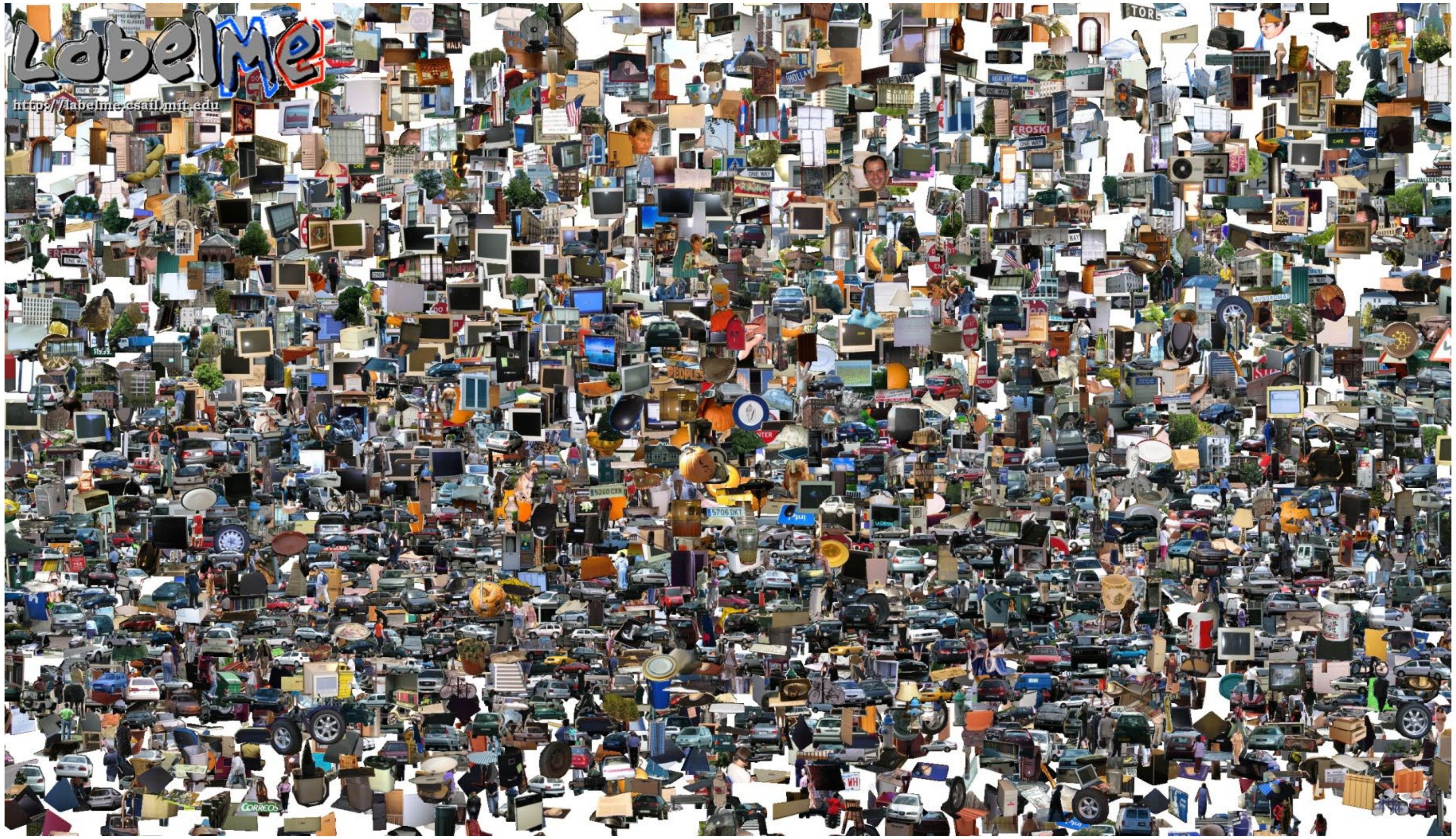
Opportunities of Scale, Encoders-Decoders



Computational Photography

Derek Hoiem, University of Illinois

Some slides from Alyosha Efros

Graphic from Antonio Torralba

Today’s class: Opportunities of scale

- Data-driven methods
  - Scene completion
  - Colorization
- Deep network representations
  - Encoder-decoder
  - CNNs and skip connections
  - U-Net

Google and massive data-driven algorithms

A.I. for the postmodern world:

- all questions have already been answered…many times, in many ways
- Google is dumb, the "intelligence" is in the data

Google Translate

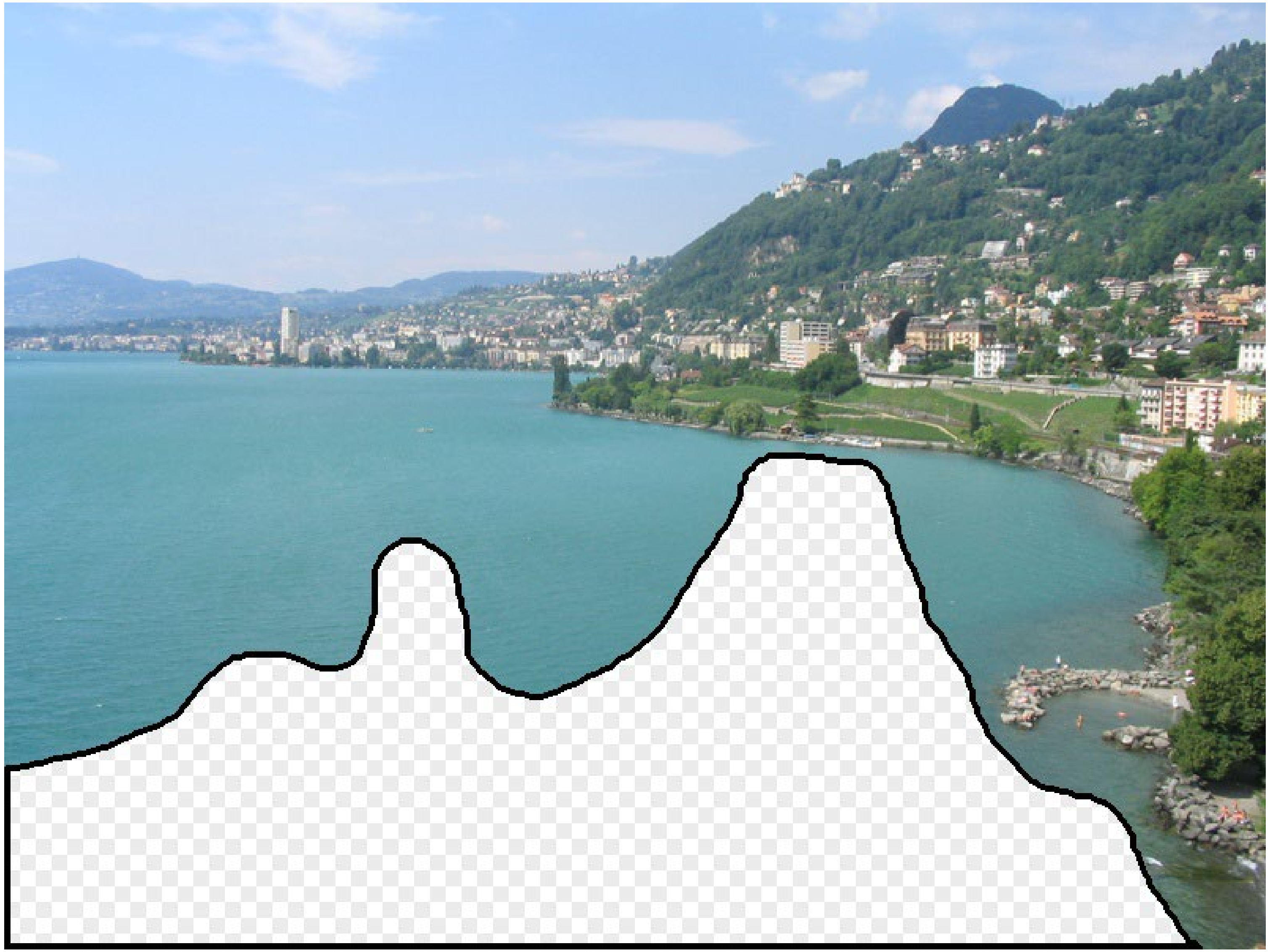
Chinese Room

- John Searle (1980)

Image Completion Example

[Hays and Efros. Scene Completion Using Millions of Photographs. SIGGRAPH 2007 and CACM October 2008.]

What should the missing region contain?



Which is the original?

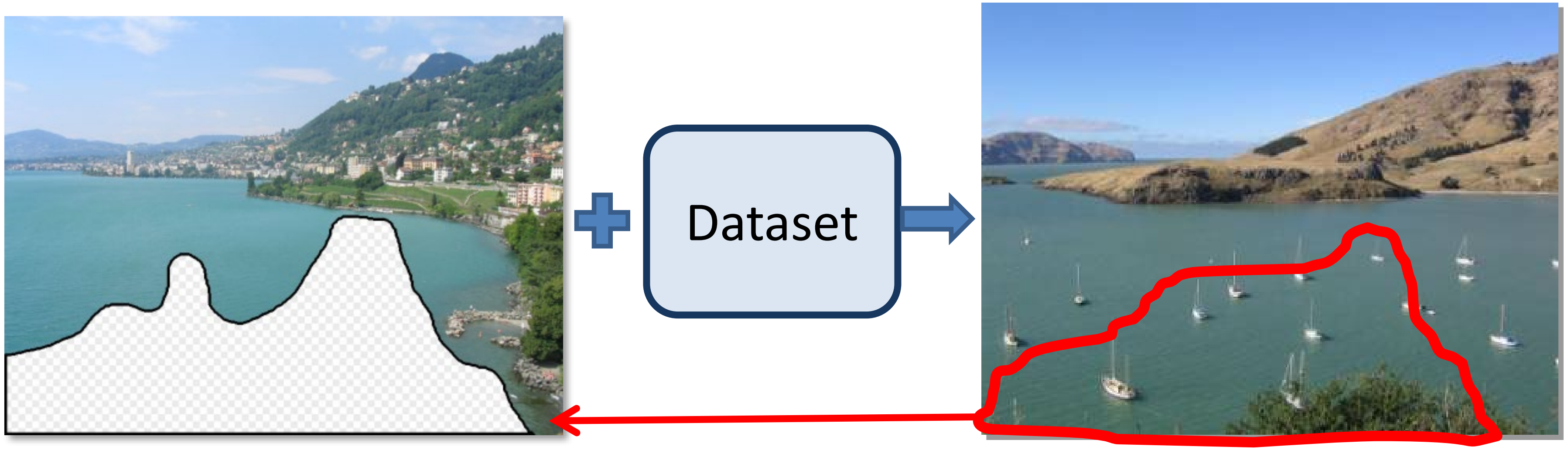
(a)

(b)

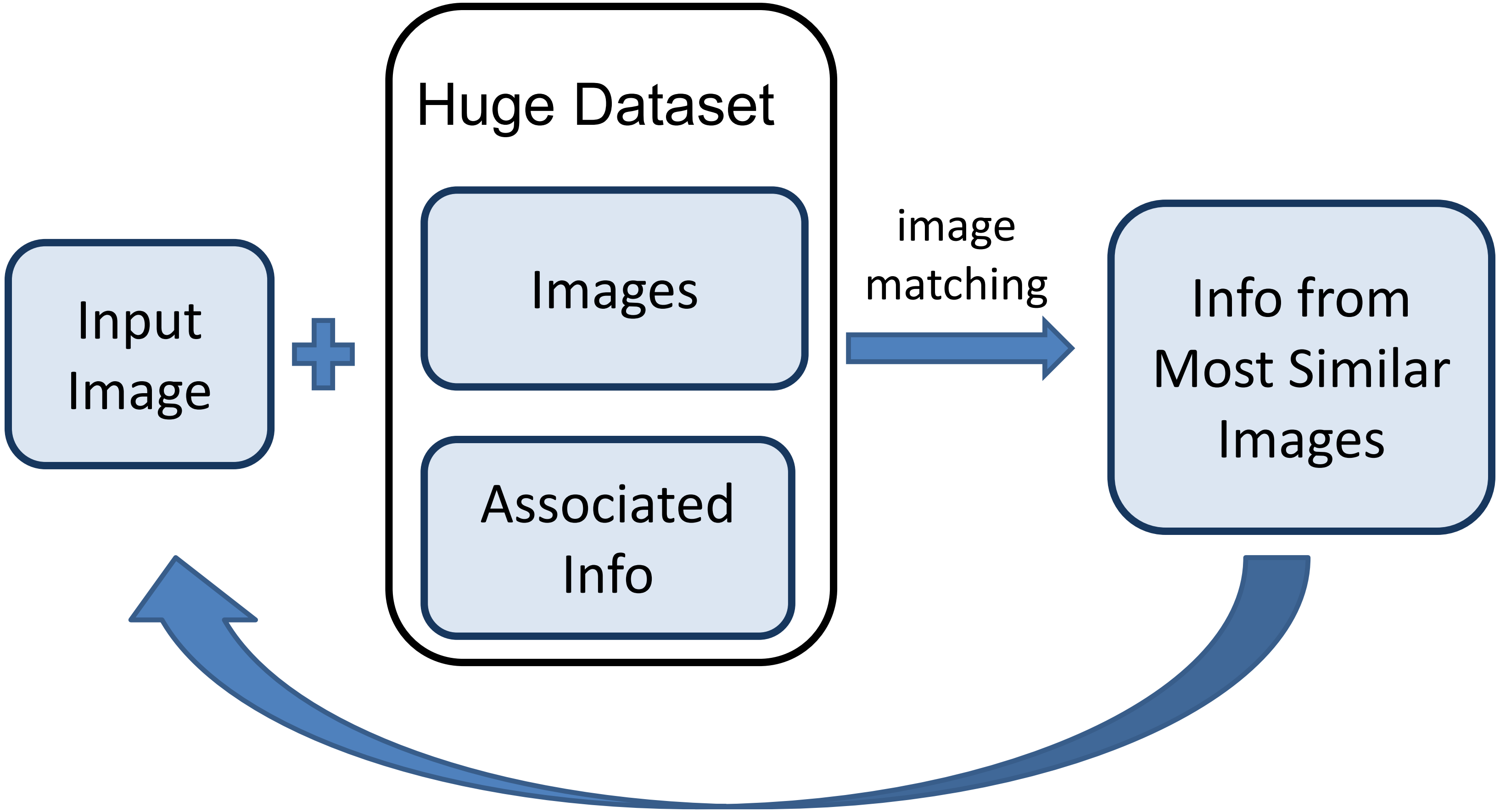
(c)

How it works

- Find a similar image from a large dataset
- Blend a region from that image into the hole

