Pixel Labelling: Depth, Super-Res + Colorization

CSE P576

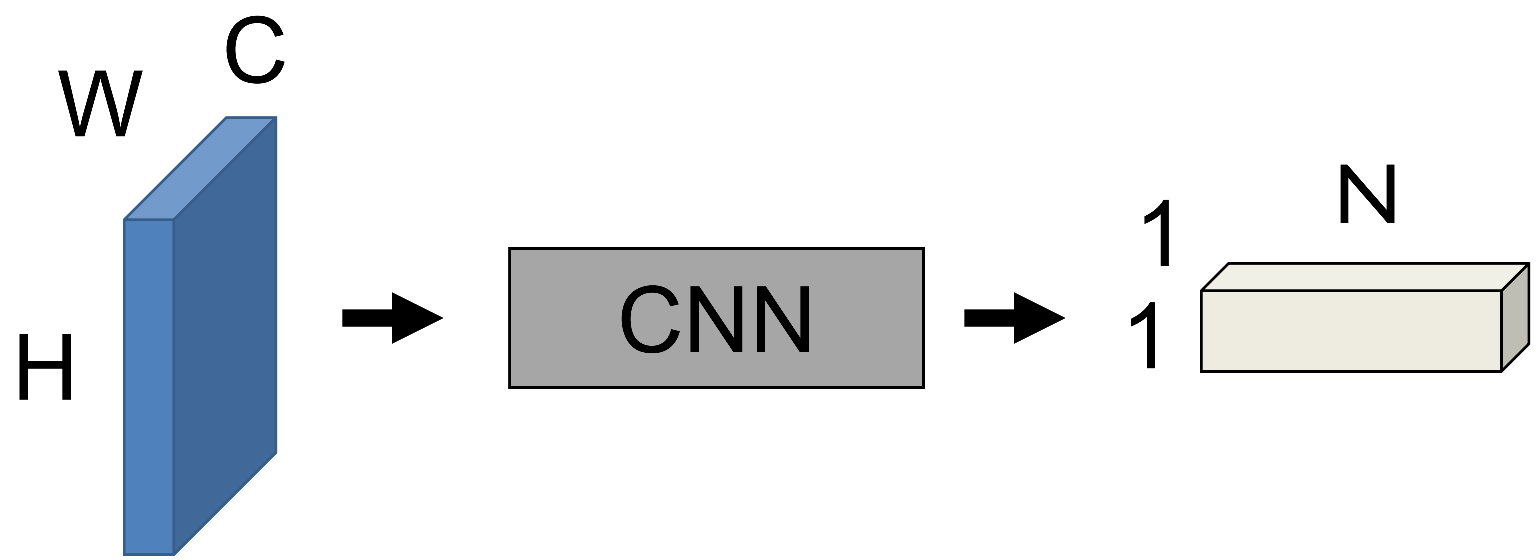
Dr. Matthew Brown

Pixel Labelling

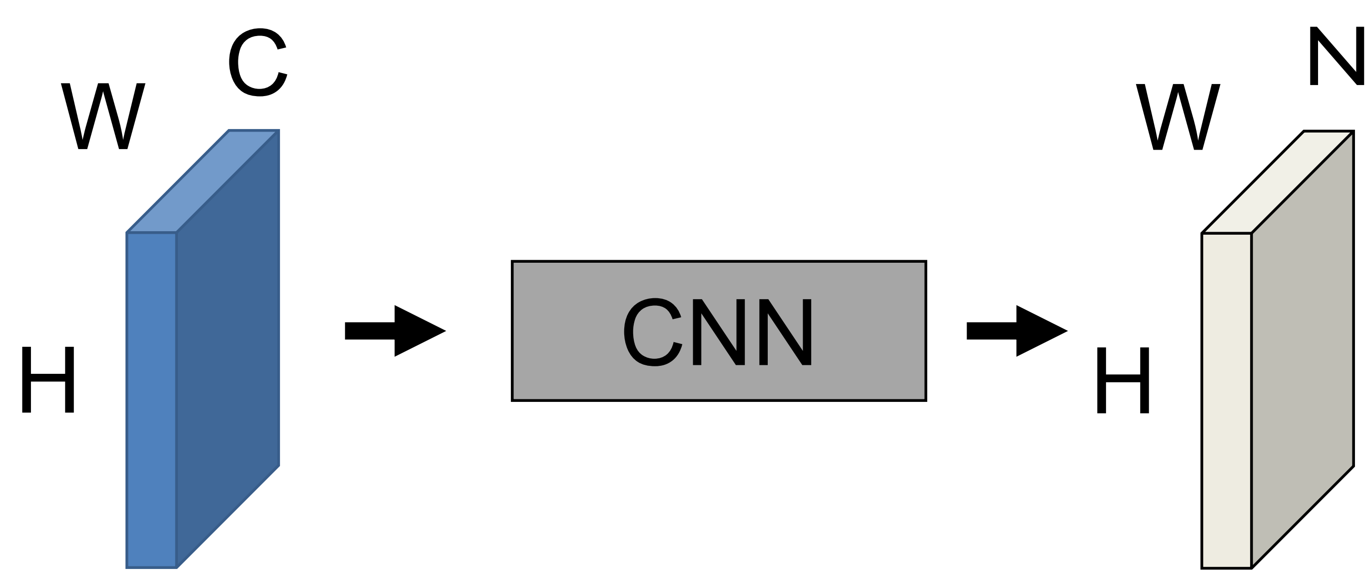
- Per-Pixel Regression + Classification, Examples, Architectures
- Depth Estimation: direct vs self supervised, pretraining
- Super-Resolution, Colorization, Image Translation

Pixel vs Image Labelling

- Image labelling, e.g., classification (N class scores per image)



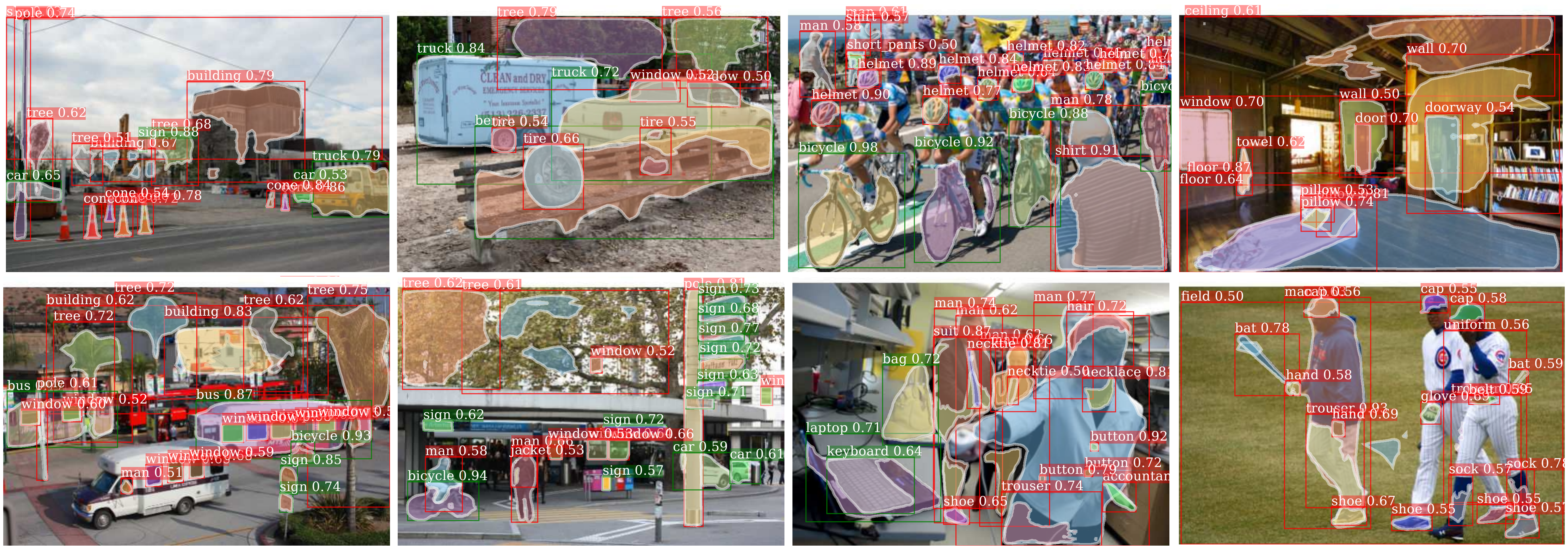
- Pixel labelling, e.g., segmentation, depth estimation, superres, (N class scores, depth, RGB value etc. per pixel)



[David Fouhey]

Segmentation

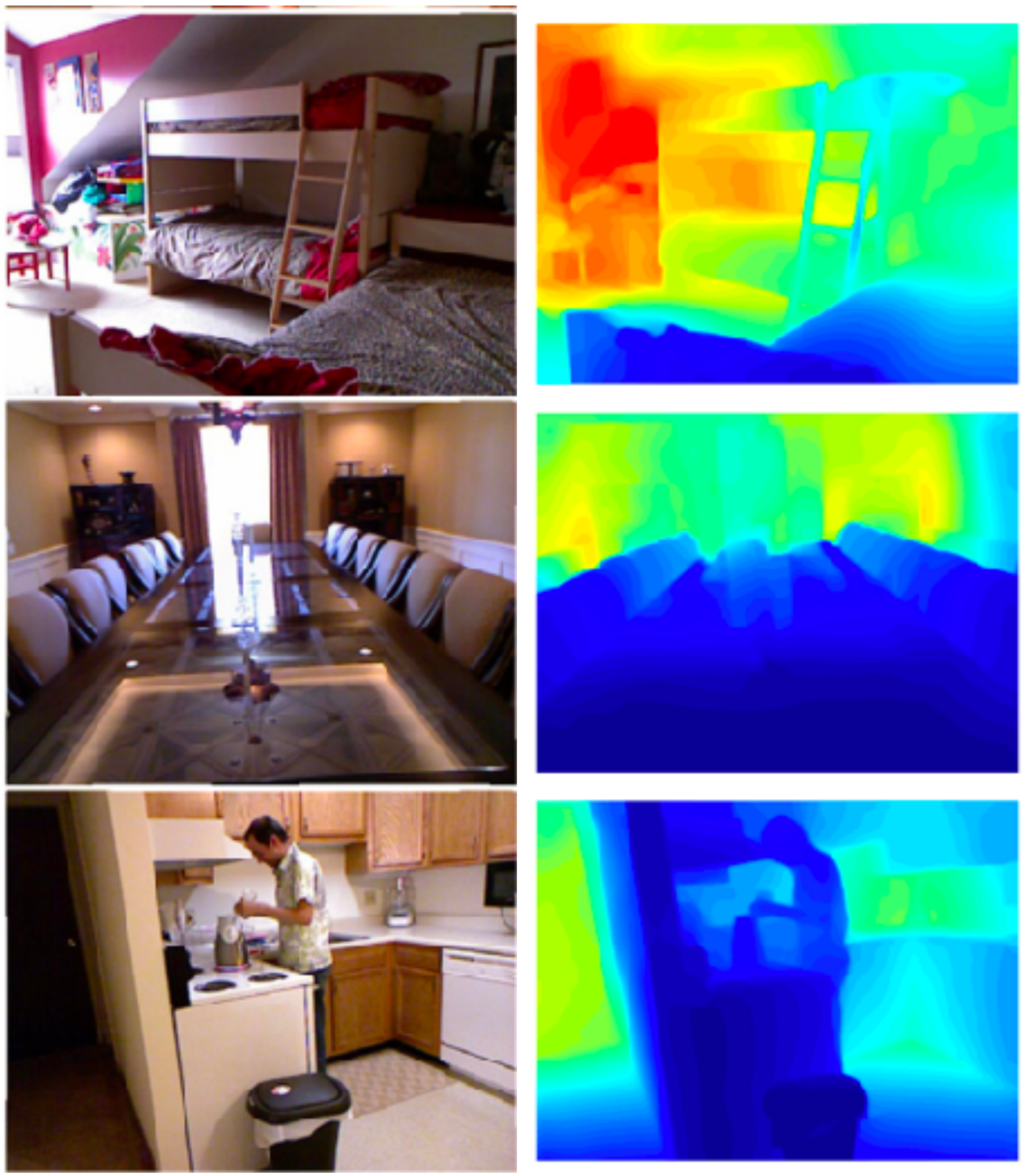
- Predict object identity and/or category per pixel



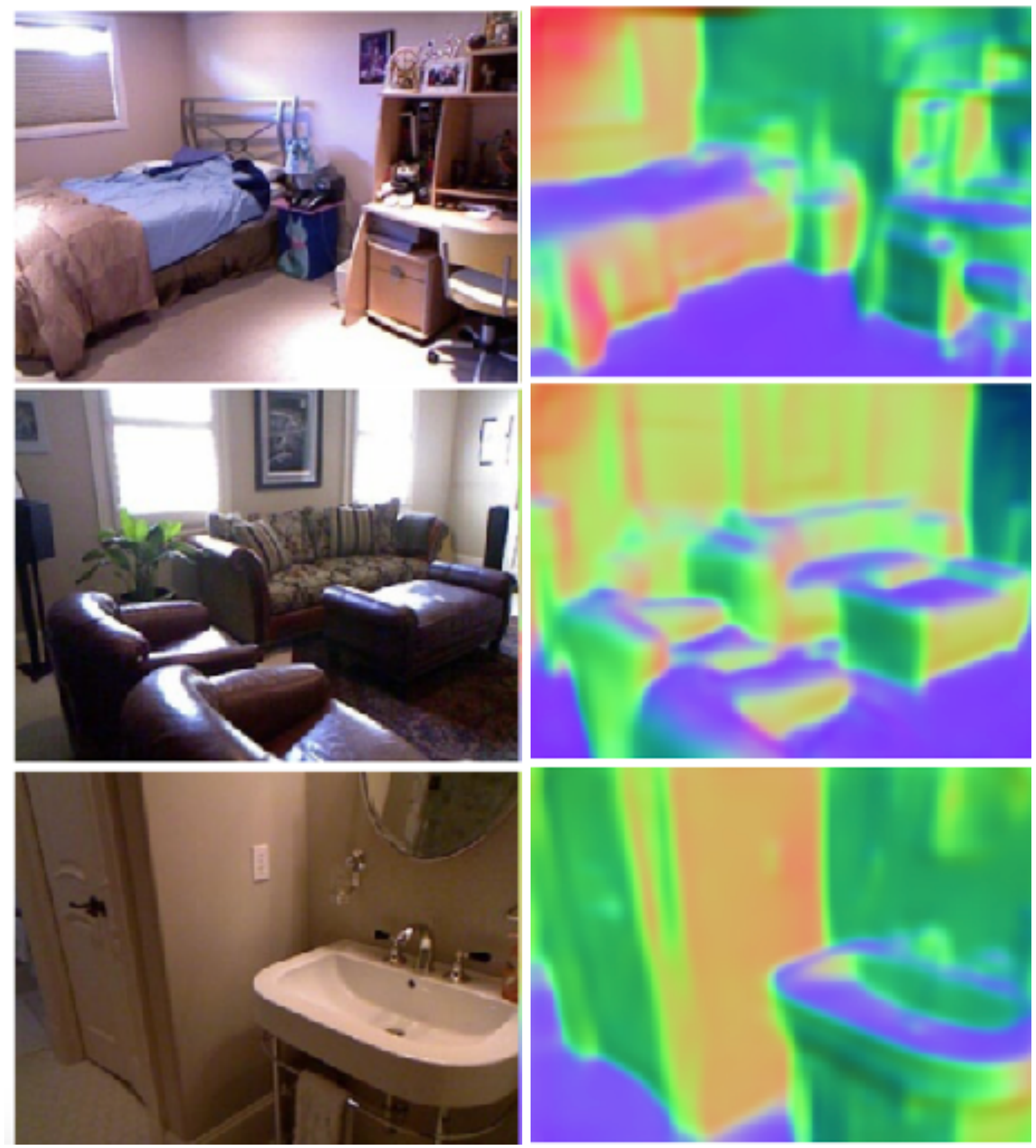
[Hu et al 2017]

