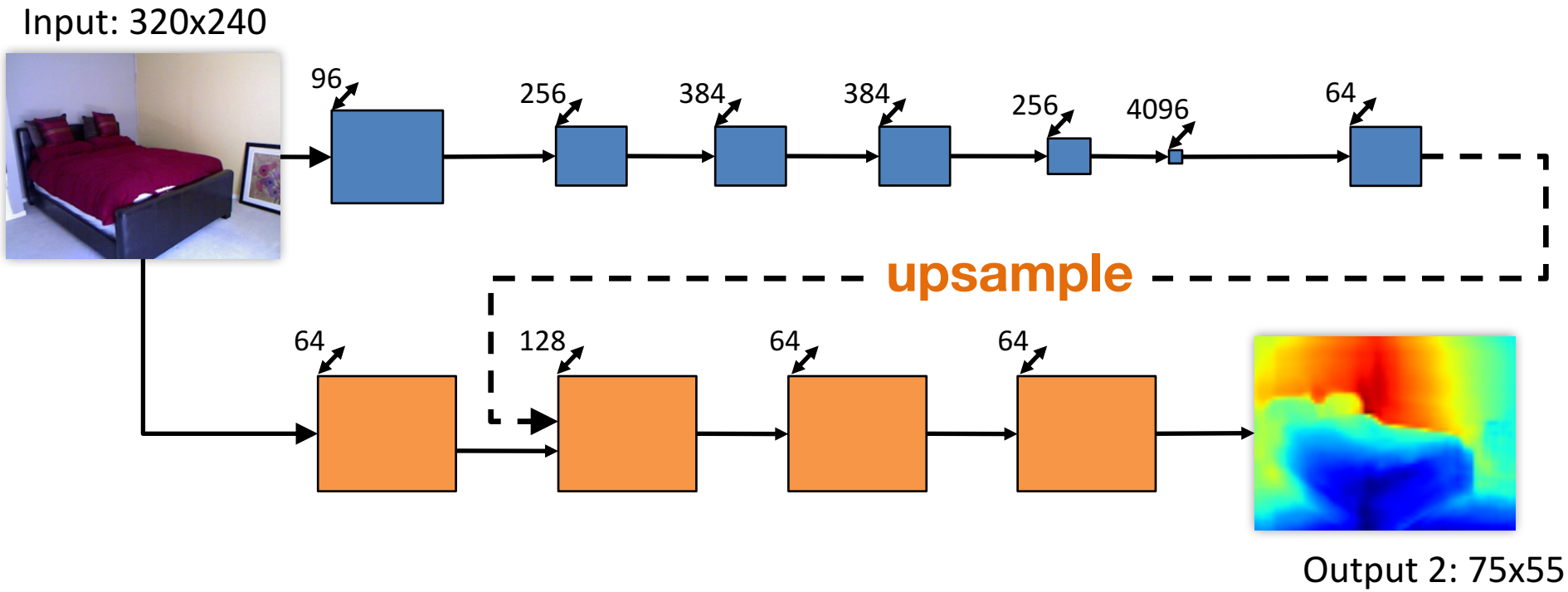
Architecture

Input: 320x240

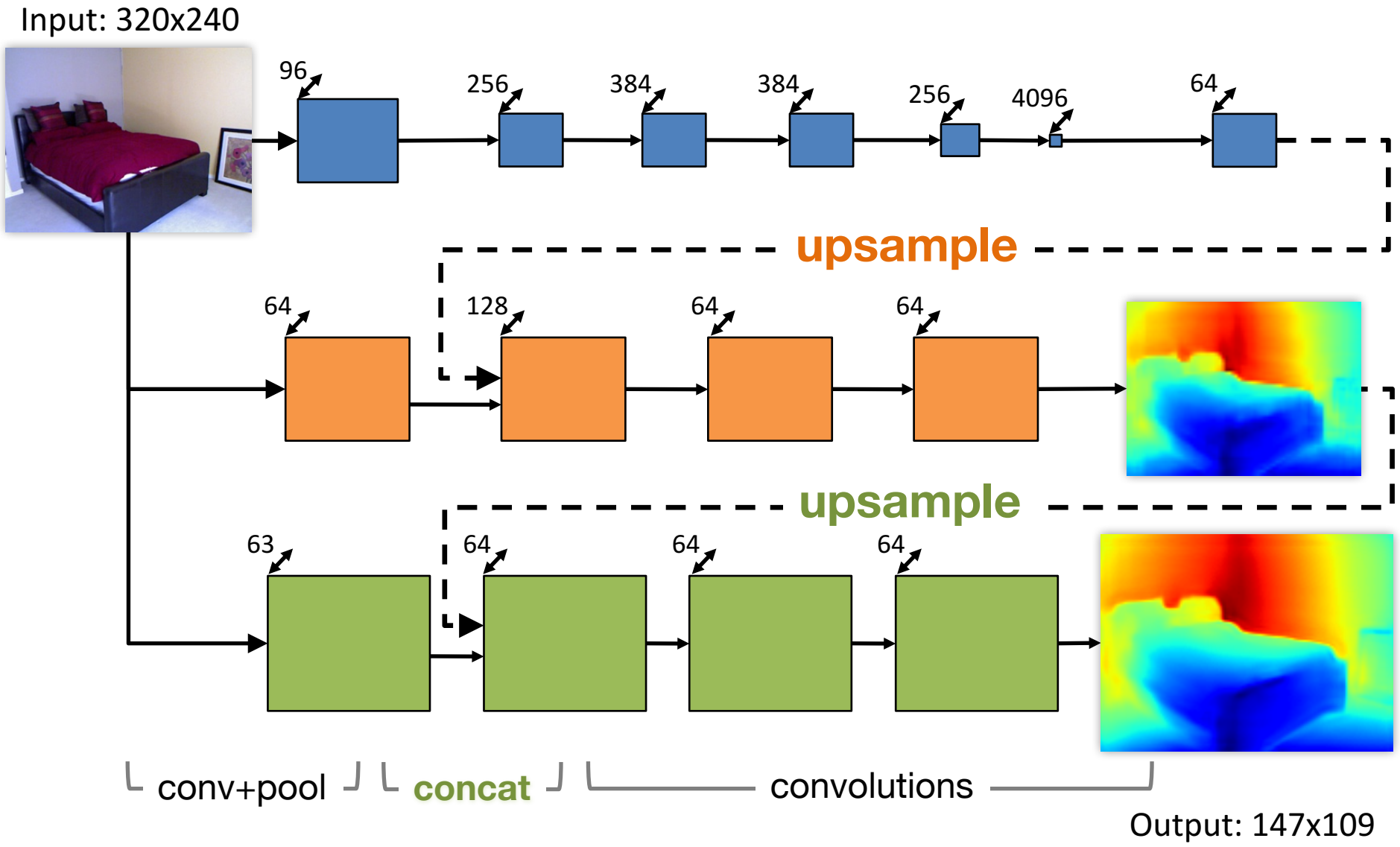


Output 1: 19x14

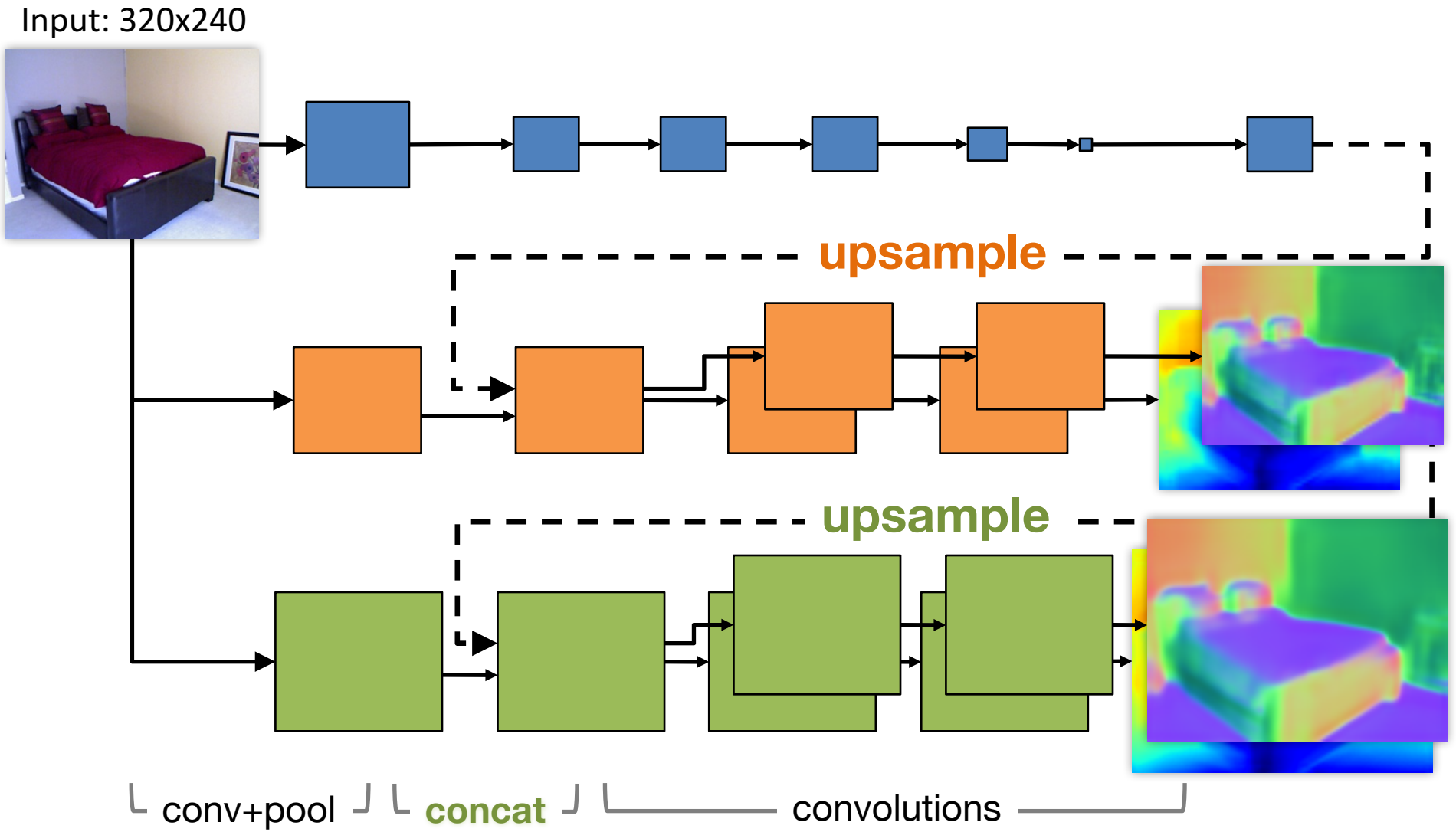
Architecture



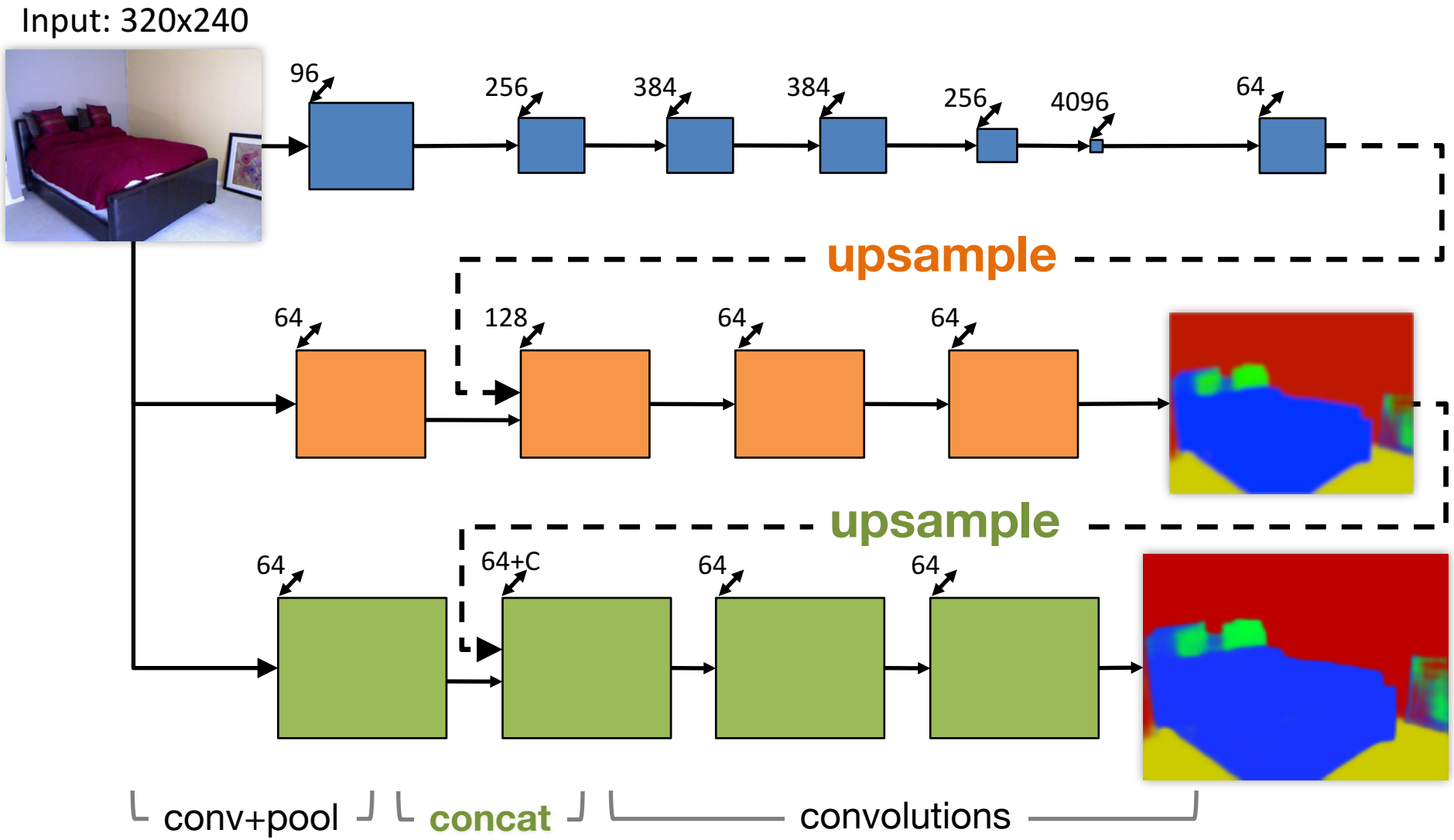
Architecture



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

Label

Training

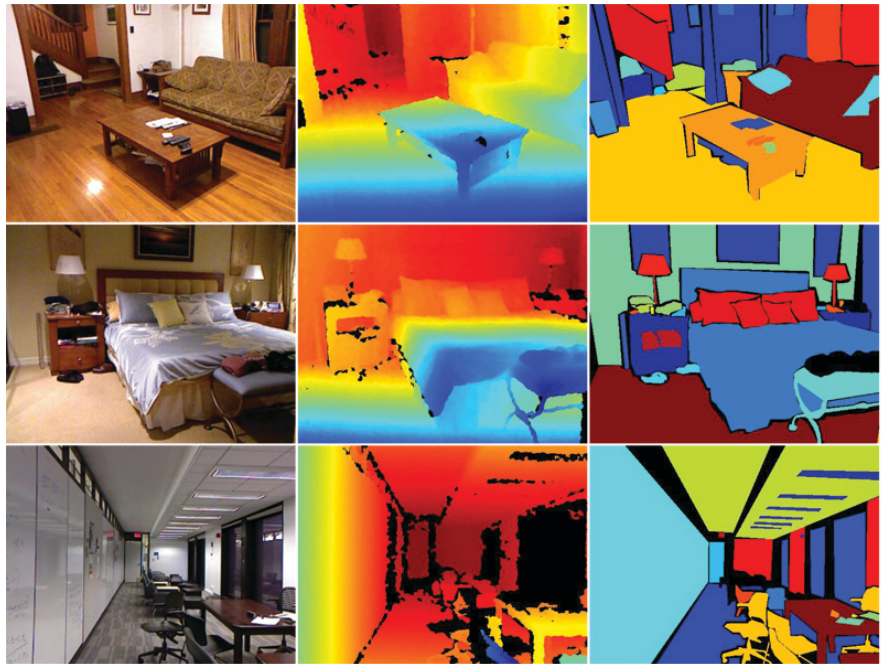
- Pre-train Alexnet/VGGnet scale 1 with Imagenet
- Scale 2 & 3 random initialization
- Joint train layers 1 & 2 for each task
  - Loss on output of layer 2

- Fix layers 1 & 2, train layer 3

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

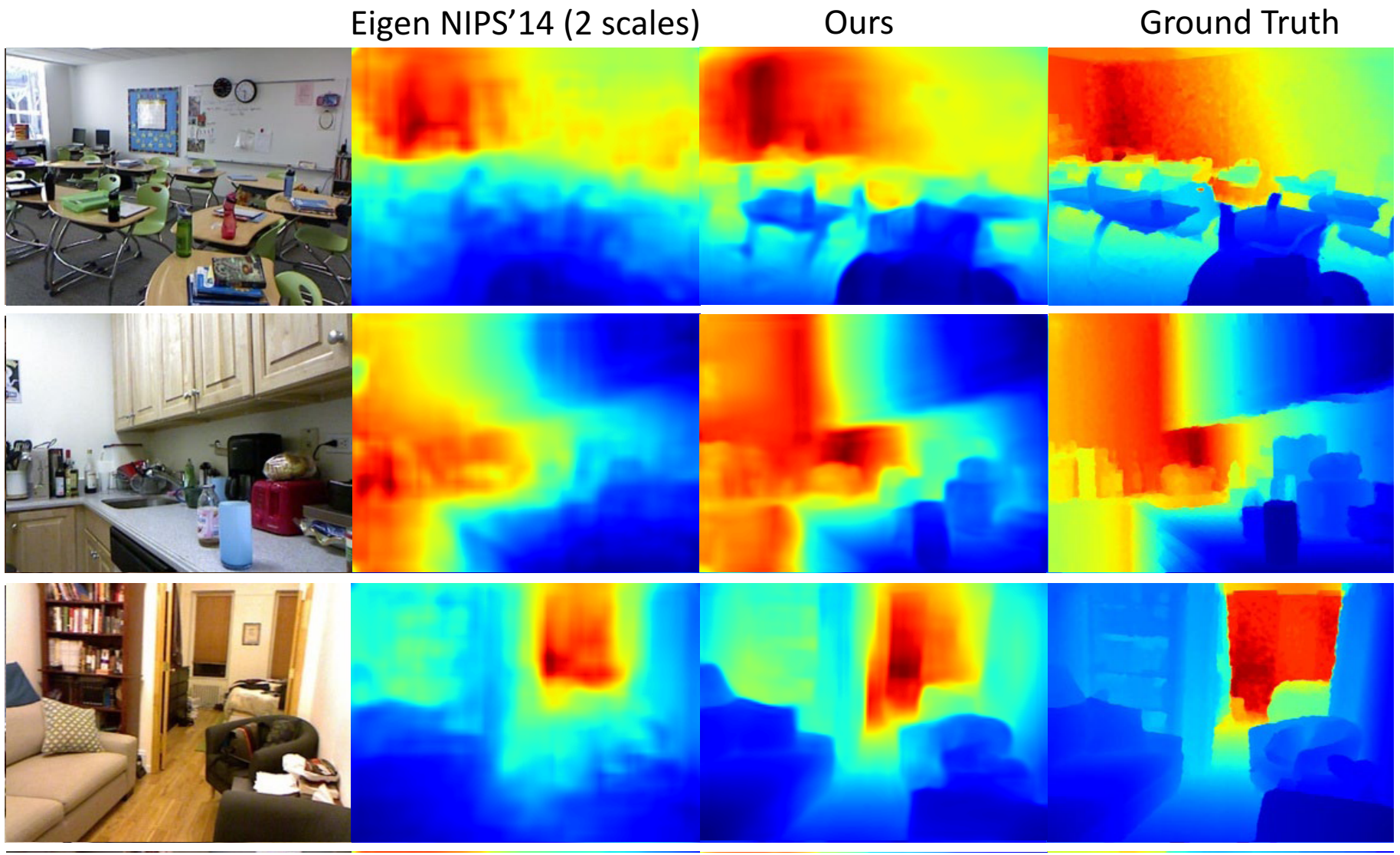
Evaluation

- NYU Depth dataset
  - RGB, Depth and per-pixel labels
  - Indoor scenes
- Supervised training of models



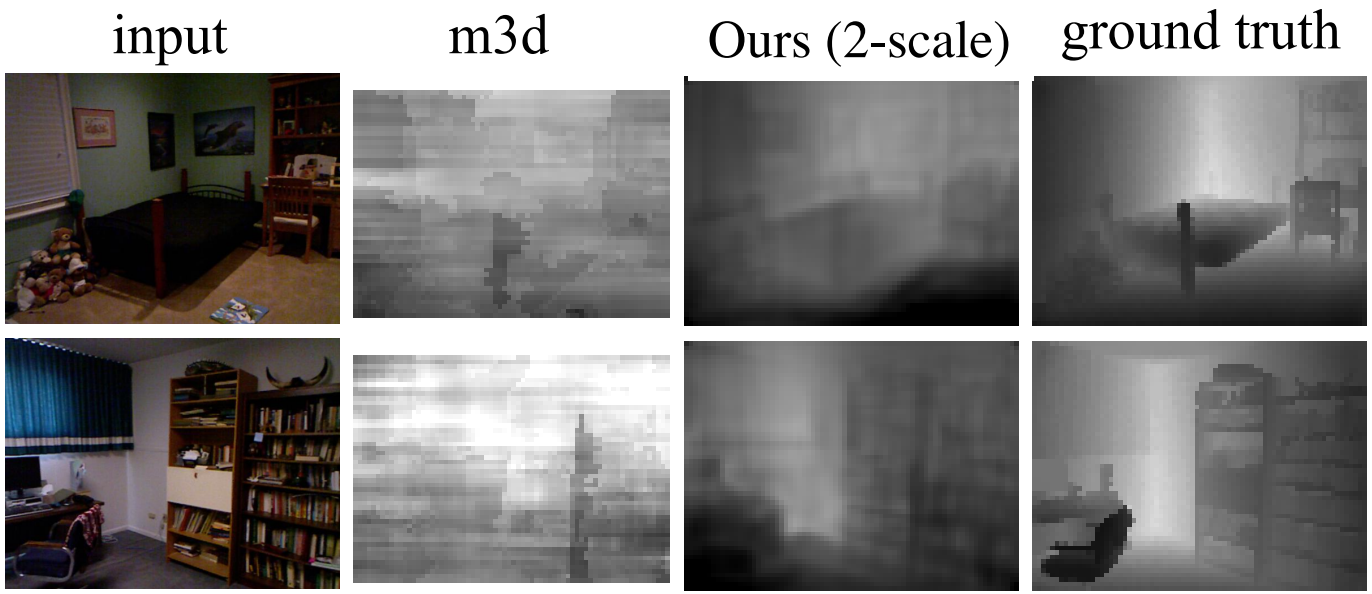
- Compare to range of other methods
  - Also on SIFTFlow and PASCAL VOC'11