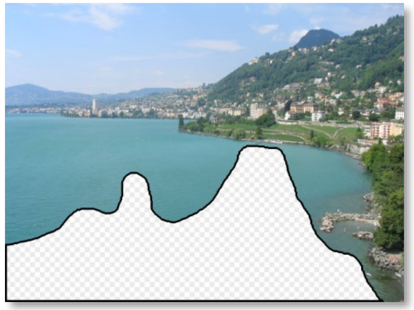


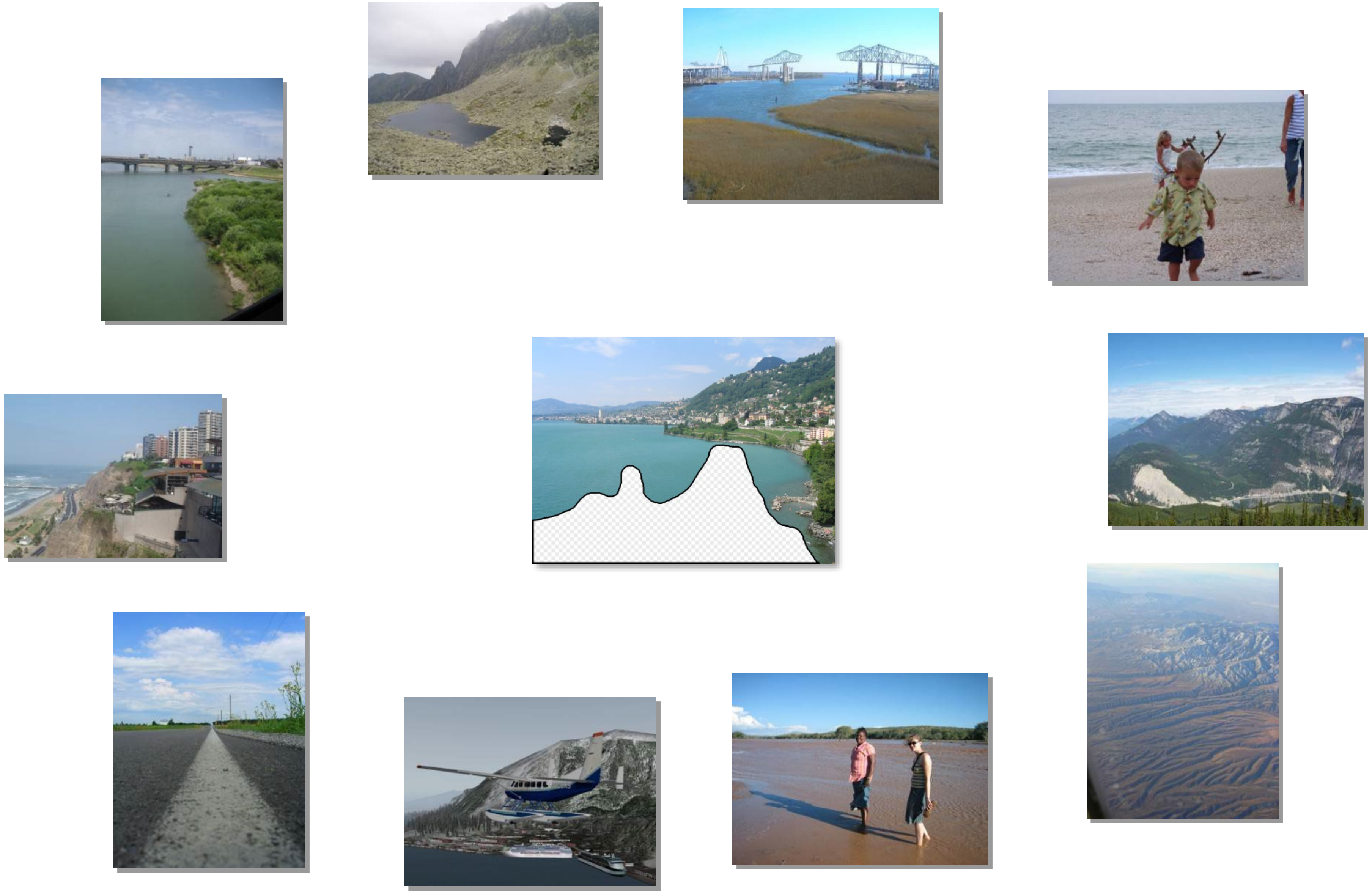
General Principal



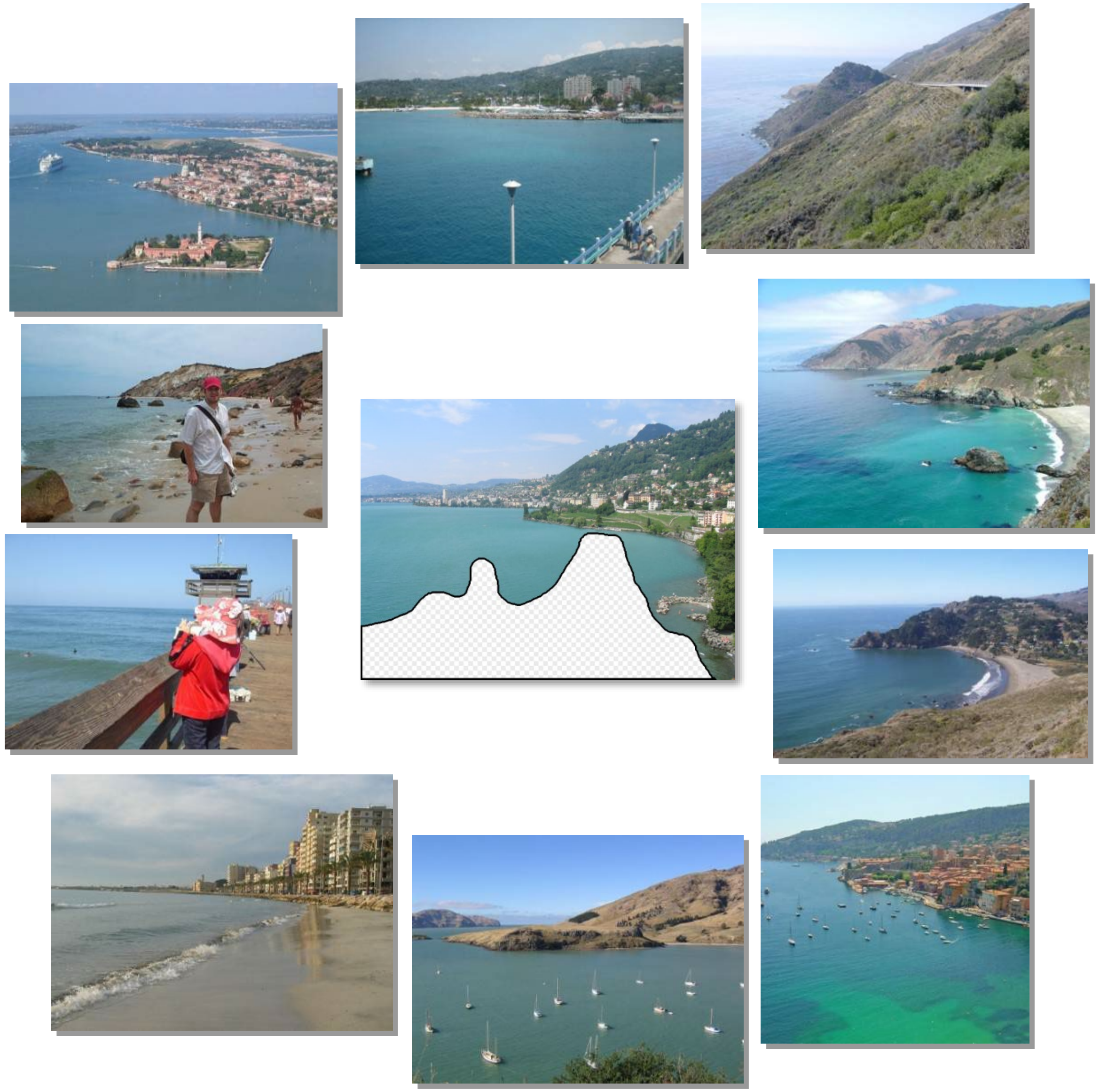
Trick: If you have enough images, the dataset will contain very similar images that you can find with simple matching methods.

How many images is enough?





Nearest neighbors from a
collection of 20 thousand images



Nearest neighbors from a collection of 2 million images

Image Data on the Internet

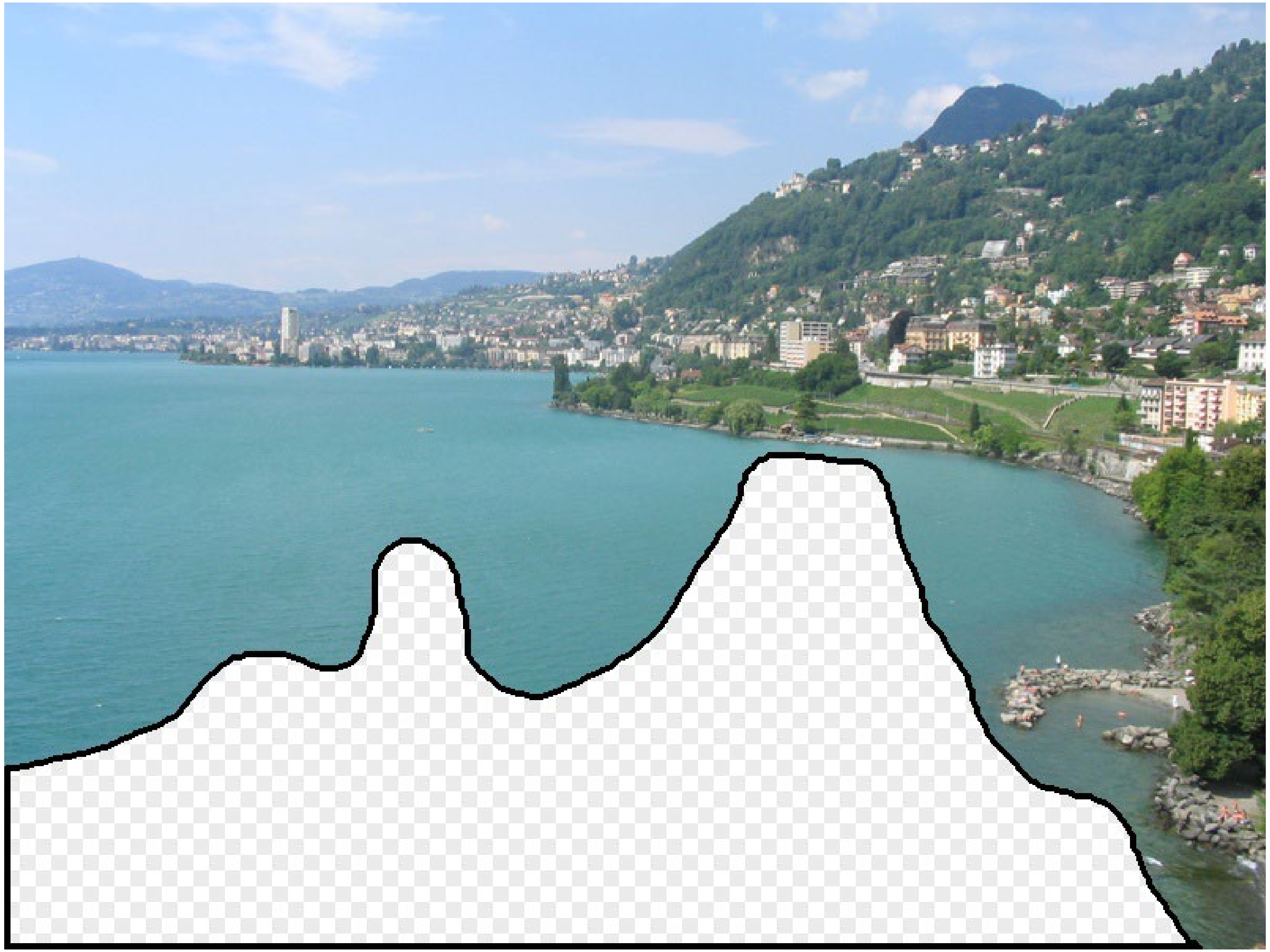
- Now: nobody counts anymore
- Facebook (2014)
  - 250 billion total, +350 million per day
- Facebook (2011)
  - 6 billion images per month
  - More than 100 petabytes of images/video
- Flickr (2010)
  - 5 billion photographs
  - 100+ million geotagged images
- Imageshack (as of 2009)
  - 20 billion
- Facebook (as of 2009)
  - 15 billion

Image completion: how it works

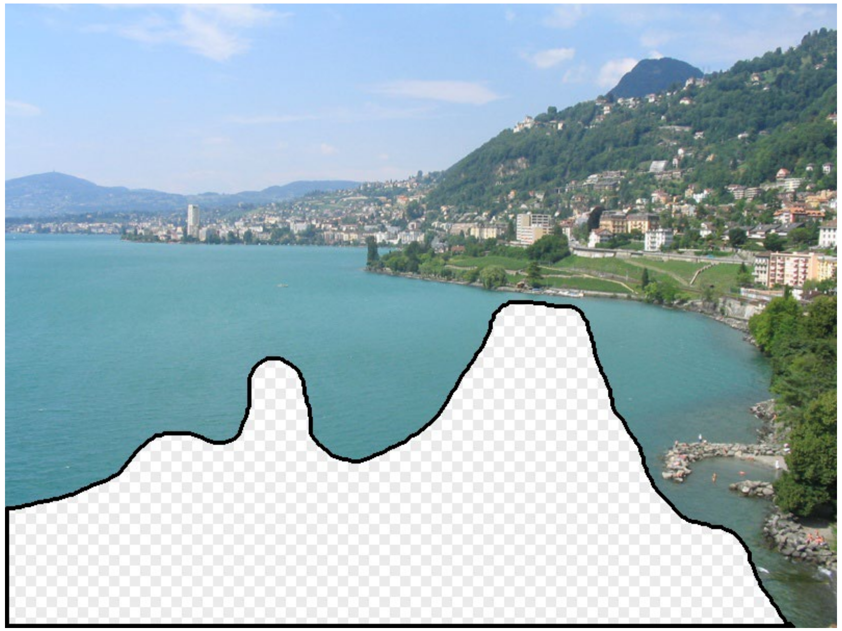
[Hays and Efros. Scene Completion Using Millions of Photographs. SIGGRAPH 2007 and CACM October 2008.]