Depth + Normals Estimation

- Predict depth or surface normal per pixel, given RGB input



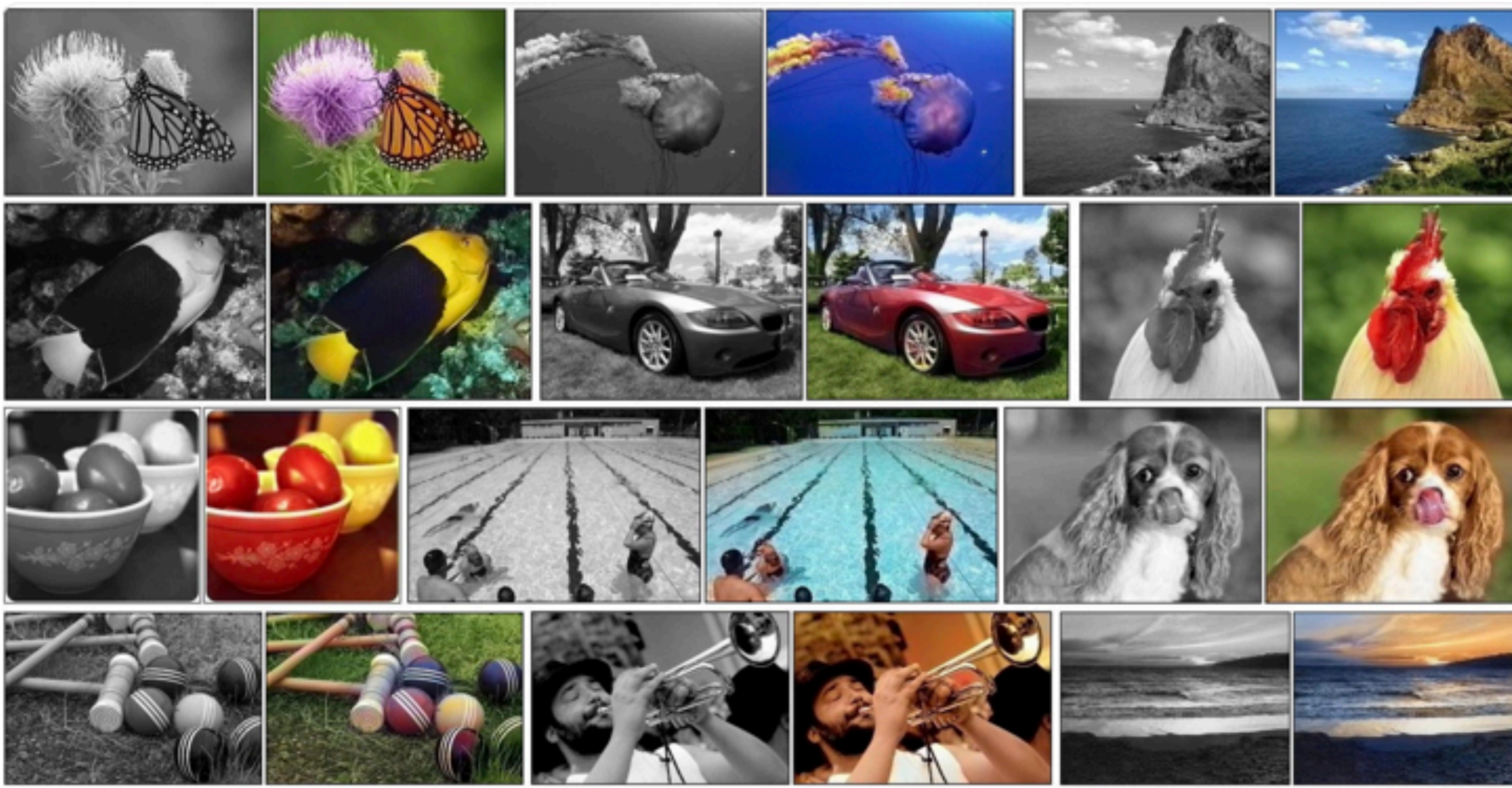
[Alhashim Wonka 2019]



[Eigen Fergus 2015]

Image Colorization

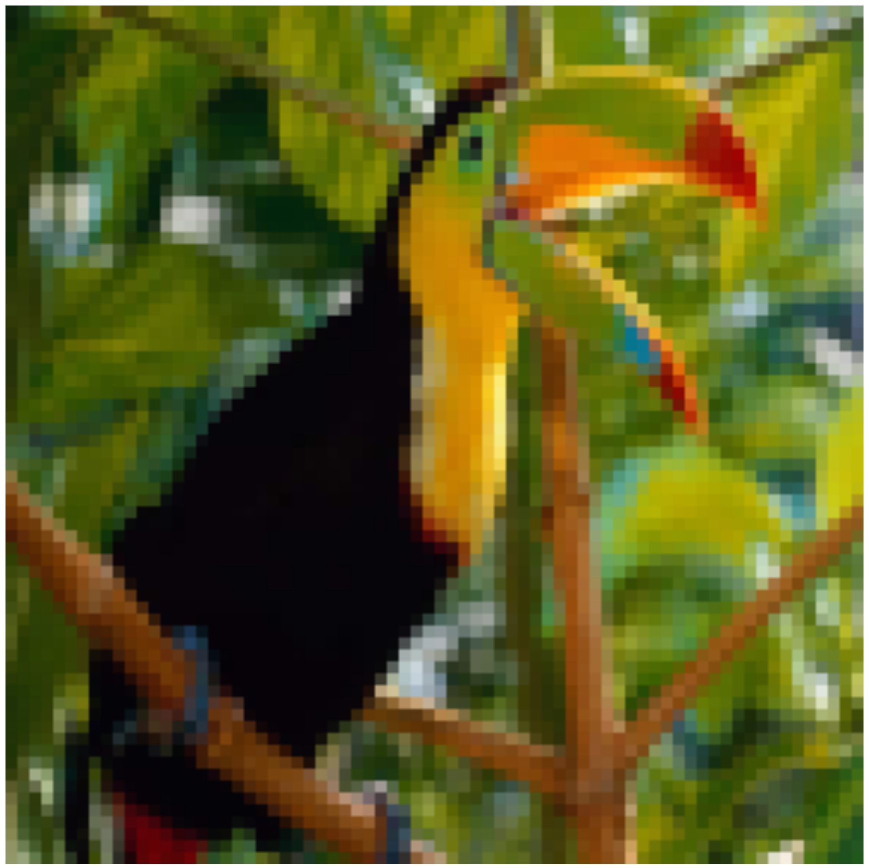
- Predict color per pixel, given grayscale input



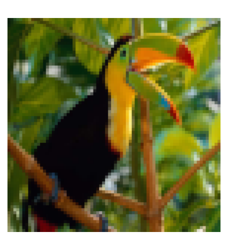
[Zhang et al. 2016]

Super-Resolution

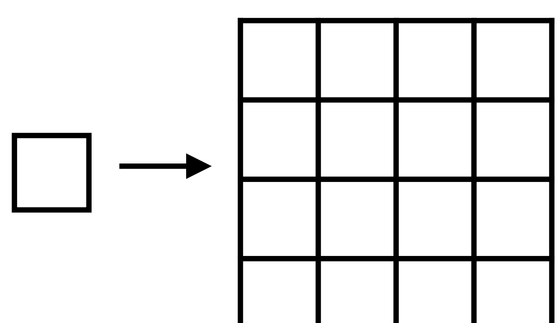
- Predict high resolution RGB, given low resolution RGB input



4 x downsampled

real size = 

bicubic upsample

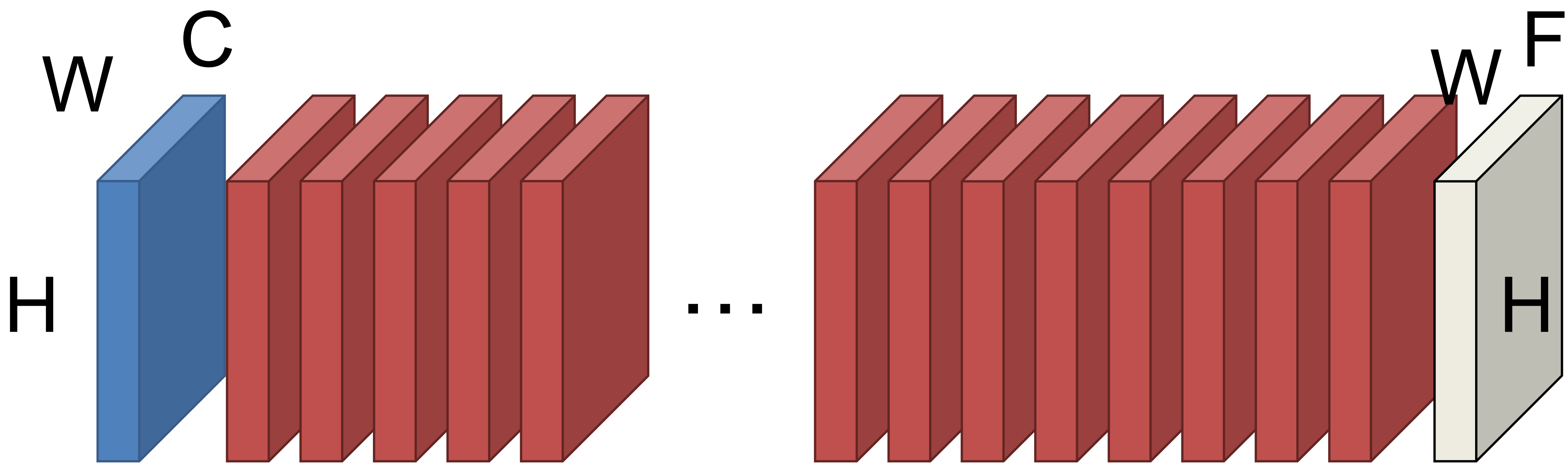


4 x superresolution

1 pixel → 16 pixels

[Ledig et al. 2017]

Why Not Stack Convolutions?



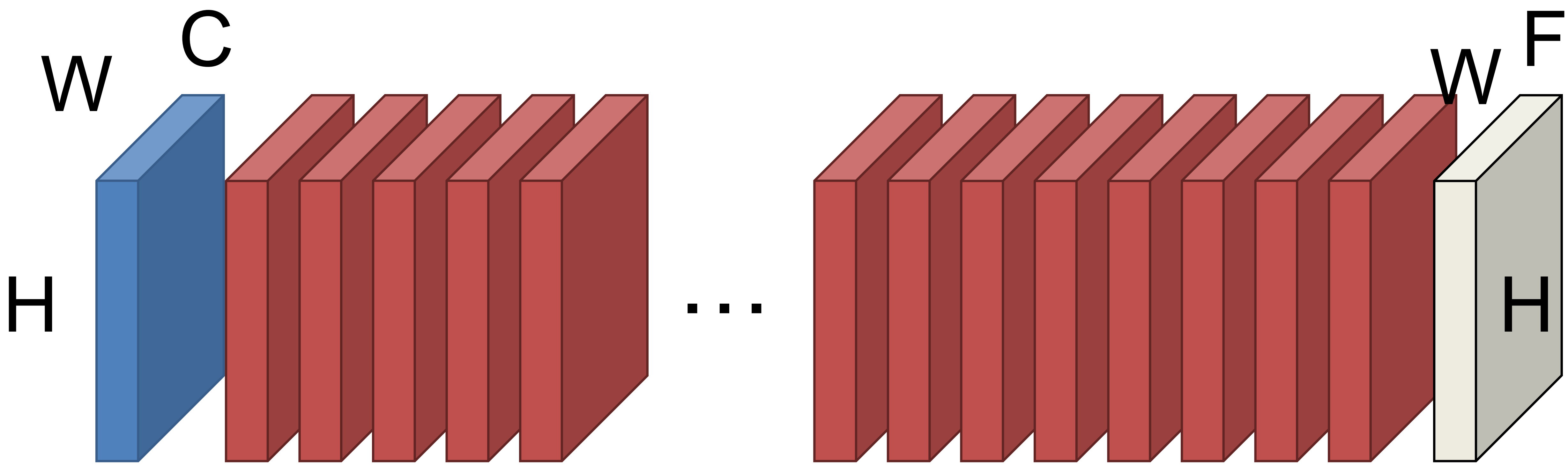
n 3x3 convs have a receptive field of 2n+1 pixels

How many convolutions until >=200 pixels?

100

[David Fouhey]

Why Not Stack Convolutions?



Suppose 200 3x3 filters/layer, H=W=400

Storage/layer/image: 200 * 400 * 400 * 4 bytes = 122MB

Uh oh!*

*100 layers, batch size of 20 = 238GB of memory!

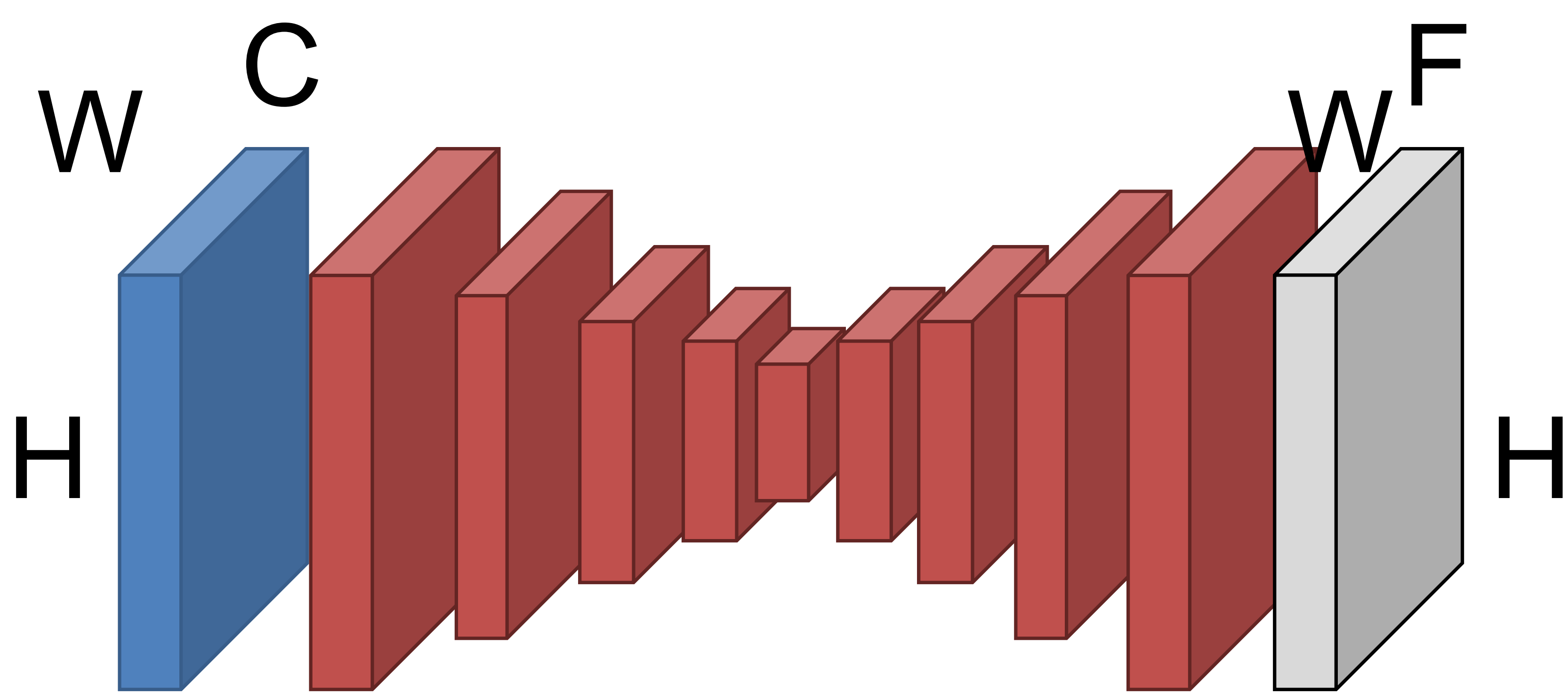
[David Fouhey]

Encoder-Decoder

Key idea: First **downsample** towards middle of network. Then **upsample** from middle.

How do we downsample?

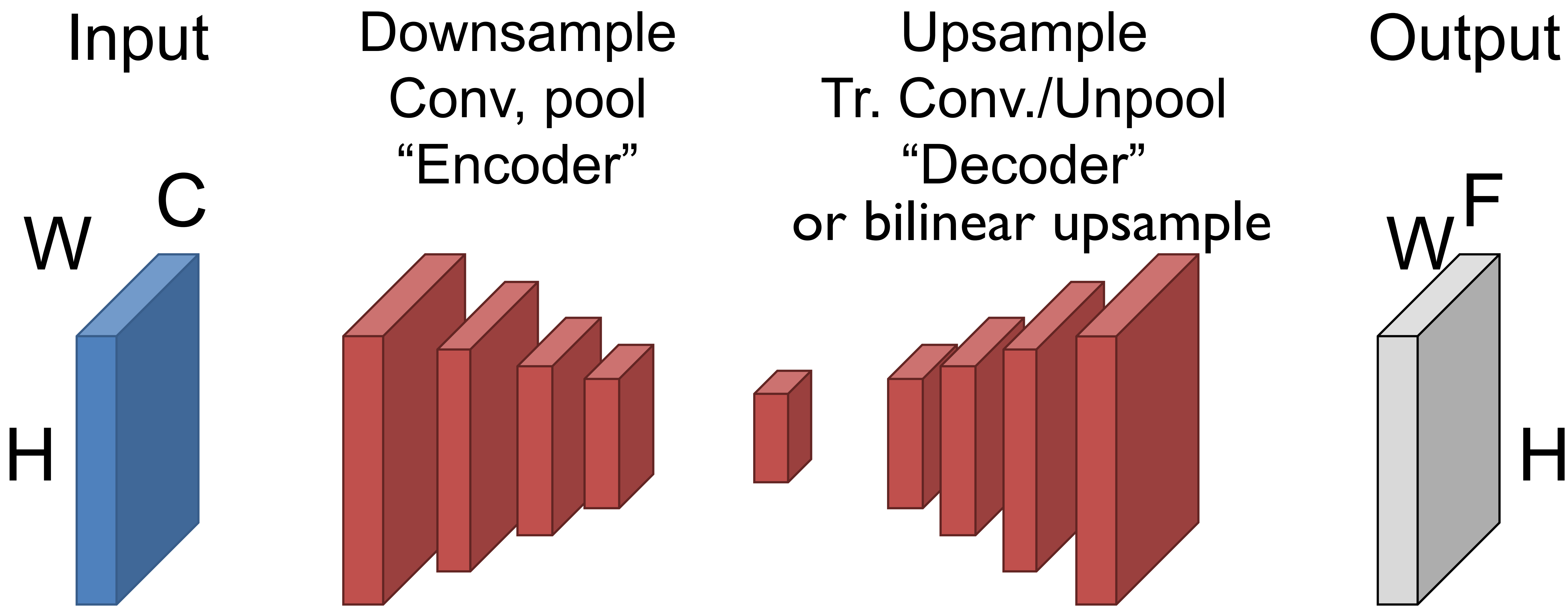
Convolutions, pooling



[David Fouhey]