Depths Comparison



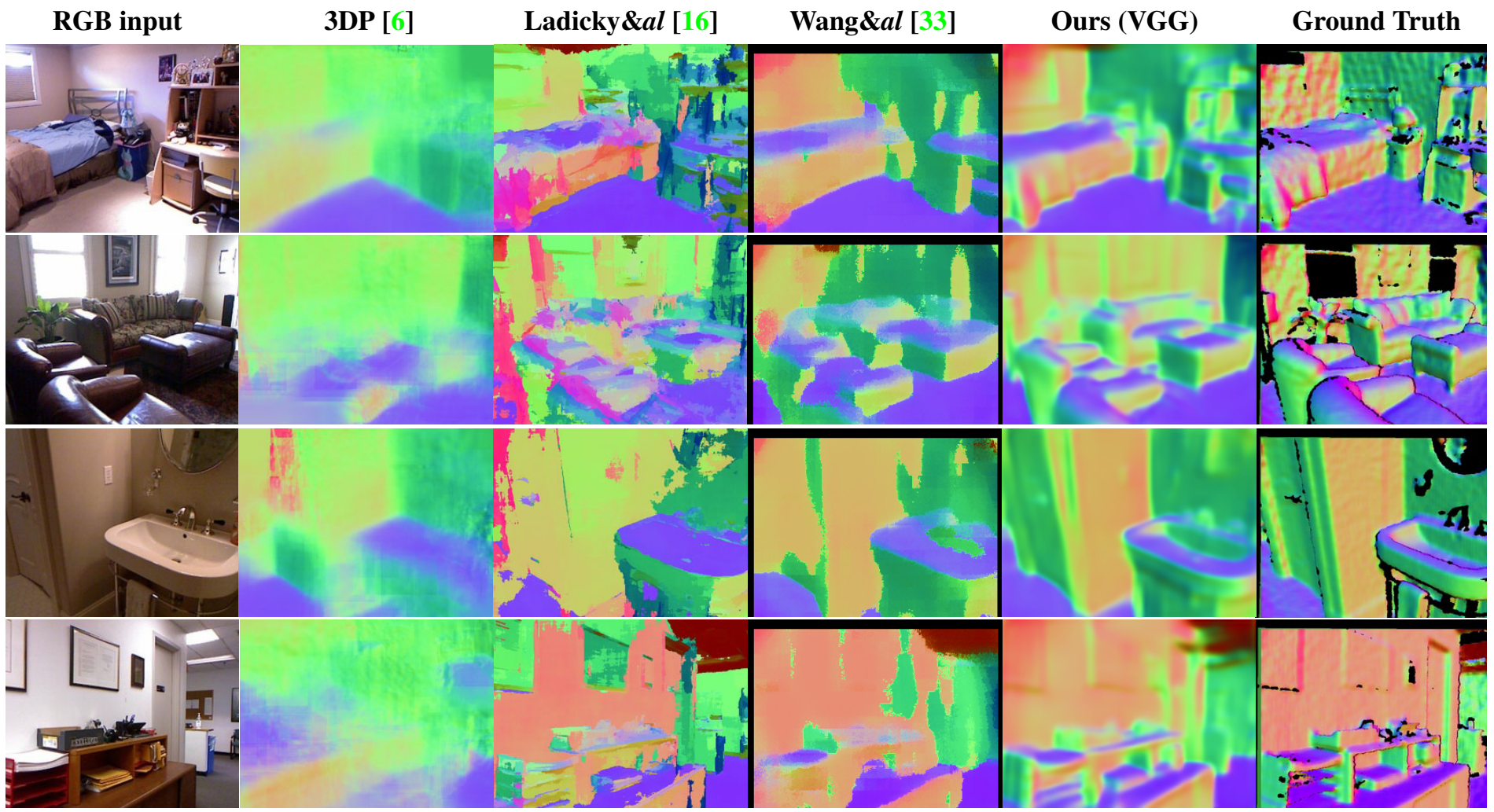
Depth Comparison

- 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

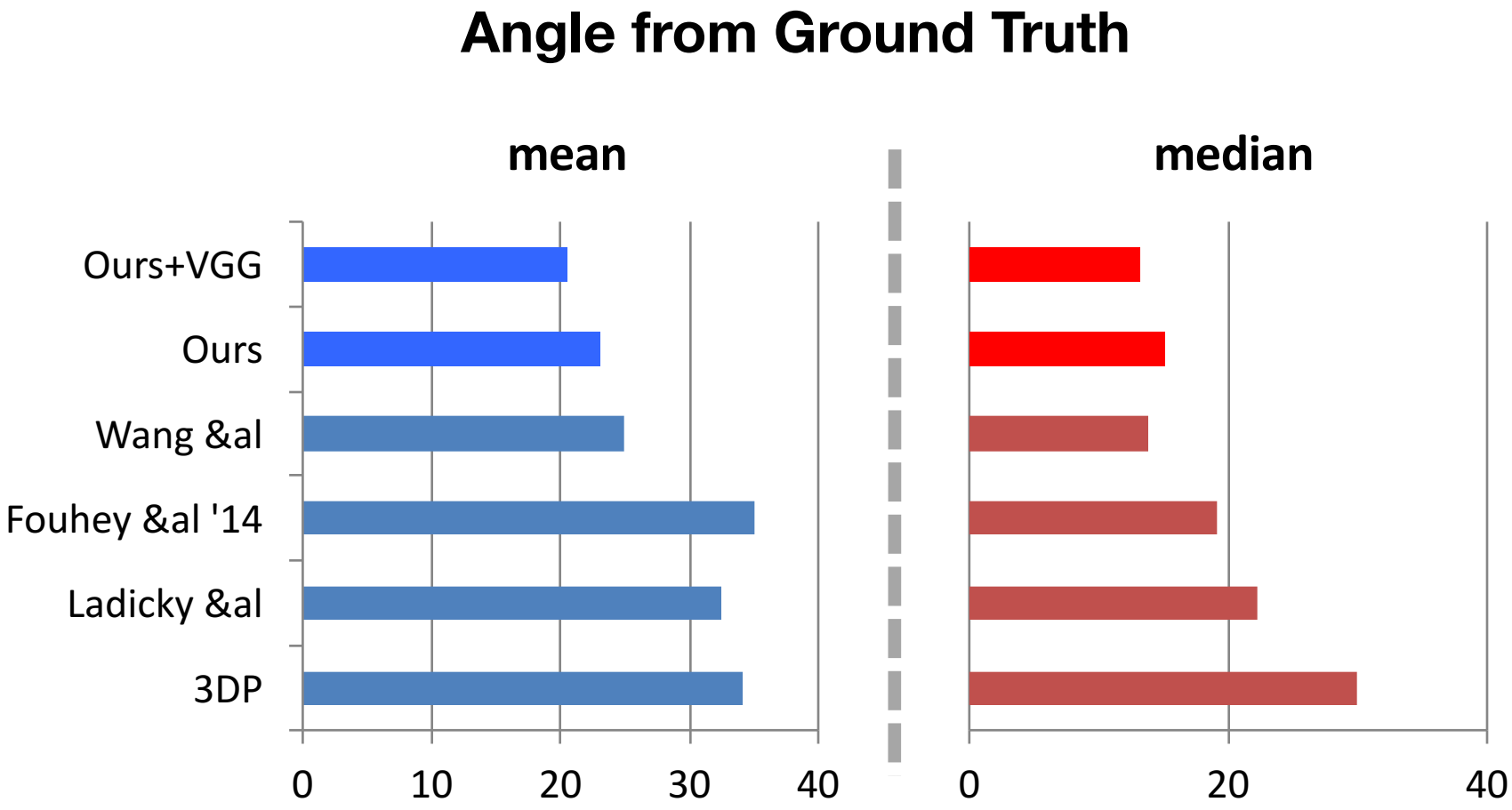


Surface Normals

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

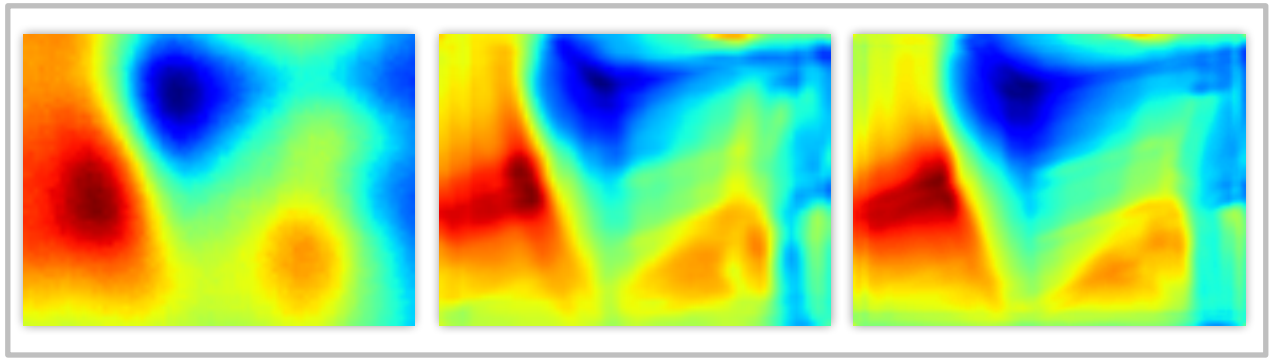


Output from each scale

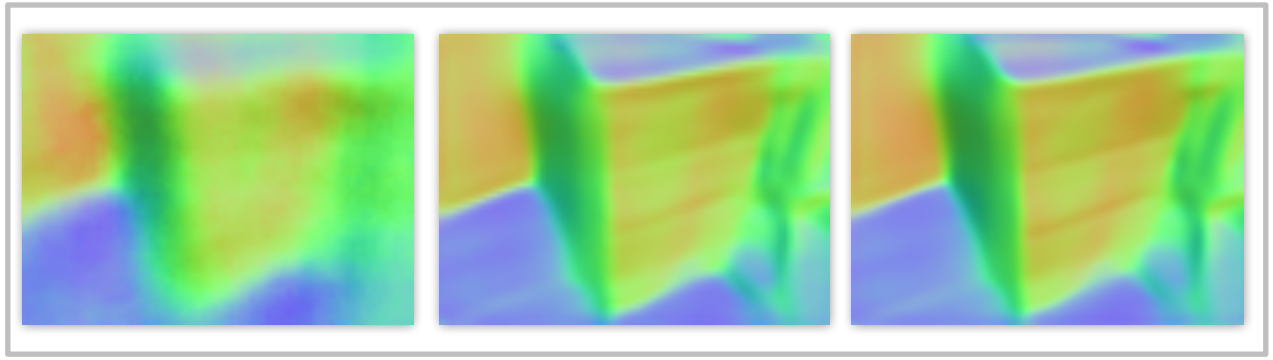
input

depth

far near

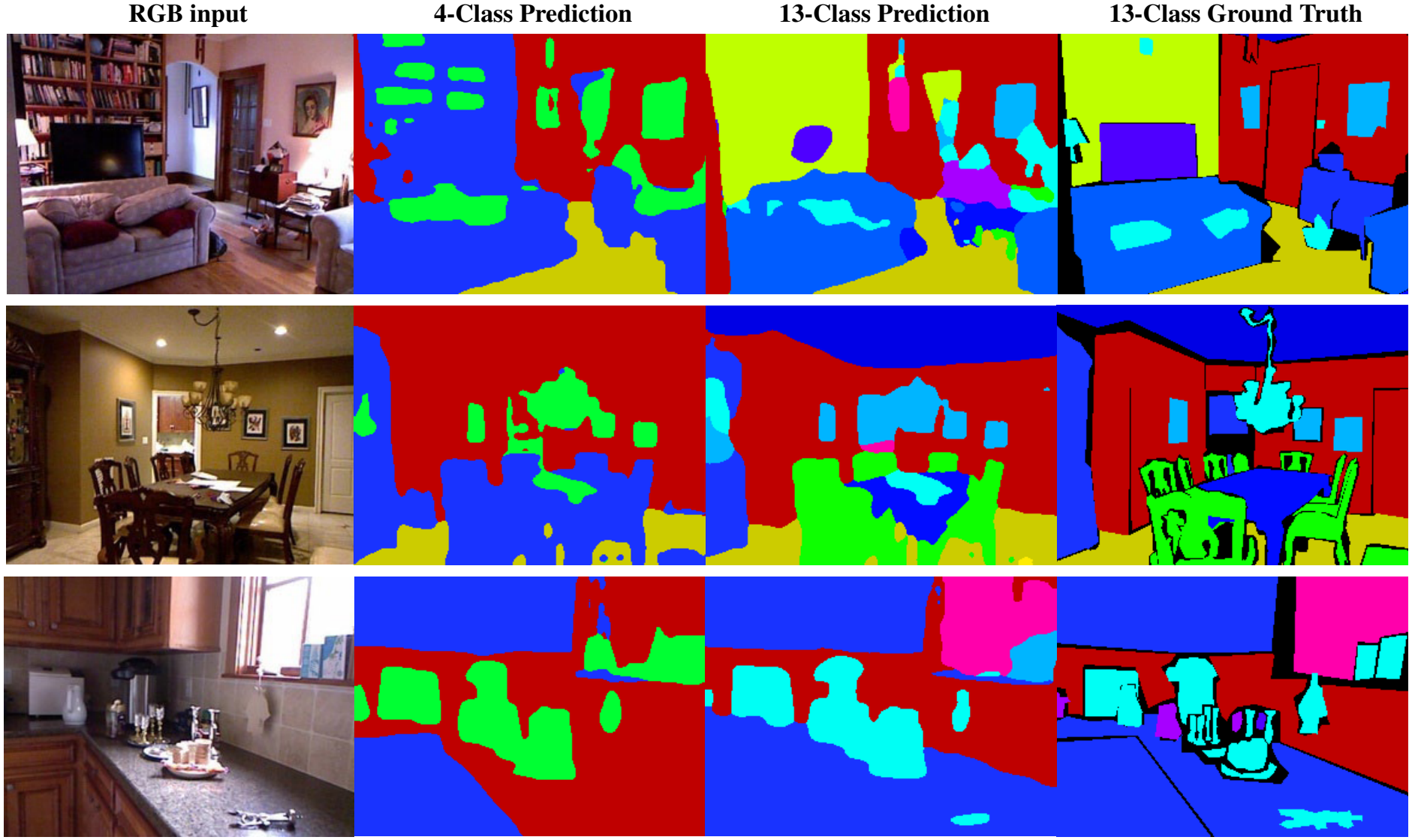


coarse **fine**



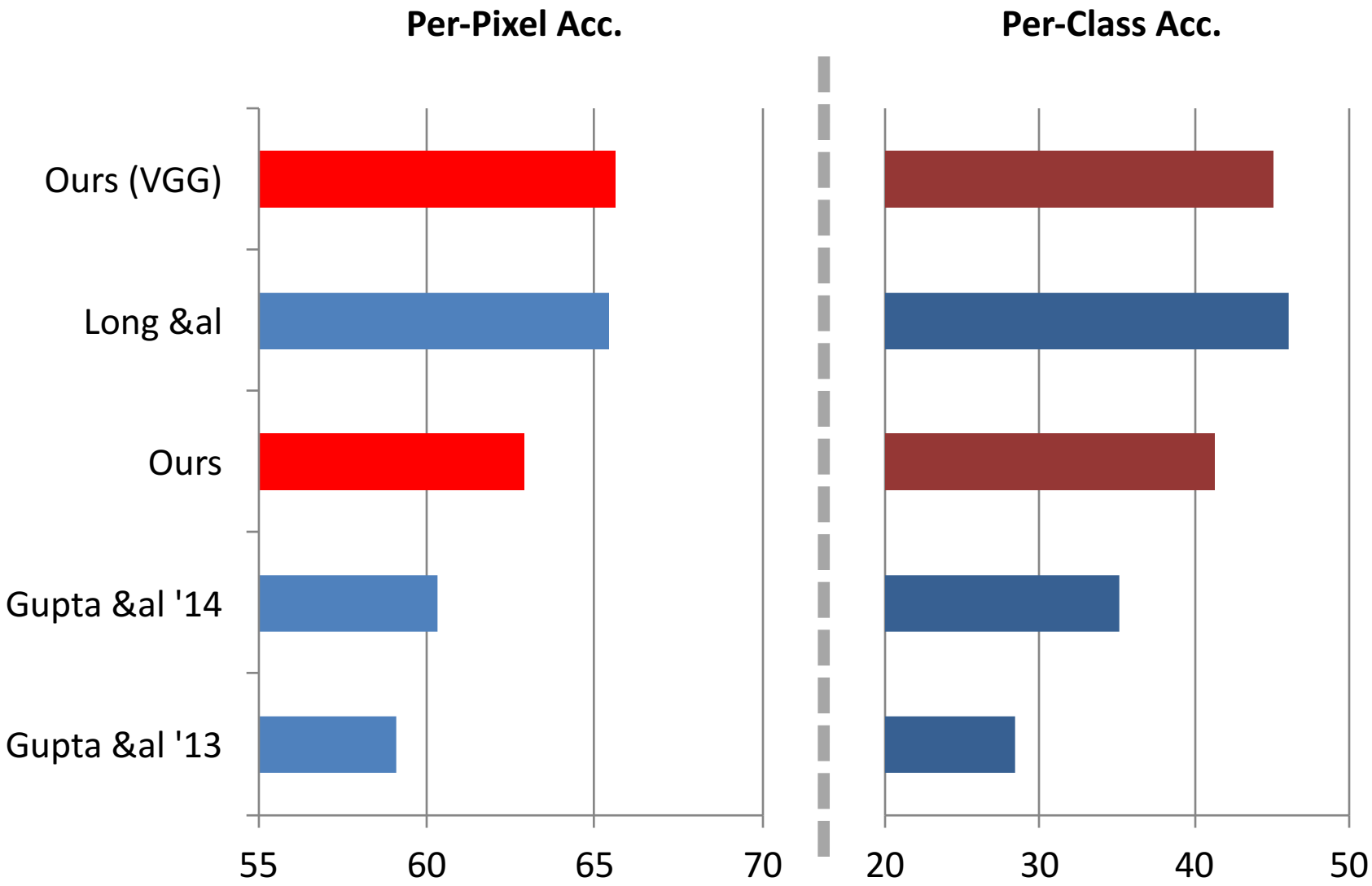
normals

Semantic Labels: NYUD



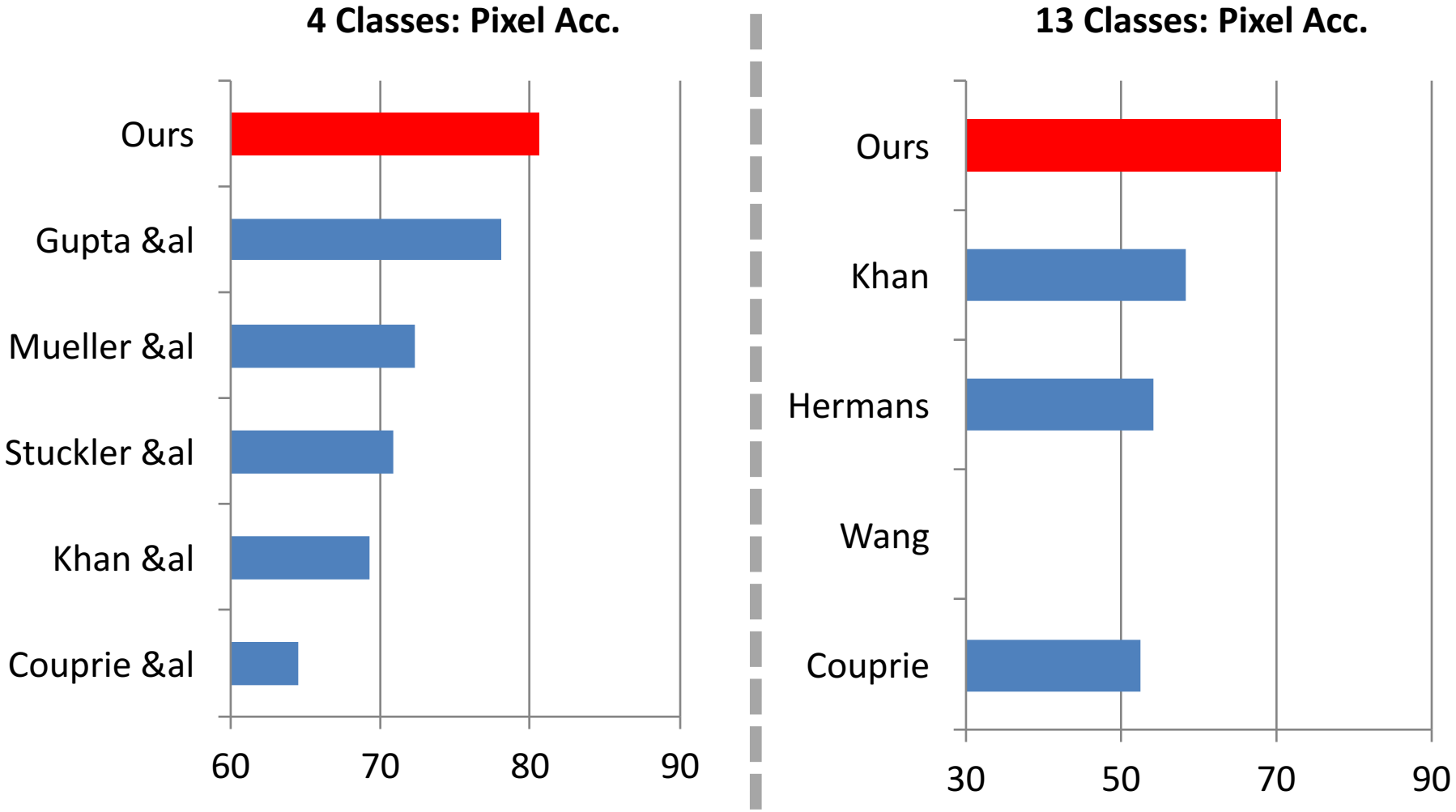
Results: NYUD 40 Classes

- 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

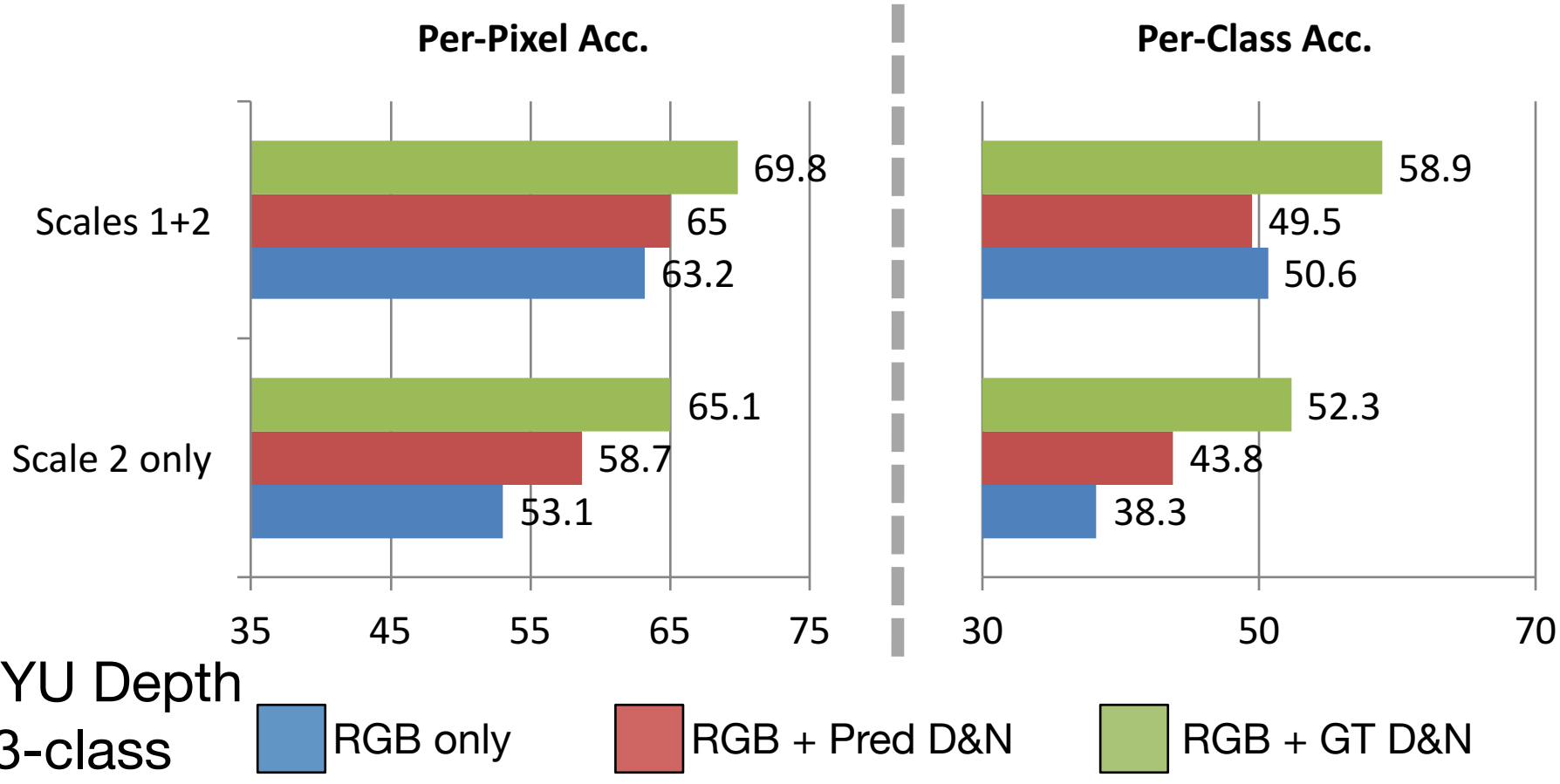
- On NYU Depth

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

Summary

- Relatively simple multi-scale model gives good results for depth, normals & labels
- Coarse interpretation of scene important for understanding depth/normals
- See ICCV 2015 paper: “Predicting Depth, Surface Normals and Semantic Labels with a Common
Multi-Scale Convolutional Architecture”, D. Eigen and R. Fergus, arXiv 1411.4734
- Code available

Overview

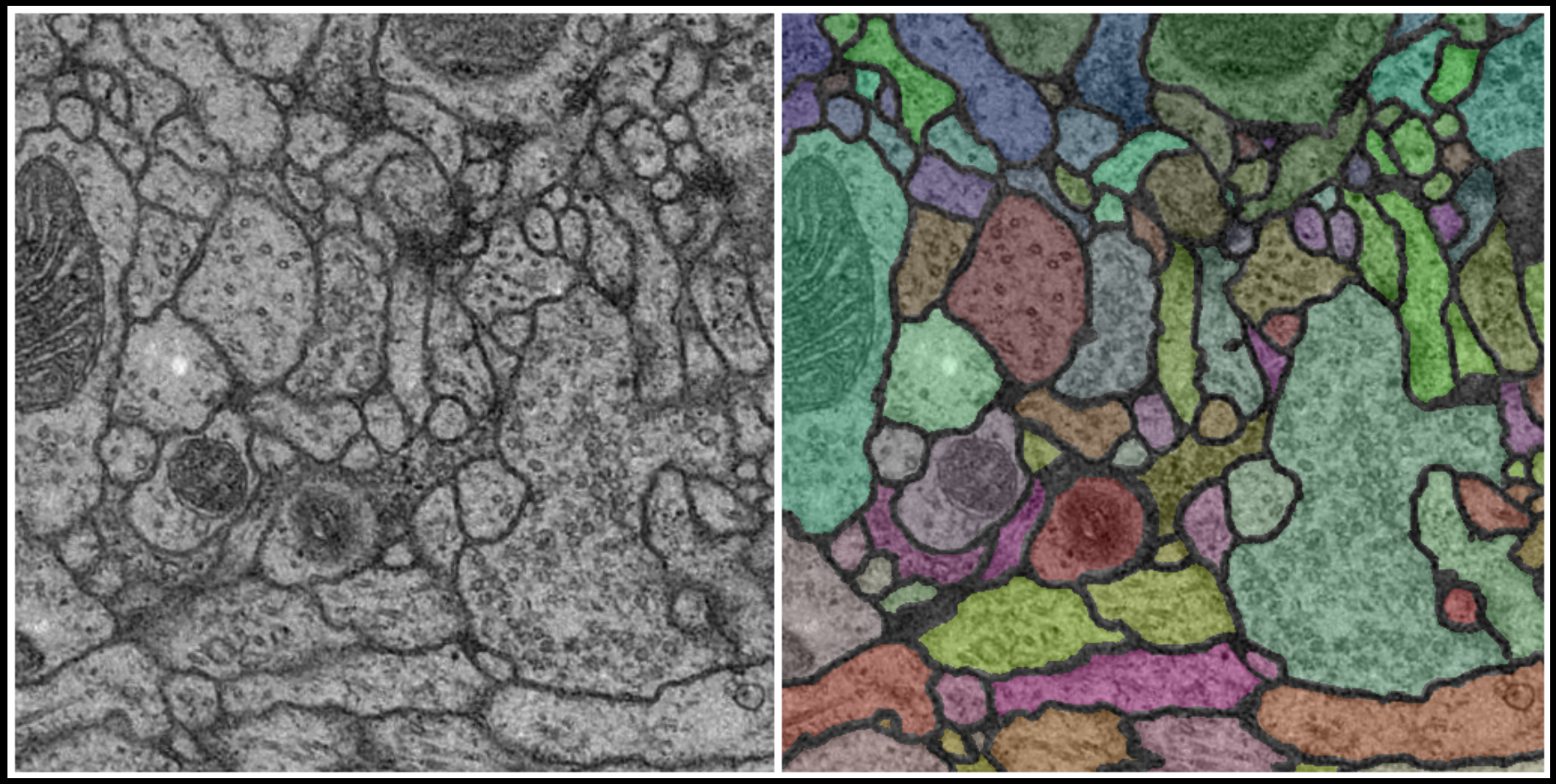
- Body pose tracking
  - Combine Convnet with graphical model [Thompson et al. NIPS 2014]

- Methods for semantic segmentation of scene