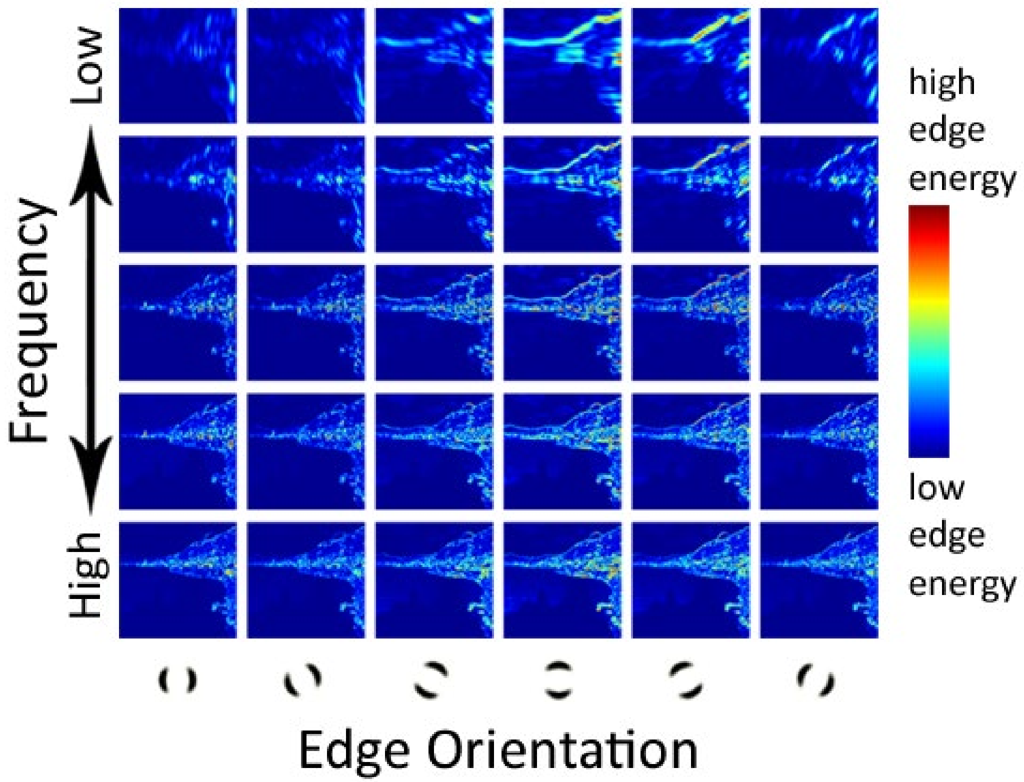
The Algorithm

Scene Matching

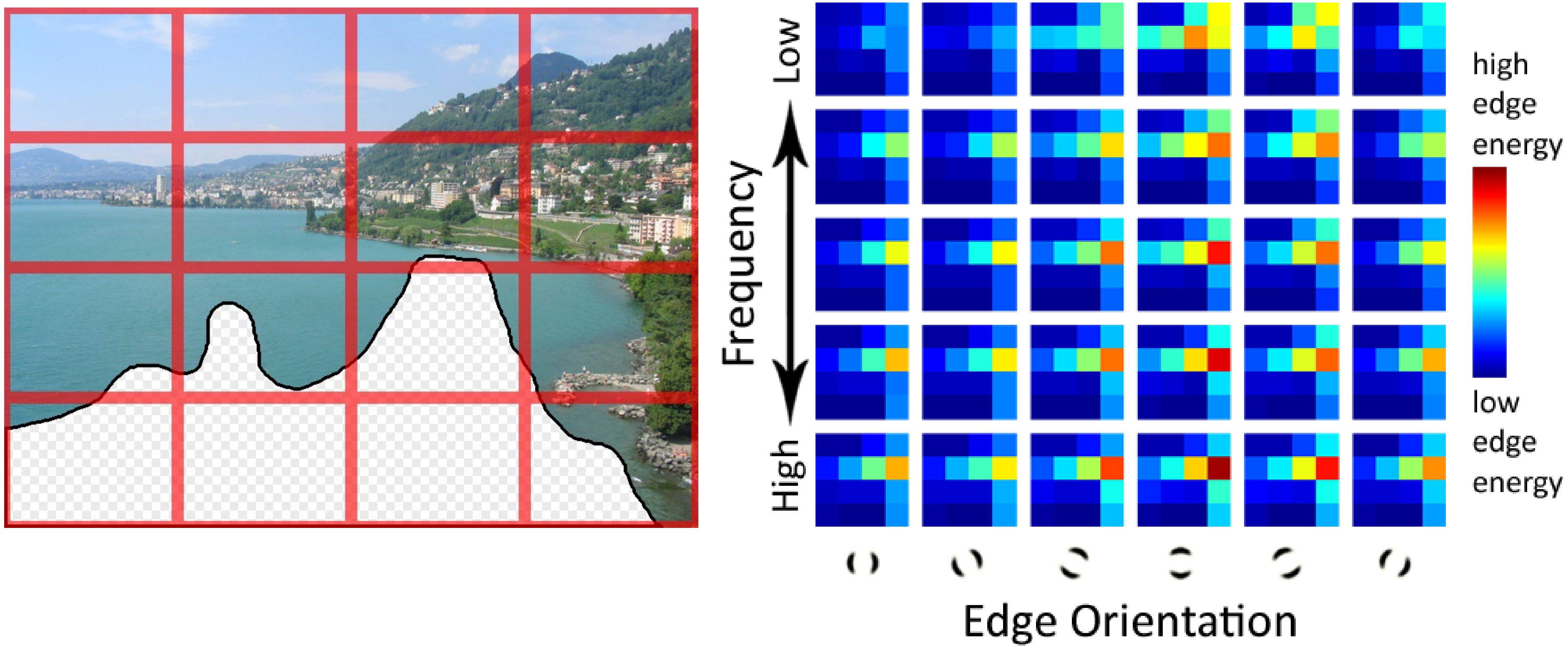


Scene Descriptor



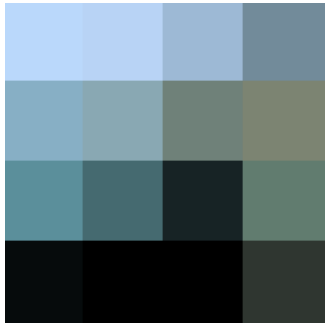


Scene Descriptor

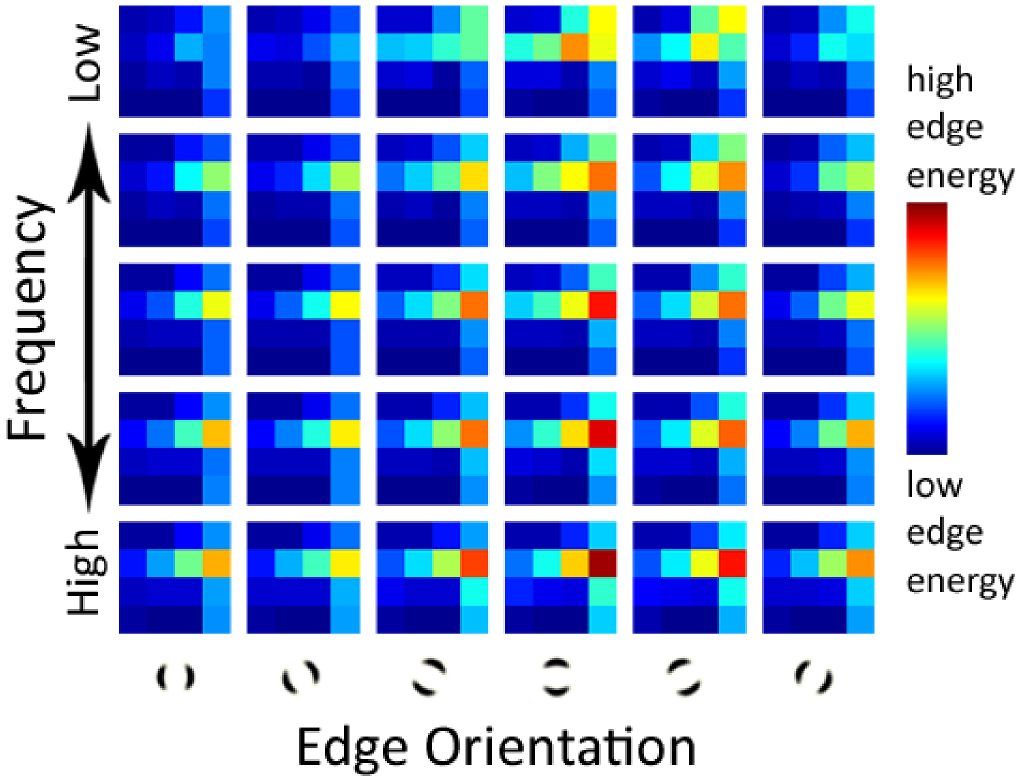


Scene Gist Descriptor
(Oliva and Torralba 2001)

Scene Descriptor

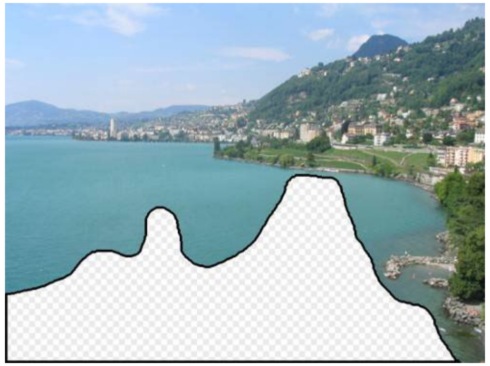


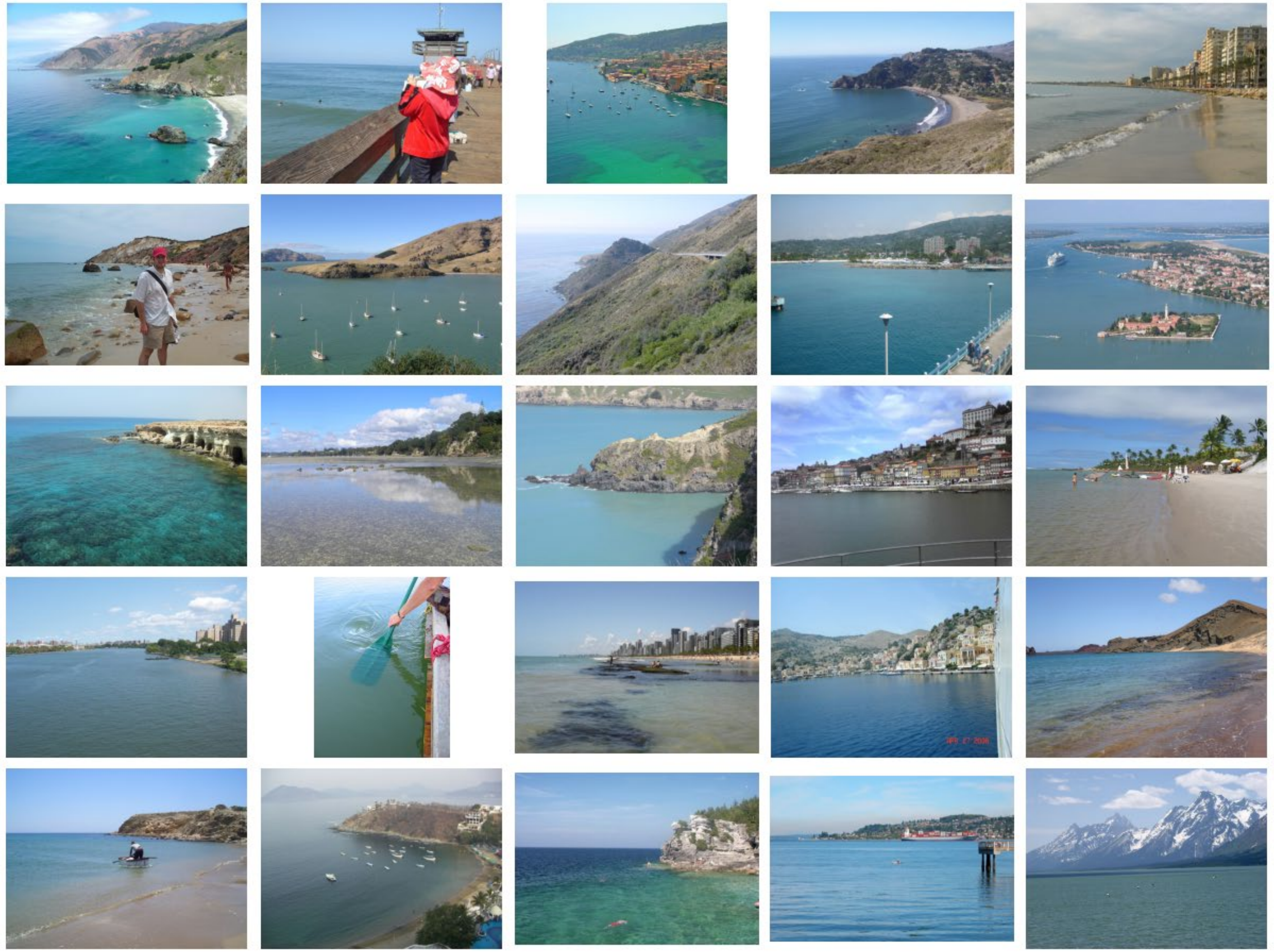
+



Scene Gist Descriptor
(Oliva and Torralba 2001)

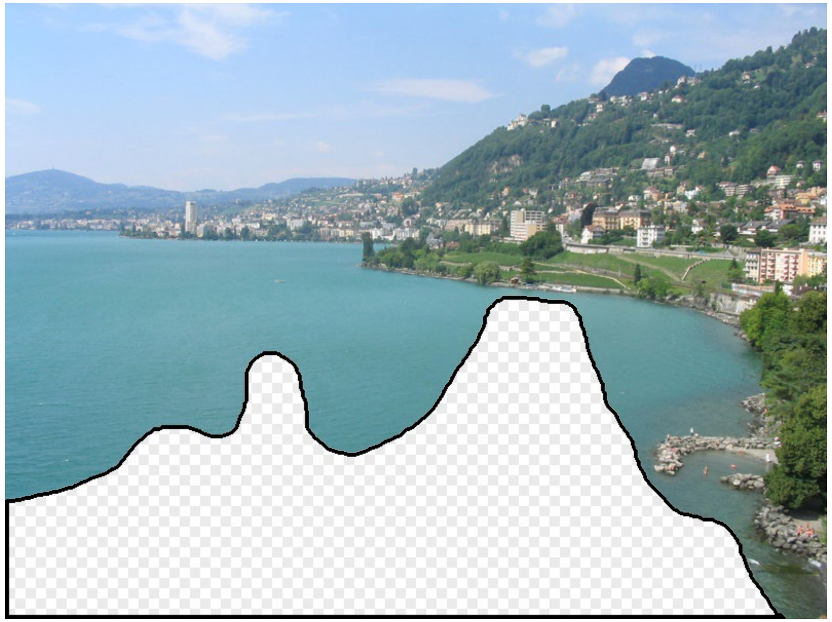
2 Million Flickr Images





… 200 total

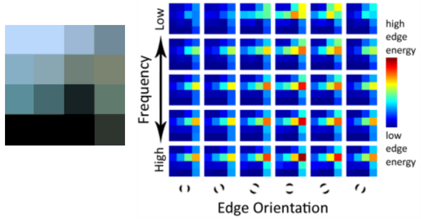
Context Matching



Graph cut + Poisson blending

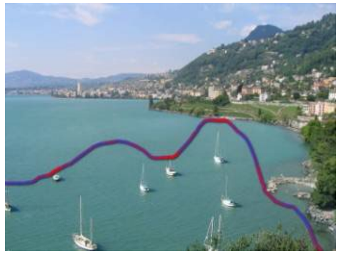
Result Ranking

We assign each of the 200 results a score which is the sum of:

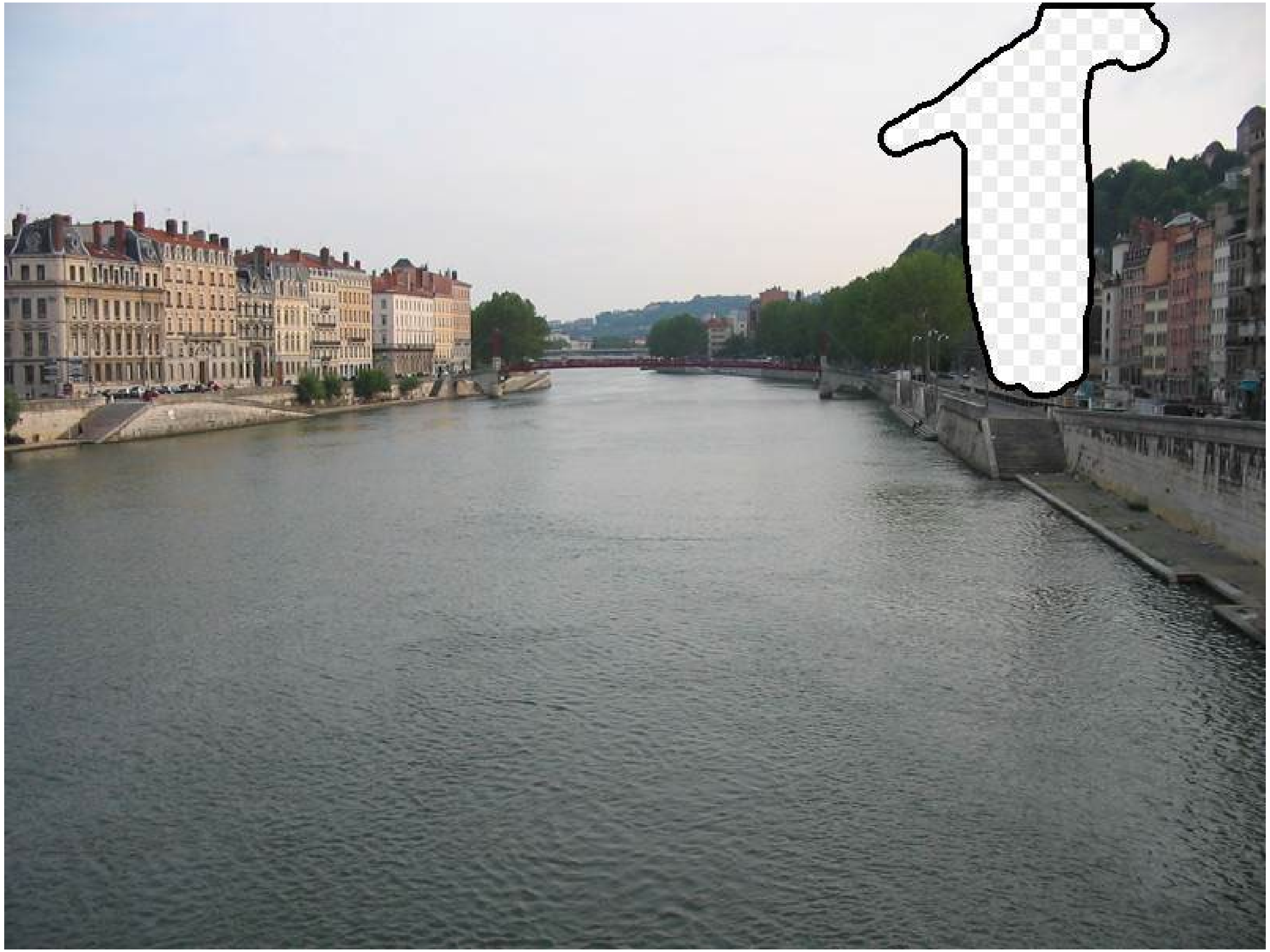


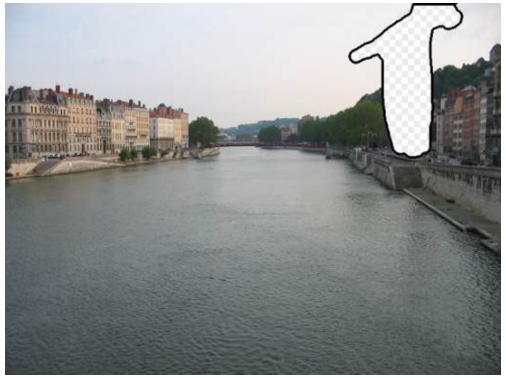
The scene matching distance

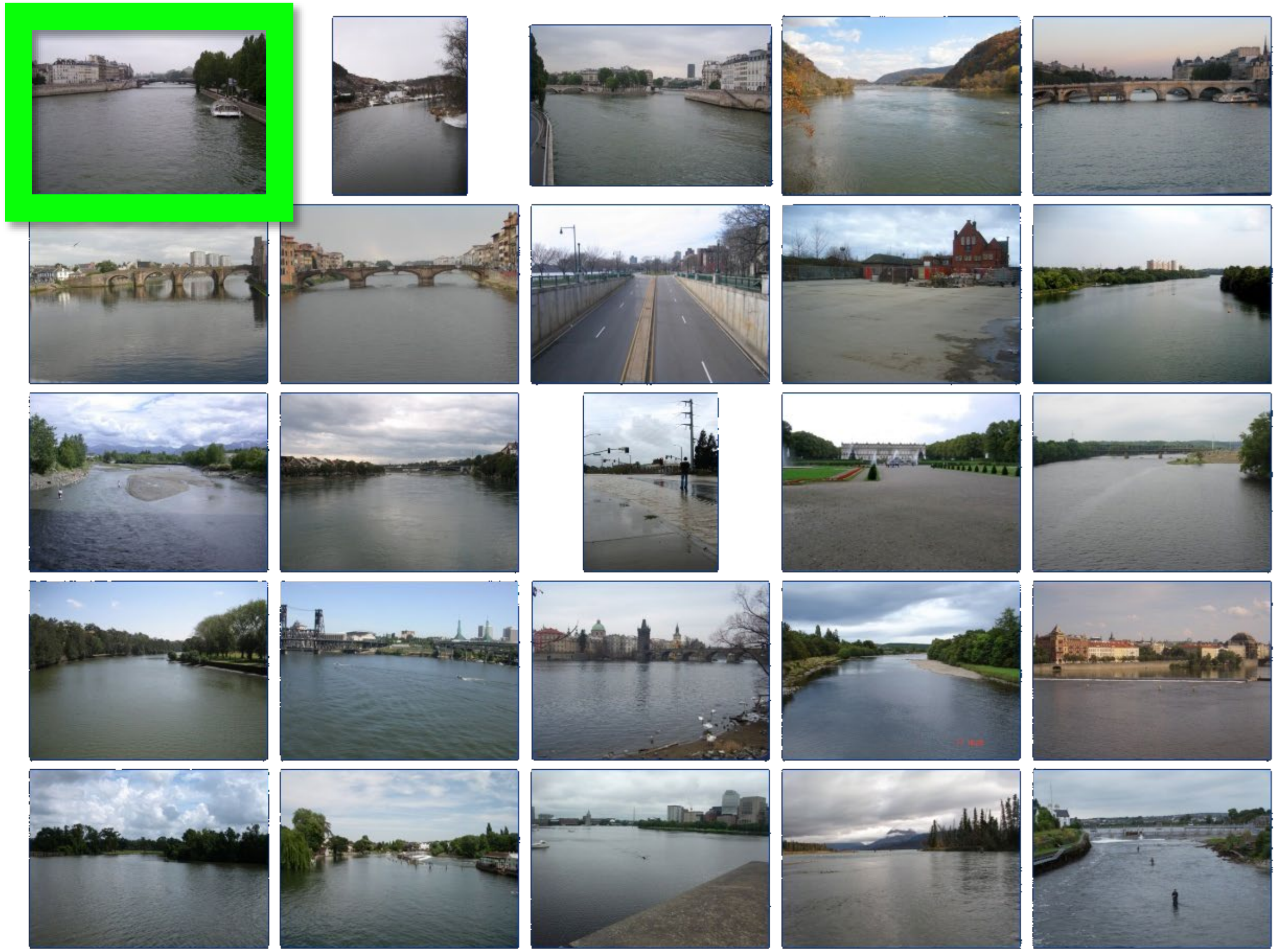
The context matching distance (color + texture)



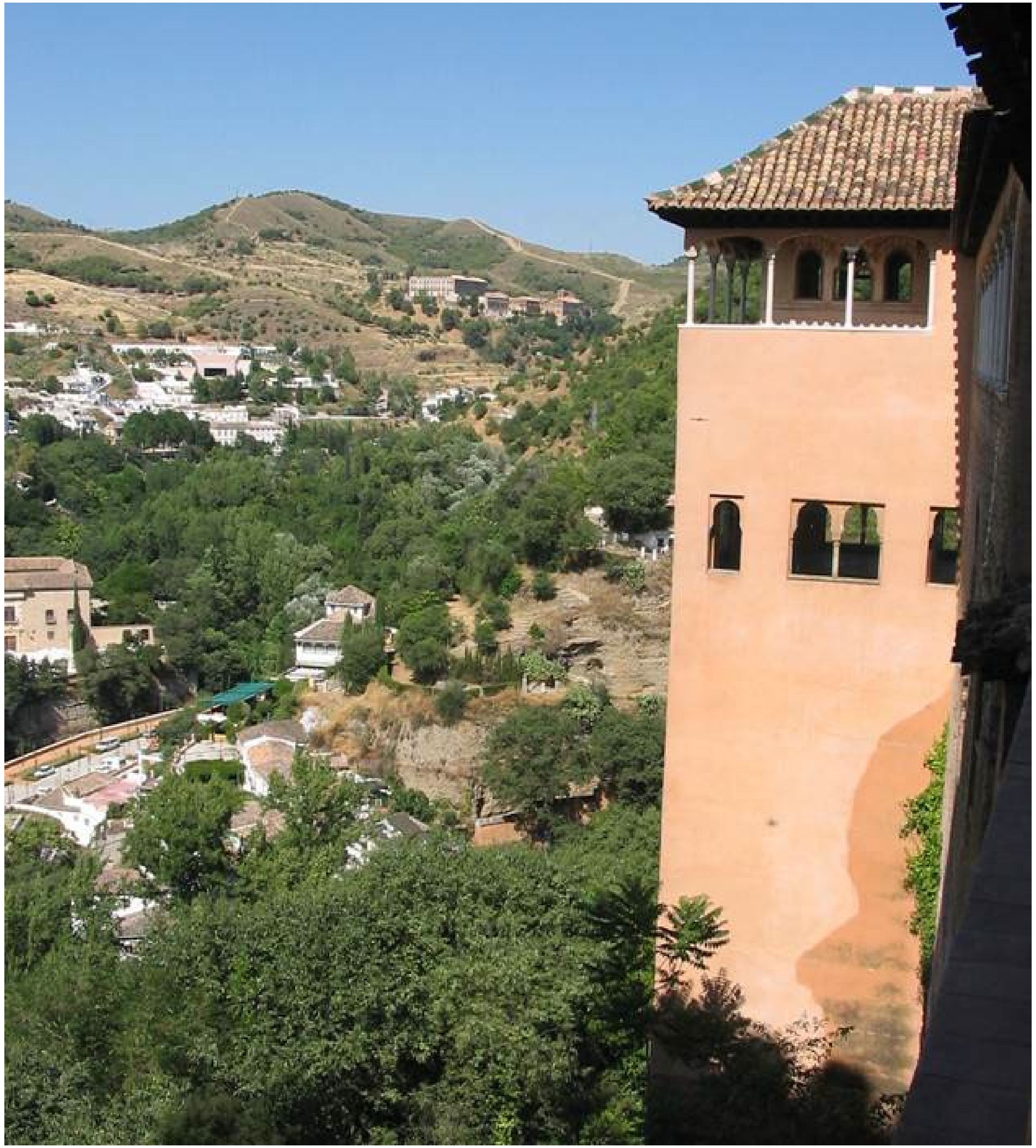
The graph cut cost

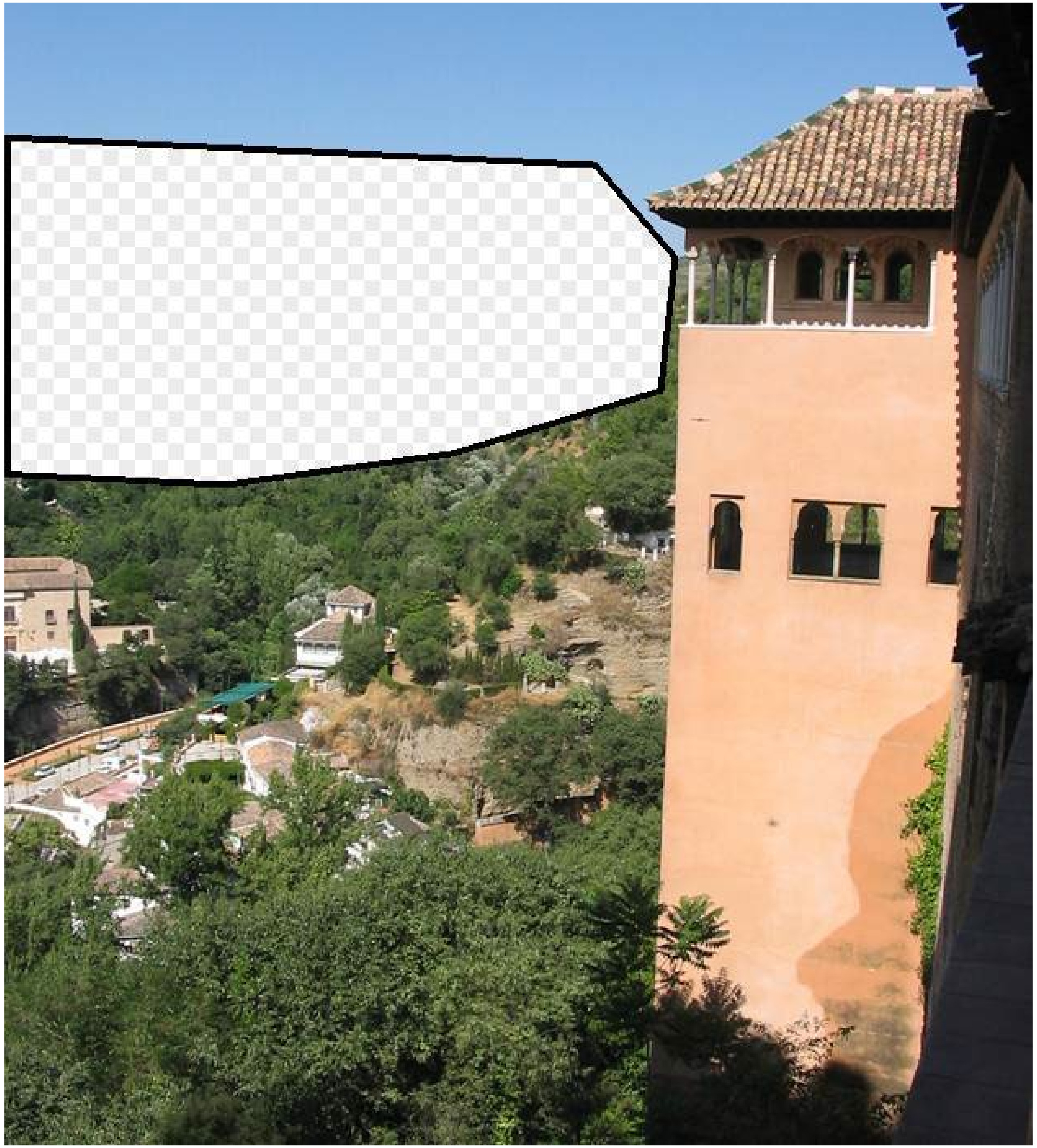


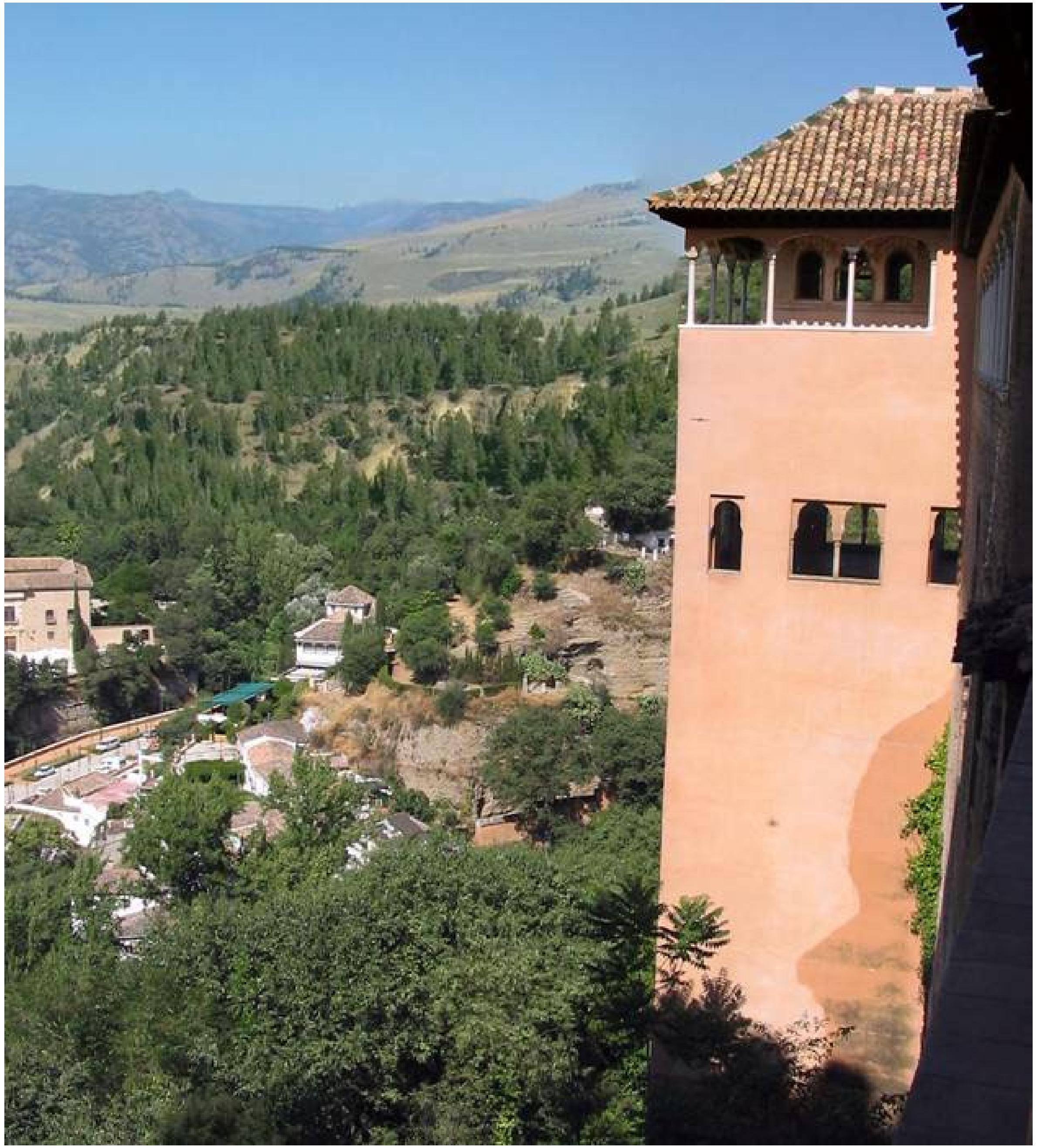




… 200 scene matches





Which is the original?