Putting it Together

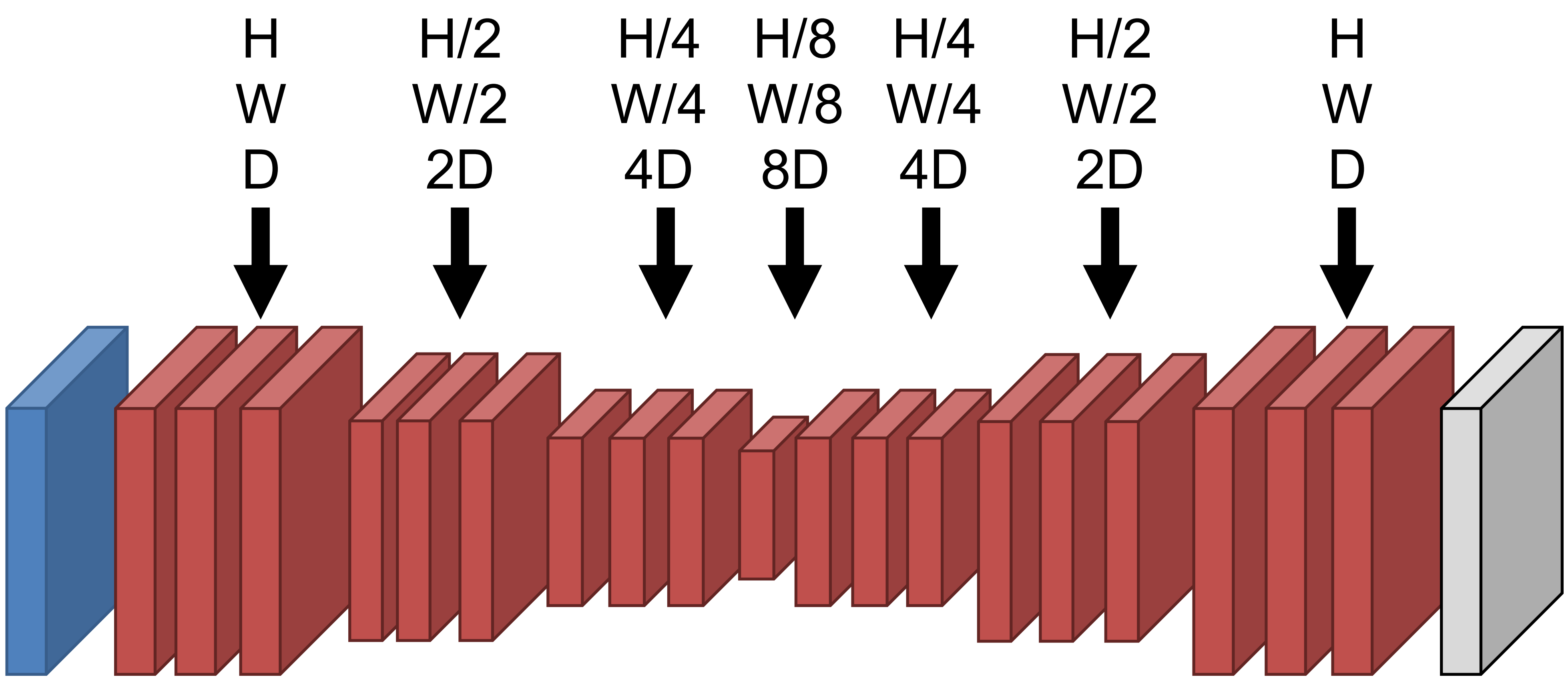
Convolutions + pooling downsample/compress/encode
Transpose convs./unpoolings upsample/uncompress/decode



[David Fouhey]

Putting It Together – Block Sizes

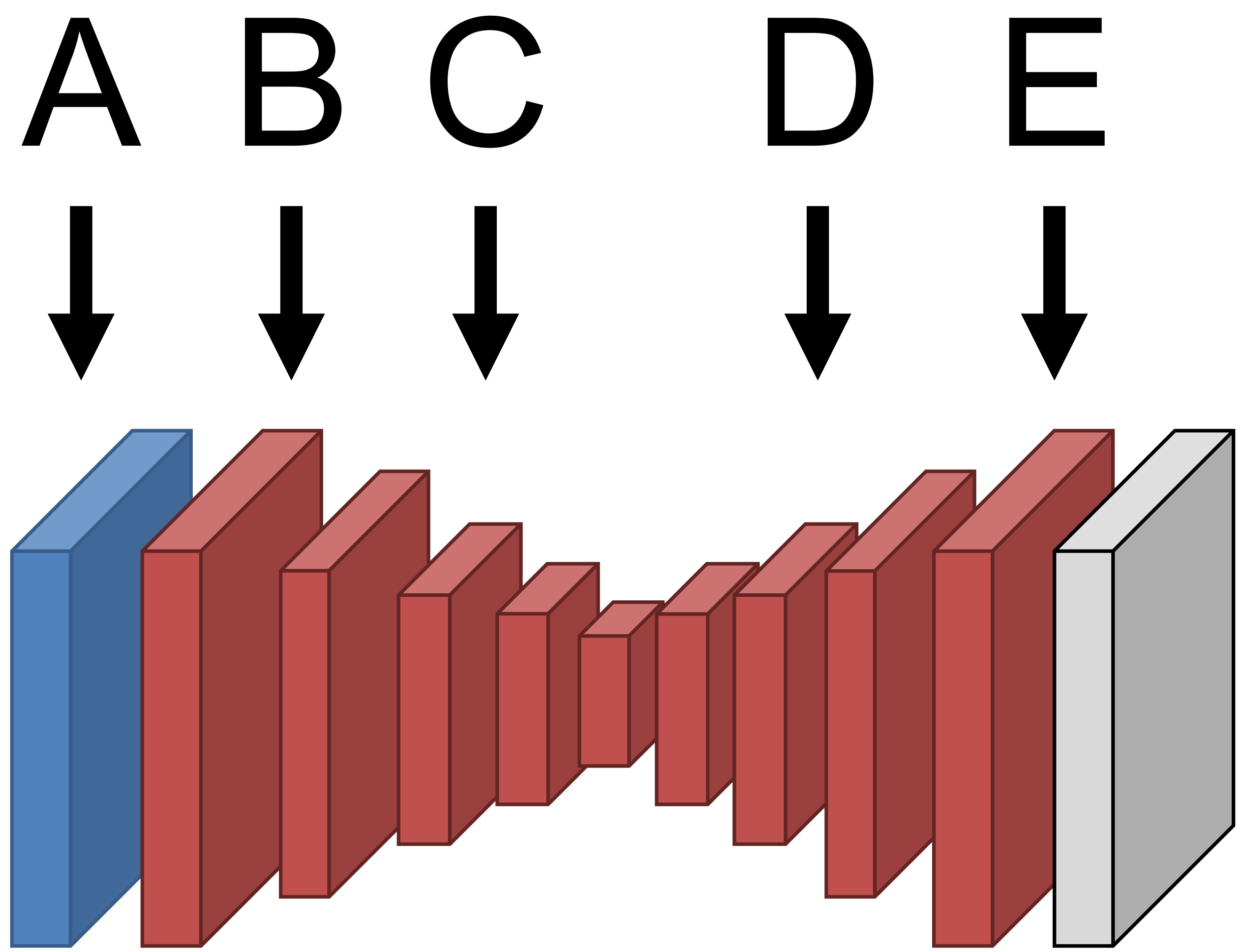
- Often multiple layers at each spatial resolution.
  - Often halve spatial resolution and double feature depth every few layers

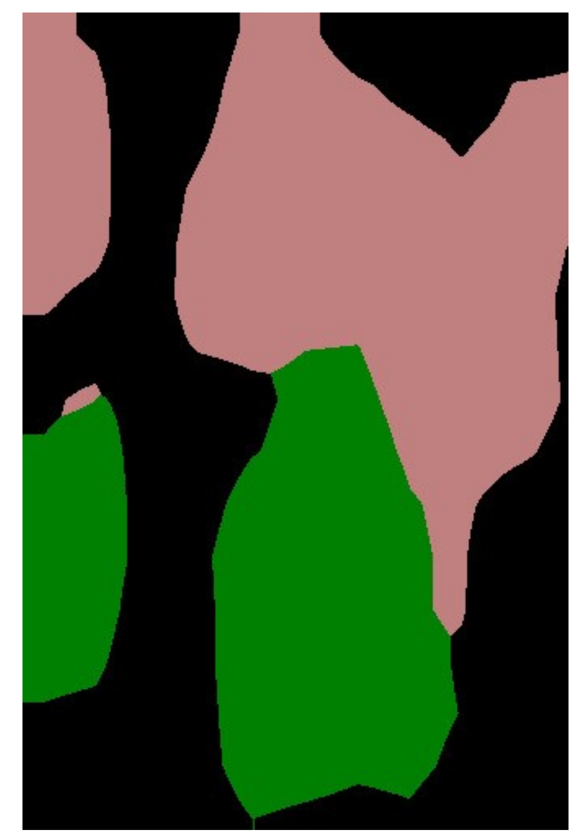


[David Fouhey]

Missing Details

Where is the useful information about the high-frequency details of the image?





Result from Long et al. *Fully Convolutional Networks For Semantic Segmentation*. CVPR 2014

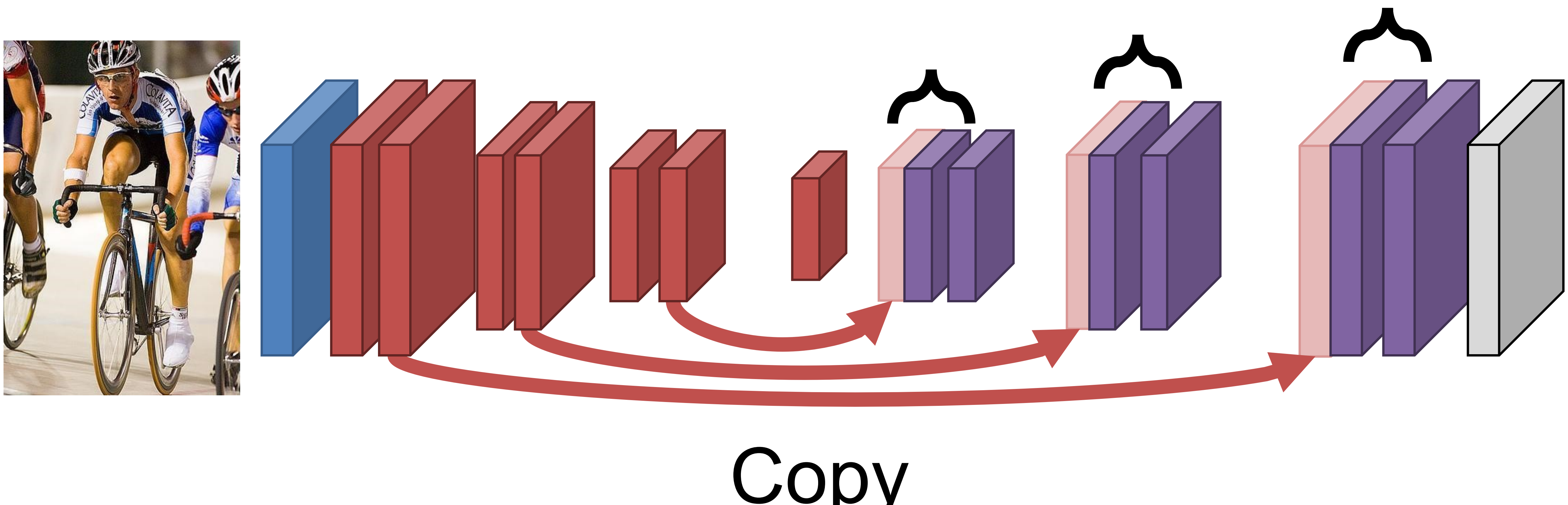
[David Fouhey]

Missing Details

How do you send details forward in the network?

You copy the activations forward.

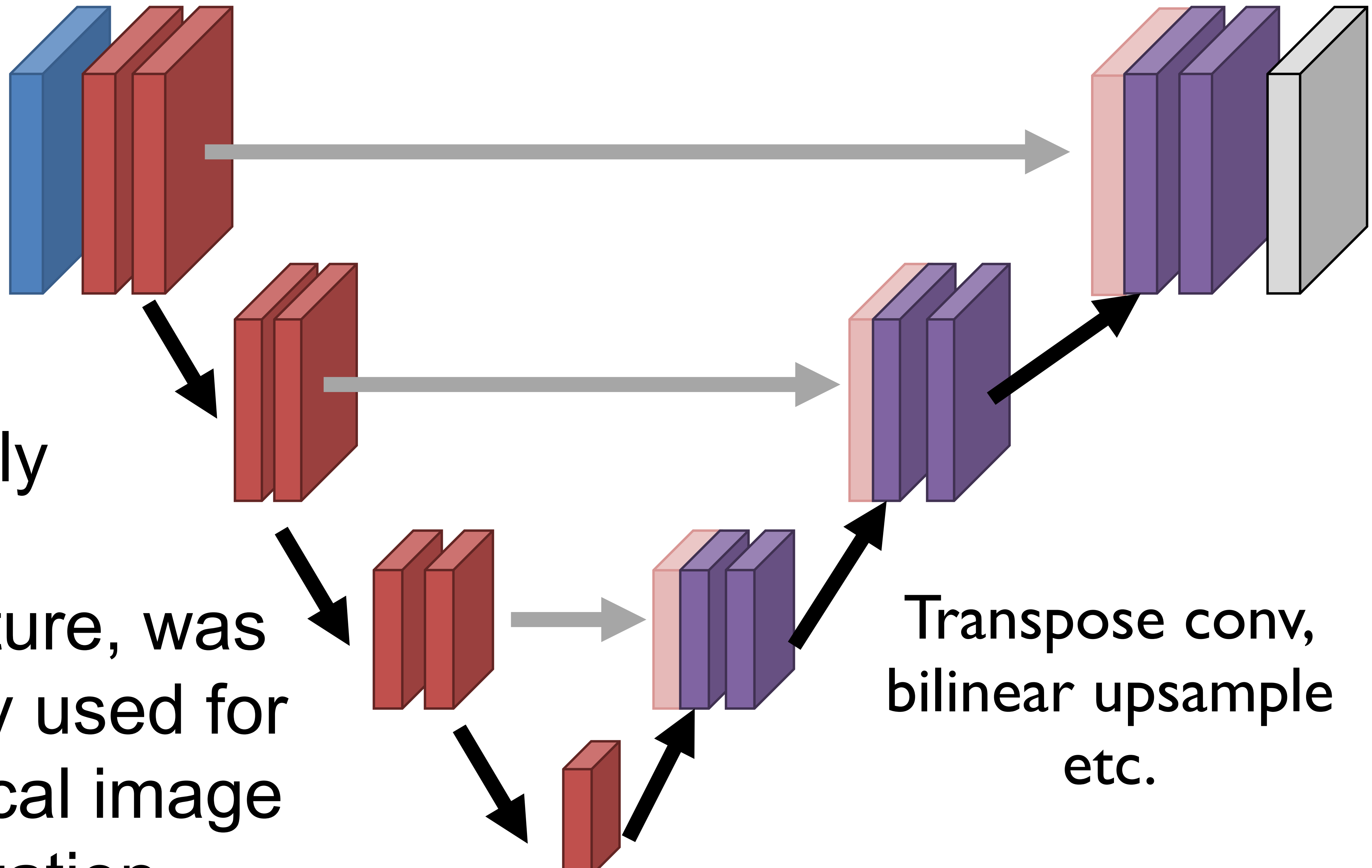
Subsequent layers at the same resolution figure out how to fuse things.



Result from Long et al. *Fully Convolutional Networks For Semantic Segmentation*. CVPR 2014

[David Fouhey]

U-Net



Extremely popular architecture, was originally used for biomedical image segmentation.

Transpose conv, bilinear upsample etc.