  - Output is now 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
  - Image colorization [Zhang et al. ECCV 2016]

Segmentation

- Ciresan et al. “DNN segment neuronal membranes…” NIPS 2012
- Turaga et al. “Maximin learning of image segmentation” NIPS 2009

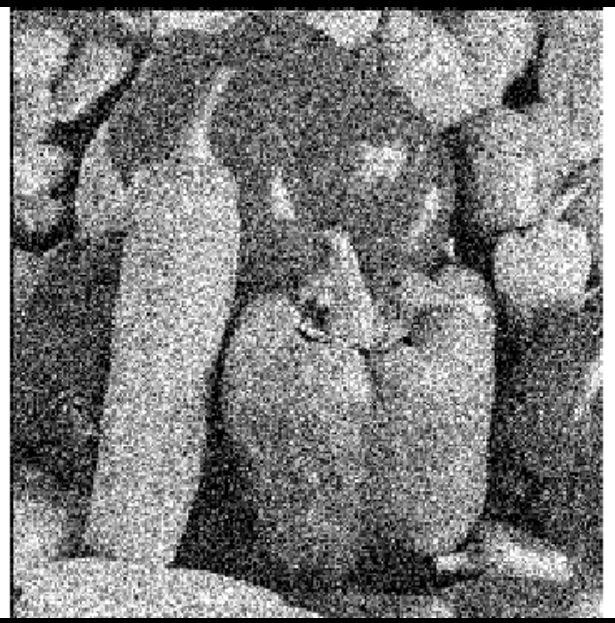


Denoising with ConvNets

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

Original

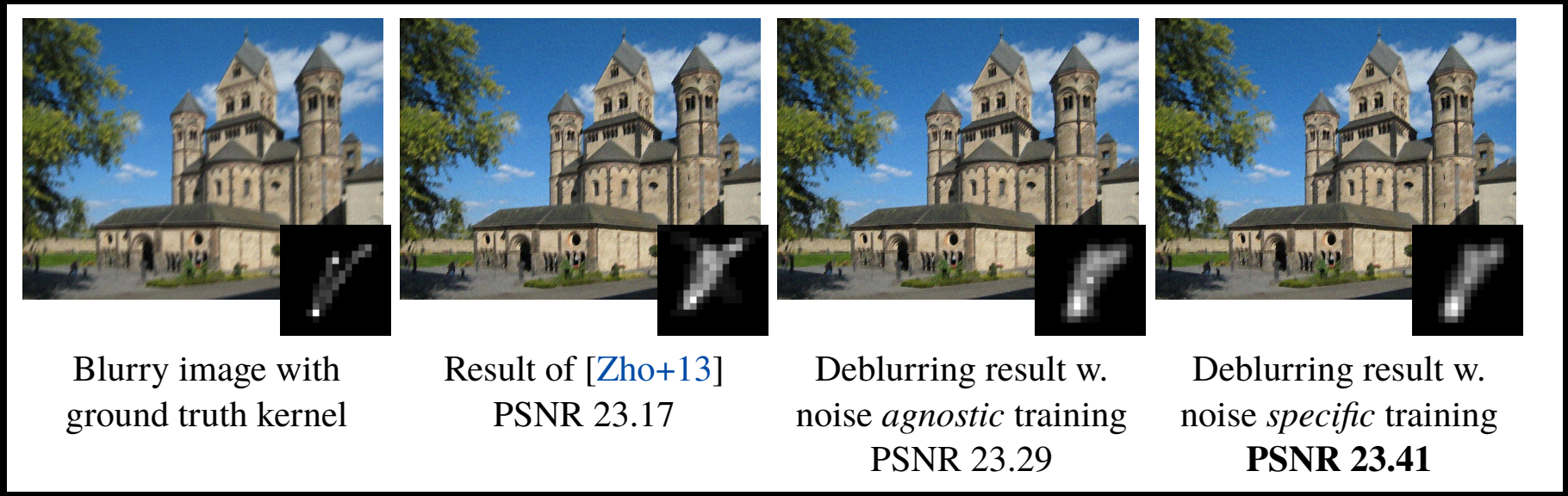
Noised



Denoised

Deblurring with Convnets

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



Blurry image with ground truth kernel

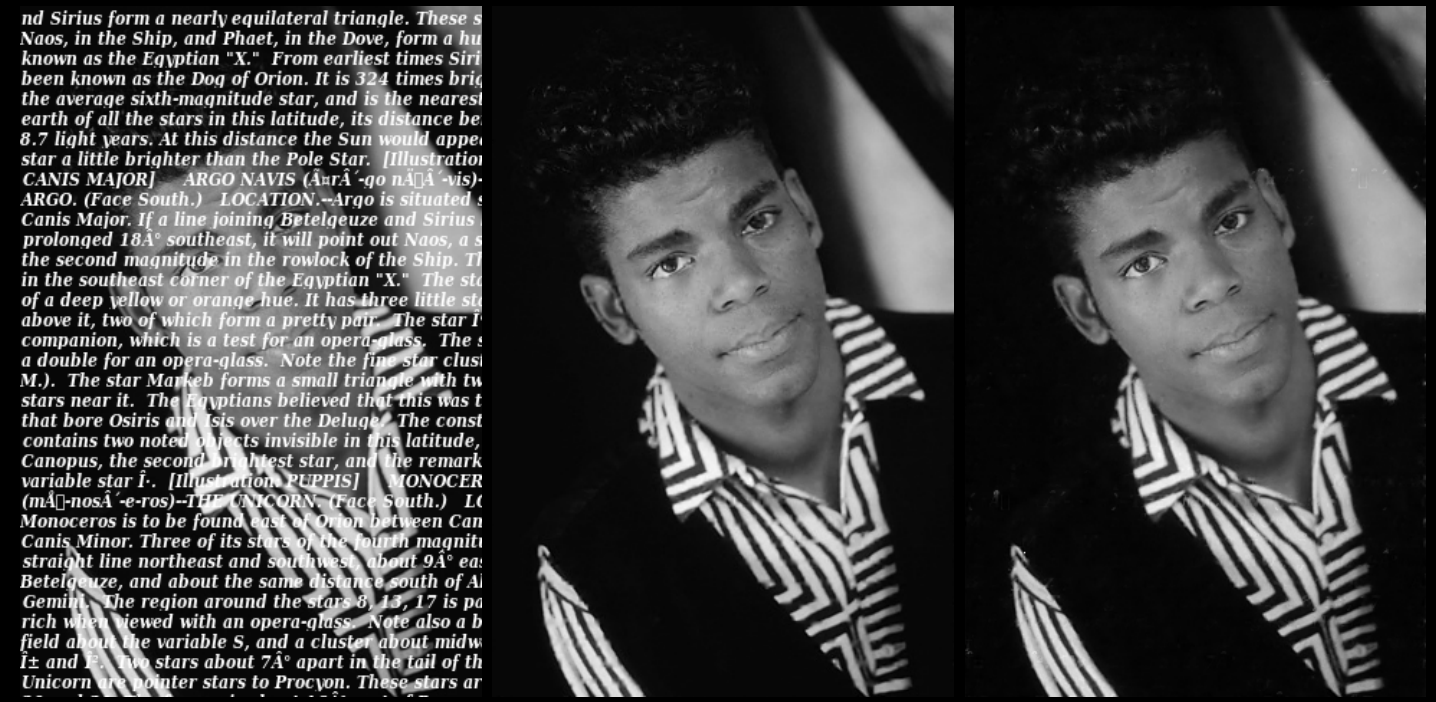
Result of [Zho+13] PSNR 23.17

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 '14 Schmid CVPR'10 Köhler et al.

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

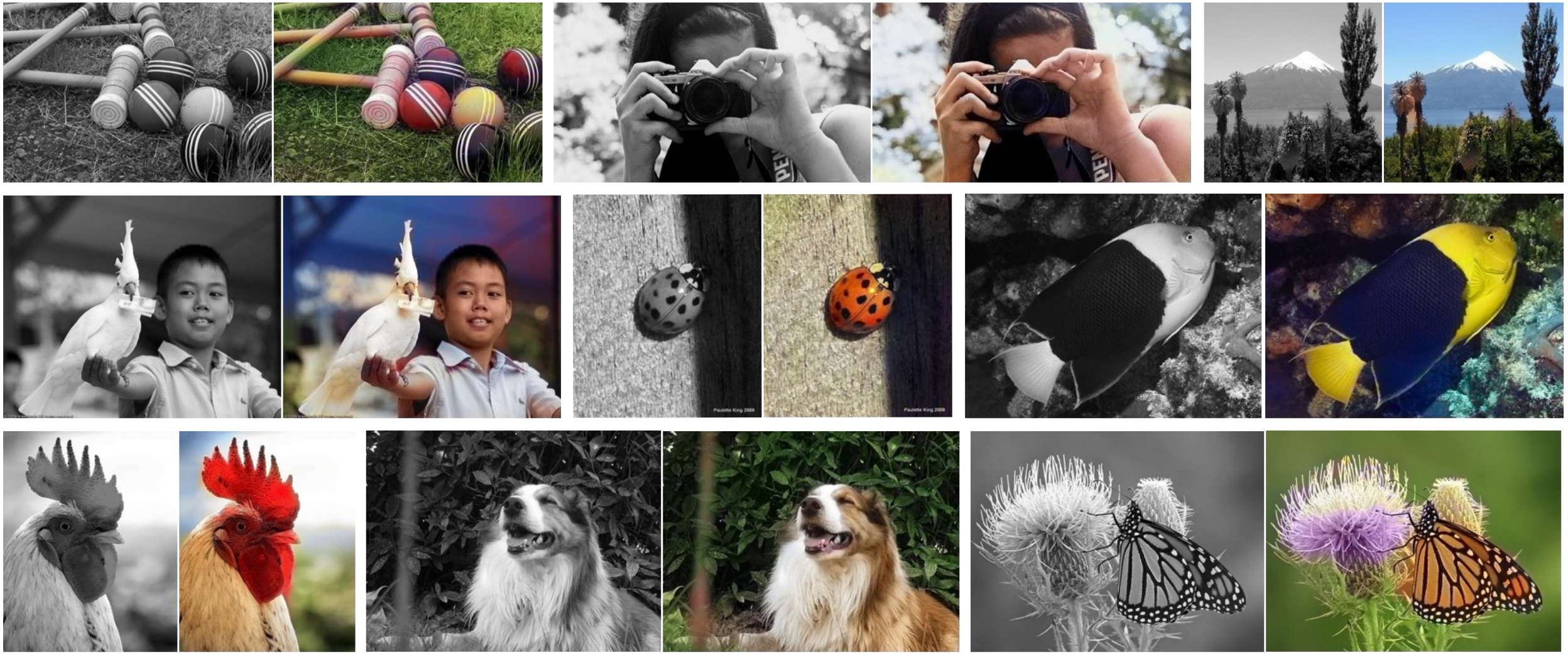
Rain Sequence

Each frame processed independently

David Eigen, Dilip Krishnan and Rob Fergus
ICCV 2013

Overview

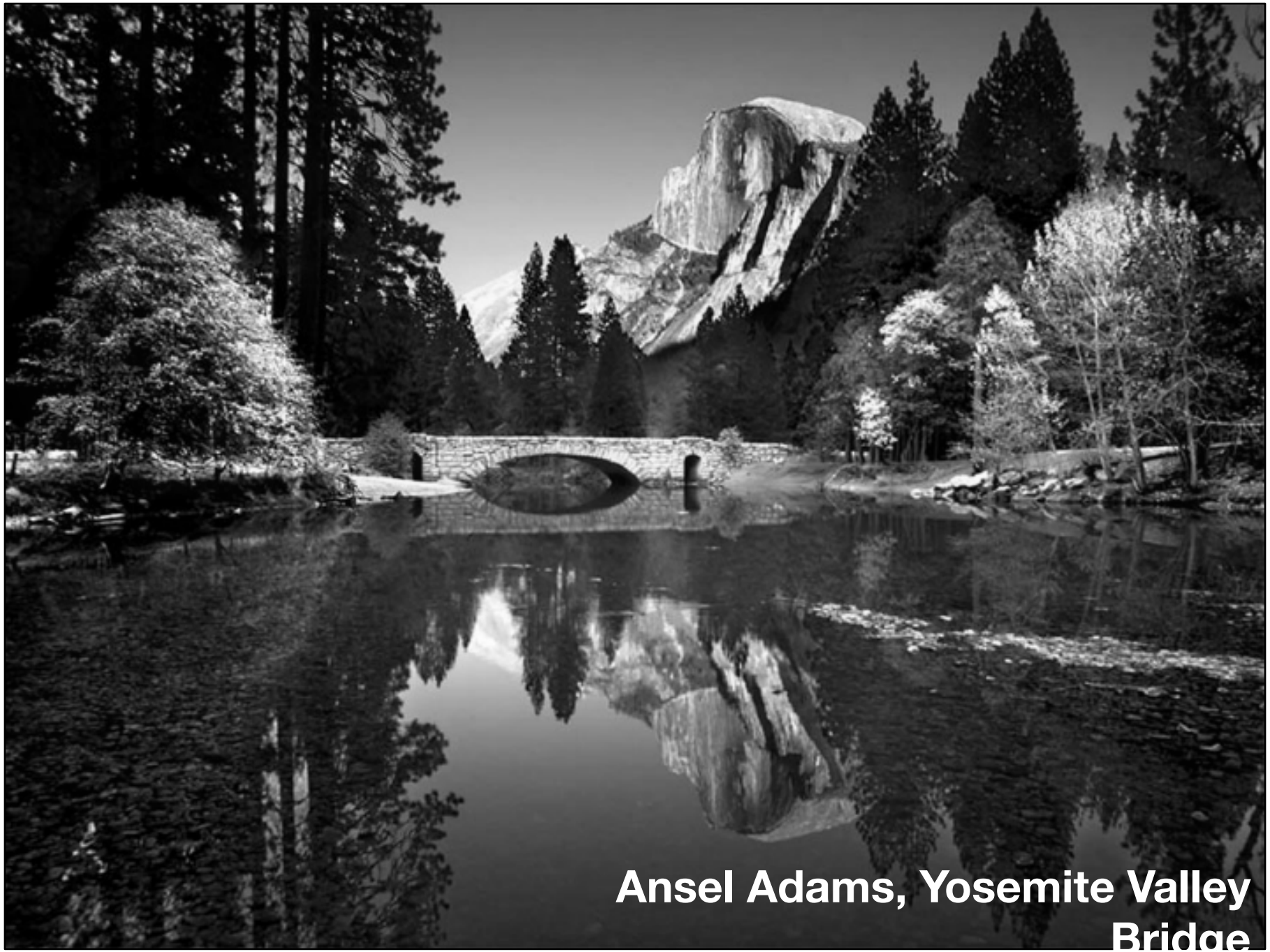
- Body pose tracking
  - Combine Convnet with graphical model [Thompson et al. NIPS 2014]
- Methods for semantic segmentation of scene
  - Output is now also an image
  - Fully Convolutional Nets [Long et al., CVPR 2015]
  - Depth, normals and semantic labels from a single image [Eigen ICCV 2015]
- Image processing with Convnets
  - Image colorization [Zhang et al. ECCV 2016]