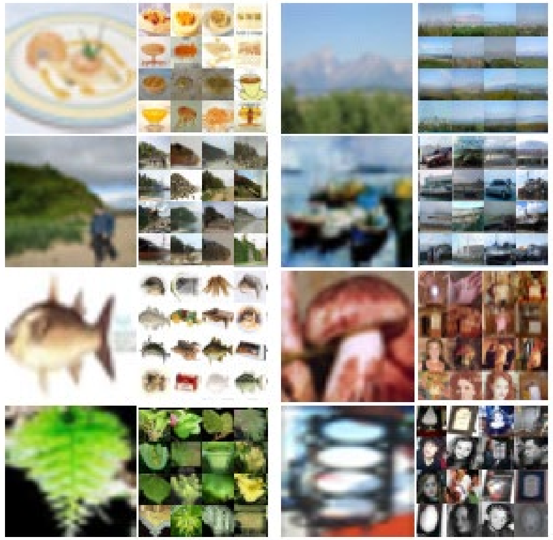

5407 BWL

Diffusion Result

Efros and Leung result

Scene Completion Result

Tiny Images



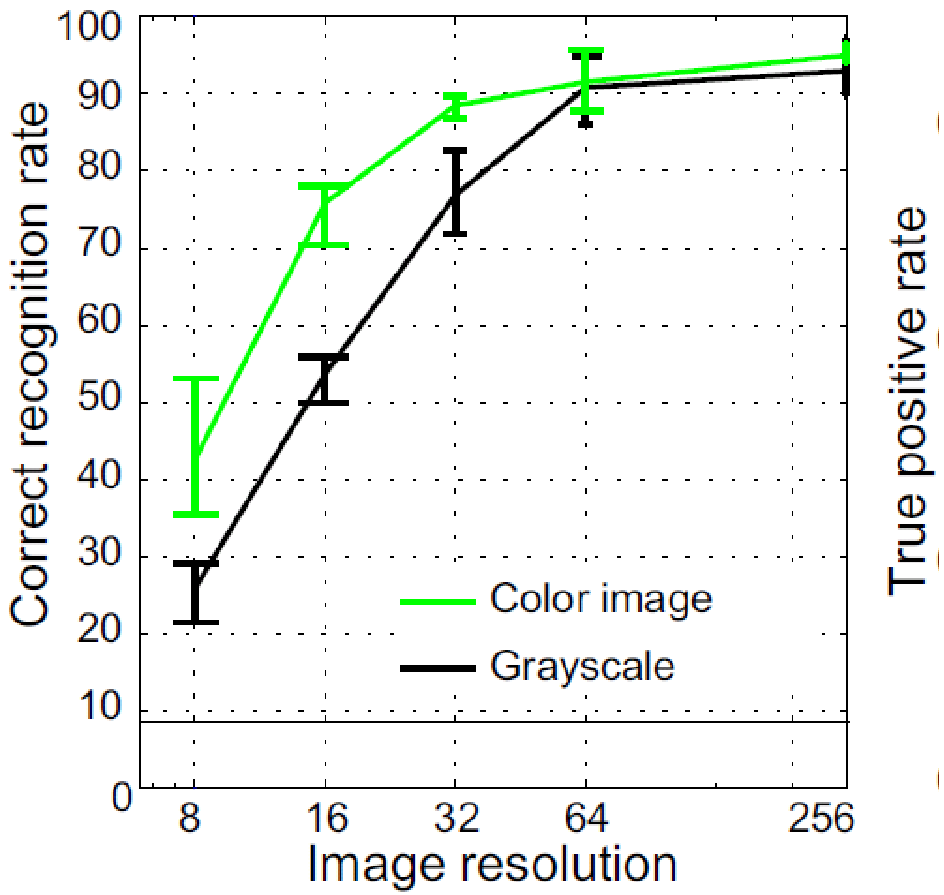
80 million tiny images: a large dataset for non-parametric object and scene recognition
Antonio Torralba, Rob Fergus and William T. Freeman. PAMI 2008.

http://groups.csail.mit.edu/vision/TinyImages/

Even 32x32 images contain a lot of information

c) Segmentation of 32x32 images

Human Scene Recognition



a) Scene recognition

Powers of 10

Number of images on my hard drive: 10^{4}

Number of images seen during my first 10 years: 10^{8}

(3 images/second * 60 * 60 * 16 * 365 * 10 = 630720000)

Number of images seen by all humanity: 10^{20}

106,456,367,669 humans[1] * 60 years * 3 images/second * 60 * 60 * 16 * 365 =

1 from http://www.prb.org/Articles/2002/HowManyPeopleHaveEverLivedonEarth.aspx

Number of photons in the universe: 10^{88}

Number of all 32x32 images: 10^{7373}

$256^{32*32*3} \sim 10^{7373}$

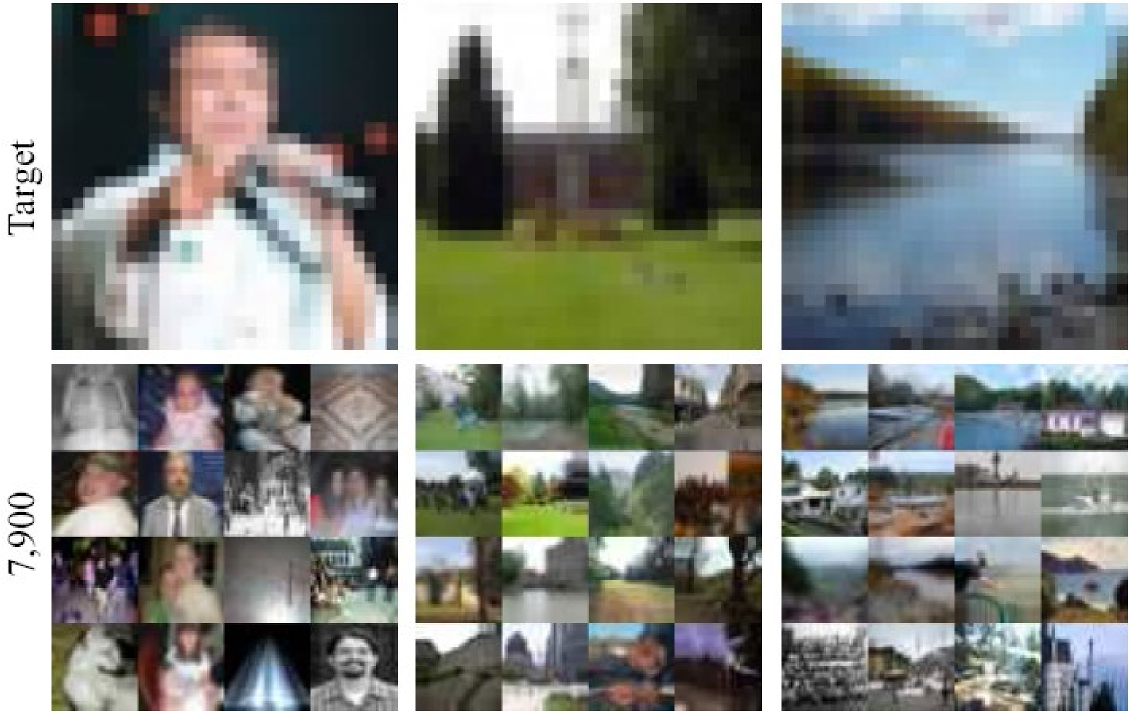
Scenes are unique

But most have other similar scenes

Lots

Of

Images

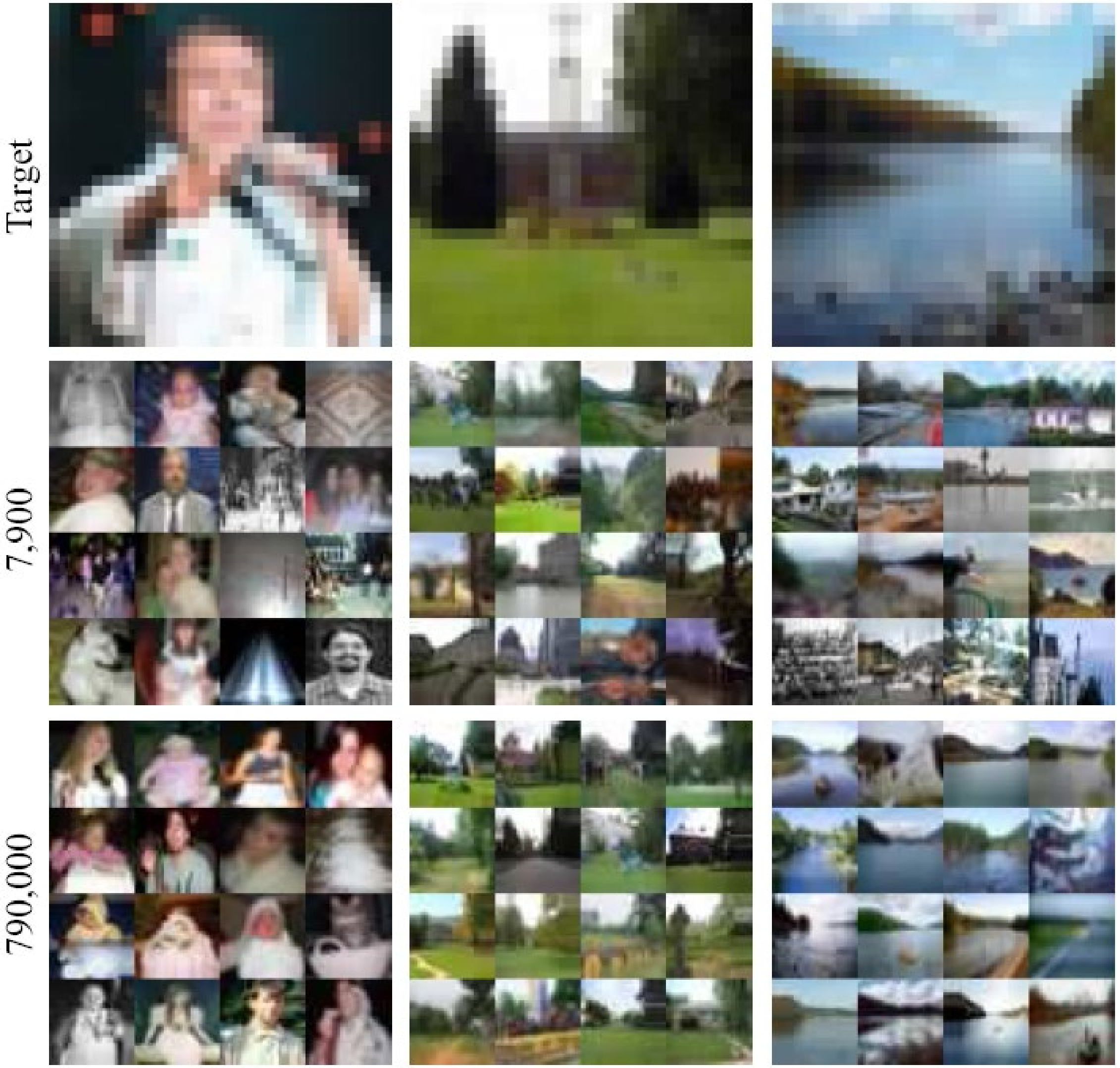


A. Torralba, R. Fergus, W.T.Freeman. PAMI 2008

Lots

Of

Images

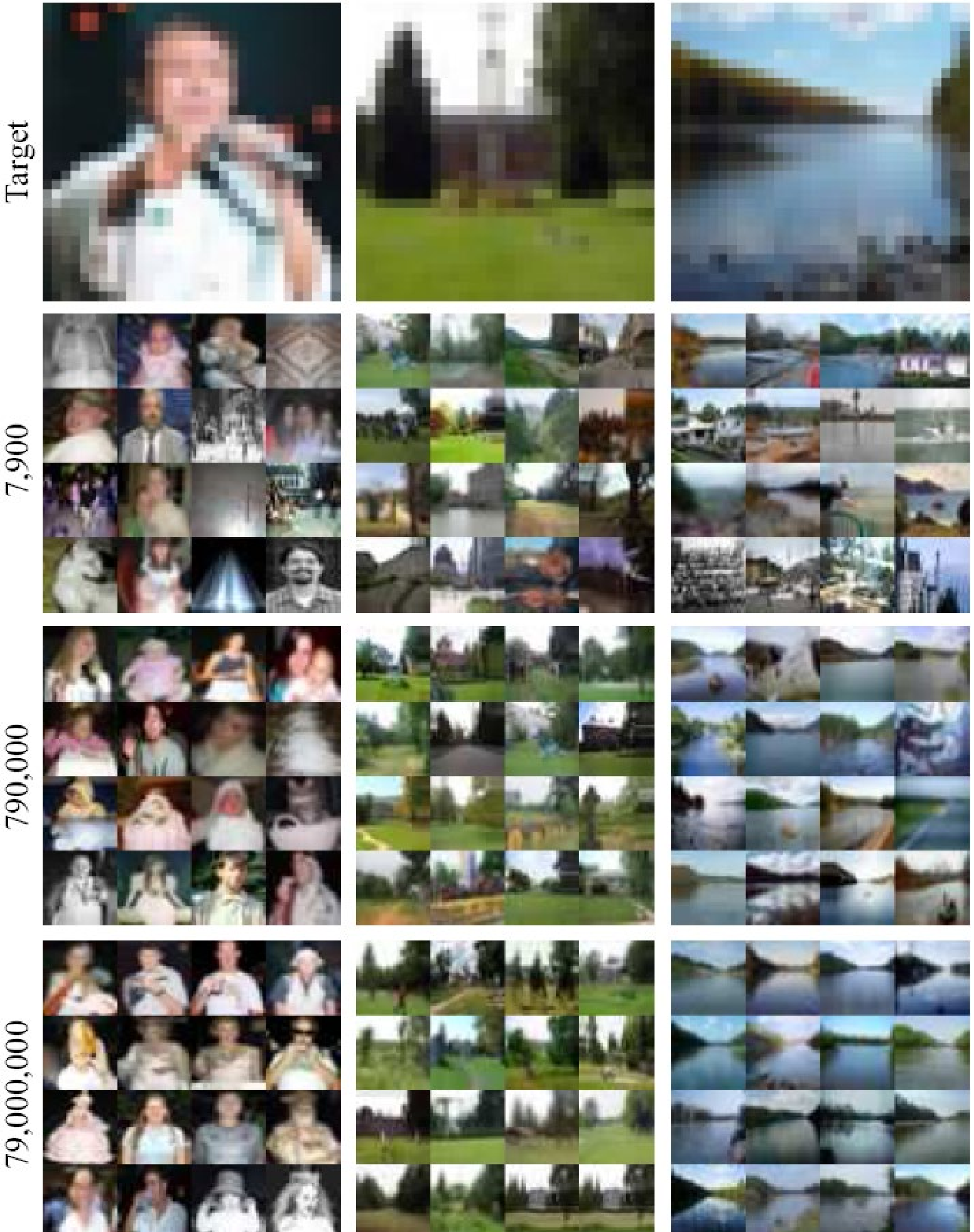


A. Torralba, R. Fergus, W.T.Freeman. PAMI 2008

Lots

Of

Images



Automatic Colorization

Input

Color Transfer

Color Transfer

Matches (gray)

Matches (w/ color)

Avg Color of Match

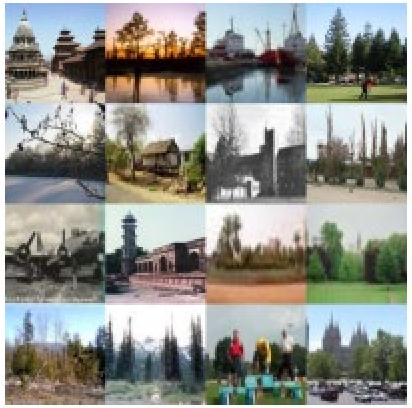
Automatic Colorization

Input

Color Transfer

Color Transfer

Matches (gray)



Matches (w/ color)