Ronneberger et al. “U-Net: Convolutional Networks for Biomedical Image Segmentation. MICCAI 2015

[David Fouhey]

Single-View Depth Estimation

[T. Zhou, A. Geiger]

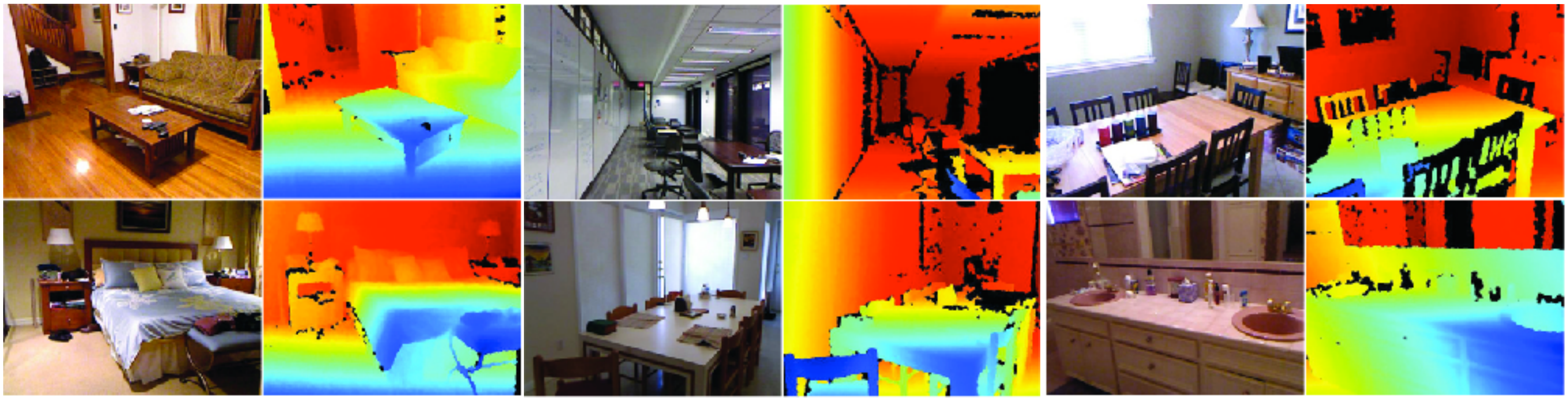
Single-View Depth Estimation

[T. Zhou, A. Geiger]

Single-View Depth Estimation

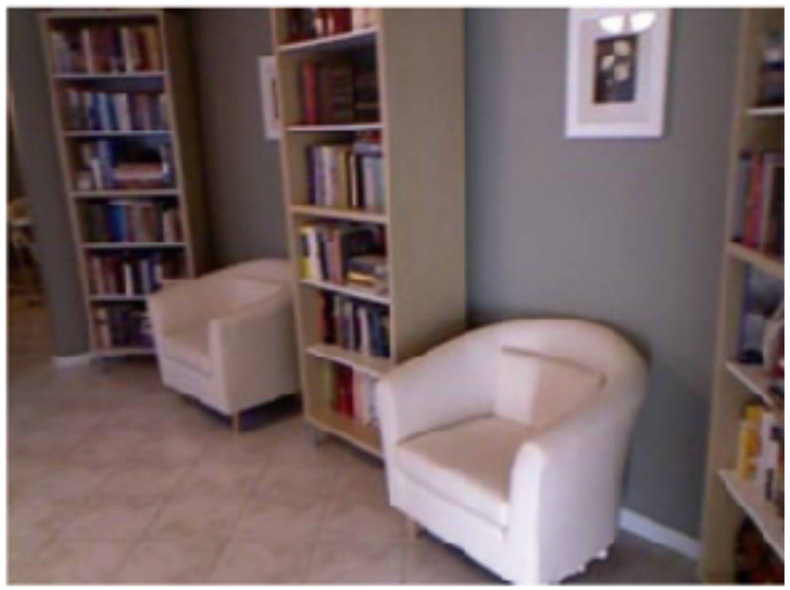
[T. Zhou, A. Geiger]

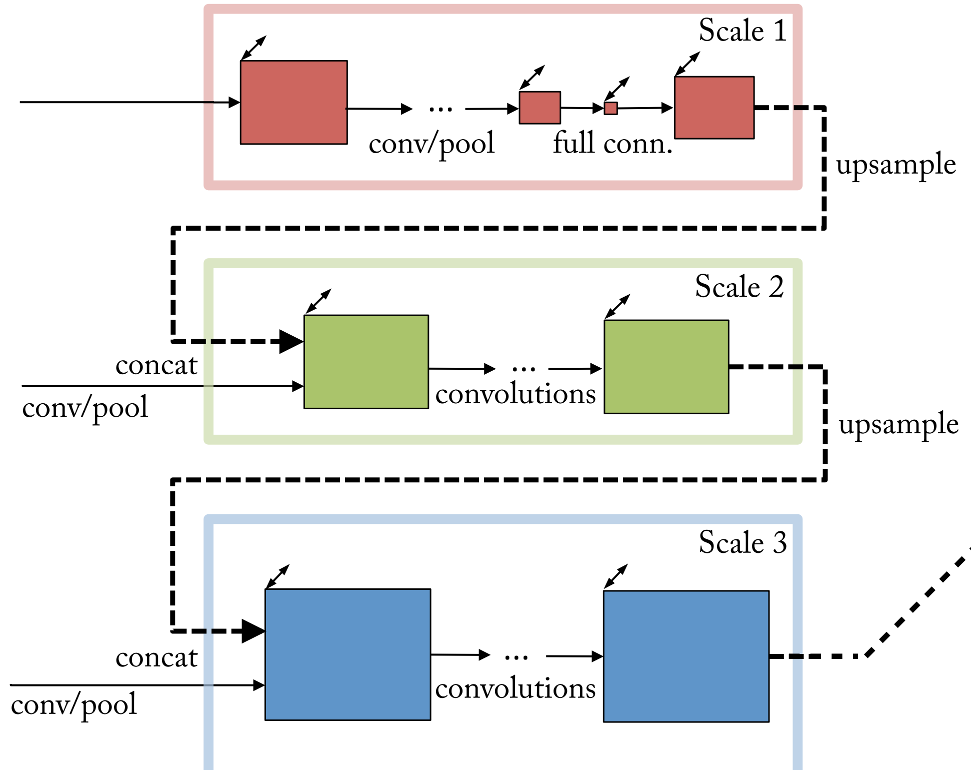
NYU Depth v2 Dataset



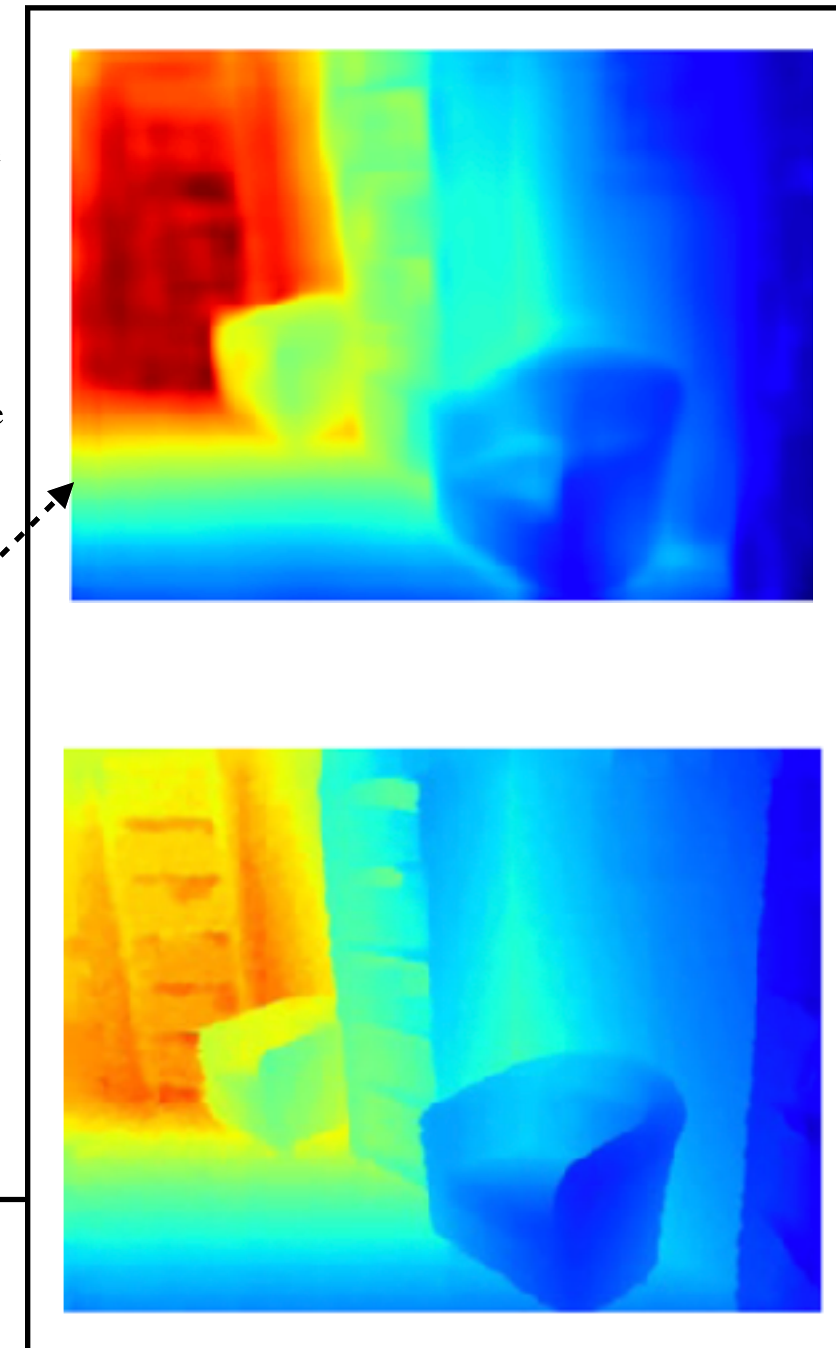
- 400K RGBD frames captured using Microsoft Kinect
- ~1500 have segmentation labels (26 classes) as well
- The dataset has depth holes, note offset between RGB and NIR cameras, and NIR dot projector, also raw RGB + D frames are not synchronized
- Synchronized and filled subset of 50K images by [Alhashim Wonka 2018] — see Project 4 description
- Limited to indoor scenes due to active NIR illumination

NYU Depth Estimation





multi-scale architecture

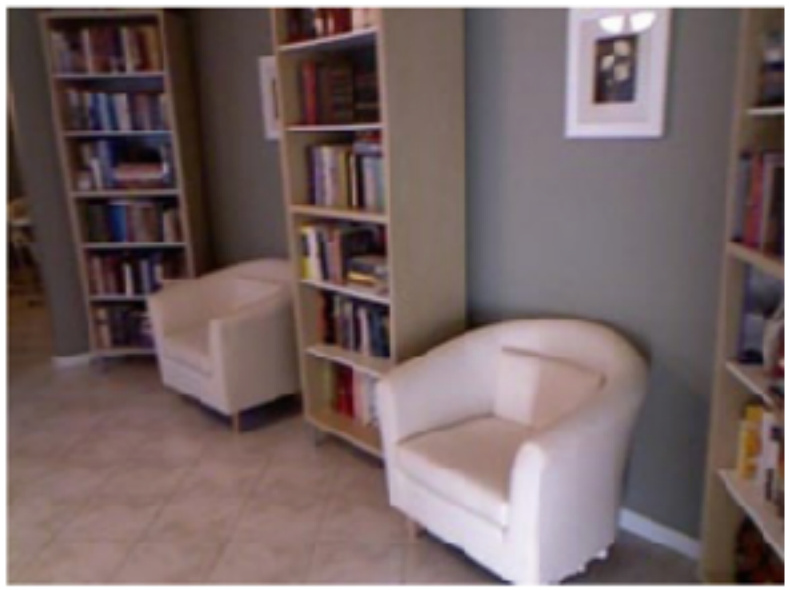


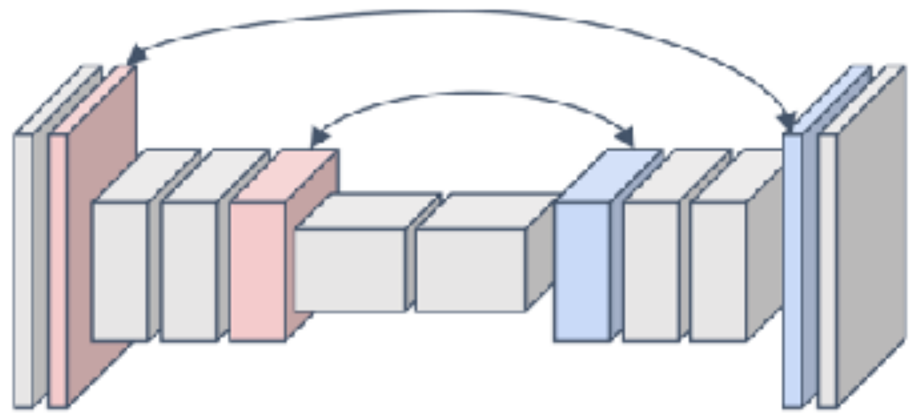
Direct supervision via Kinect RGB+D

Loss, e.g., L2

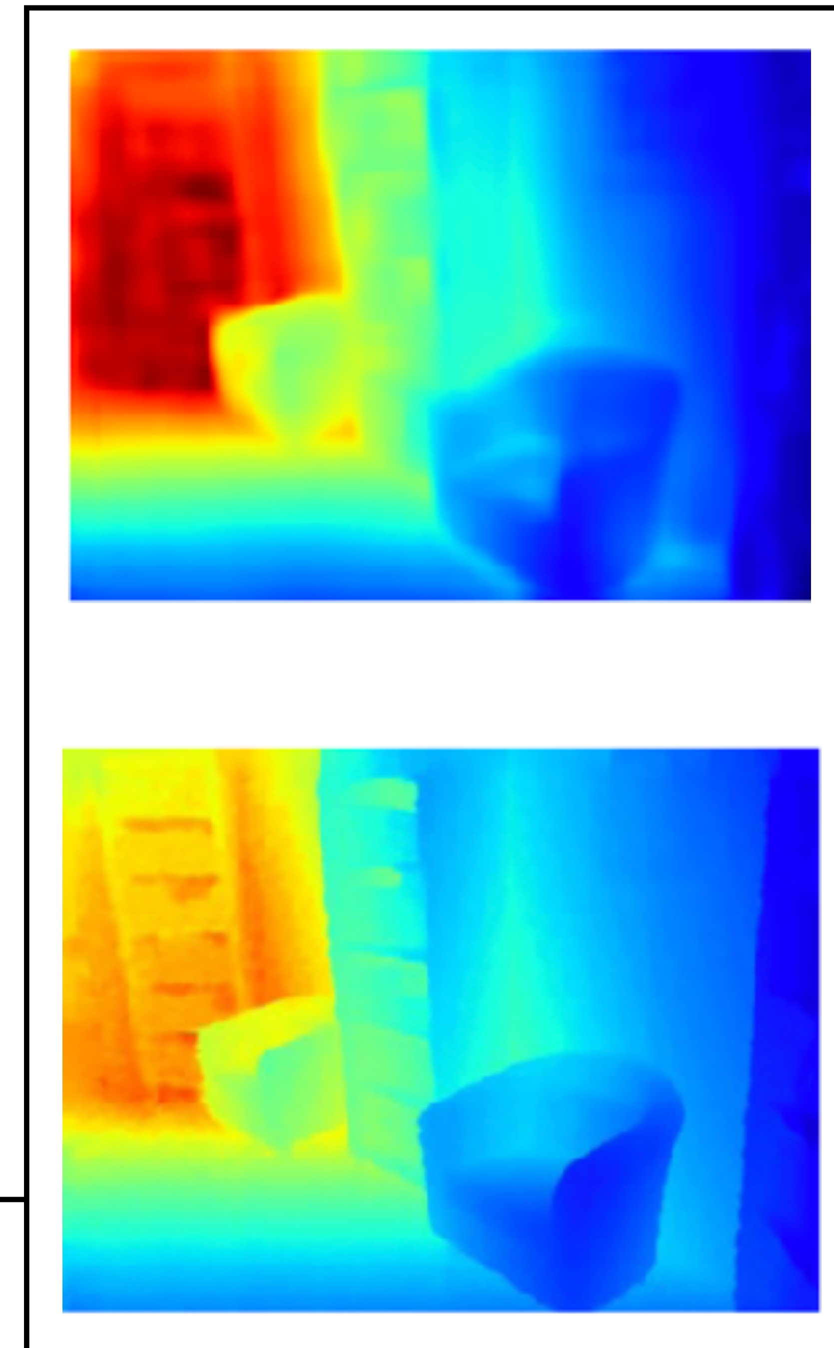
[Eigen Fergus 2015]

NYU Depth Estimation





U-Net with skip connections



Direct supervision via Kinect RGB+D

Loss, e.g., L2

NYU Depth Estimation

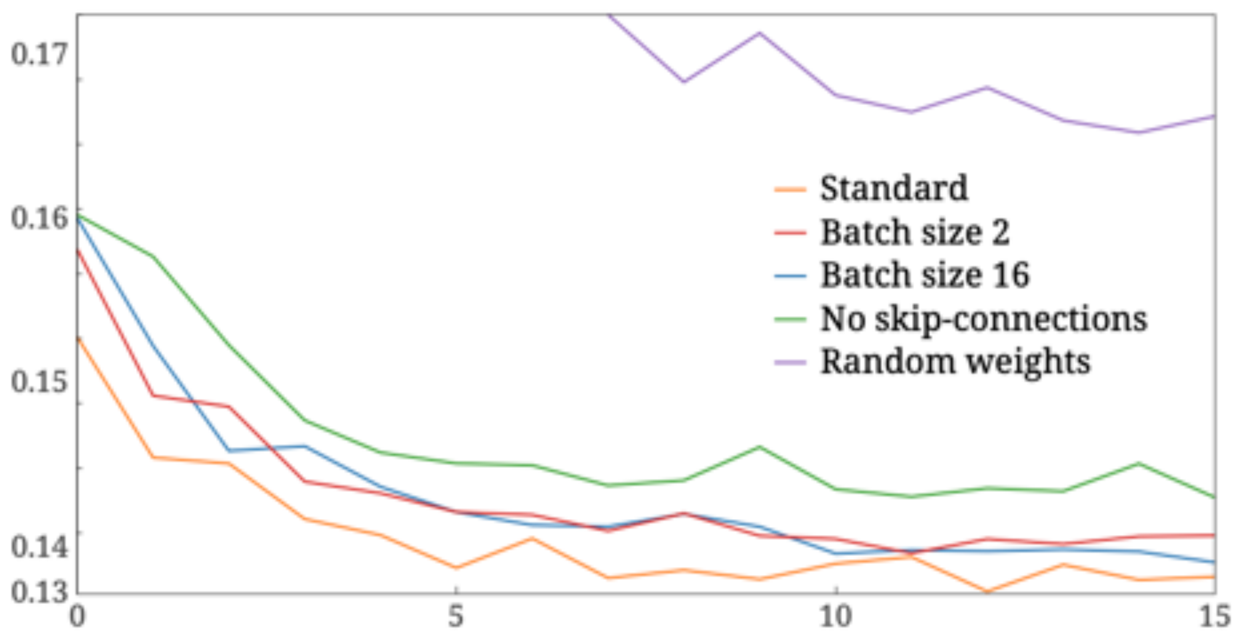
- ImageNet Pretrained DenseNet 169 with skip connections