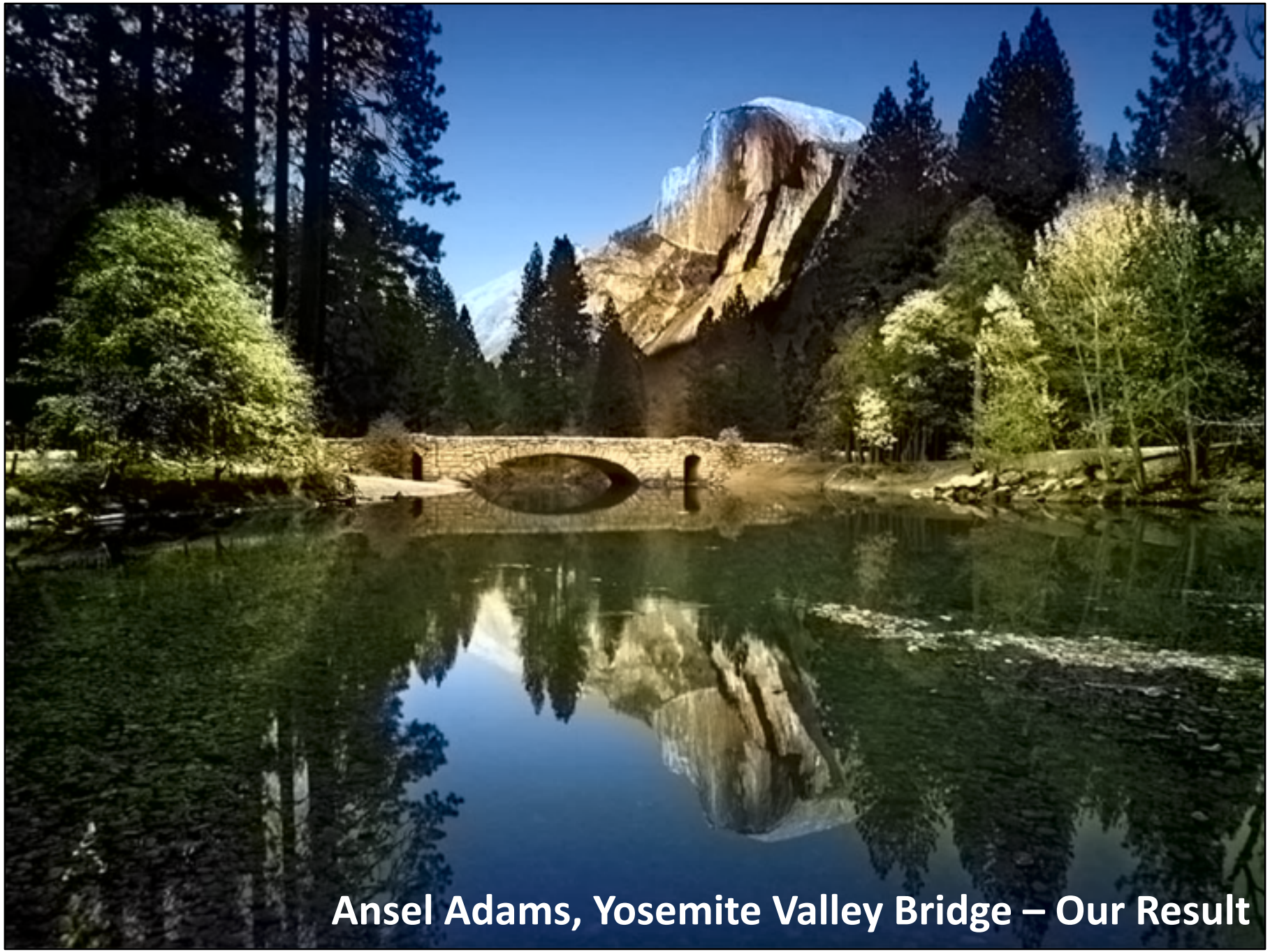


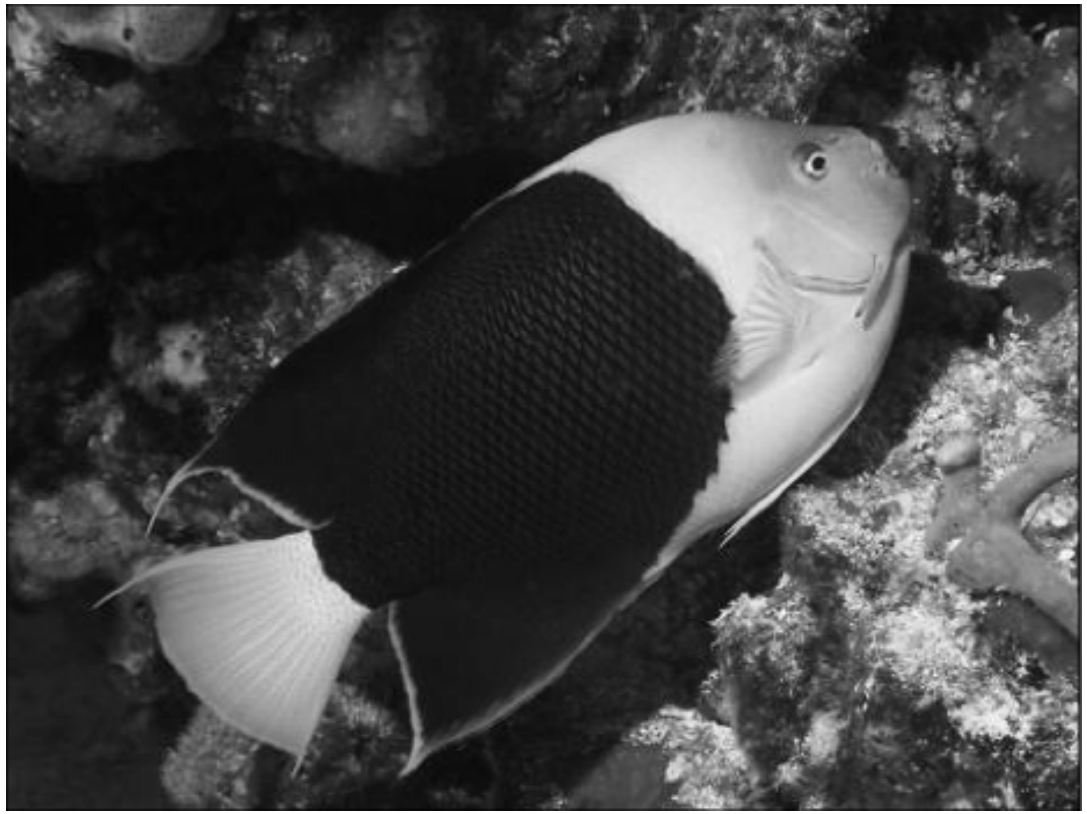
Colorful Image Colorization

Richard Zhang, Phillip Isola, Alexei (Alyosha) Efros

richzhang.github.io/colorization


Ansel Adams, Yosemite Valley
Bridge


Ansel Adams, Yosemite Valley Bridge – Our Result



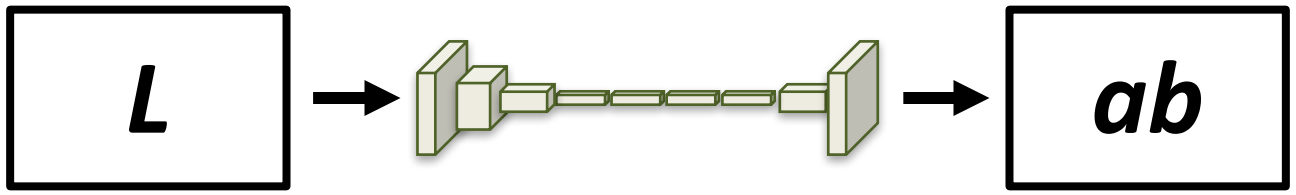
$\mathcal{F}$

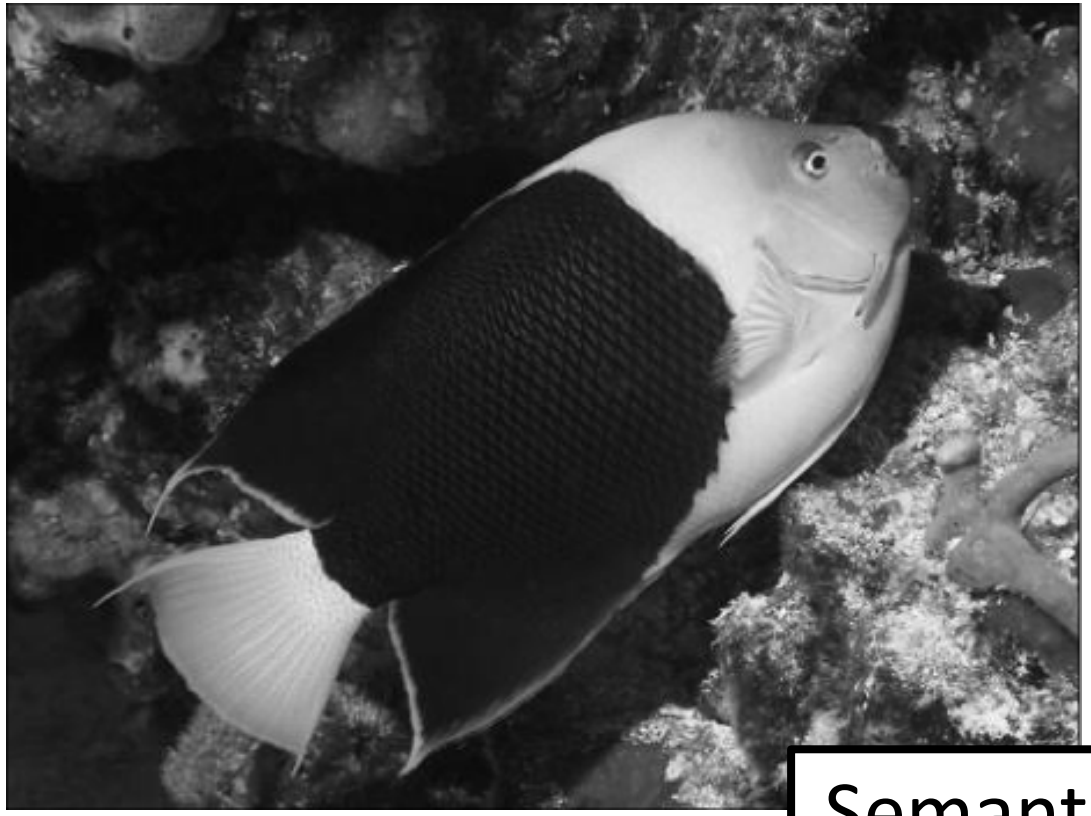
Grayscale image: *L* channel

$$\mathbf{X} \in \mathbb{R}^{H \times W \times 1}$$

Color information: *ab* channels

$$\hat{\mathbf{Y}} \in \mathbb{R}^{H \times W \times 2}$$



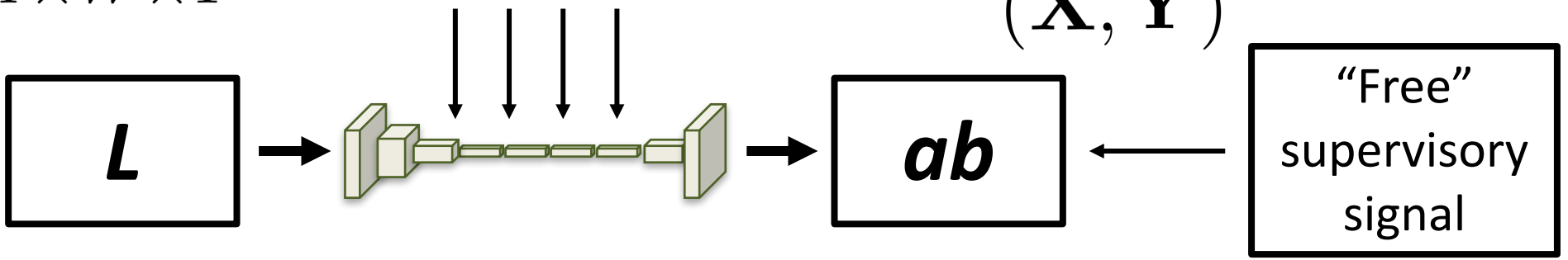


$\mathcal{F}$

Semantics? Higher-level abstraction?

Grayscale image: *L* ch ncatenate (L,ab)

$\mathbf{X} \in \mathbb{R}^{H \times W \times 1}$ $(\mathbf{X}, \hat{\mathbf{Y}})$



Inherent Ambiguity

Grayscale

Inherent Ambiguity

Our Output

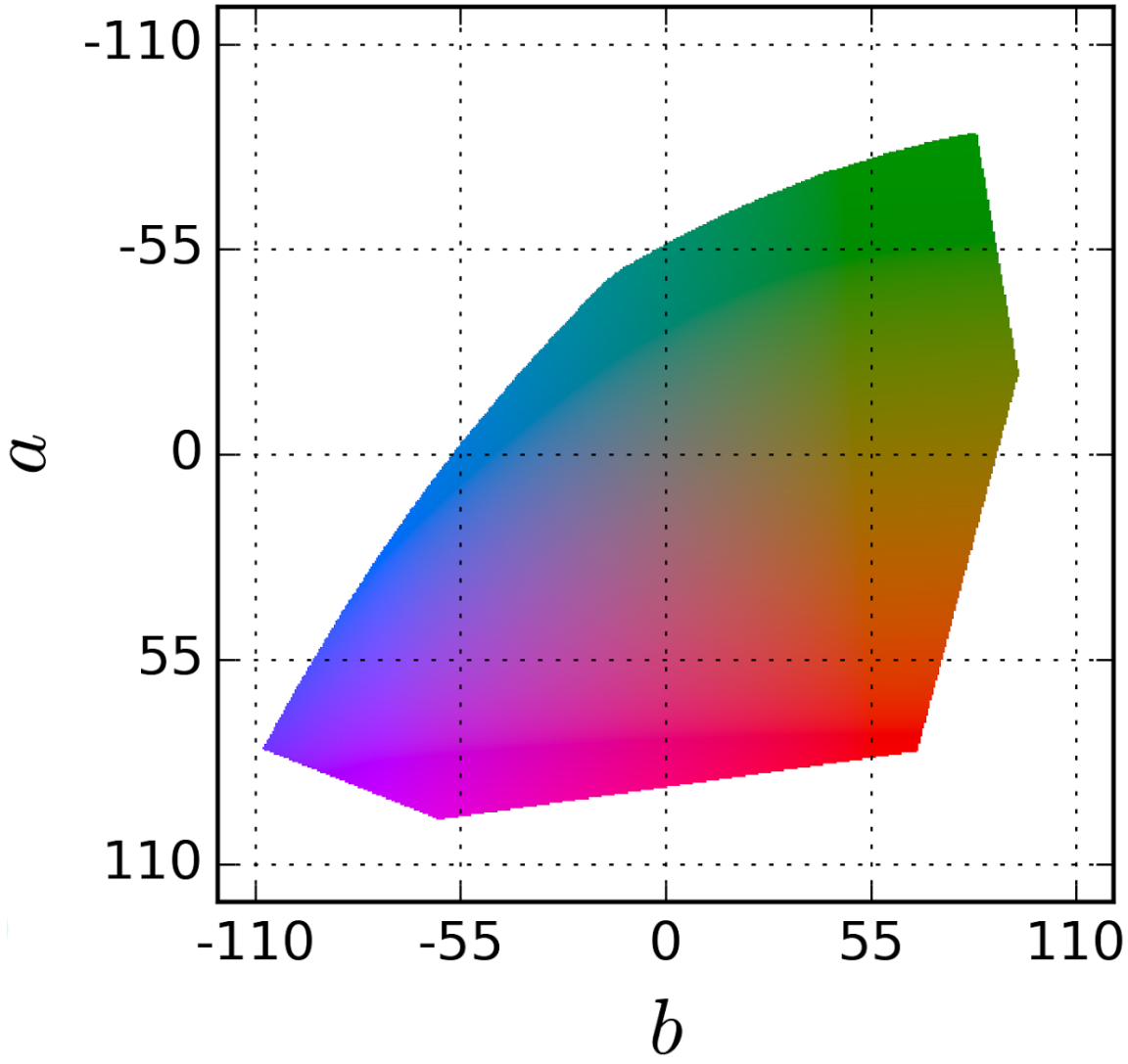
Ground Truth

Better Loss Function

- Regression with L2 loss inadequate

$$\mathrm{L}_2(\widehat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{2}\sum_{h,w} \|\mathbf{Y}_{h,w} - \widehat{\mathbf{Y}}_{h,w}\|_2^2$$

Colors in *ab* space
(continuous)