Avg Color of Match

Encoder – Decoder view

Encoder

Decoder

Input

Output

Encoder: maps input to a new representation

Images with similar encodings should have similar outputs

Decoder: maps encoded representation to output

Encoder – Decoder: simple example

Input

Encoder

Decoder

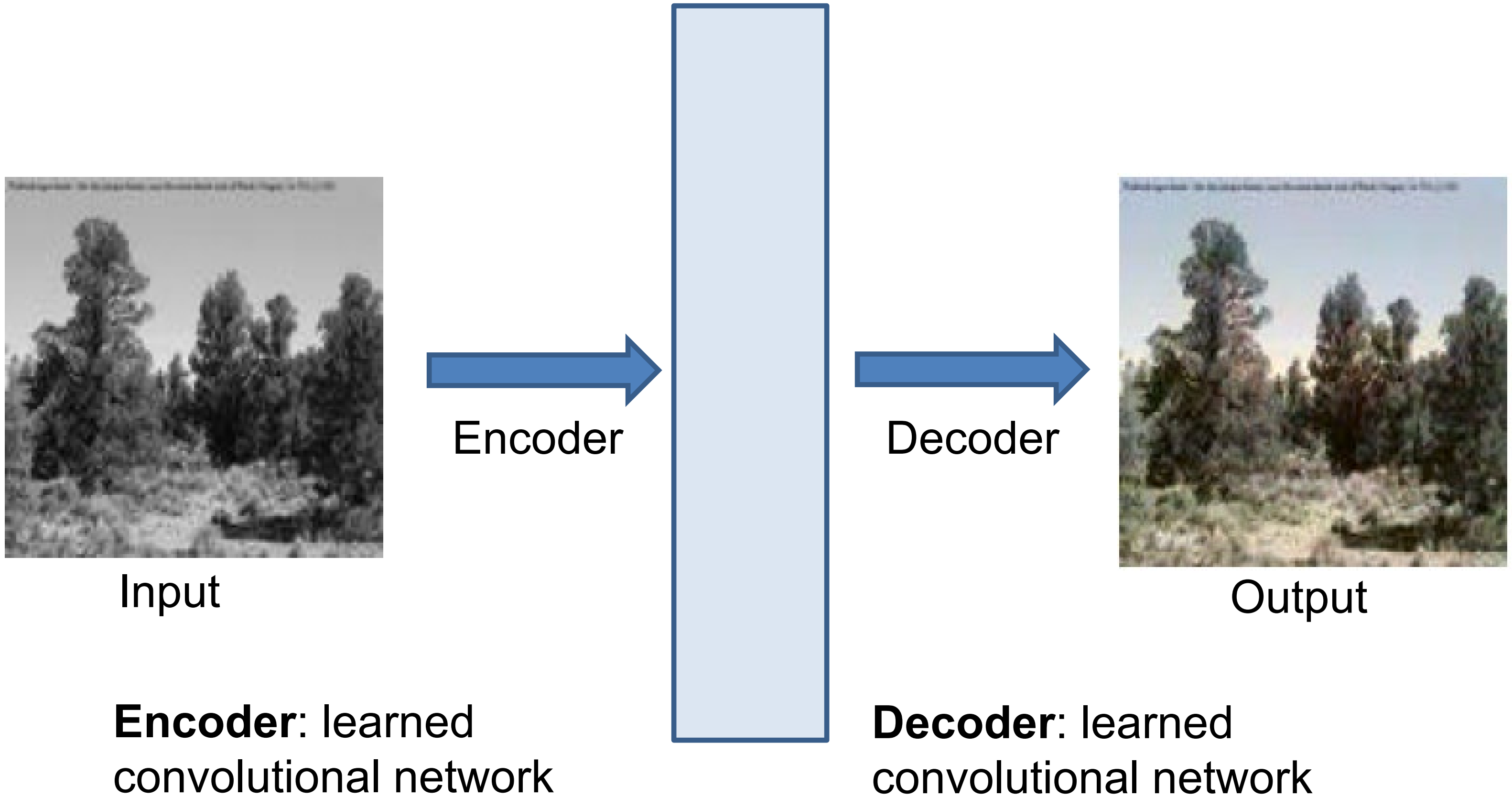
Output

Encoder: shrink image to 32x32 → 1024x1 intensity vector

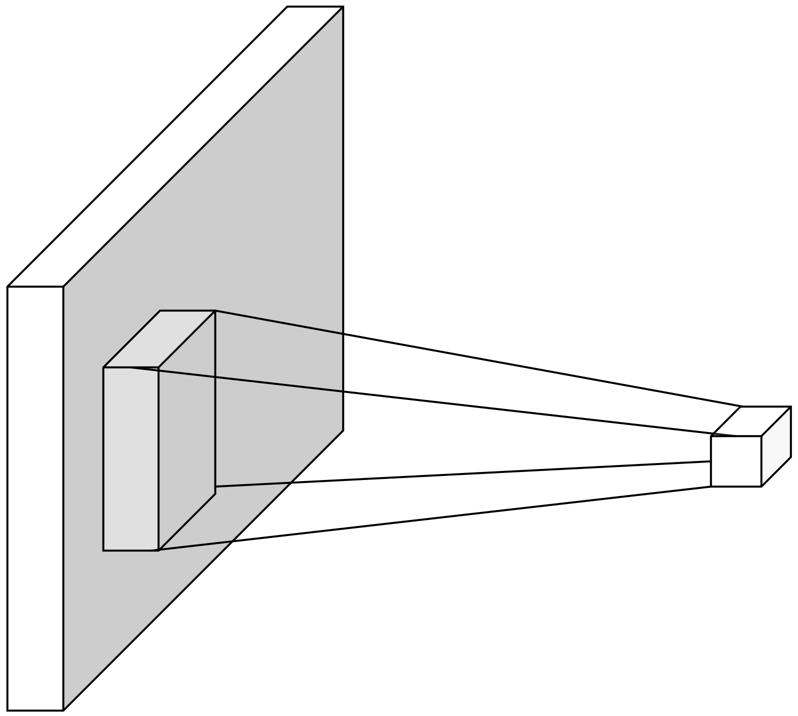
Decoder: retrieve nearest neighbor in database and copy color channels

Encoder – Decoder: deep network

Learn parameters of convolutional networks so that encoding / decoding satisfies some training objective for training samples