[DenseNet Huang et al 2018] [Alhashim Wonka 2019]

Depth Estimation: Pre-Training

- ImageNet Pretrained DenseNet 169 with skip connections

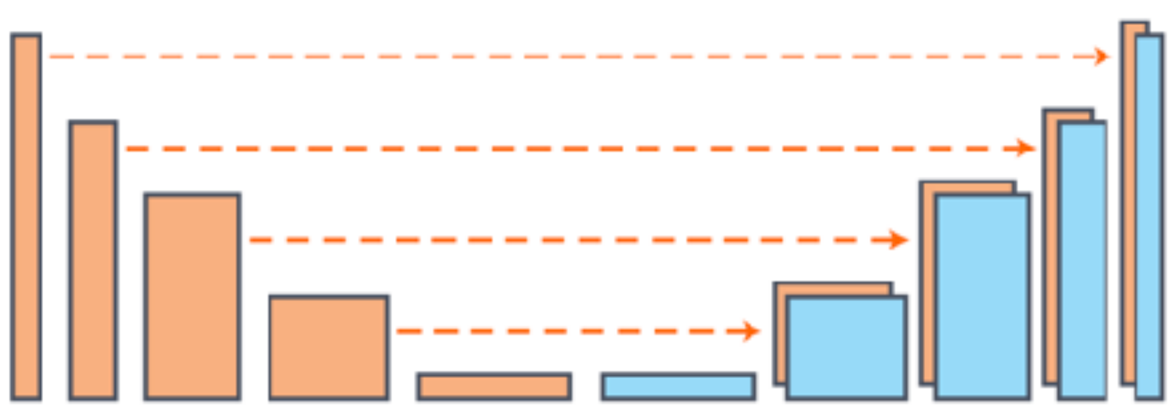
loss



epoch

No pre-training

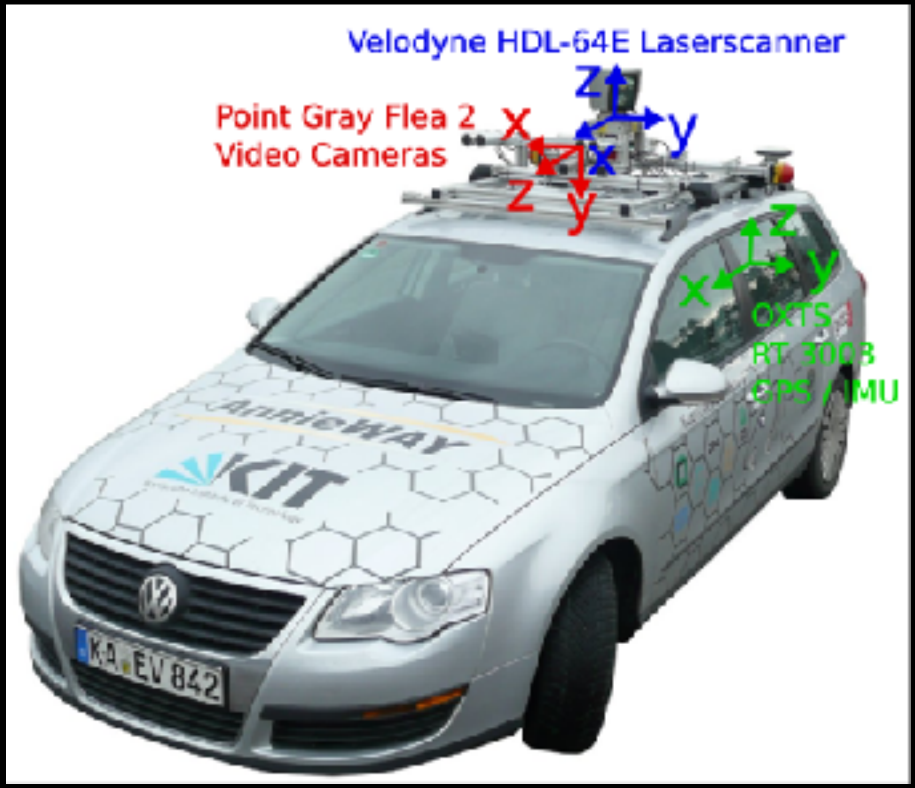
No skip connections

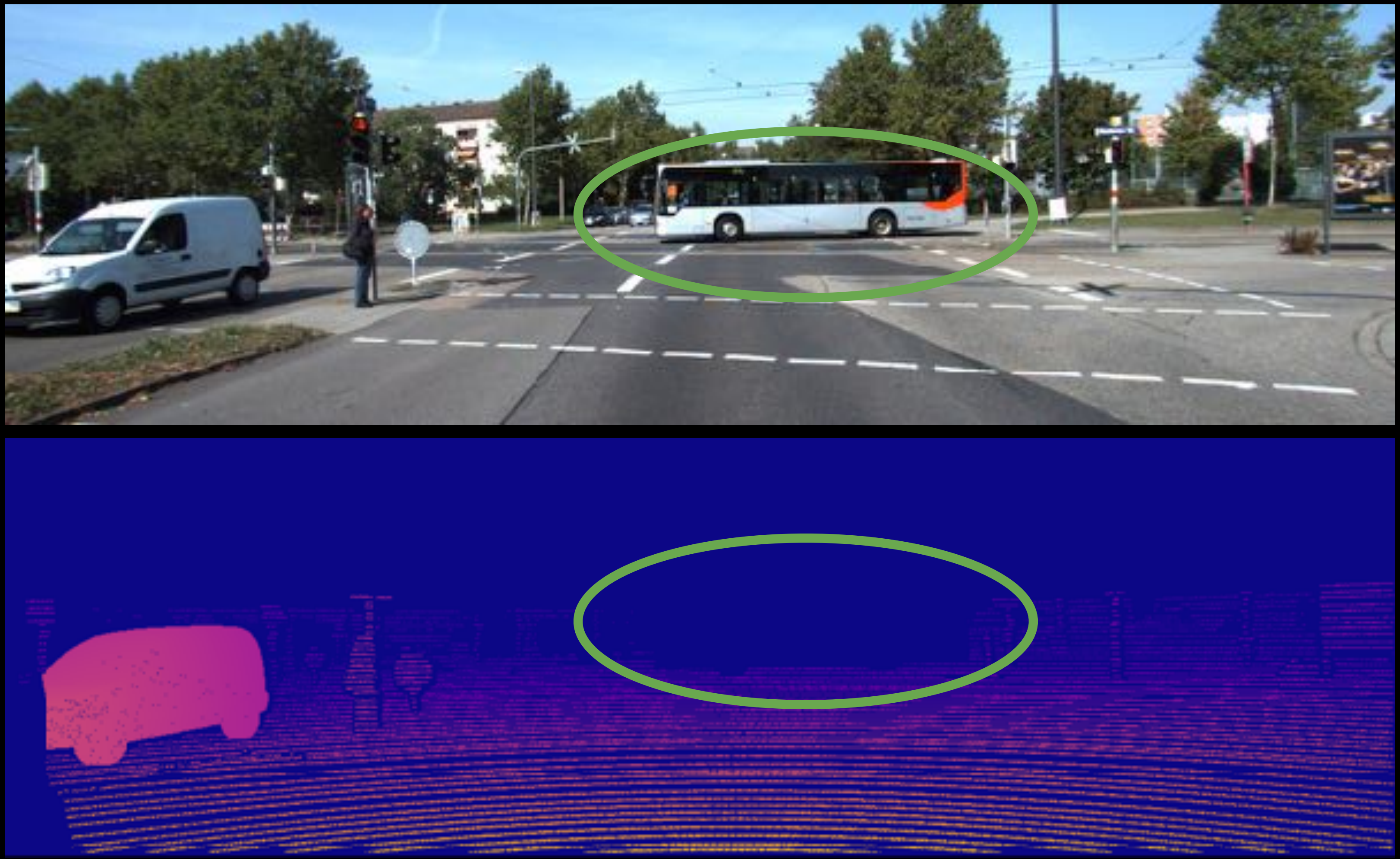


orange = pretrained Densenet 169
decoder blocks = bilinear ↑2 → 2 x conv

[DenseNet Huang et al 2018] [Alhashim Wonka 2019]

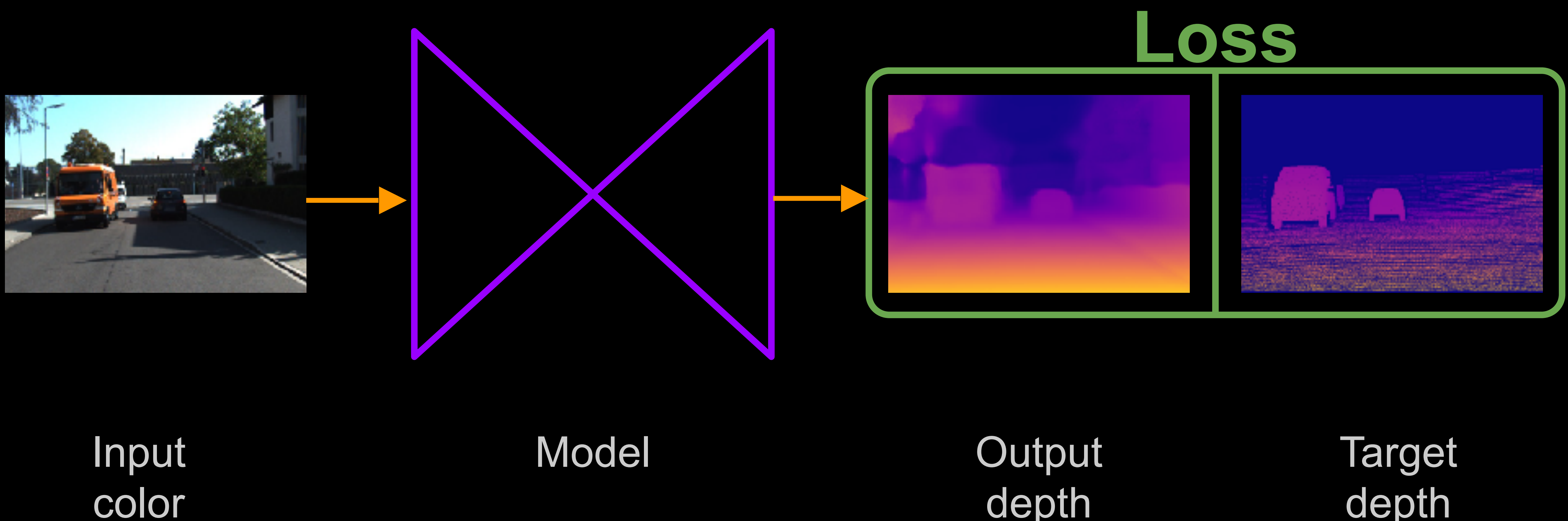
KITTI 2015



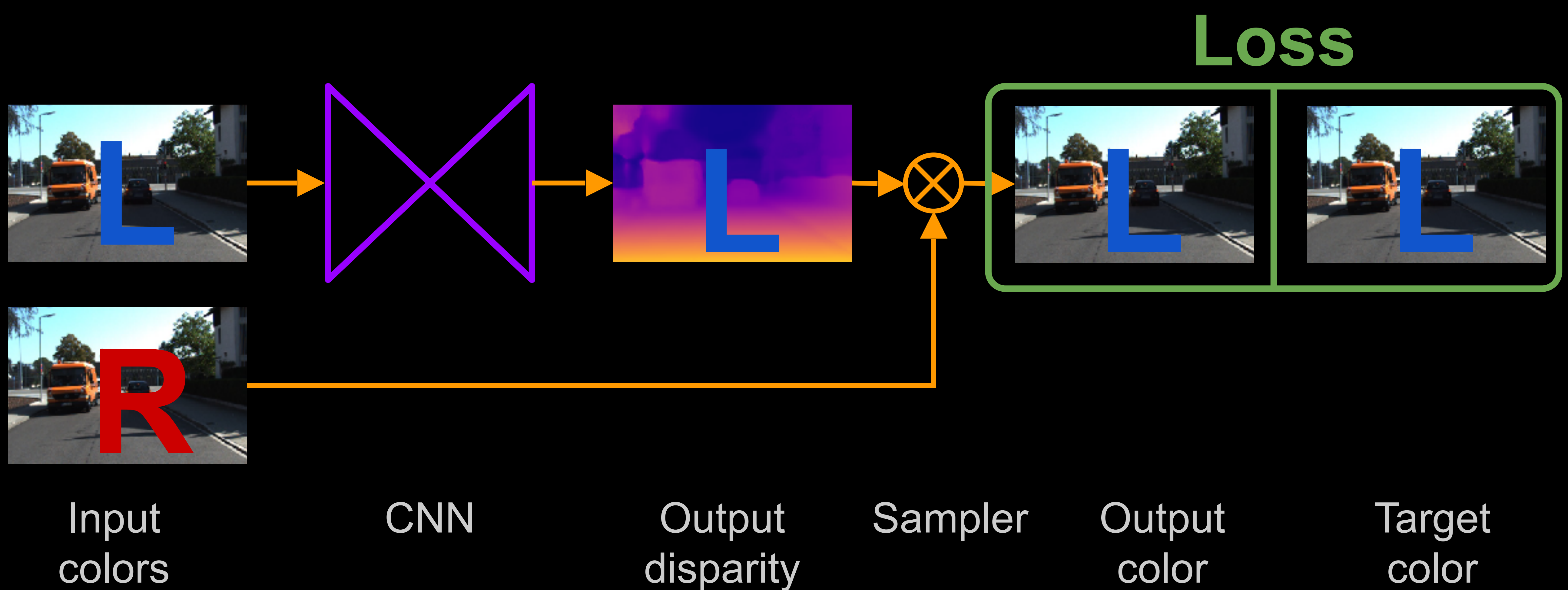


http://www.cvlibs.net/datasets/kitti/ [Slides: Clement Godard]

Supervised Depth Estimation



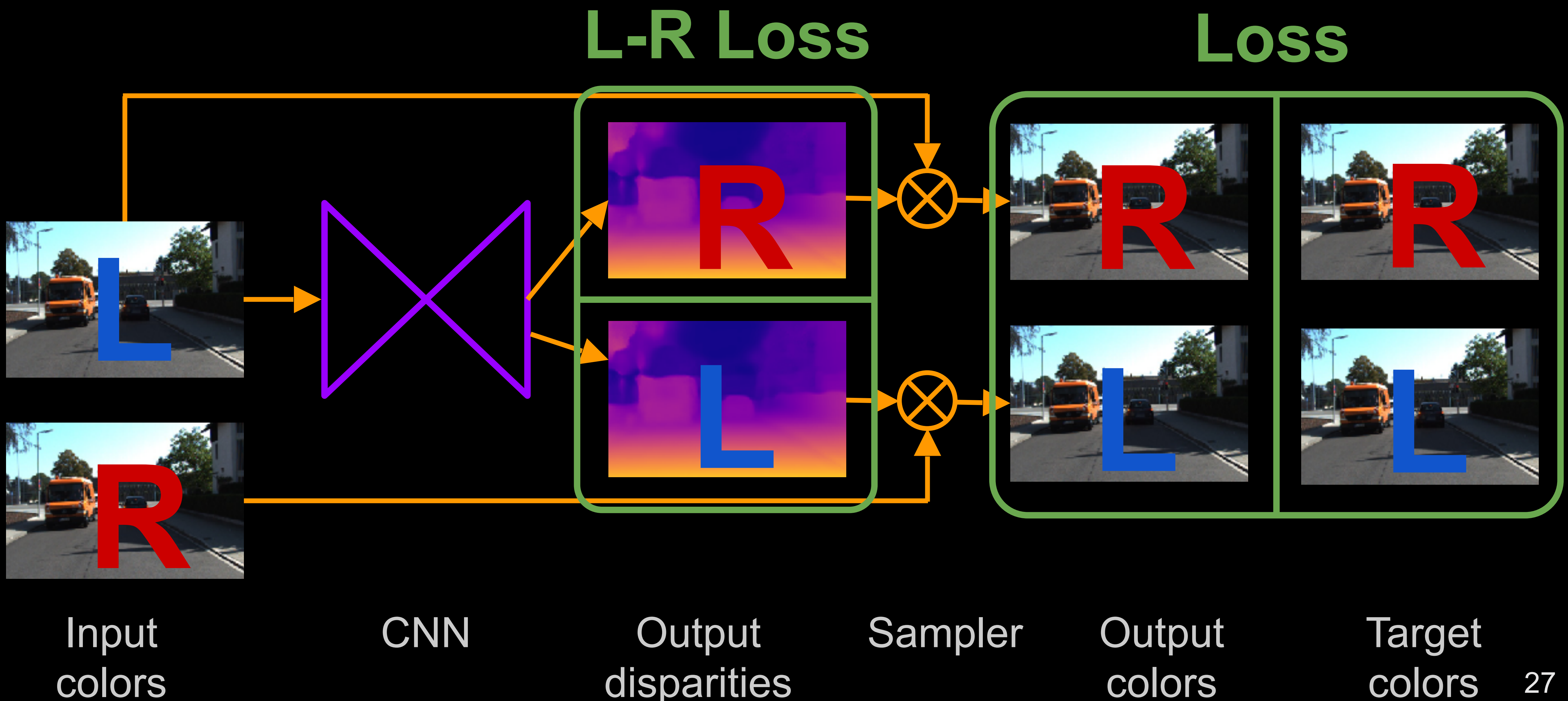
Unsupervised Depth Estimation - Concept



Note: sampling must be differentiable (dpixel/ddepth), e.g., bilinear

[Godard et al. 2016] [Garg et al 2016]