Better Loss Function

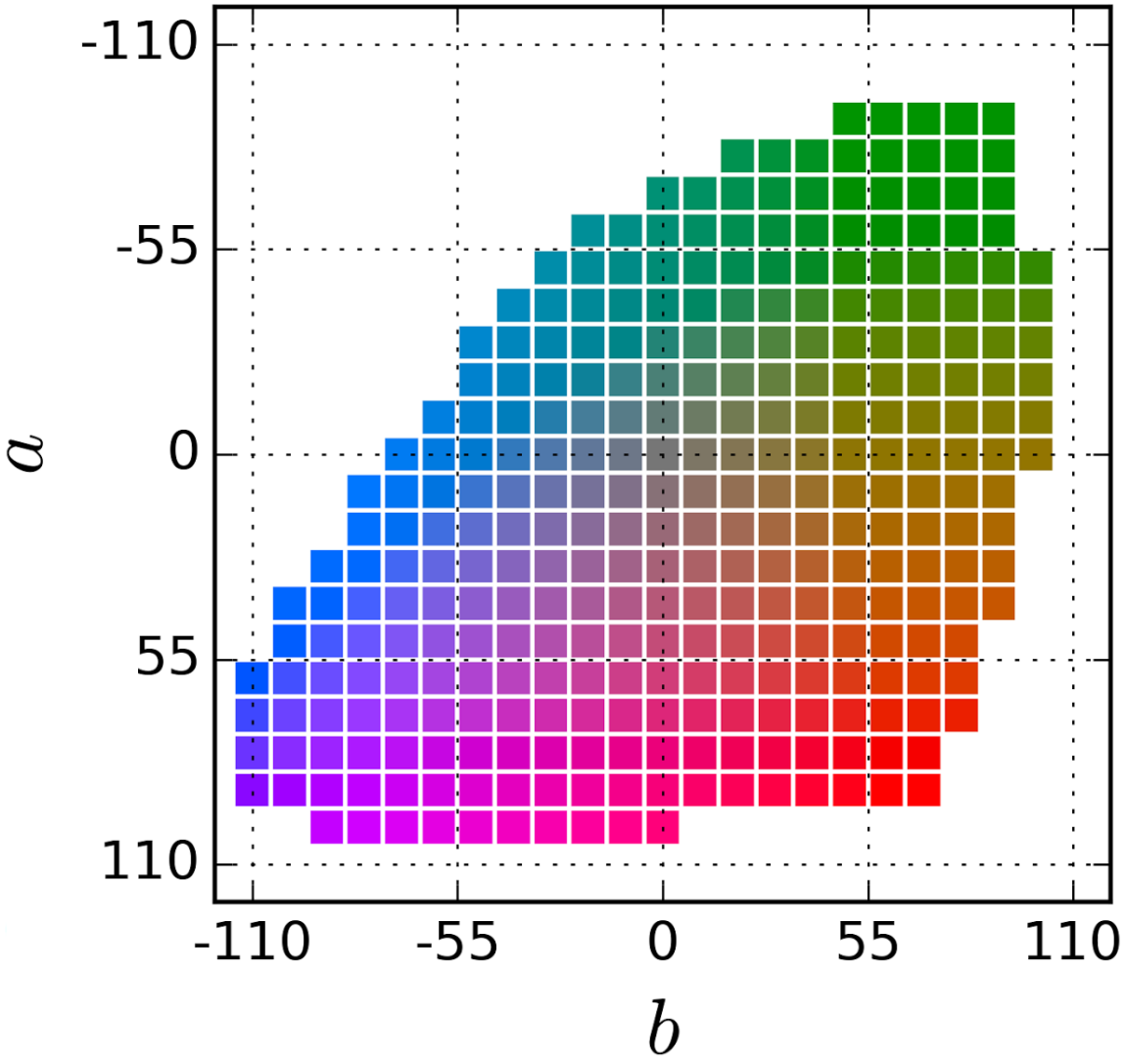
- Regression with L2 loss inadequate

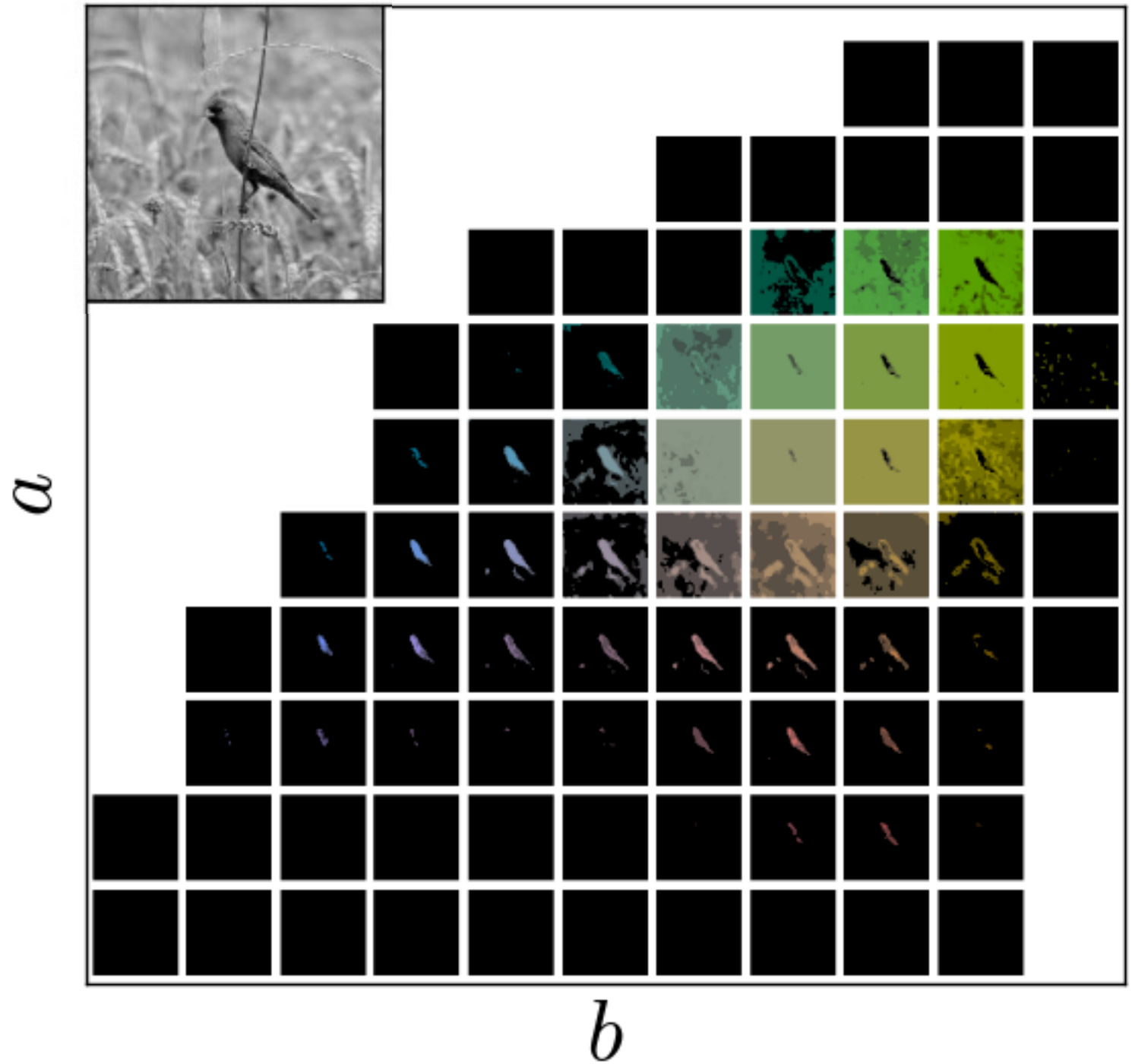
$$\mathrm{L}_2(\widehat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{2}\sum_{h,w} \|\mathbf{Y}_{h,w} - \widehat{\mathbf{Y}}_{h,w}\|_2^2$$

- Use **multinomial classification**

$$\mathrm{L}(\widehat{\mathbf{Z}}, \mathbf{Z}) = -\frac{1}{HW}\sum_{h,w}\sum_{q} \mathbf{Z}_{h,w,q} \log(\widehat{\mathbf{Z}}_{h,w,q})$$

Colors in *ab* space
(discrete)




a
b

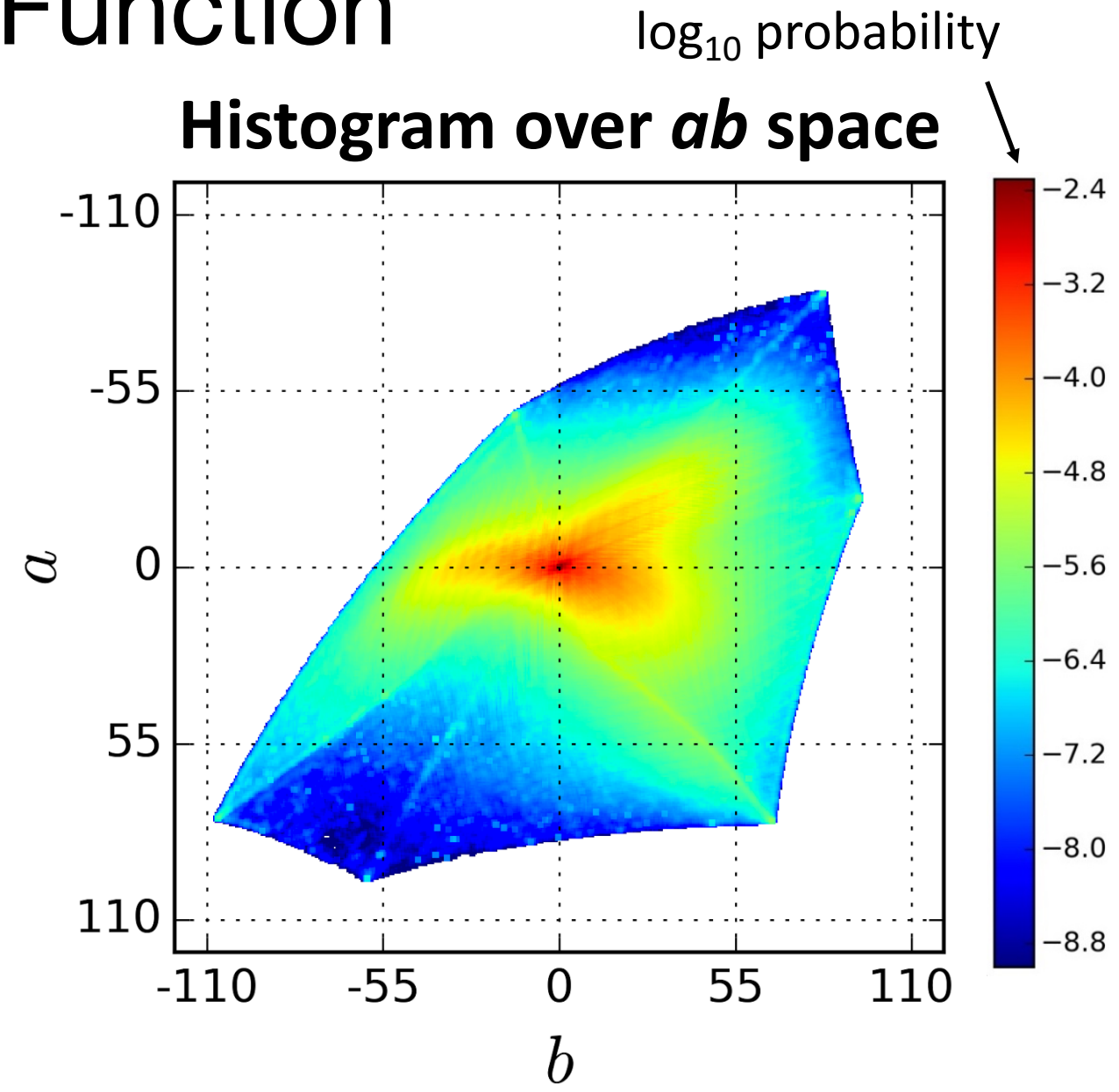
Better Loss Function

- Regression with L2 loss inadequate

$$\mathrm{L}_2(\widehat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{2} \sum_{h,w} \|\mathbf{Y}_{h,w} - \widehat{\mathbf{Y}}_{h,w}\|_2^2$$

- Use **multinomial classification**

$$\mathrm{L}(\widehat{\mathbf{Z}}, \mathbf{Z}) = -\frac{1}{HW} \sum_{h,w} \sum_{q} \mathbf{Z}_{h,w,q} \log(\widehat{\mathbf{Z}}_{h,w,q})$$



Better Loss Function

- Regression with L2 loss inadequate

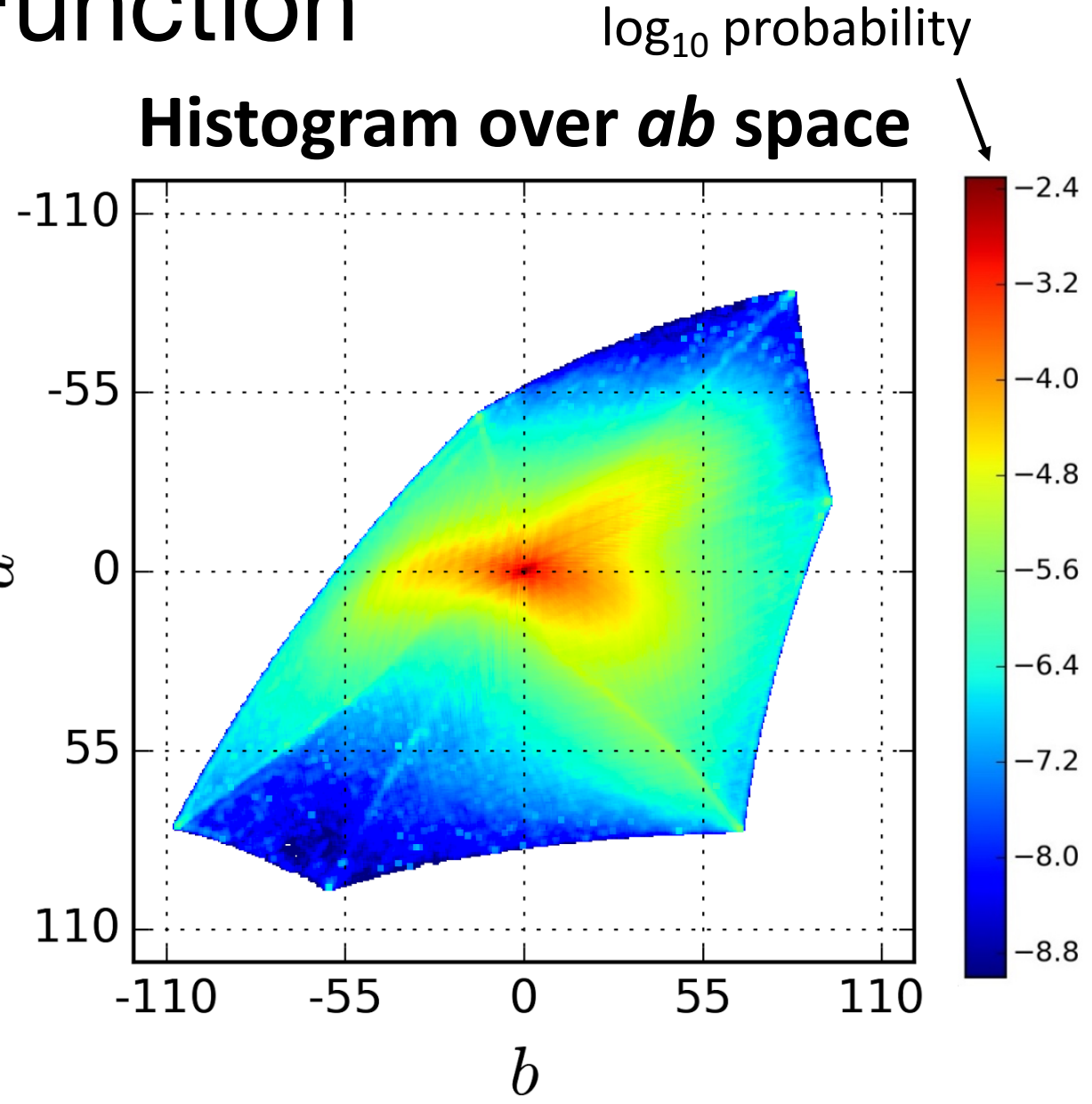
$$\mathrm{L}_2(\widehat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{2} \sum_{h,w} \|\mathbf{Y}_{h,w} - \widehat{\mathbf{Y}}_{h,w}\|_2^2$$

- Use **multinomial classification**

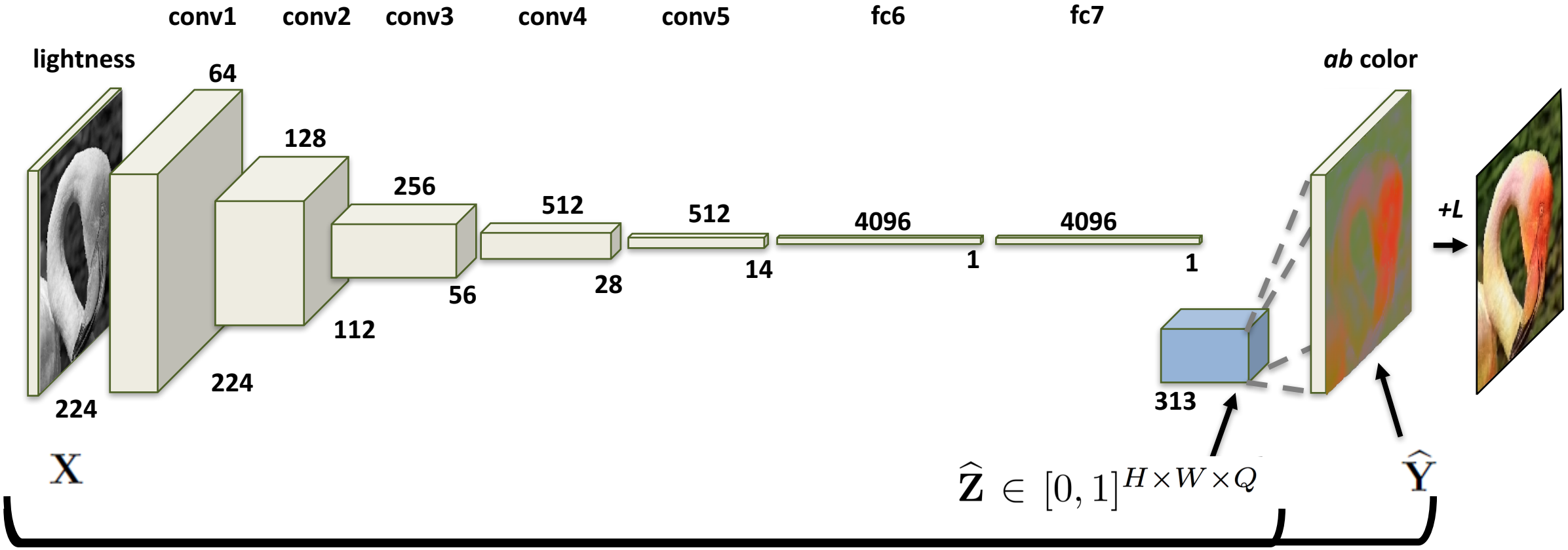
$$\mathrm{L}(\widehat{\mathbf{Z}}, \mathbf{Z}) = -\frac{1}{HW} \sum_{h,w} \sum_{q} \mathbf{Z}_{h,w,q} \log(\widehat{\mathbf{Z}}_{h,w,q})$$

- **Class rebalancing** to encourage learning of *rare* colors

$$\mathrm{L}(\widehat{\mathbf{Z}}, \mathbf{Z}) = -\frac{1}{HW} \sum_{h,w} v(\mathbf{Z}_{h,w}) \sum_{q} \mathbf{Z}_{h,w,q} \log(\widehat{\mathbf{Z}}_{h,w,q})$$

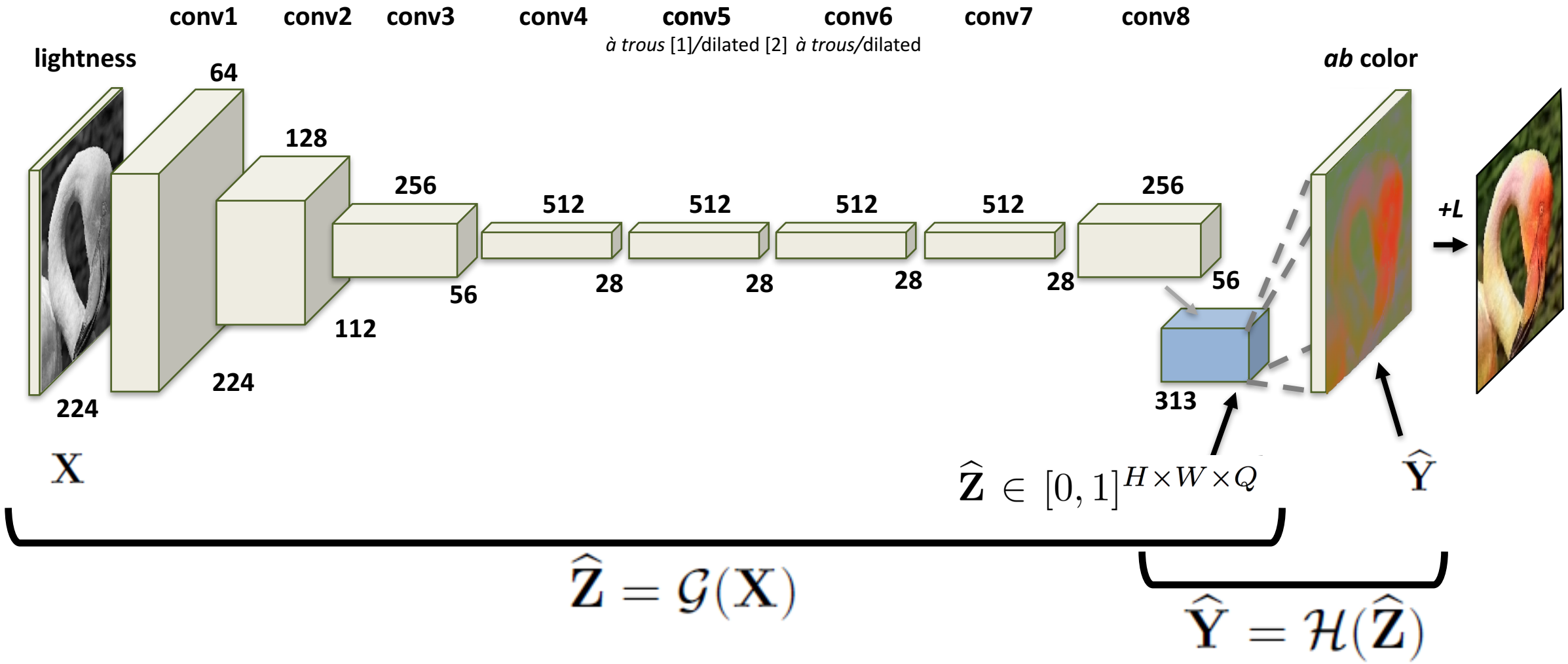


Network Architecture



$$\hat{\mathbf{Y}} = \mathcal{F}(\mathbf{X})$$

Network Architecture



[1] Chen *et al.* In arXiv, 2016.
[2] Yu and Koltun. In ICLR, 2016

Grayscale input

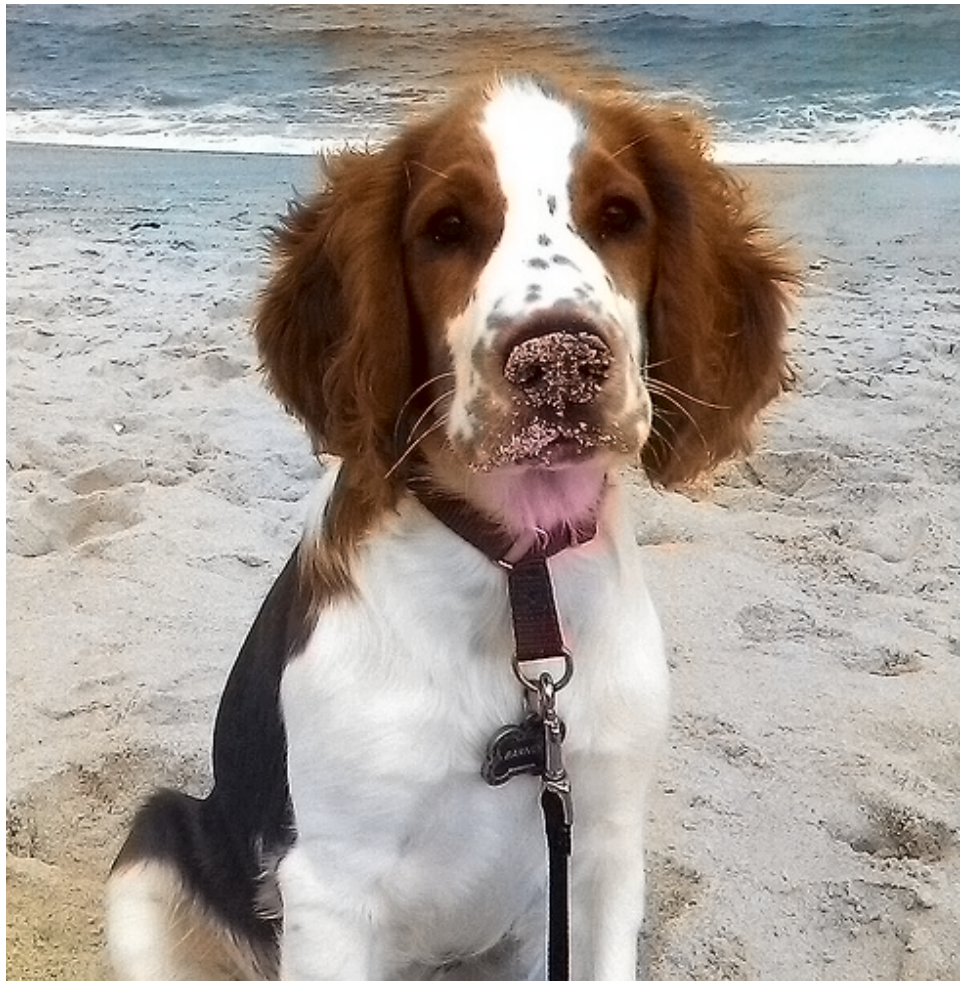
Ground truth

L2 Regression

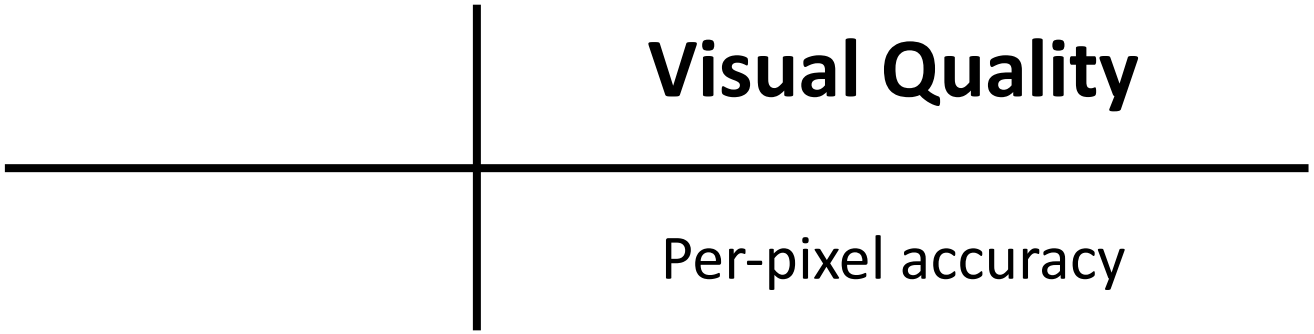
Class w/ Rebalancing

Failure Cases

Biases



Evaluation



Evaluation

	Visual Quality	Representation Learning
Quantitative	Per-pixel accuracy Perceptual realism Semantic interpretability	Task generalization ImageNet classification Task & dataset generalization PASCAL classification, detection, segmentation
Qualitative	Low-level stimuli Legacy grayscale photos	Hidden unit activations