Convolutional network

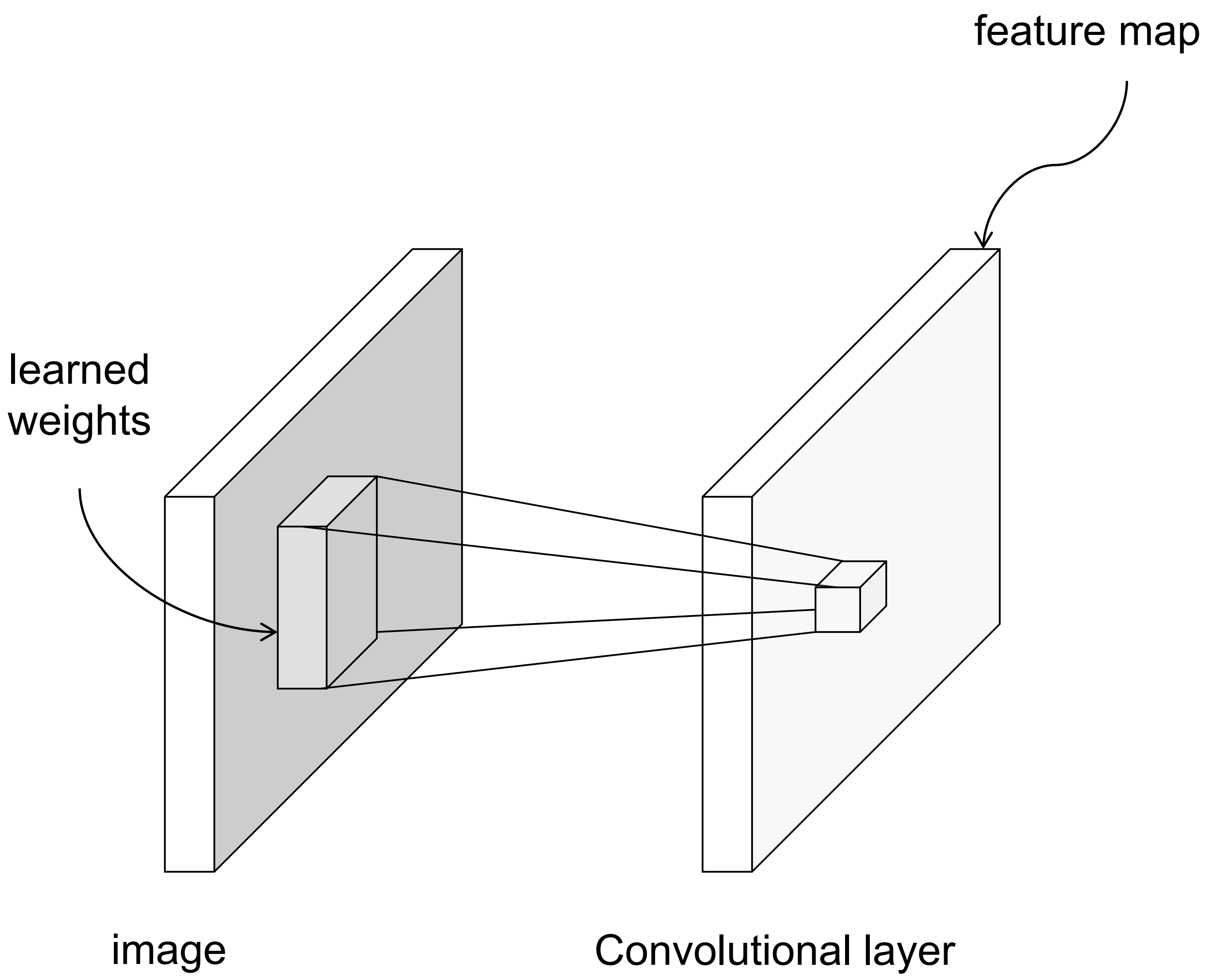


image

Convolutional layer

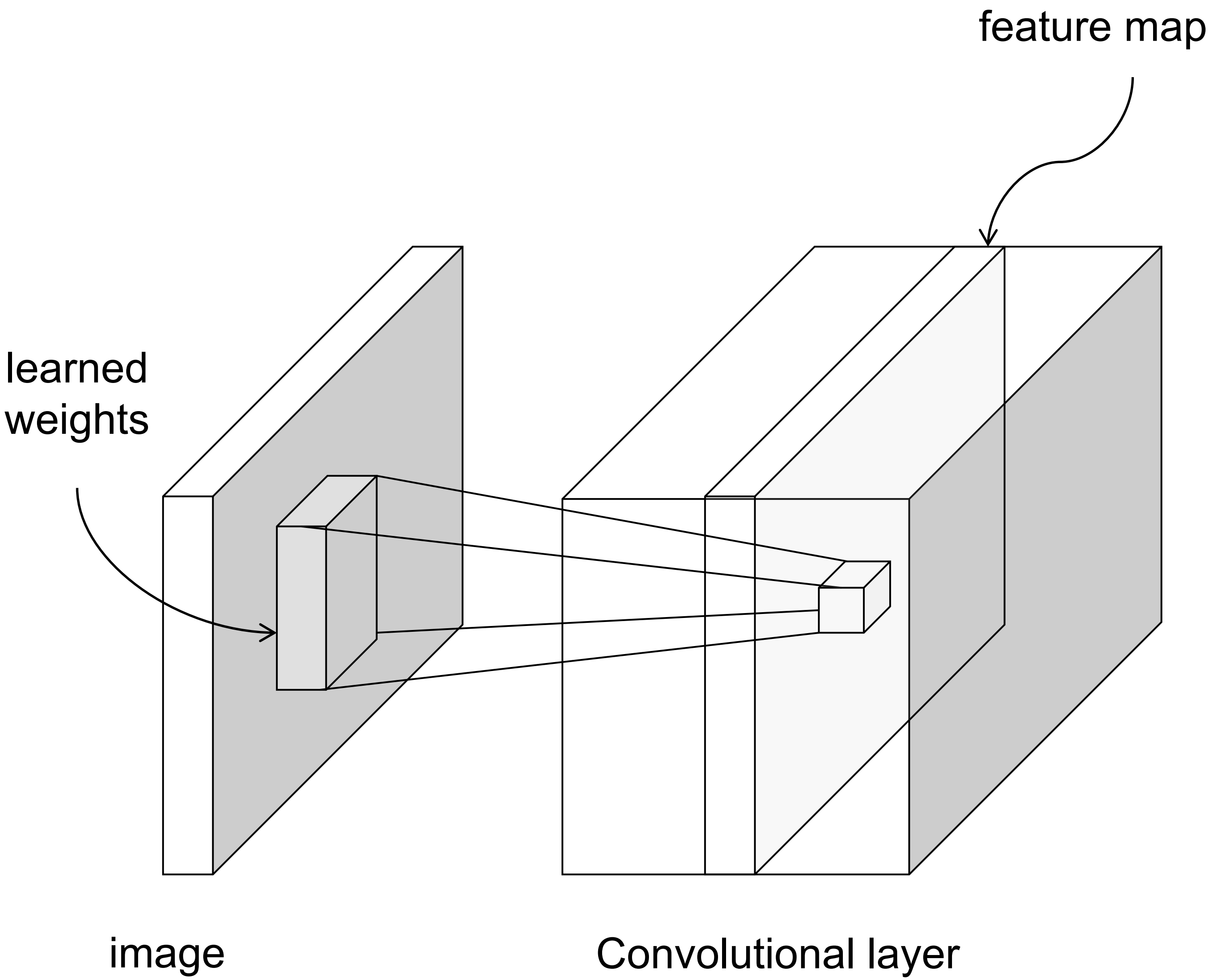
Slide: Lazebnik

Convolutional network



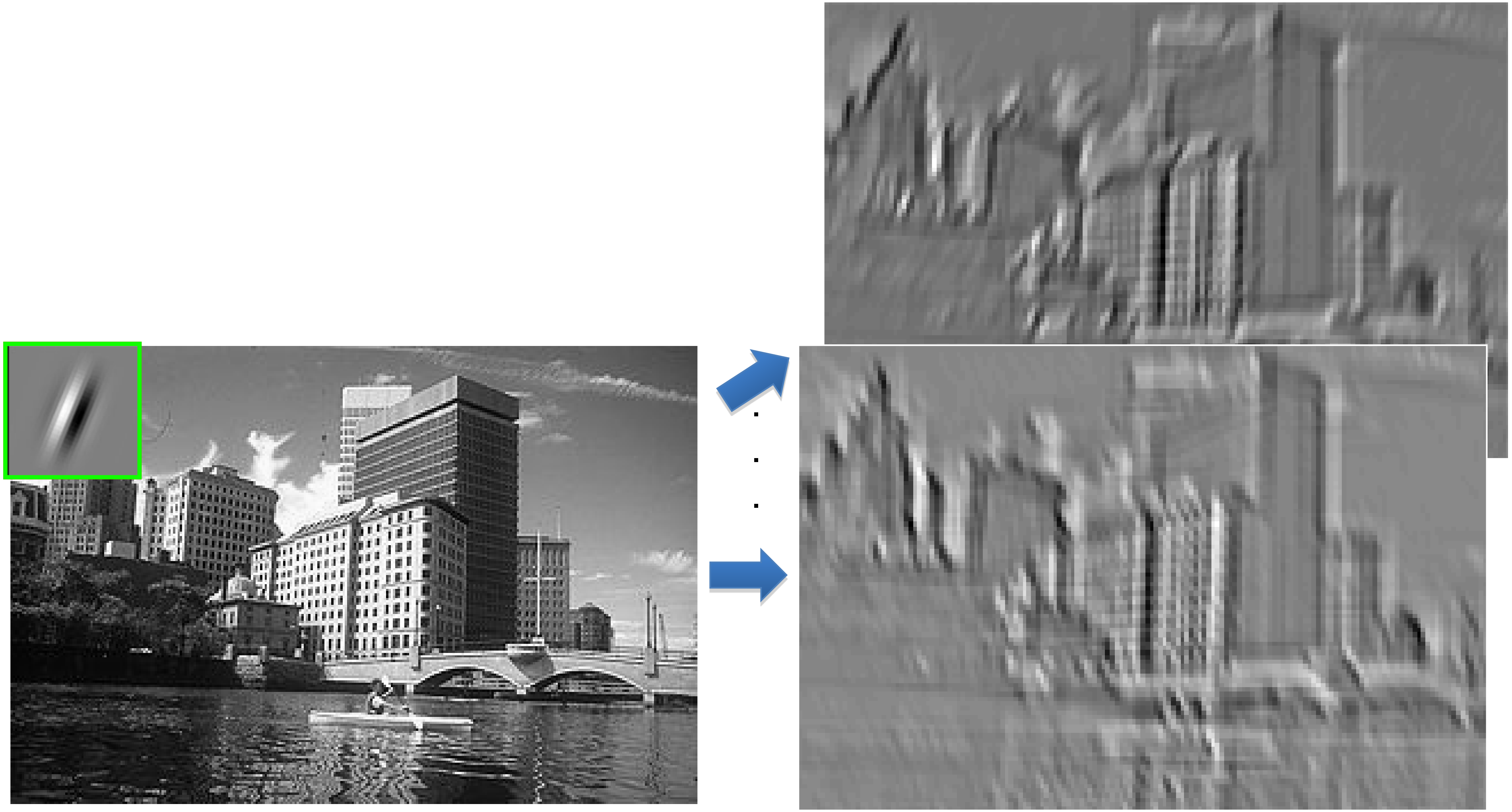
Slide: Lazebnik

Convolutional network



Slide: Lazebnik

Convolution as feature extraction

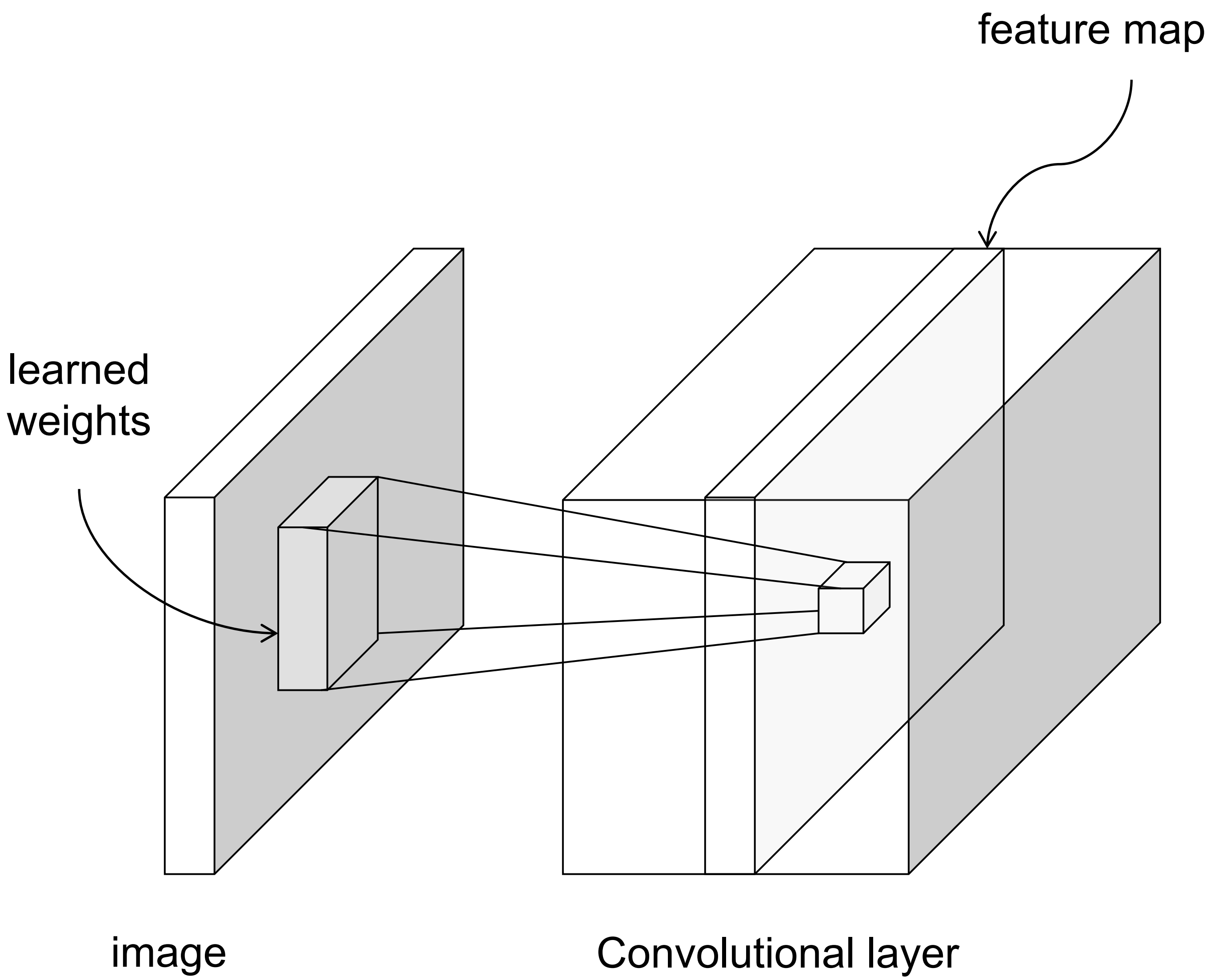


Input

Feature Map

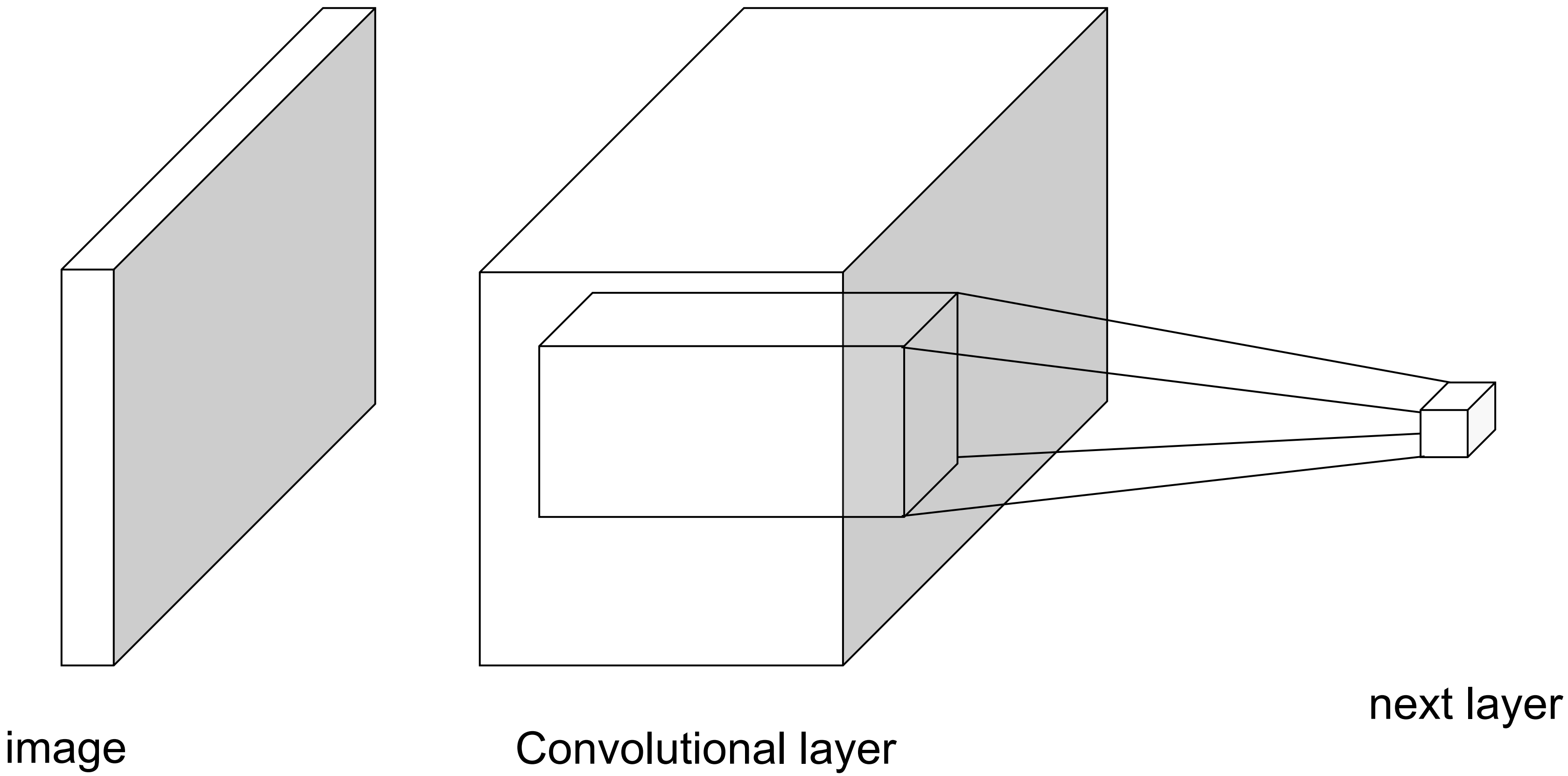
Slide: Lazebnik

From fully connected to convolutional networks



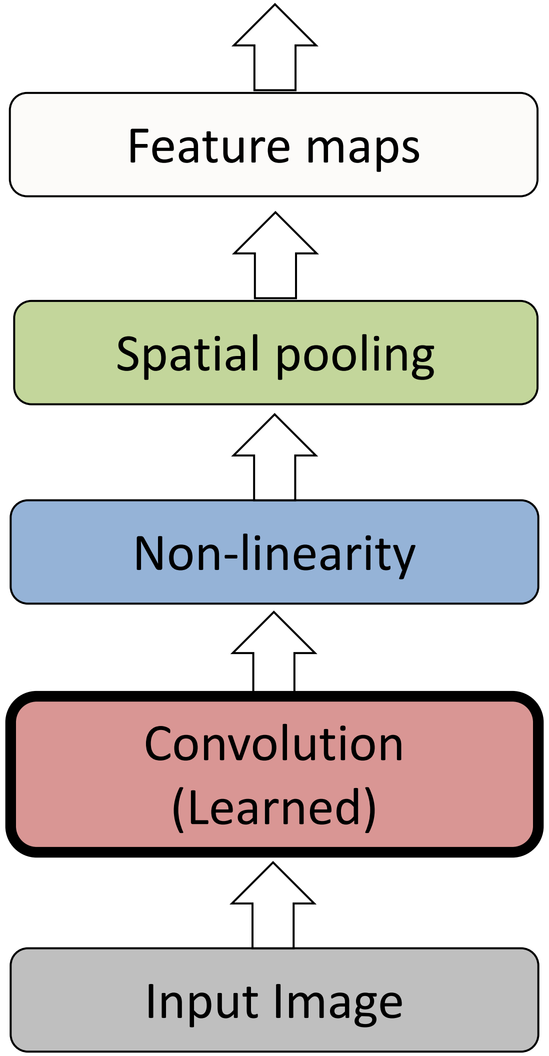
Slide: Lazebnik

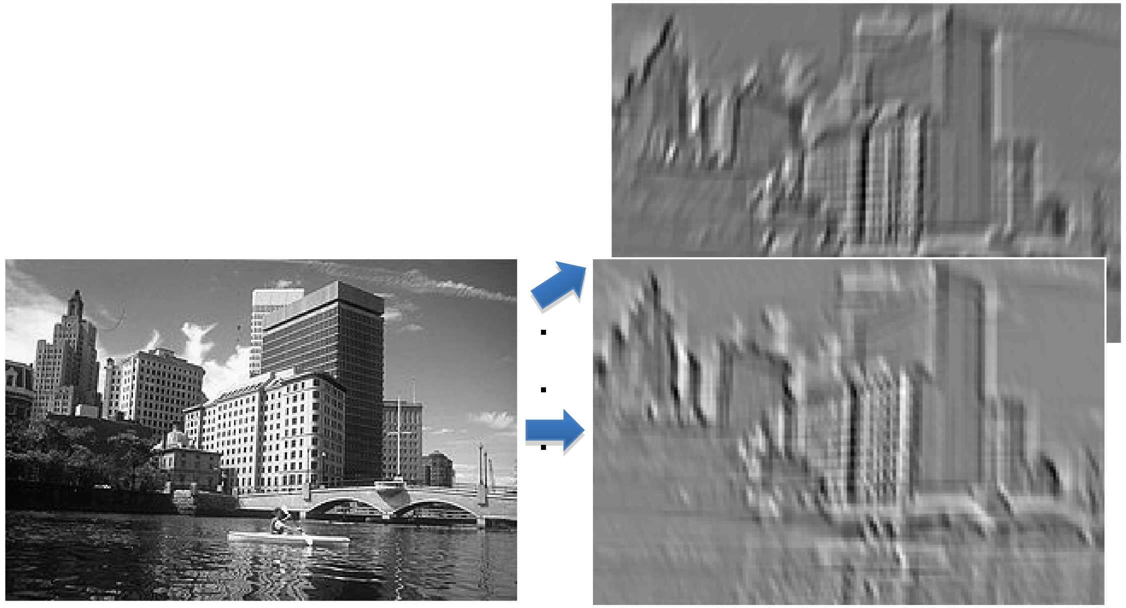
From fully connected to convolutional networks



Slide: Lazebnik

Key operations in a CNN





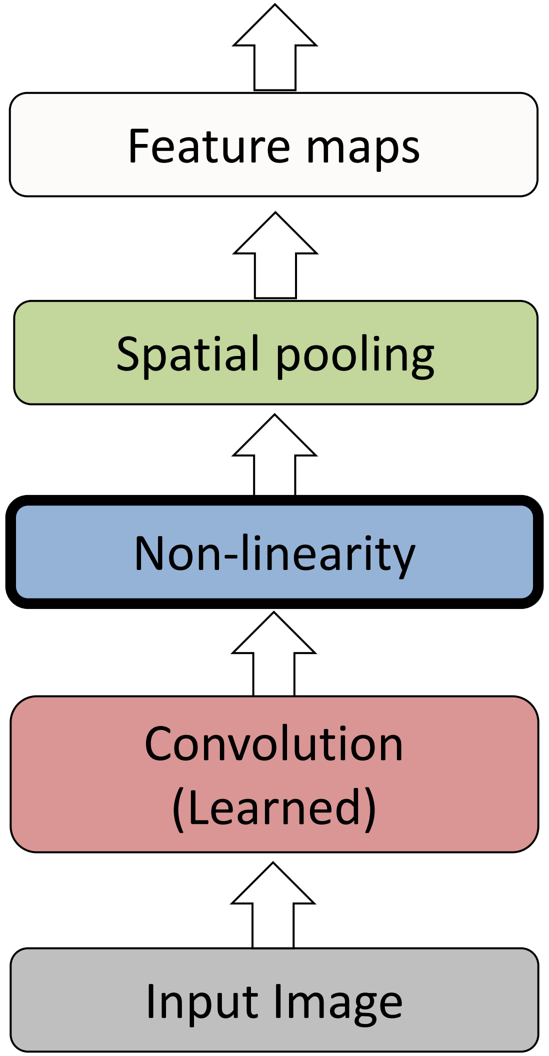
Input

Feature Map

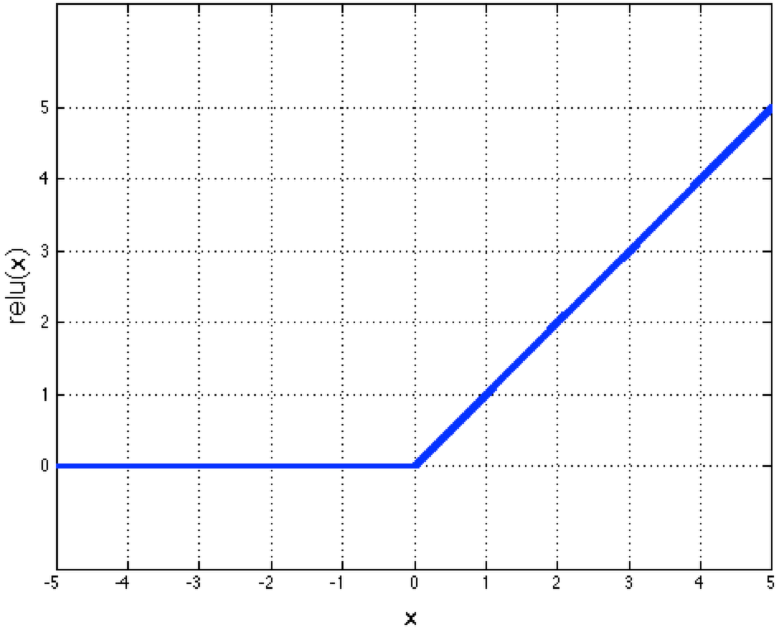
Source: R. Fergus, Y. LeCun

Slide: Lazebnik

Key operations

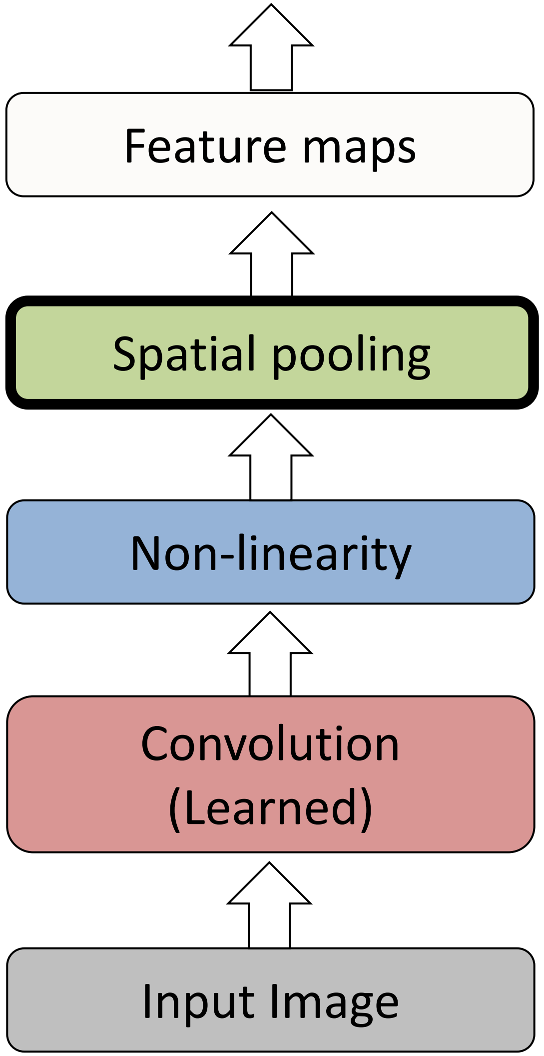


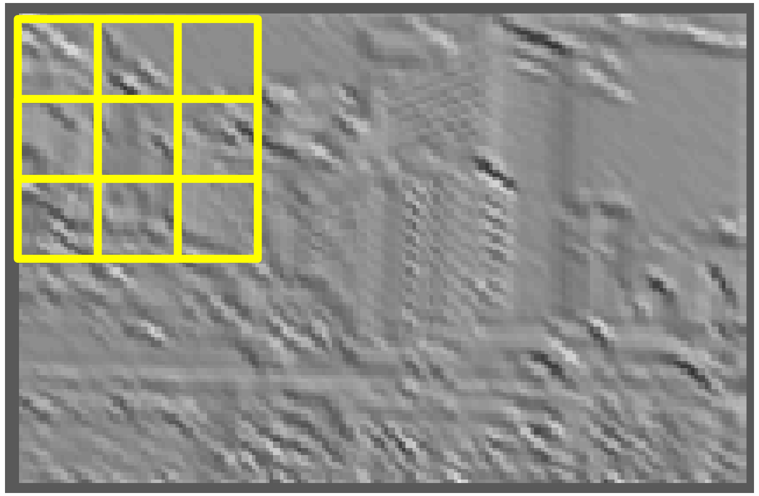
Rectified Linear Unit (ReLU)