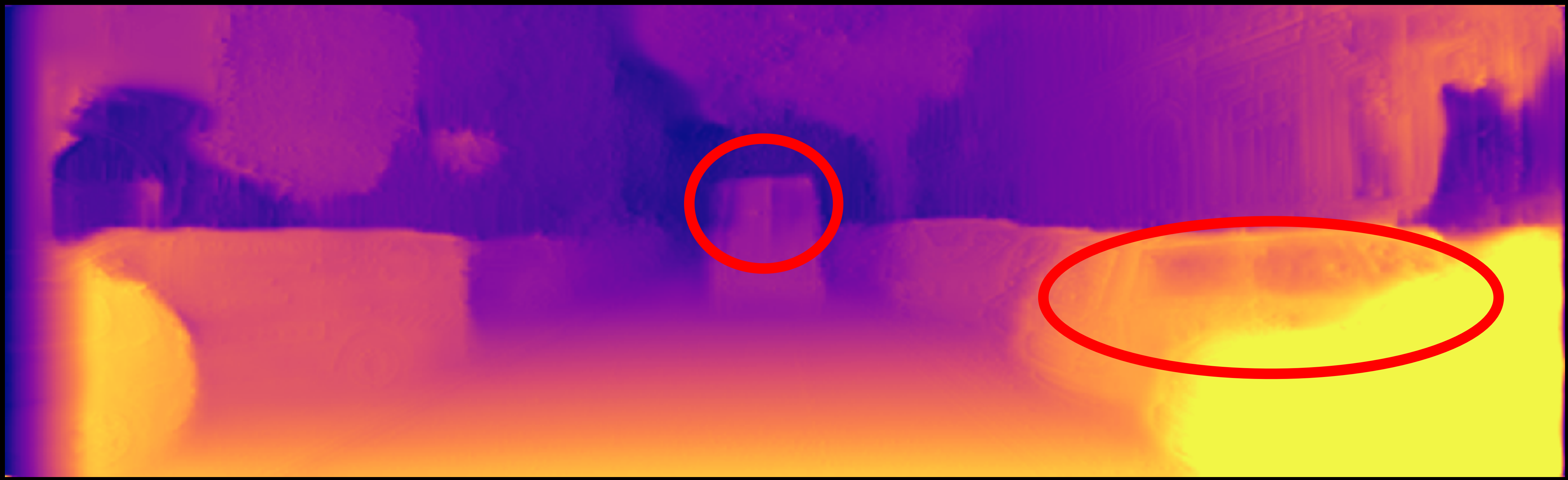
Unsupervised Depth: Left-Right Consistency Loss



Input

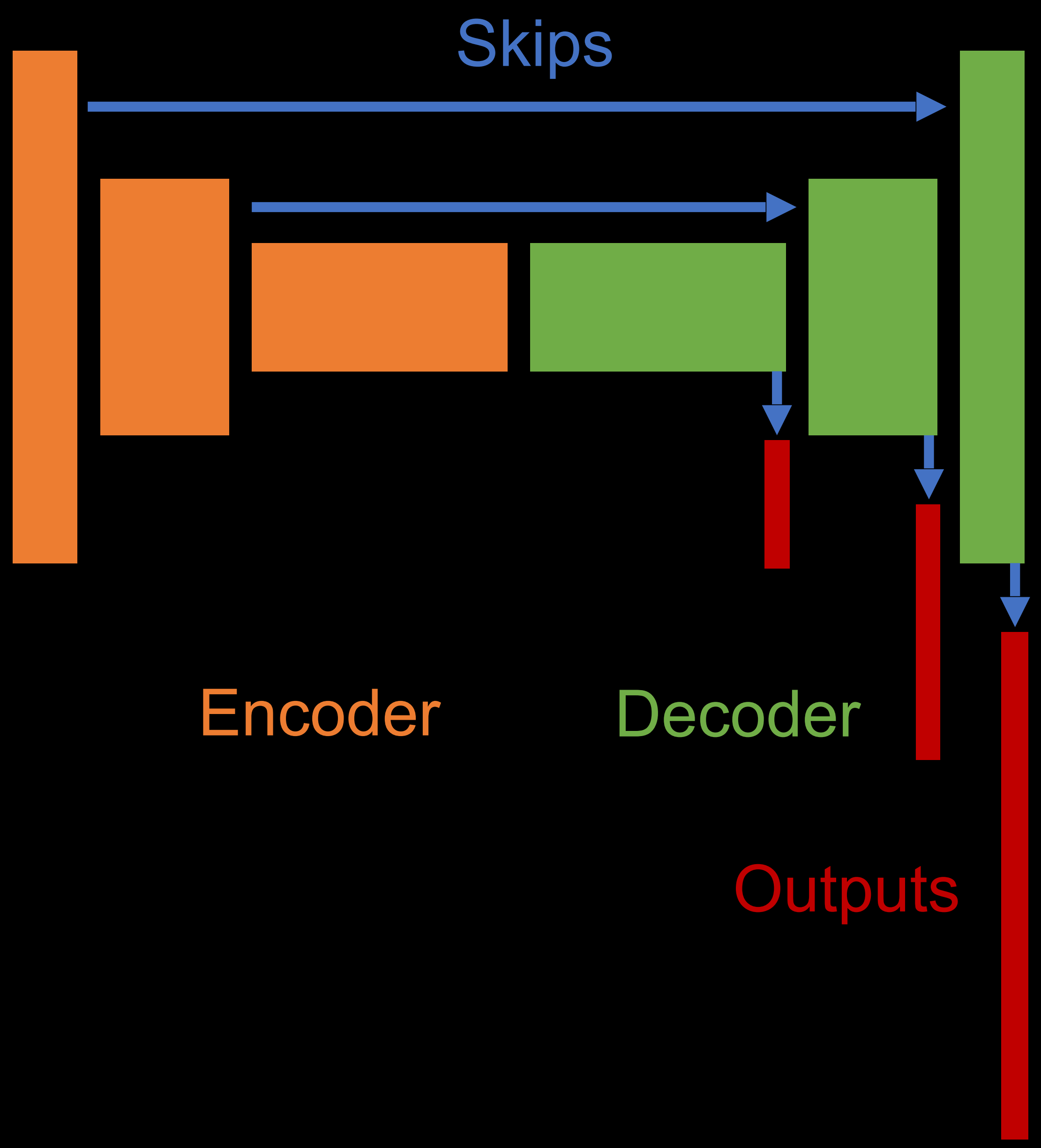
Without Left-Right Consistency

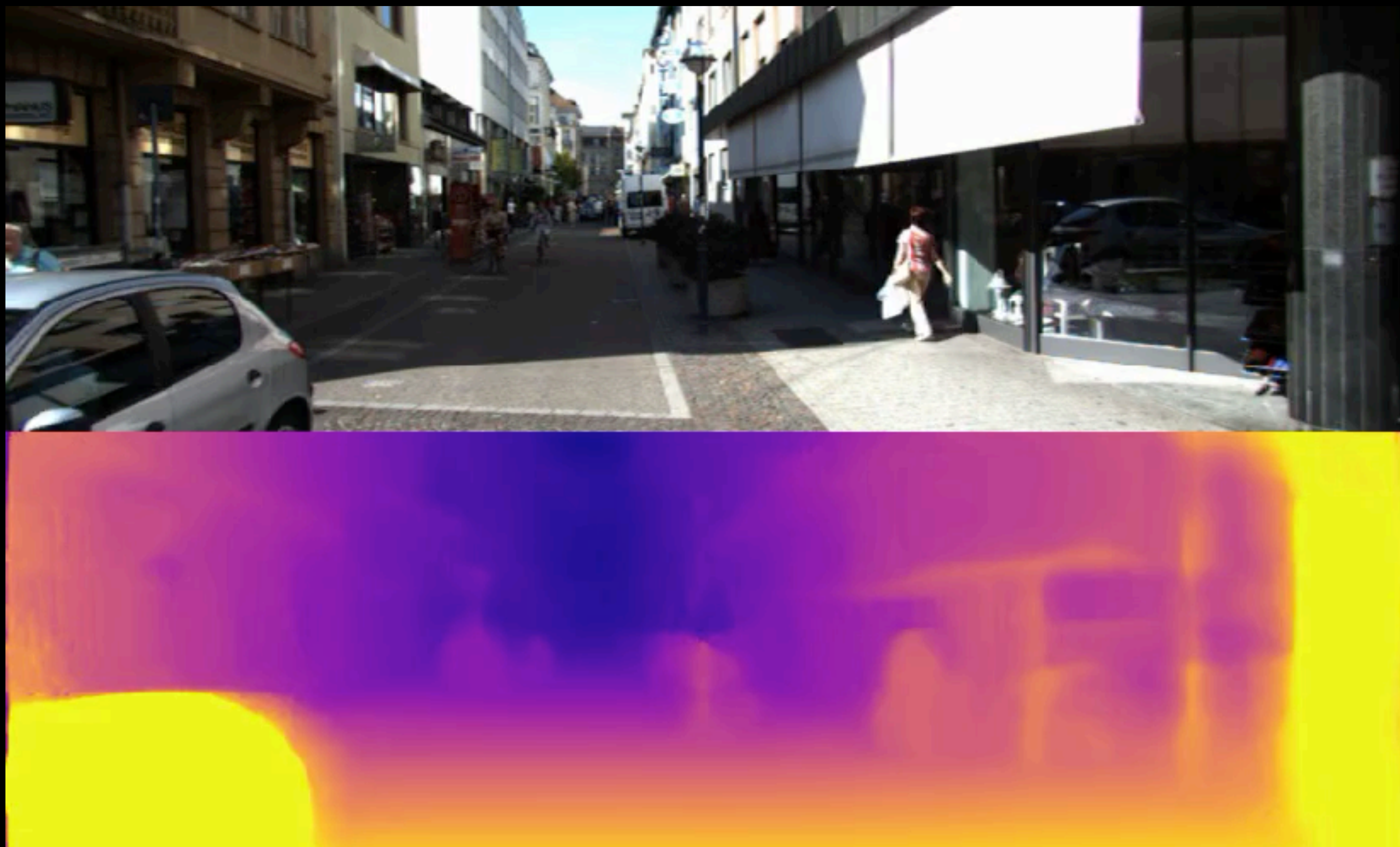
With Left-Right consistency



Architecture

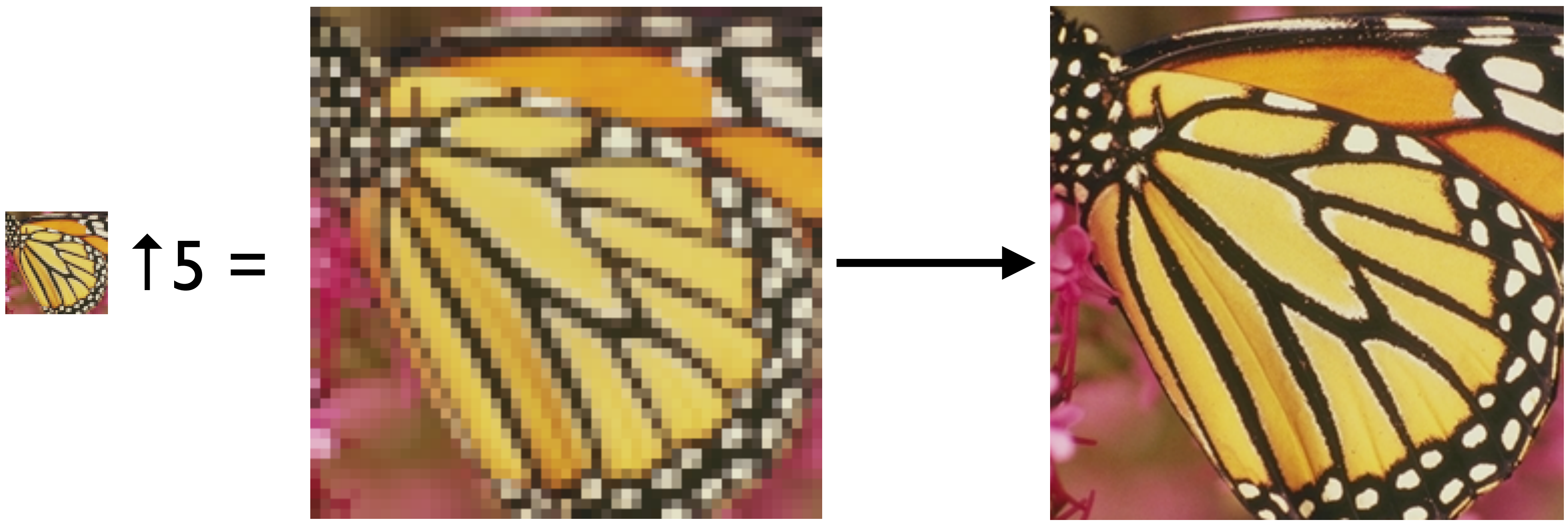
- Fully convolutional
  - Choose your favorite encoder
- Skip connections
  - Similar to DispNet and FlowNet
- Multiscale generation
  - And Loss!
- Fast!
  - ~30fps on a Titan X





Super-Resolution

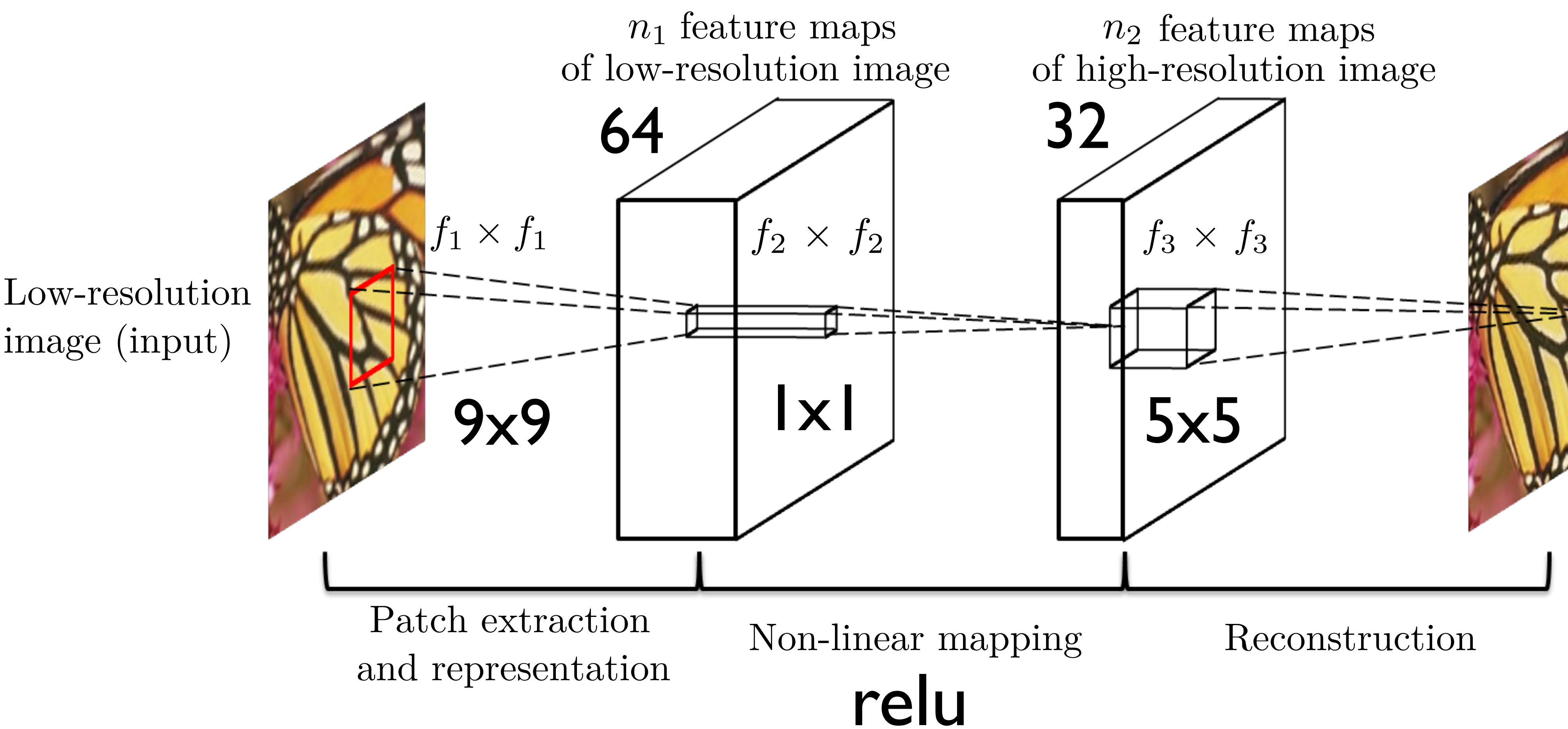
- Increase the spatial resolution of an image



- Super-res algorithms use knowledge of image statistics to predict a likely high resolution version given low-res input
- Training data is easy — just downsample images!