Evaluation

	Visual Quality	Representation Learning
Quantitative	Per-pixel accuracy **Perceptual realism** Semantic interpretability	Task generalization ImageNet classification Task & dataset generalization PASCAL classification, detection, segmentation
Qualitative	Low-level stimuli Legacy grayscale photos	Hidden unit activations

Perceptual Realism / Amazon Mechanical Turk Test

clap if “fake”

clap if “fake”

Fake, 0% fooled

clap if “fake”

clap if “fake”

Fake, 55% fooled

clap if “fake”

clap if “fake”

Fake, 58% fooled

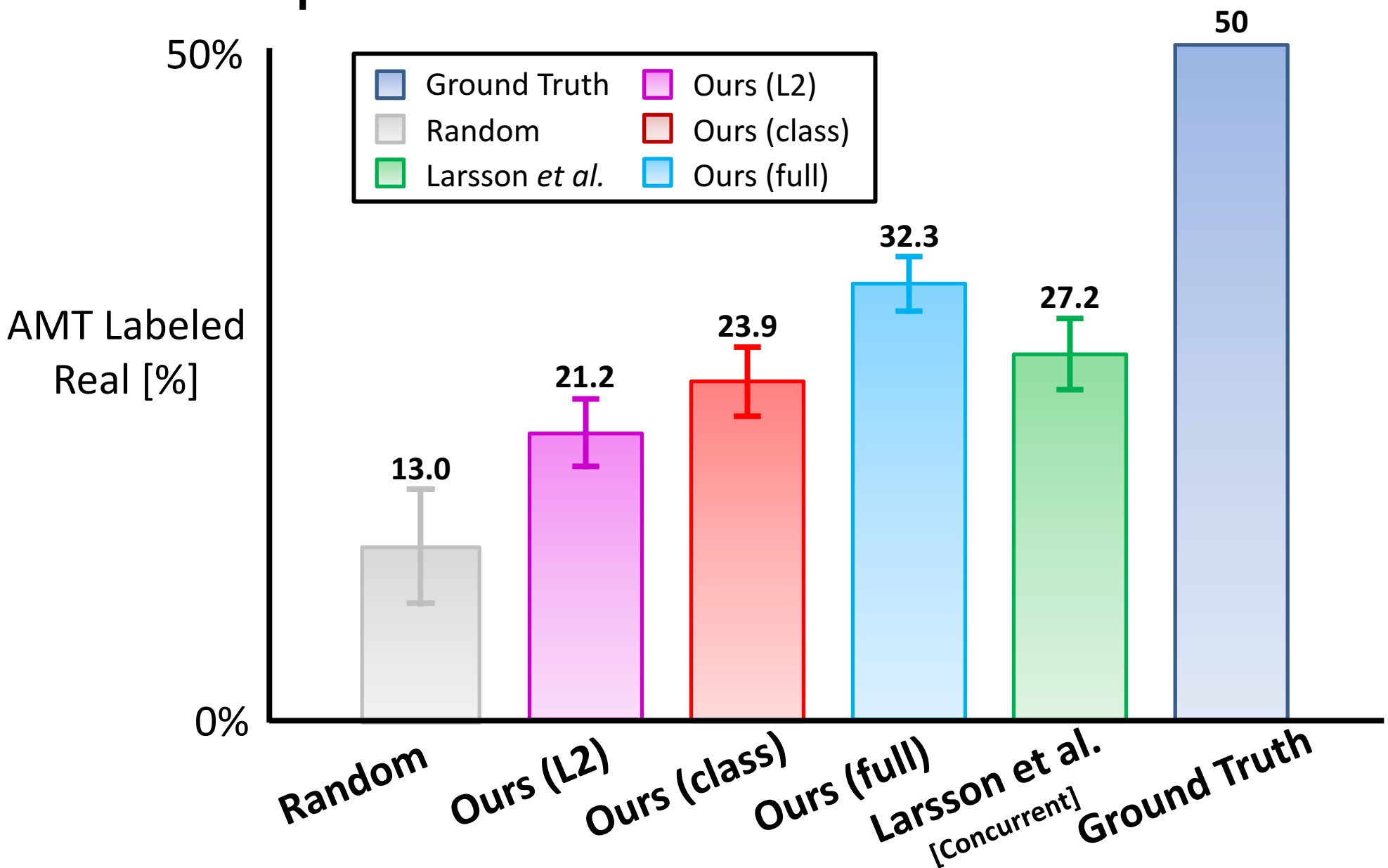
from Reddit /u/SherySantucci

Recolorized by Reddit ColorizeBot

Photo taken by Reddit /u/Timteroo, Mural from street artist Eduardo Kobra

Recolorized by Reddit ColorizeBot

Perceptual Realism Test

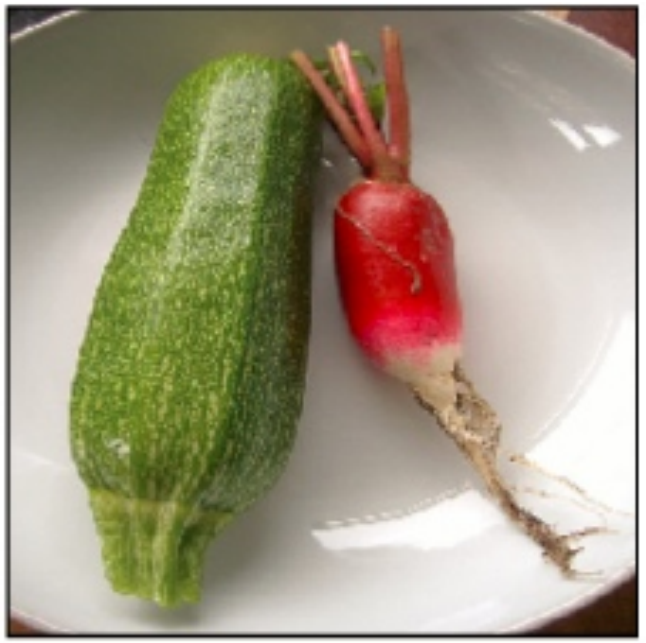


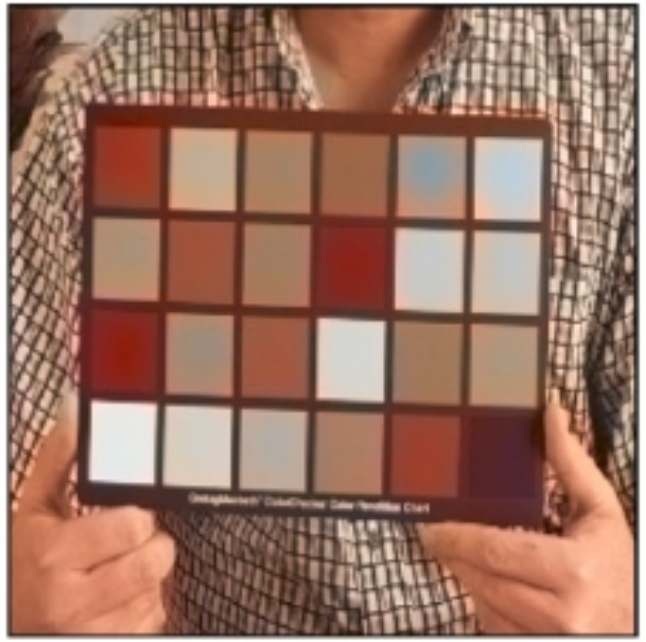
1600 images tested per algorithm

Input

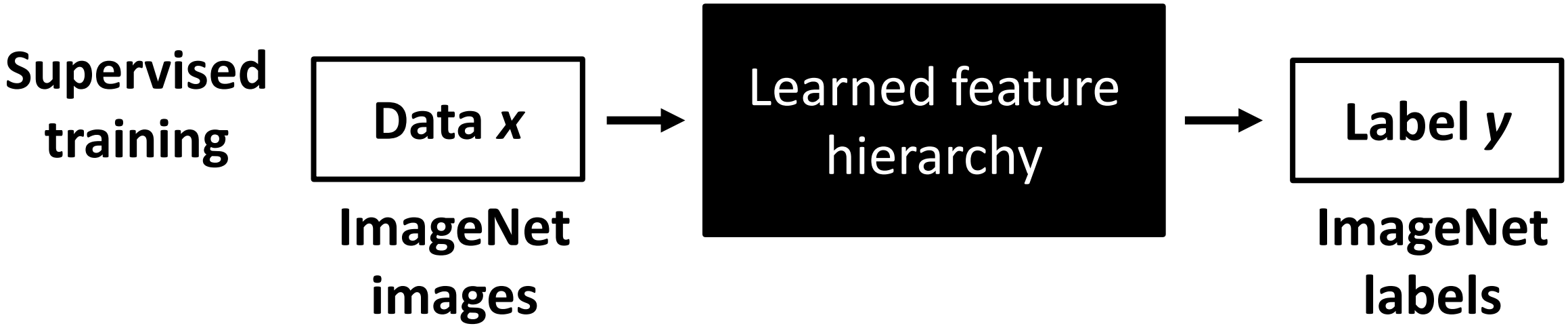
Ground Truth

Output

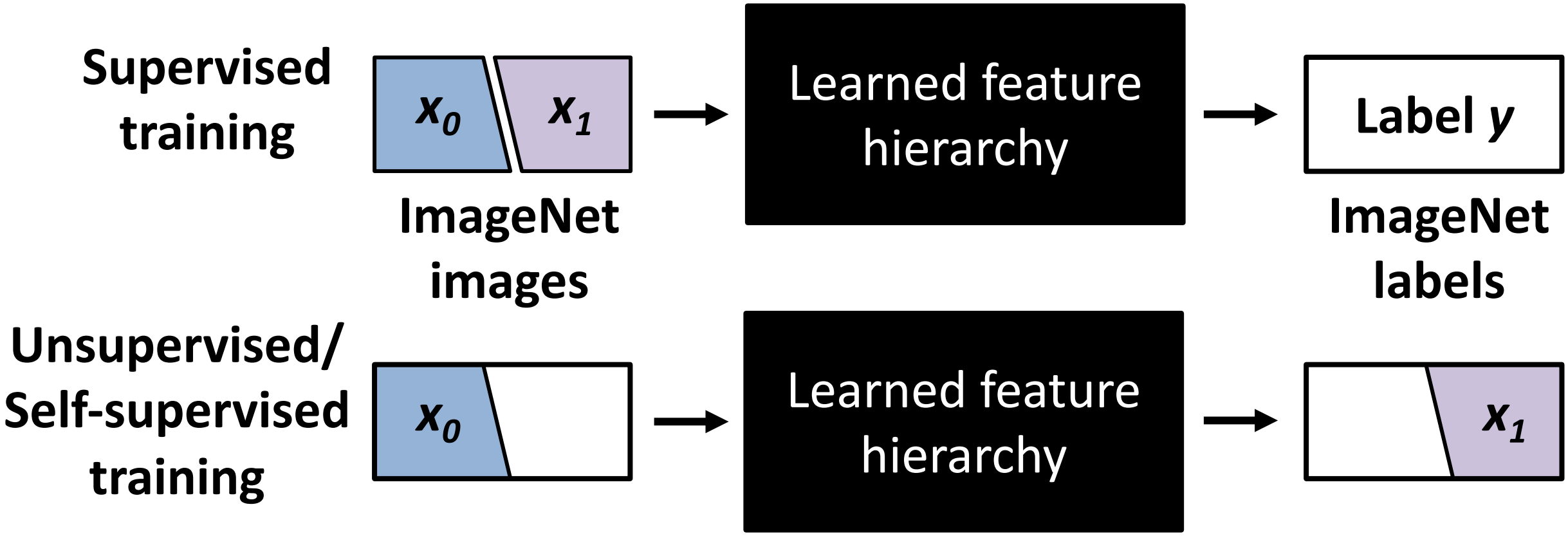




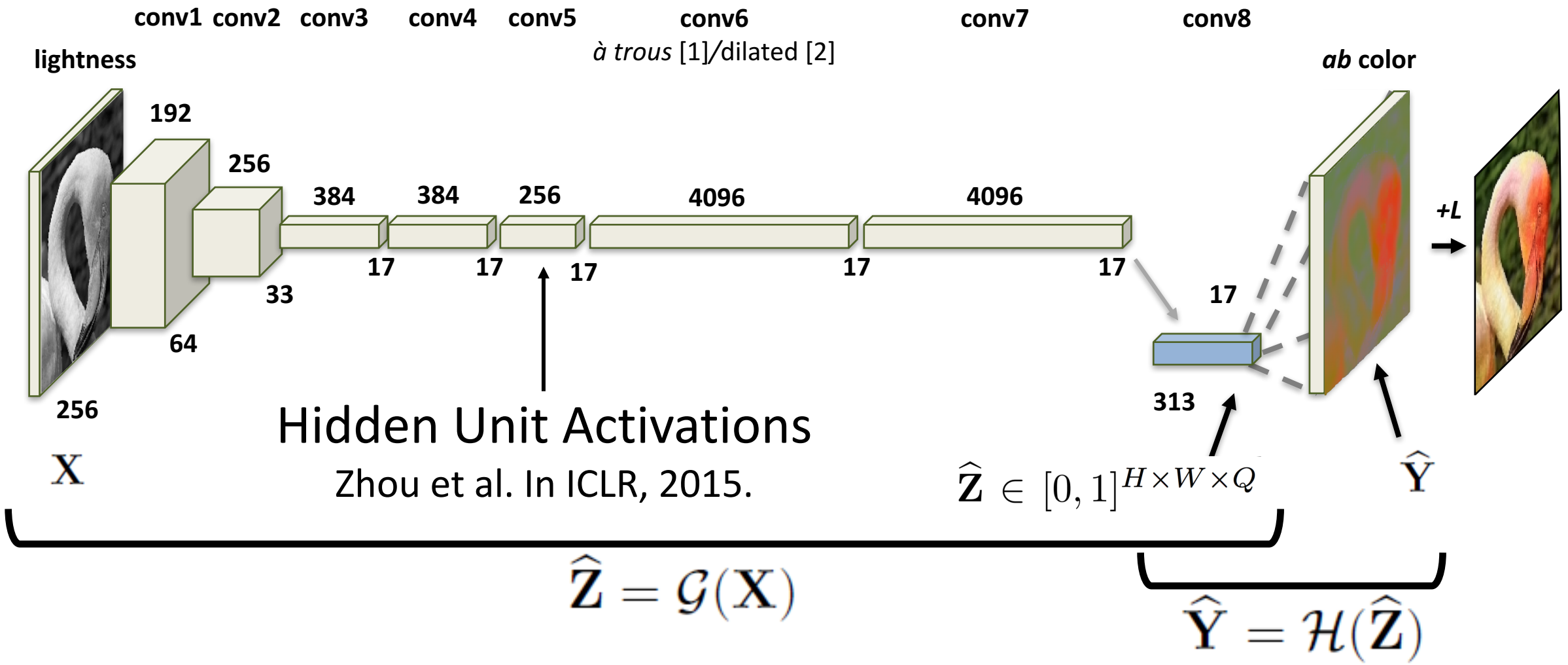
Predicting Labels from Data



Predicting Data from Data

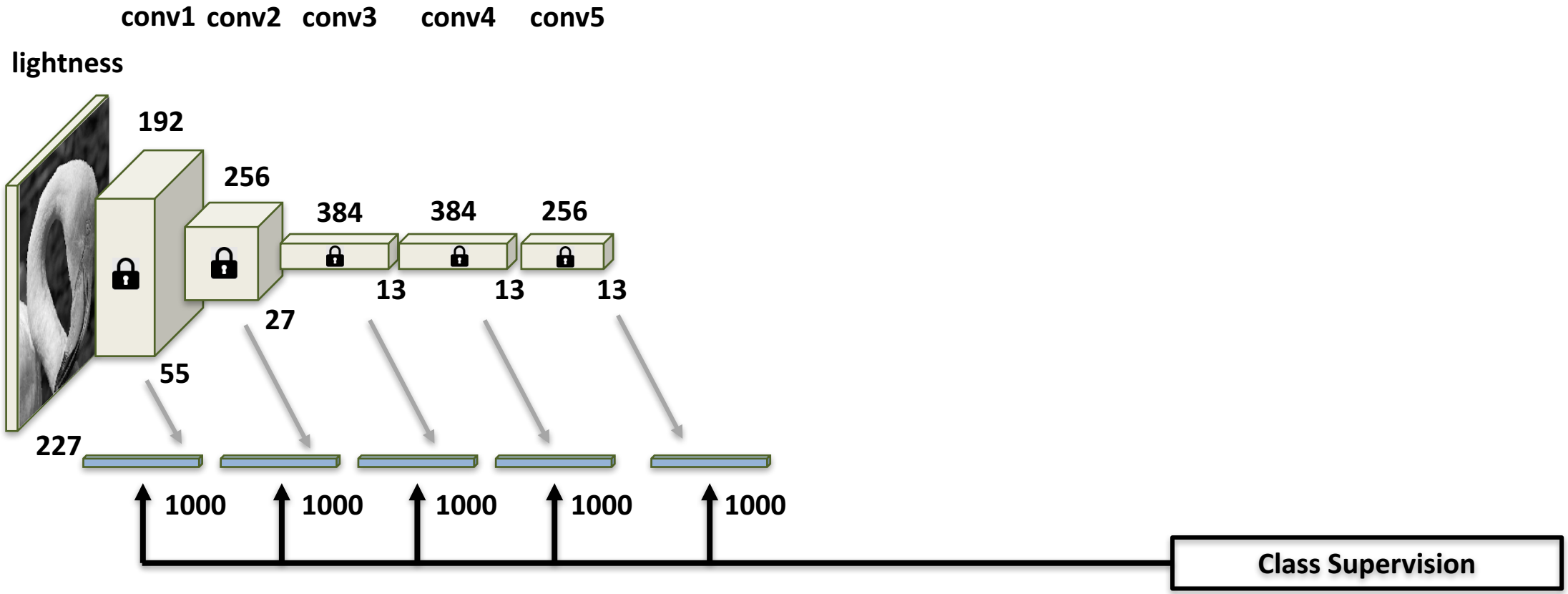


Cross-Channel Encoder



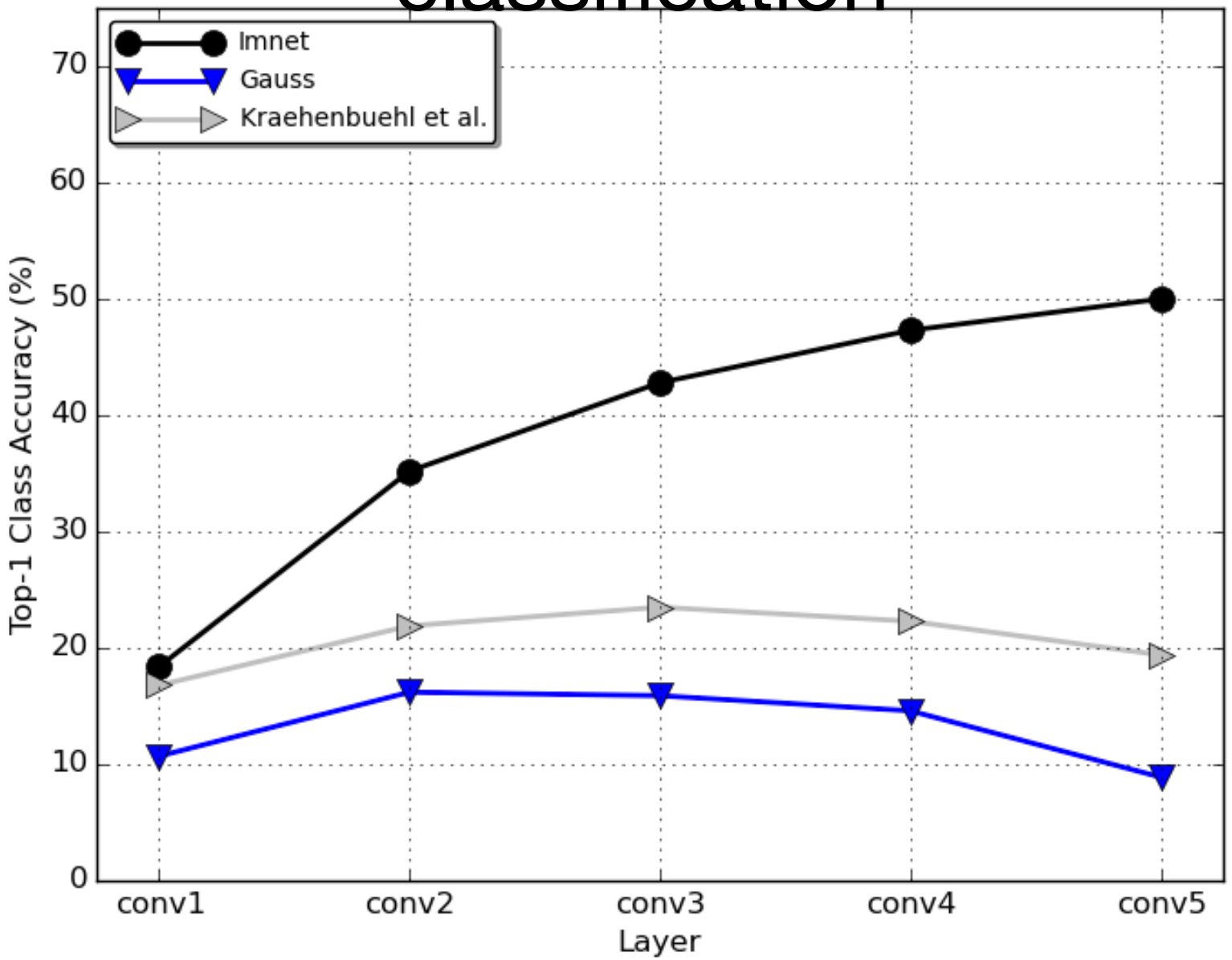
[1] Chen *et al.* In arXiv, 2016.
[2] Yu and Koltun. In ICLR, 2016

Task Generalization: ILSVRC linear classification

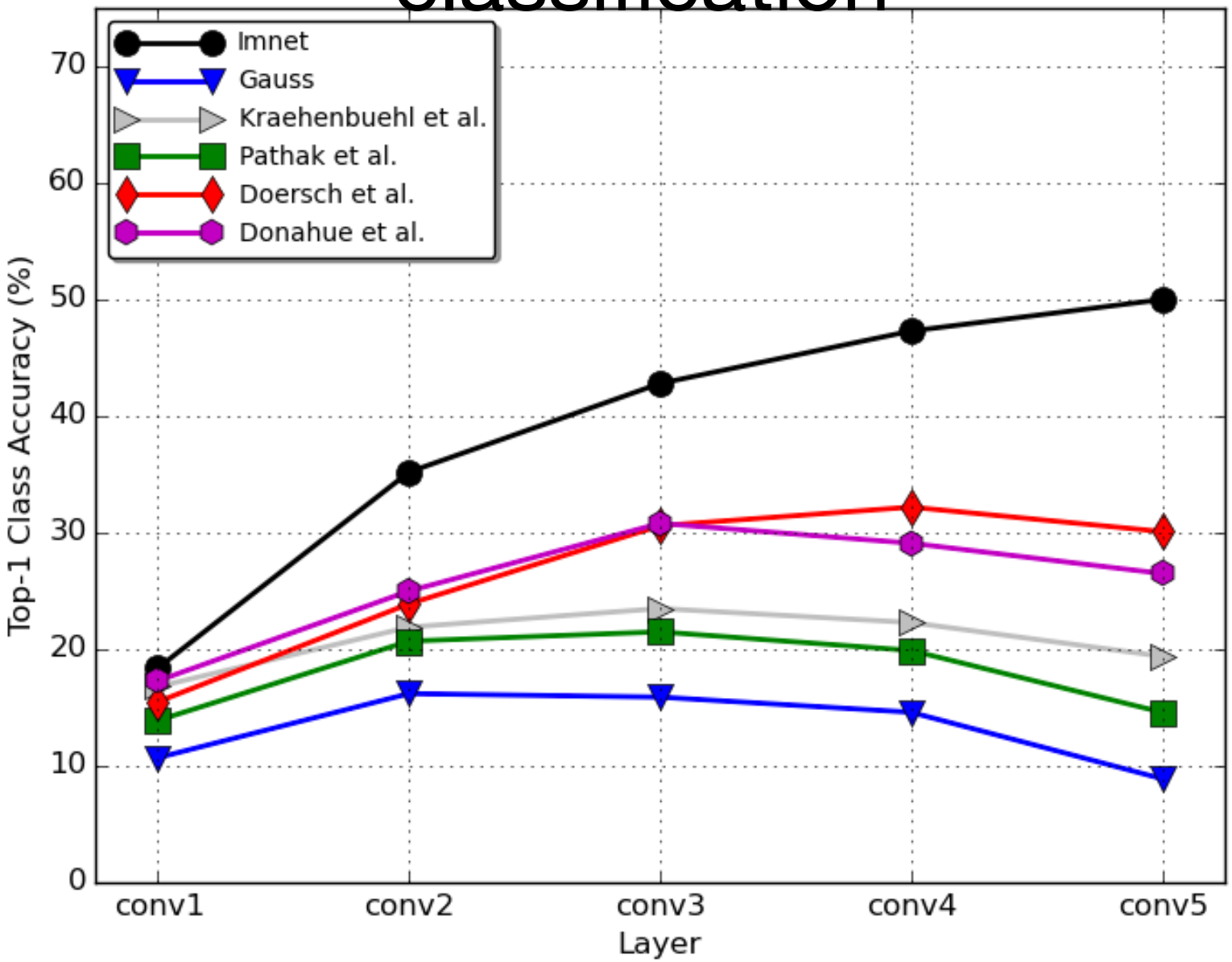


Are semantic classes ***linearly separable*** in the learned feature space?

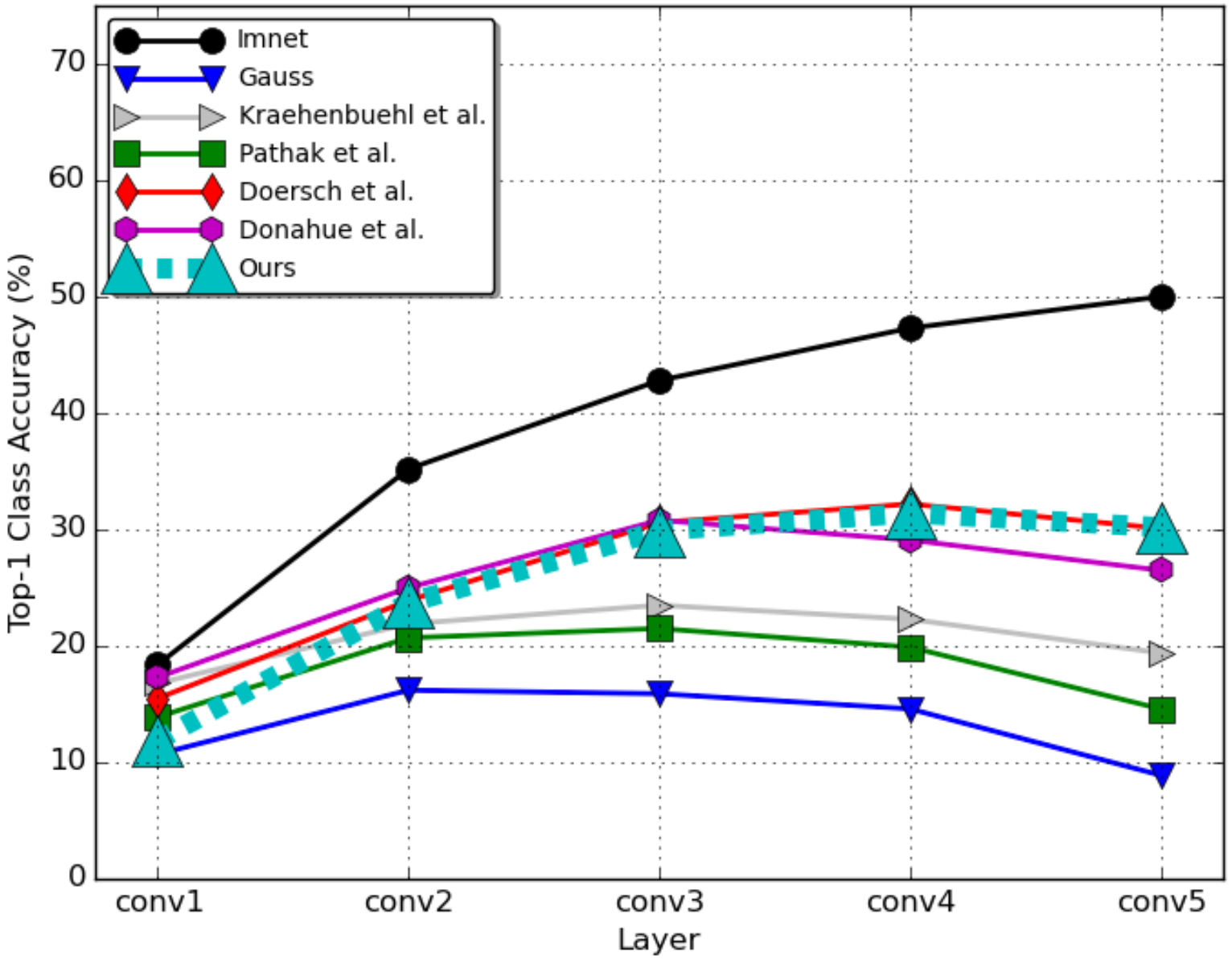
Task Generalization: ILSVRC linear classification



Task Generalization: ILSVRC linear classification



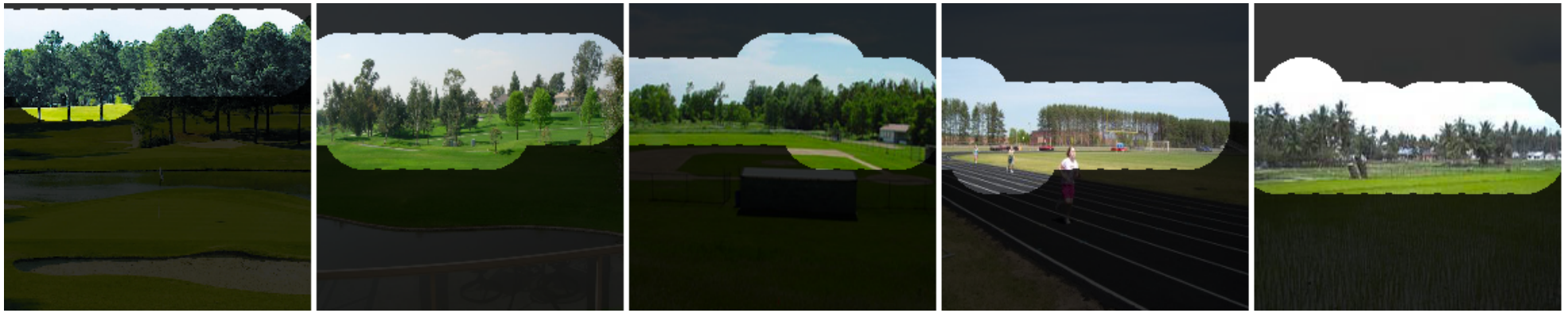
Task Generalization: ILSVRC linear classification



Hidden Unit (conv5) Activations

sky

trees

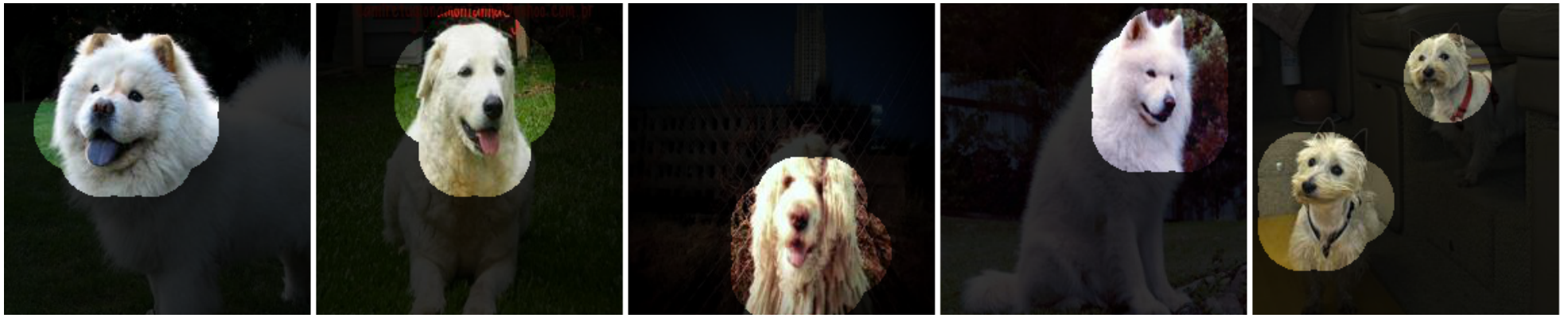


water

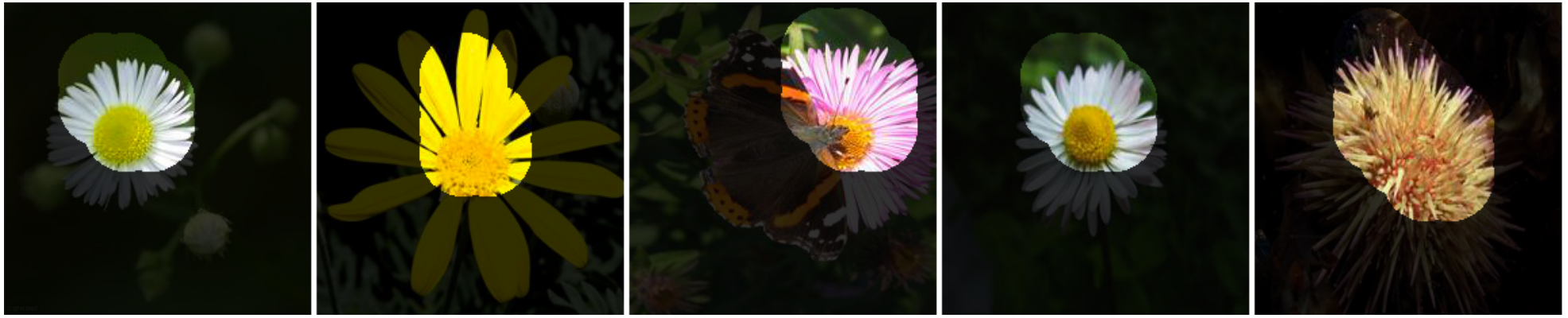
Hidden Unit (conv5) Activations

faces

dog faces

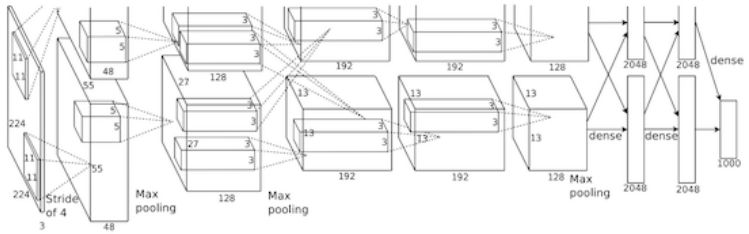


flowers



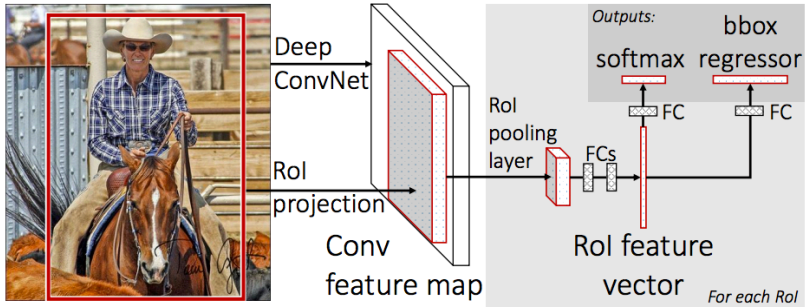
Dataset & Task Generalization on PASCAL VOC

Does the feature representation ***transfer*** to other datasets and tasks?



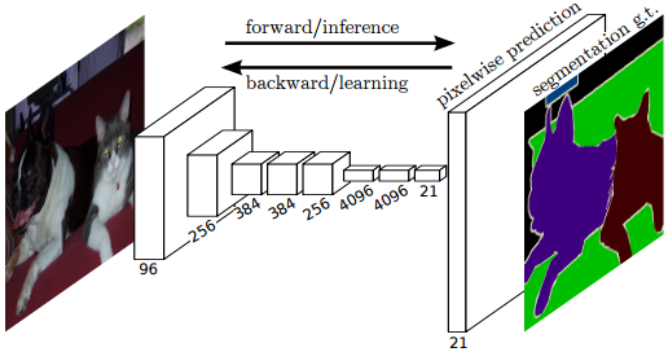
Classification

Krähenbühl et al. In ICLR, 2016.



Detection

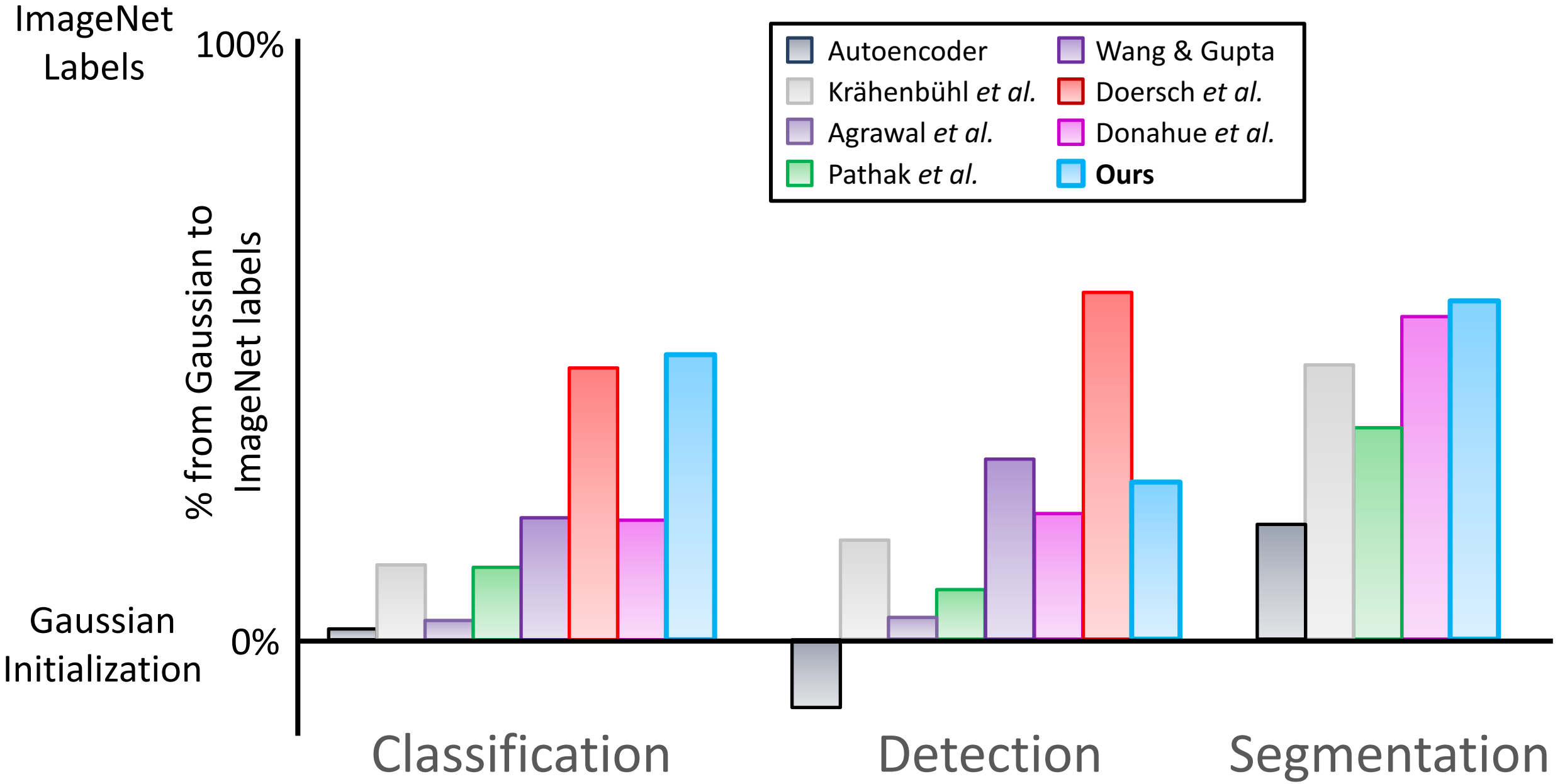
Fast R-CNN. Girshick. In ICCV, 2015.