Super-Resolution: SRCNN

- Small networks (e.g., 3 layers) generate reasonable results



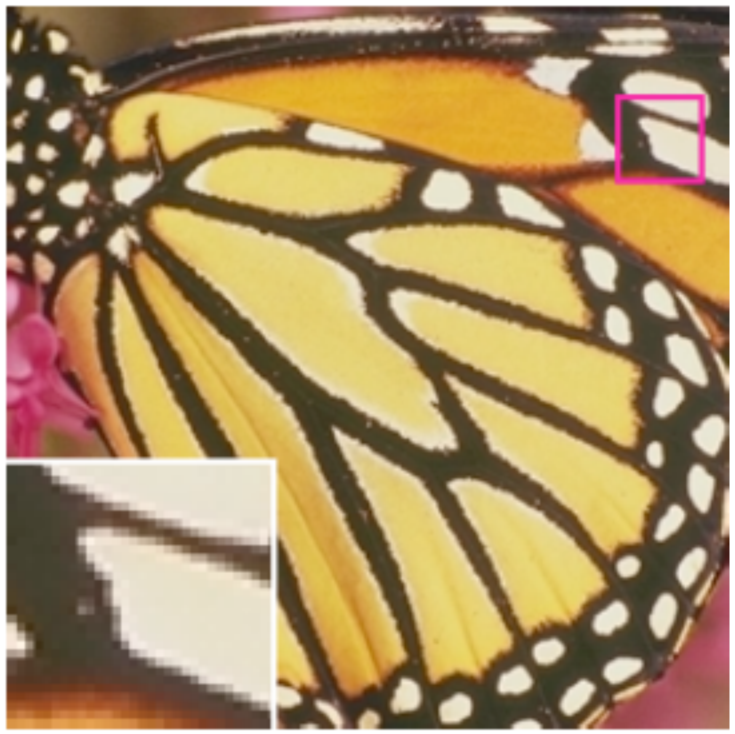
Train using
L2 loss vs
ground truth

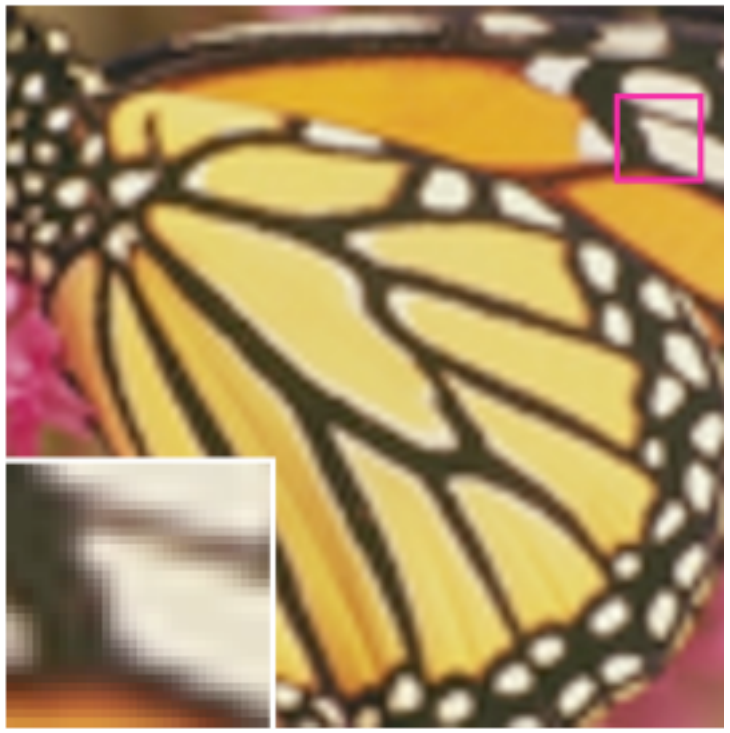
What does this suggest about super-resolution?

[SRCNN, Dong et al 2014]

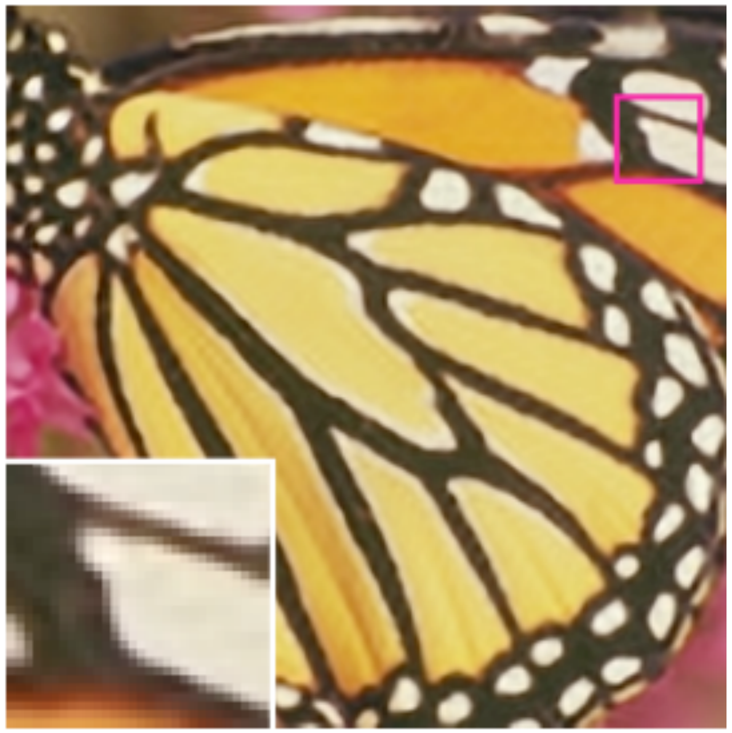
Super-Resolution: SRCNN

- Small networks (e.g., 3 layers) generate reasonable results

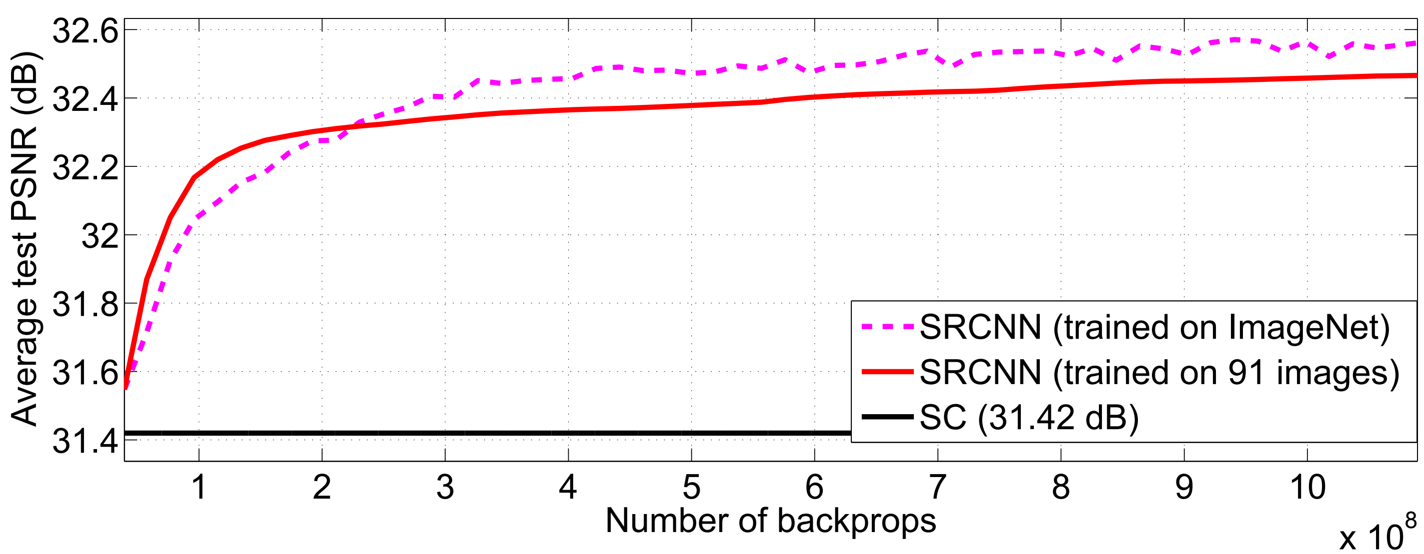




bicubic = 24.04dB



SRCNN = 27.95dB



Can be trained using a small image set (e.g., 91 images)

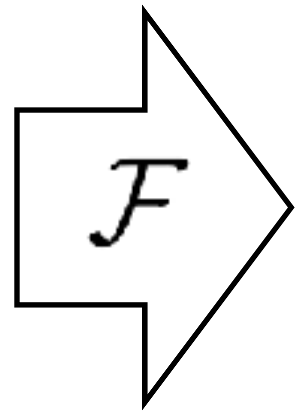
Super-Resolution

- Small networks are generally good at sharpening edges and can work well for small factor (e.g., 2) super-resolution
- Better results can be achieved by using deeper networks, + more sophisticated loss functions (perceptual loss, GANs)

Original Bicubic SRCNN Johnson et al*

*12-layer, residual conn., fully conv, VGG loss [Johnson et al. 2016]

Image Colorization

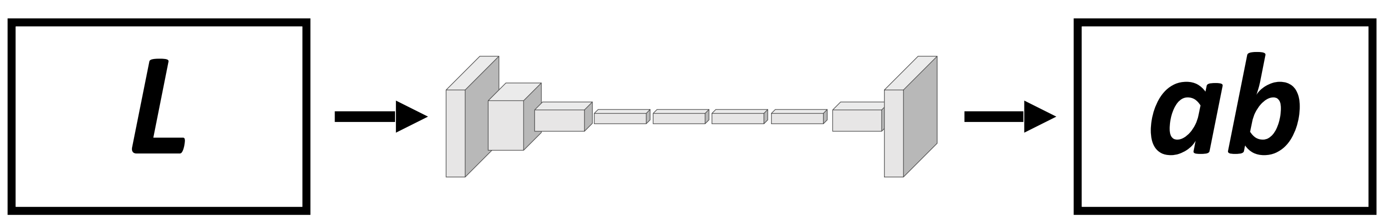


Grayscale: *L* channel

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

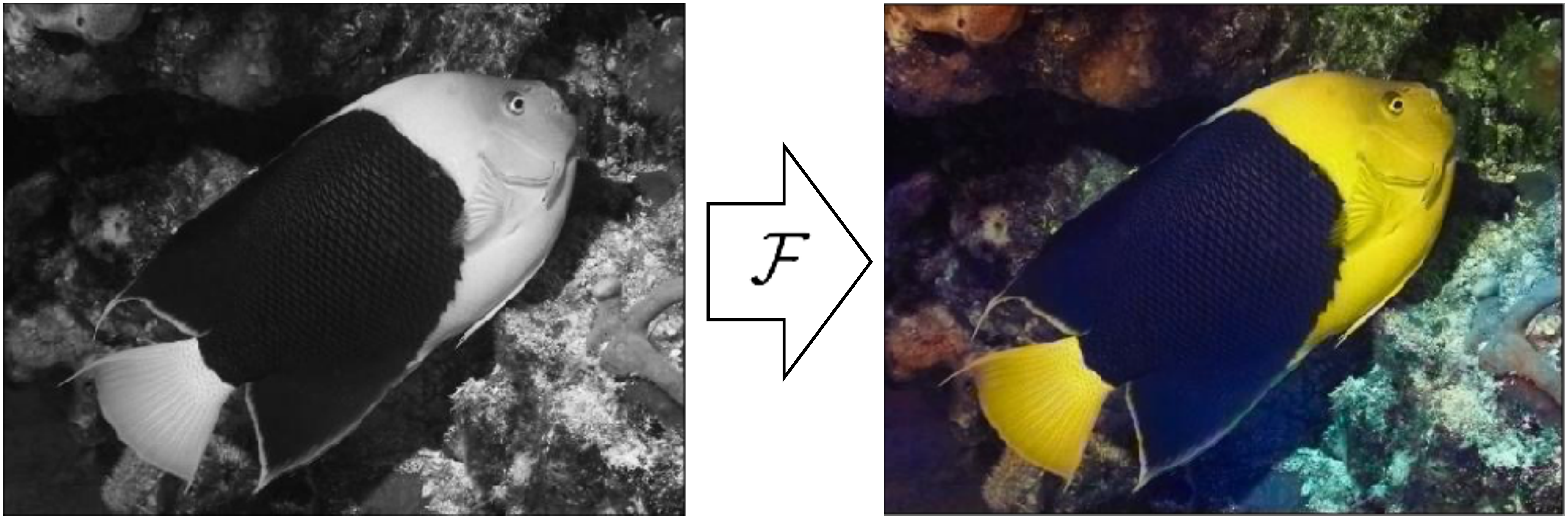
Color: *ab* channels

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



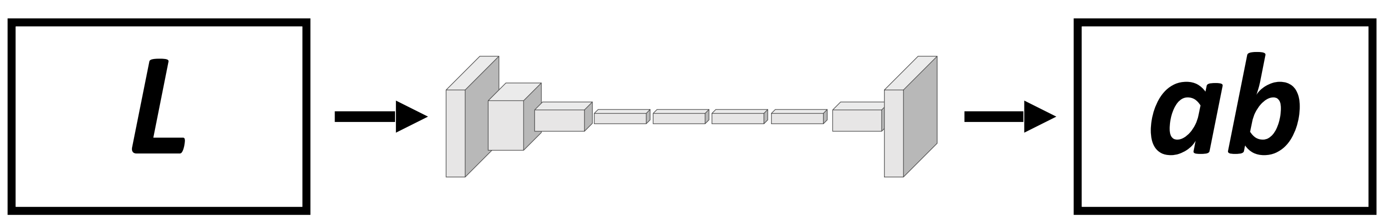
[Zhang et al. 2016]

Image Colorization



Grayscale: *L* channel $\mathbf{X} \in \mathbb{R}^{H \times W \times 1}$

Color: *ab* channels $\widehat{\mathbf{Y}} \in \mathbb{R}^{H \times W \times 2}$



[Zhang et al. 2016]

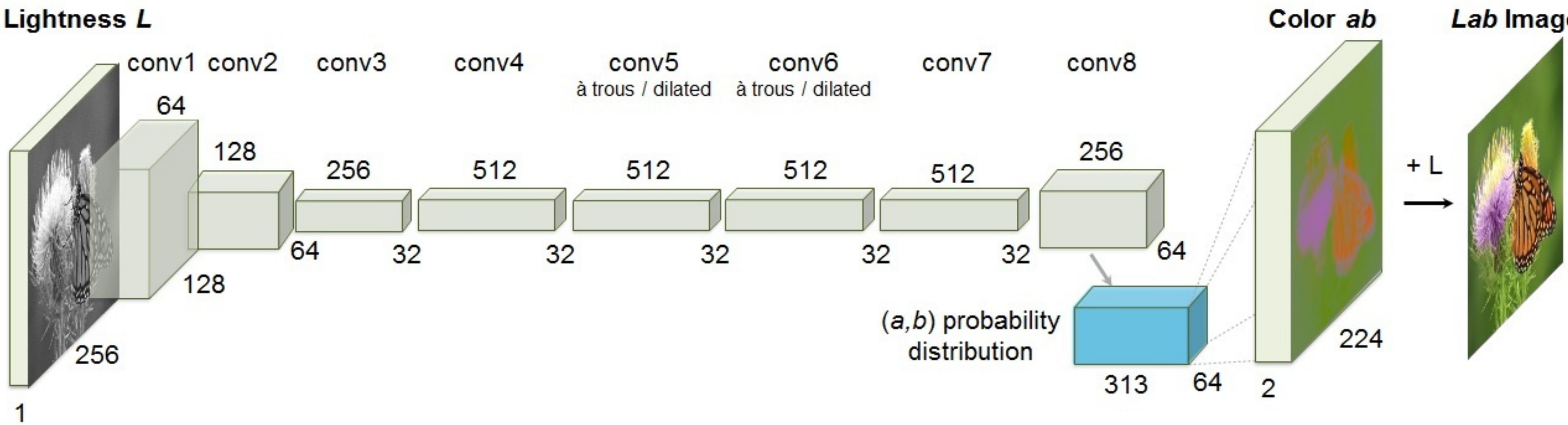
Colorization Challenges

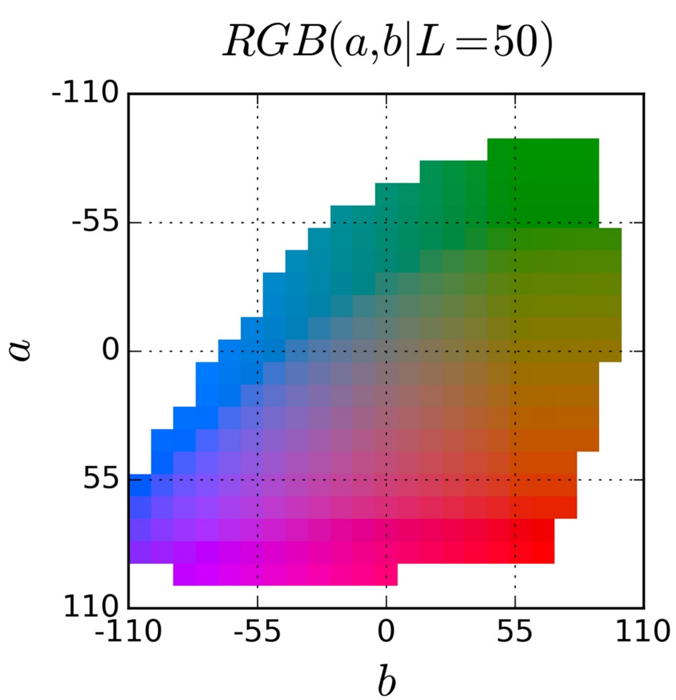
- Many colors may be possible for an object (multimodal)
- Object colors should be consistent for the whole object

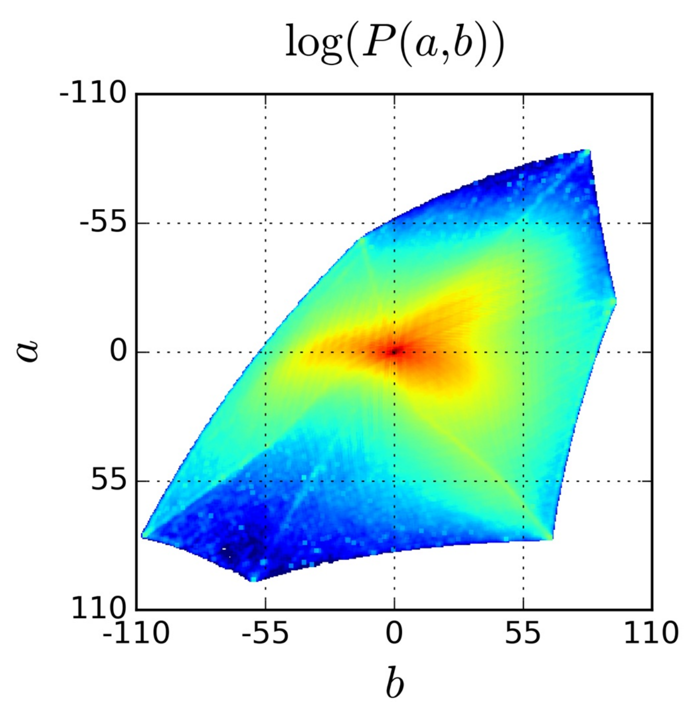

How might this affect our model?