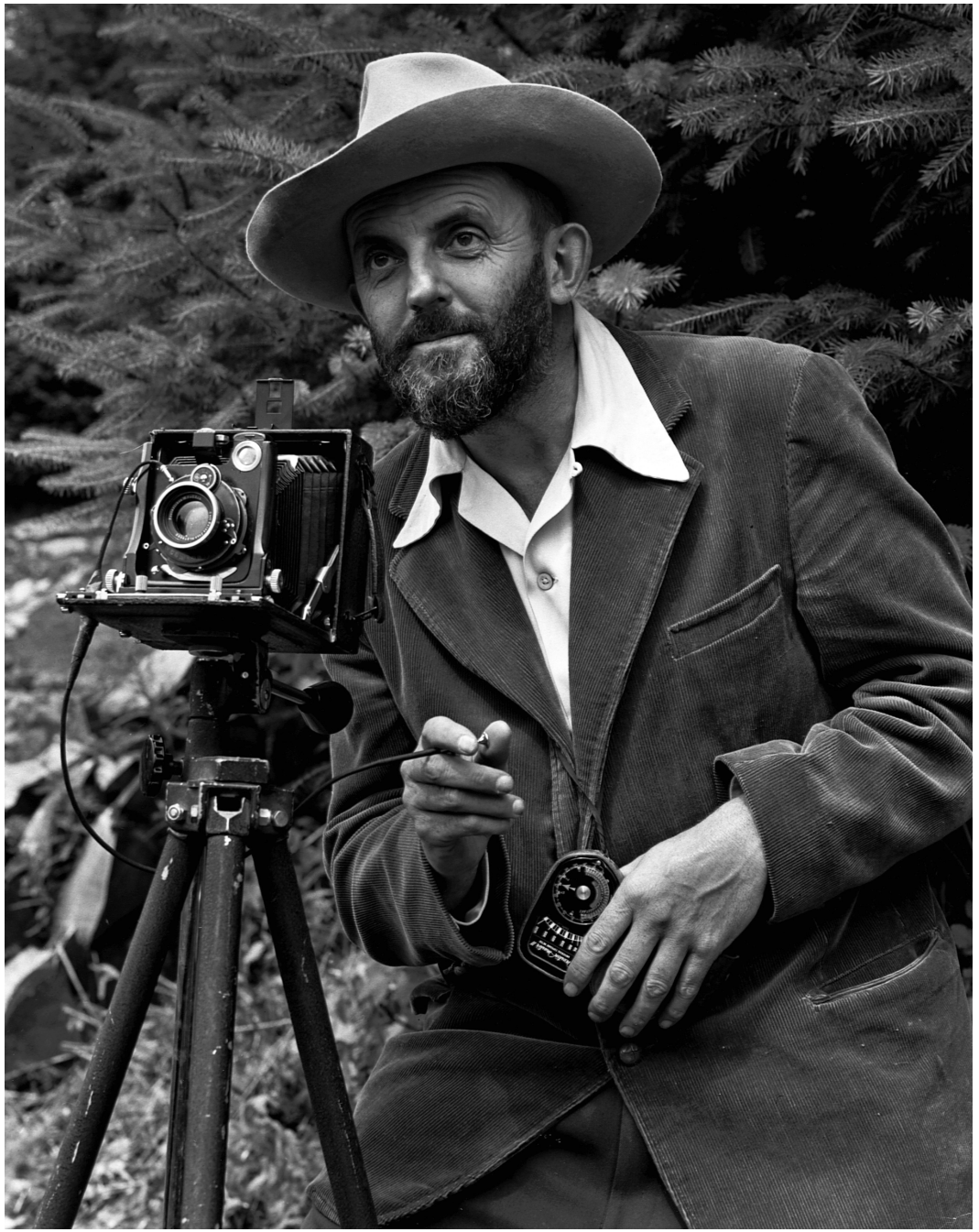
Segmentation

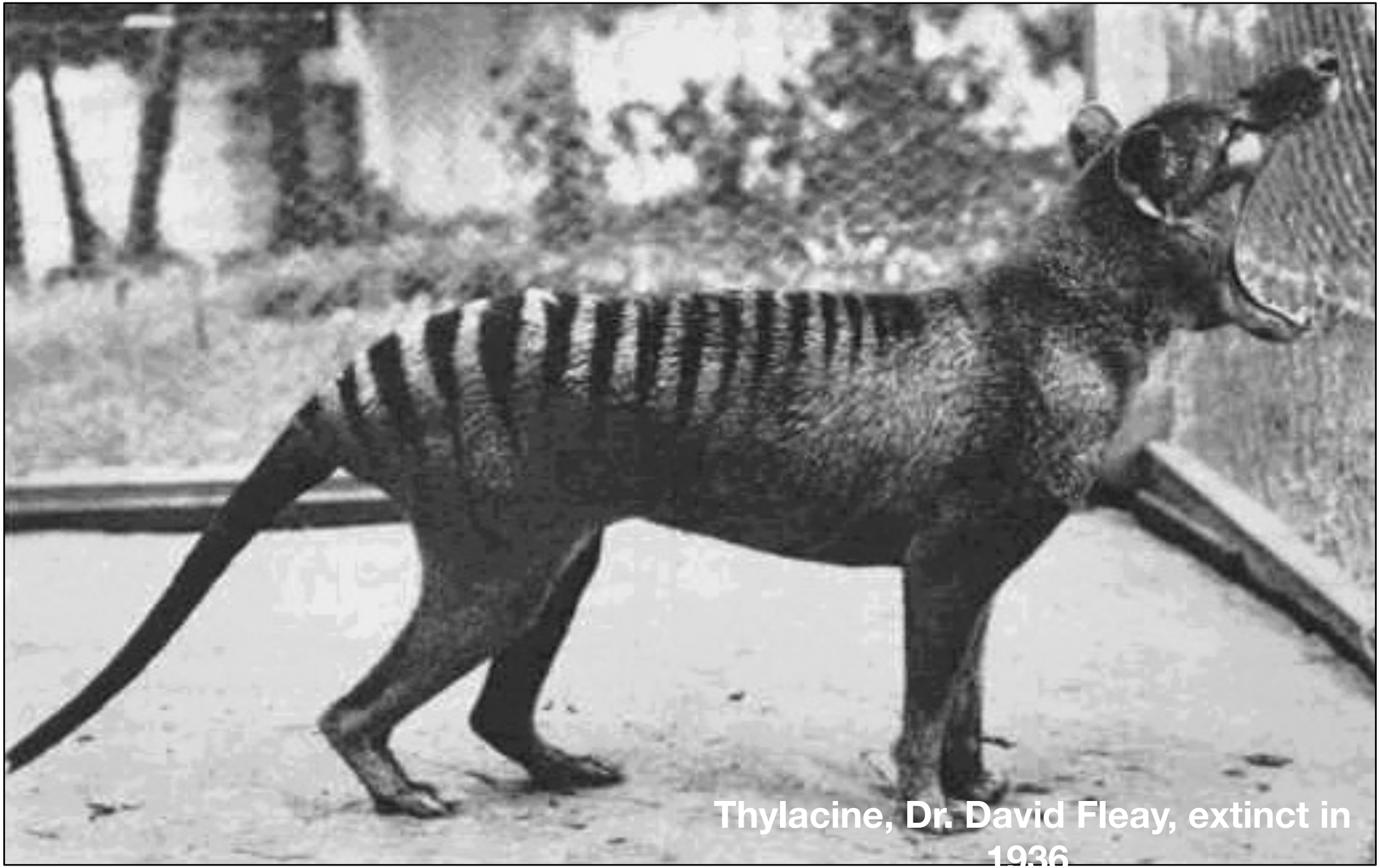
FCNs. Long et al. In CVPR, 2015.

Dataset & Task Generalization on PASCAL VOC



Does the method work on ***legacy*** black and white photos?



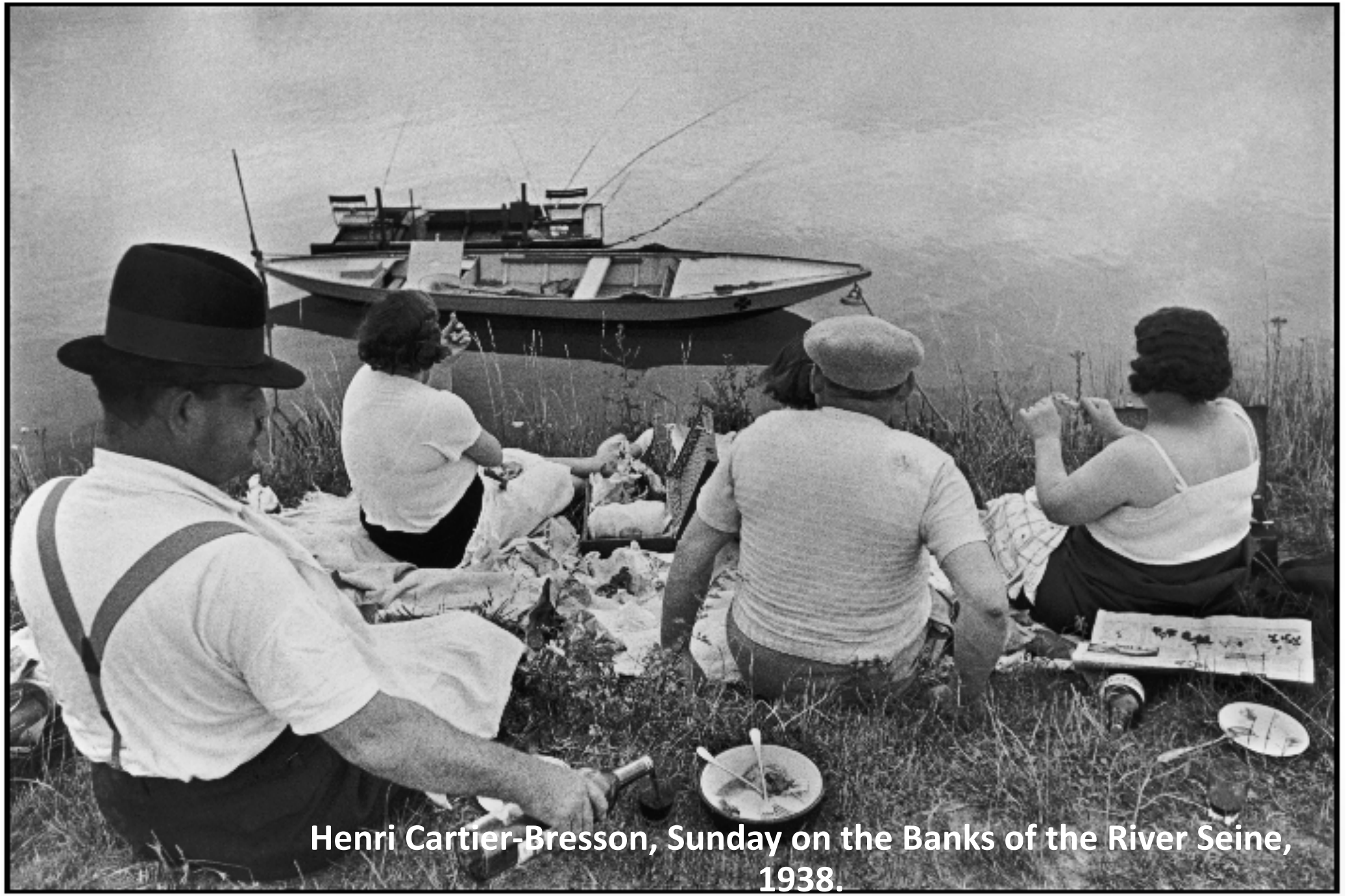


Thylacine, Dr. David Fleay, extinct in 1936

Thylacine, Dr. David Fleay, extinct in 1936

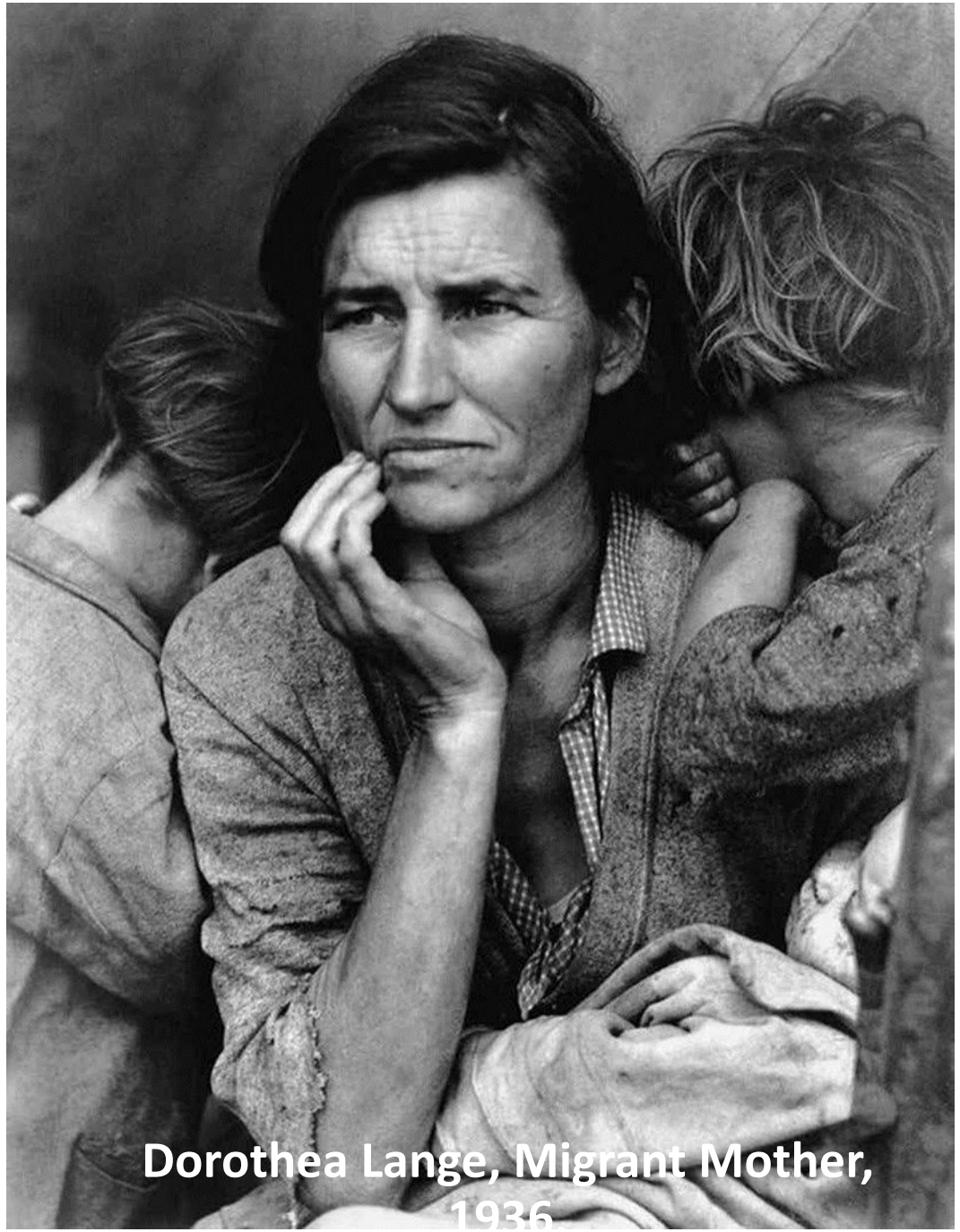
Amateur Family Photo, 1956

Amateur Family Photo, 1956

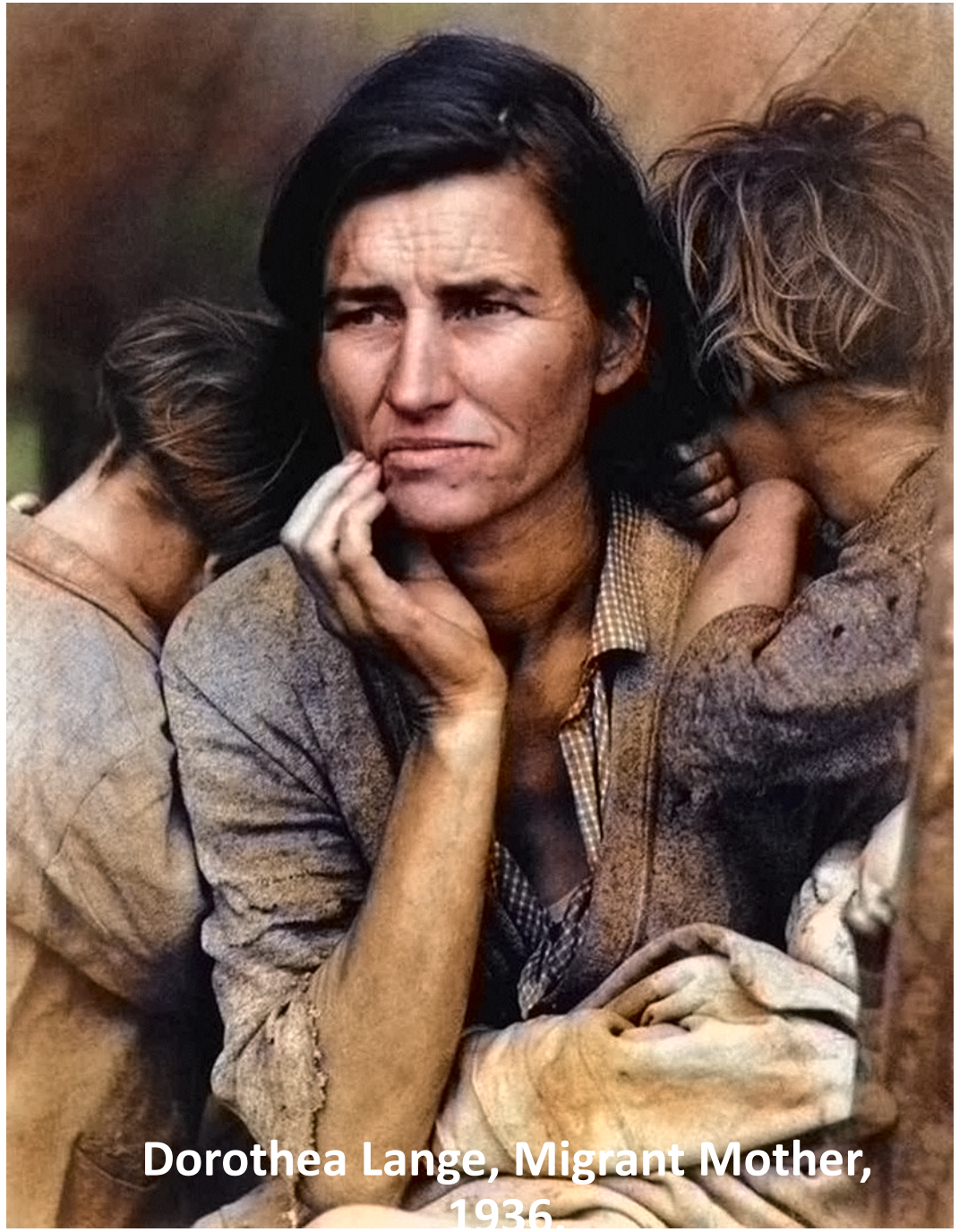


Henri Cartier-Bresson, Sunday on the Banks of the River Seine, 1938.

Henri Cartier-Bresson, Sunday on the Banks of the River Seine, 1938.



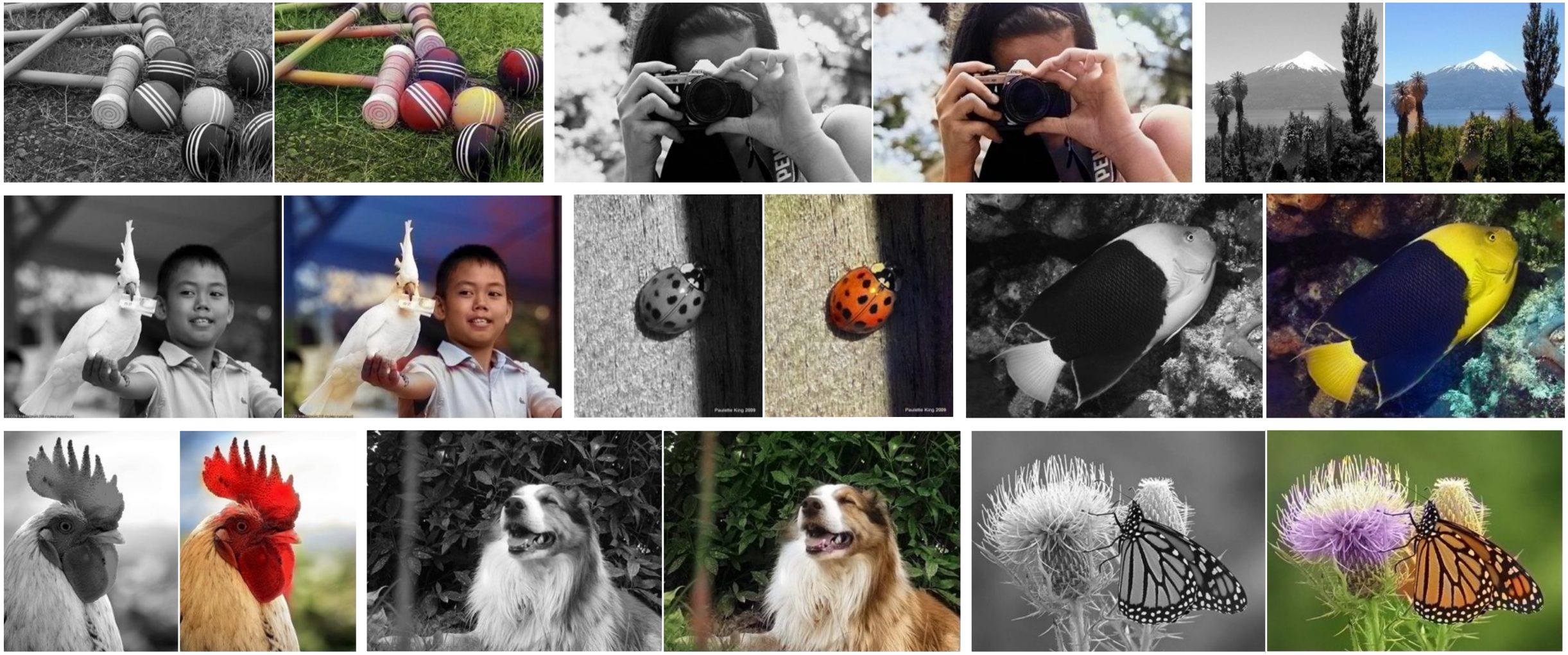
Dorothea Lange, Migrant Mother, 1936



Dorothea Lange, Migrant Mother, 1936

Additional Information

- Demo
  - http://demos.algorithmia.com/colorize-photos/
- Reddit ColorizeBot
  - Type “colorizebot” under any image post
- Code
  - https://github.com/richzhang/colorization
- Website – full paper, user examples, visualizations
  - http://richzhang.github.io/colorization



For the full paper, additional examples and our model:
richzhang.github.io/colorization