Colorful Image Colorization

- Zhang et al. predict a distribution of color by quantizing a,b



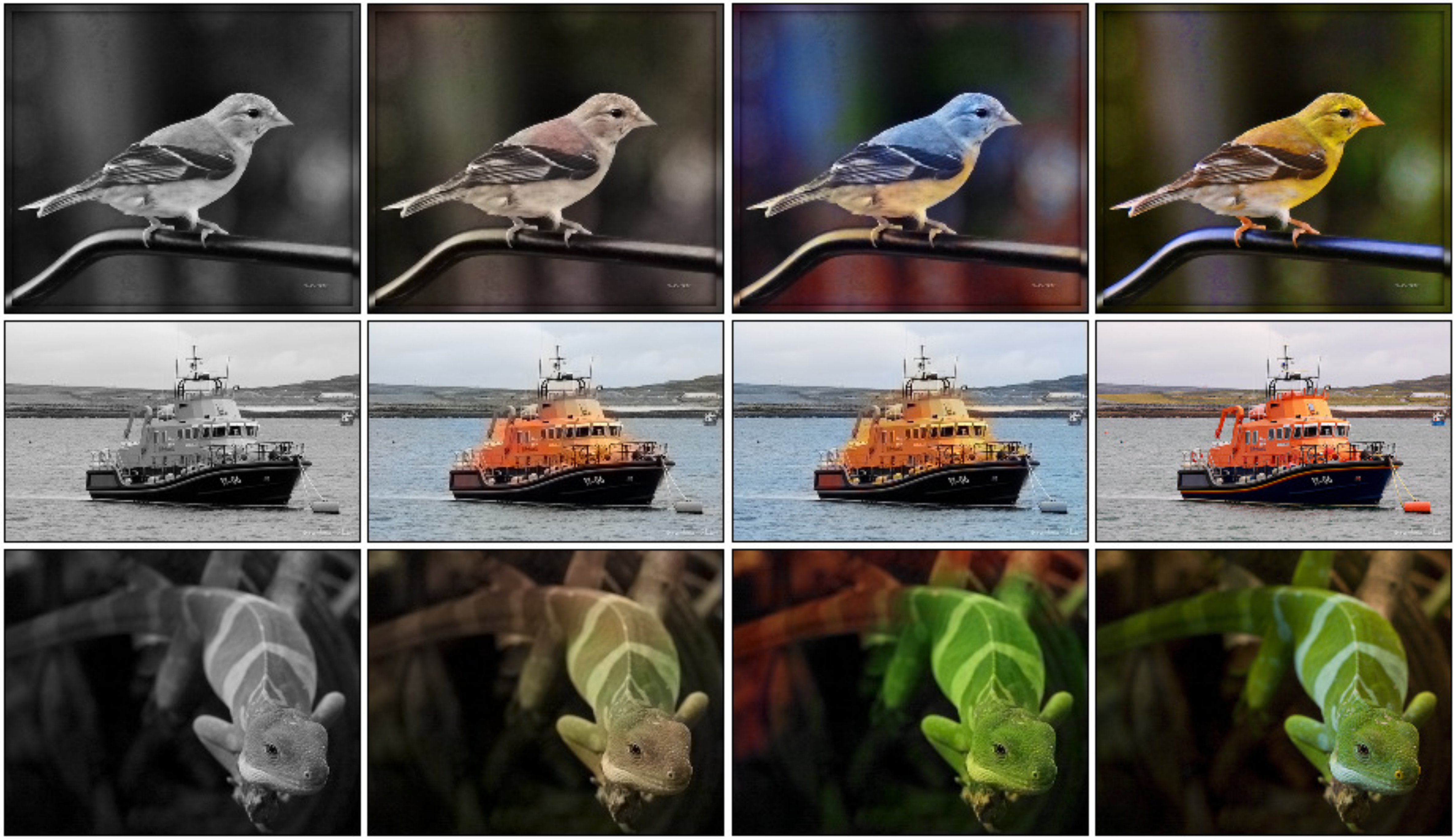




Loss is cross entropy, with an additional weighting to penalise desaturated values

[Zhang et al. 2016]

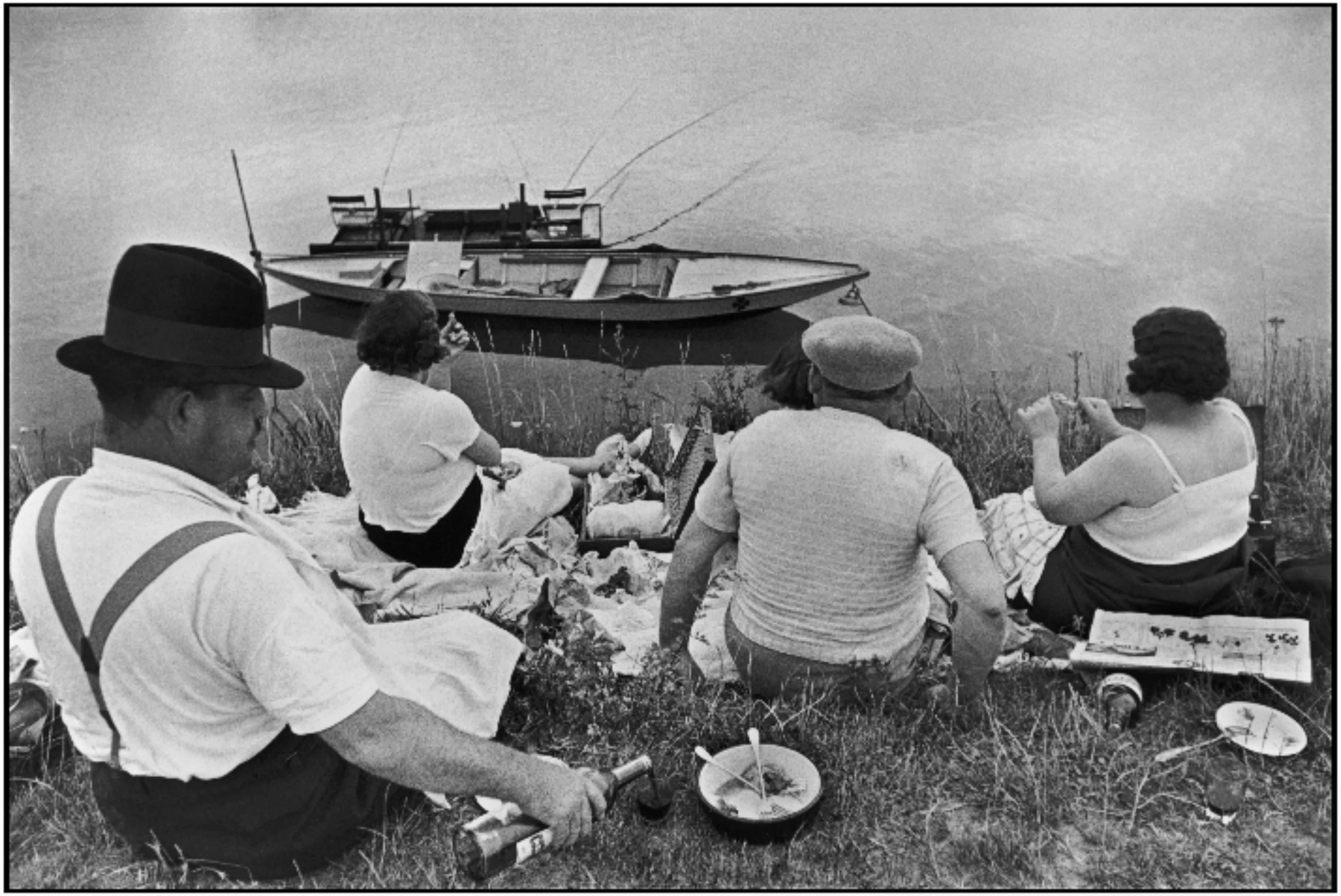
Colorful Image Colorization



Input | Regression (L2) | Zhang et al | Ground Truth

[Ansel Adams, Yosemite Valley Bridge]

[Ansel Adams, Yosemite Valley Bridge]

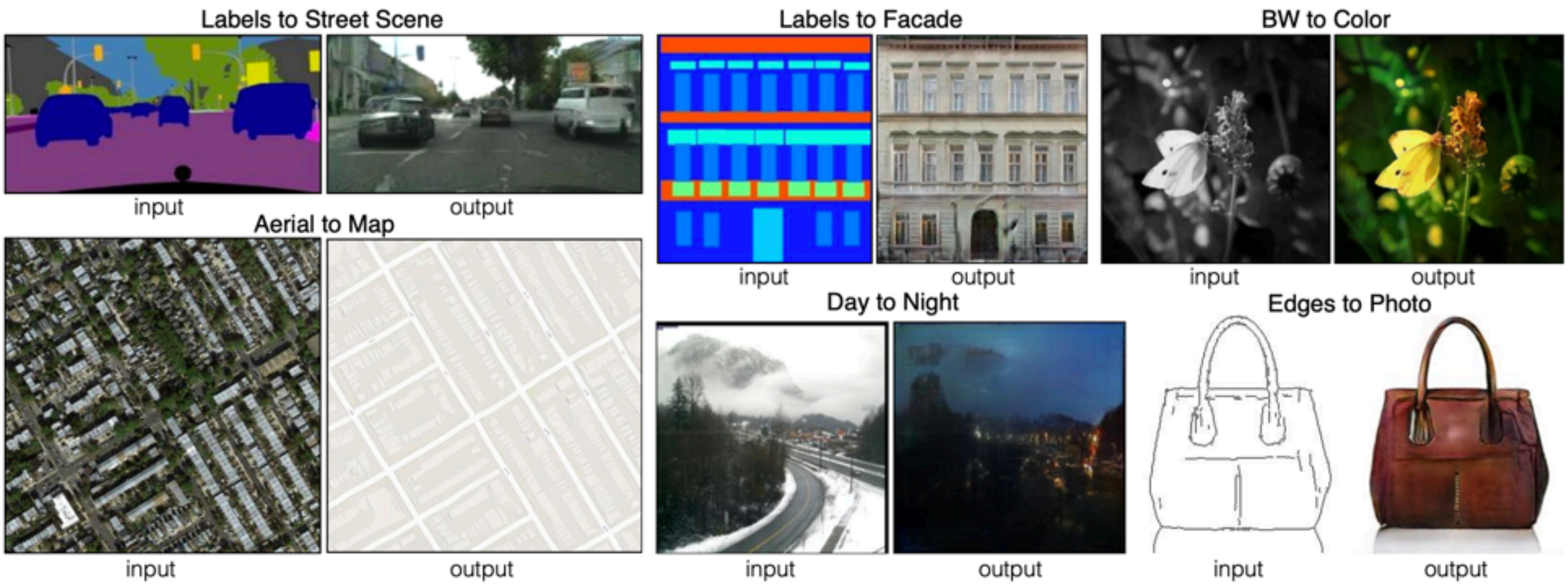


[Henri Cartier-Bresson, Sunday on the Banks of the River Seine, 1938]

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

Image Translation

- Many problems in vision/graphics can be viewed as image translation problems

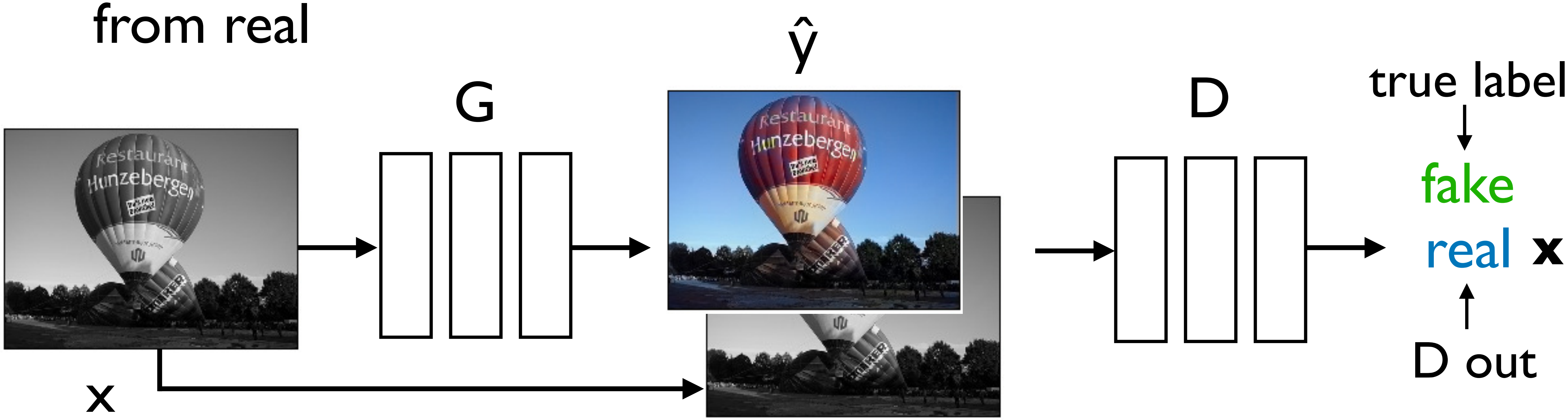


Can we build a general machine to translate images?

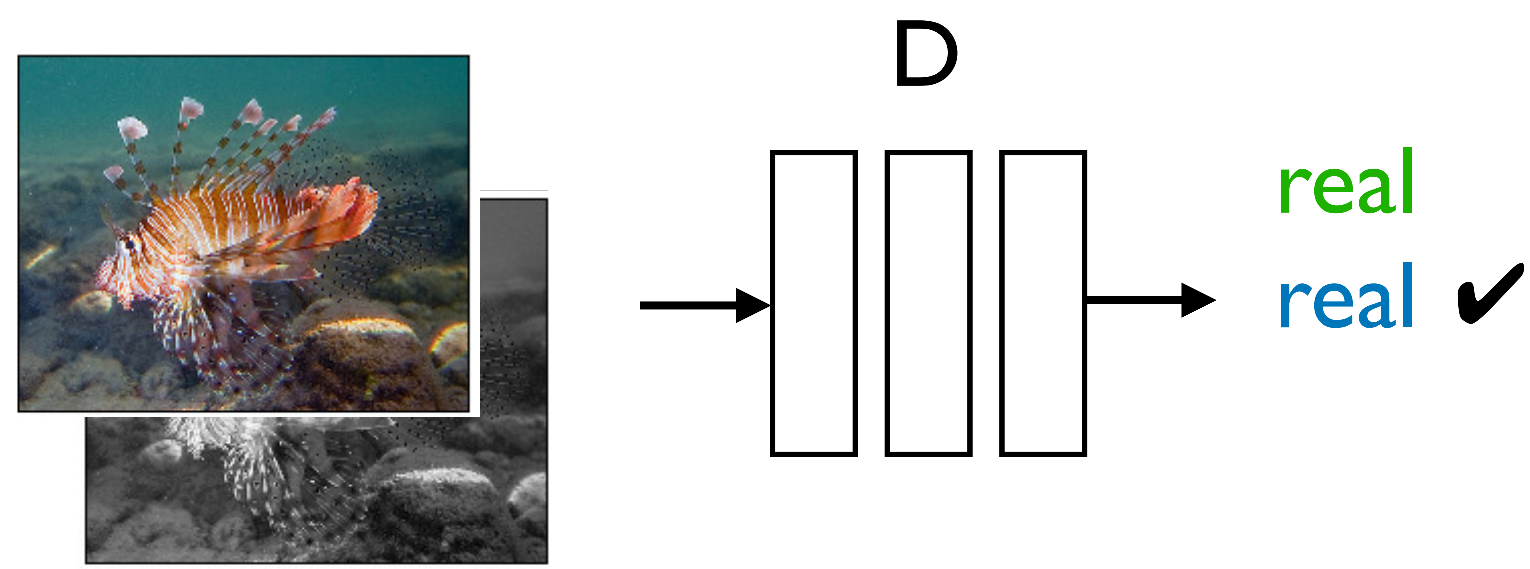
[pix2pix, Isola et al. 2018]

Image Translation

- e.g., translation from grey to color should be indistinguishable from real



Note: pix2pix has an additional supervisory L1 loss = $|y-\hat{y}|$



This is a (conditional) Generative Adversarial Network

Next Lecture

- 3D Deep Learning, Generative Adversarial Networks