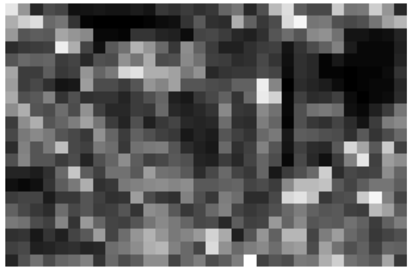
Source: R. Fergus, Y. LeCun

Key operations





Max



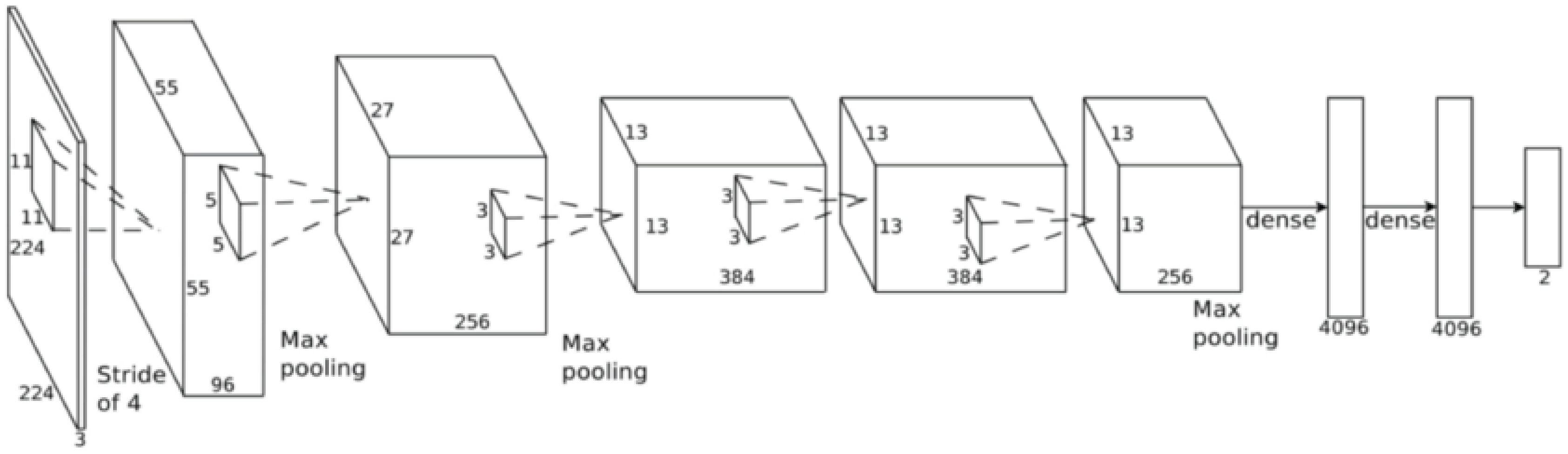
Source: R. Fergus, Y. LeCun

Slide: Lazebnik

AlexNet architecture

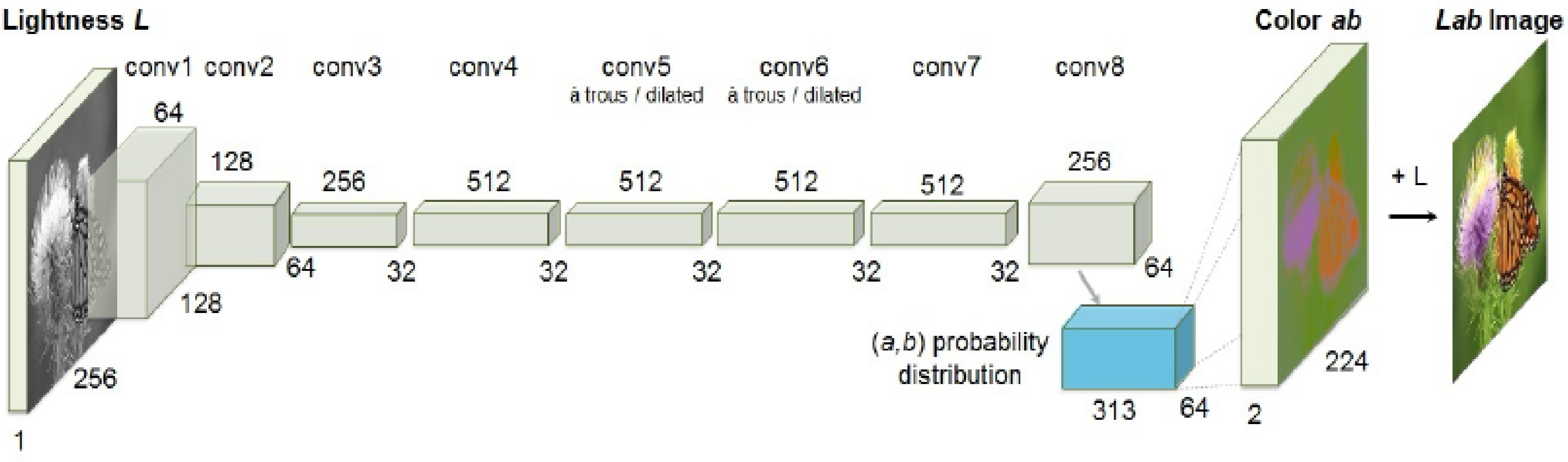
Create encoding by passing image through a series of steps

1. Feature generation
   a. Apply filters
   b. ReLU: Zero out negative values
   c. Downsample or “pool” by taking average or max response
2. Vectorize and add dense neural network layers



AlexNet: achieved good results on ImageNet in 2012 to convince computer vision researchers of potential

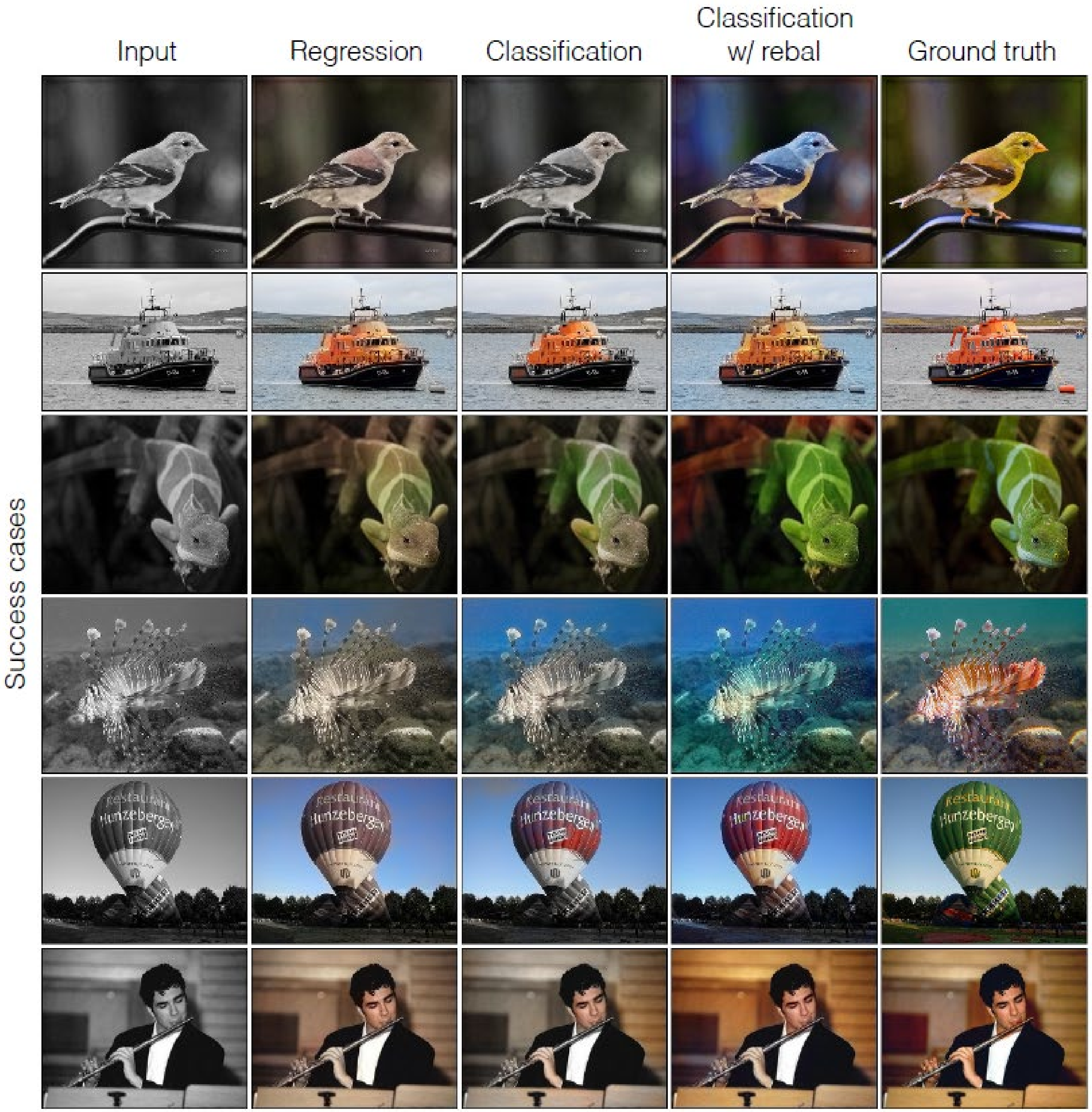
Colorful image colorization



Each conv layer refers to a block of 2 or 3 repeated conv and ReLU layers, followed by a BatchNorm. No pooling-- all changes in resolution are achieved through spatial downsampling or upsampling between conv blocks.

- Divide color (ab) values into bins and classify each pixel
- Loss puts more emphasis on correctly predicting rare colors (otherwise, prior for gray is too strong)
- Final inference is annealed mean of distribution (makes probabilities more peaky before taking mean, so that color is not averaged out too much)

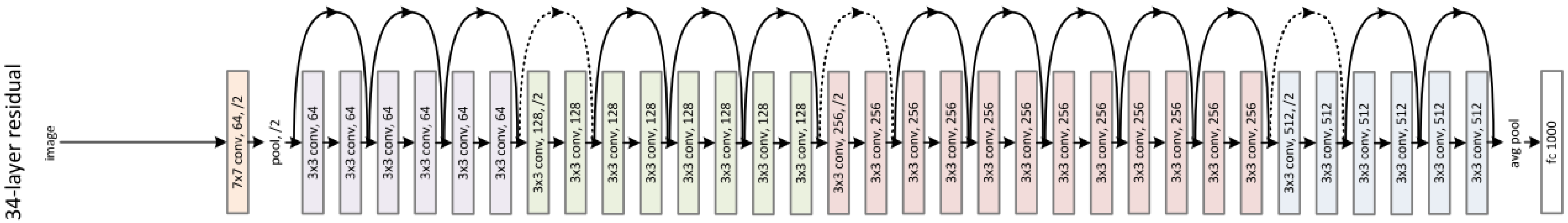
Zhang et al. ECCV 2016
https://richzhang.github.io/colorization/


Input
Regression
Classification
Classification w/ rebal
Ground truth
Success cases


Input
Regression
Classification
Classification w/ rebal
Ground truth
Failure cases

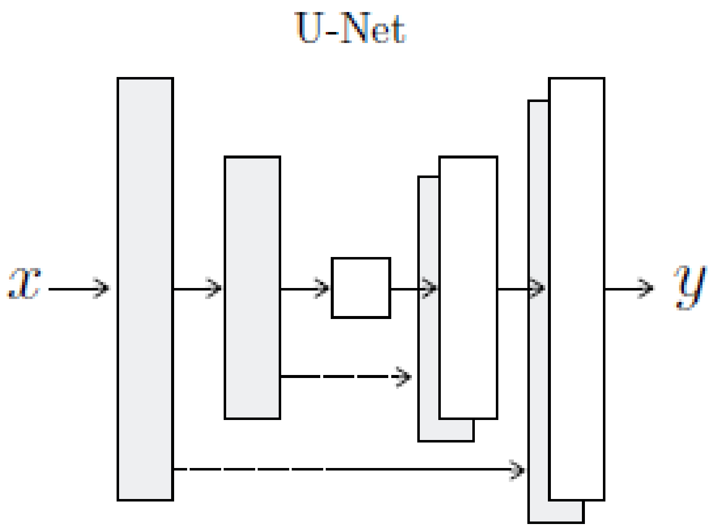
ResNet introduces “skip” connections

- Layers add their response to previous layer outputs so they don’t need to re-encode it
- Network is more compact and easier to train



ResNet Architecture

U-Net Architecture



The "U-Net" is an encoder-decoder with skip connections between mirrored layers in the encoder and decoder stacks.

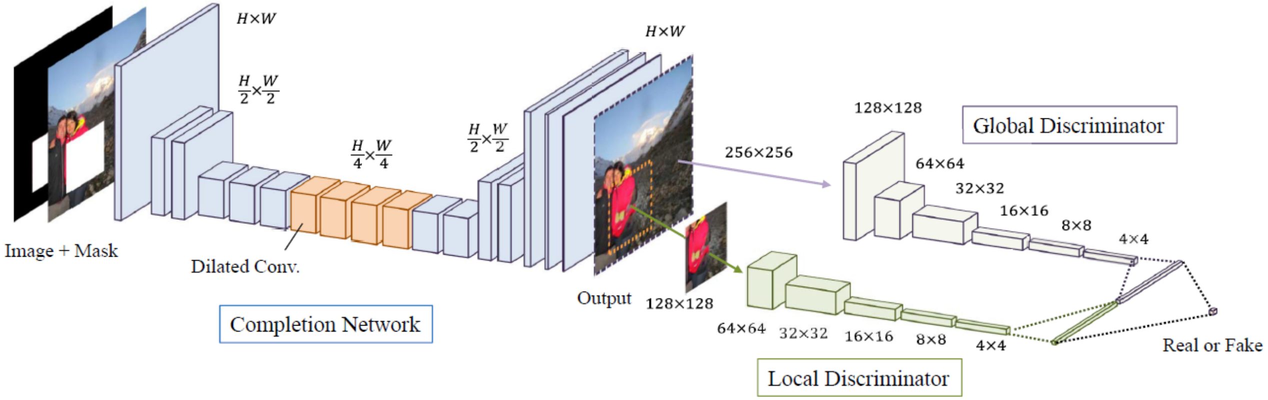
O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In MICCAI, 2015.

Fig from Isola et al. 2017

Globally and Locally Consistent Image Completion

SATOSHI IIZUKA, Waseda University
EDGAR SIMO-SERRA, Waseda University
HIROSHI ISHIKAWA, Waseda University

Fig. 1. Image completion results by our approach. The masked area is shown in white. Our approach can generate novel fragments that are not present elsewhere in the image, such as needed for completing faces; this is not possible with patch-based methods.



SIGGRAPH 2017

Why deep networks work

- "**End-to-end training**": feature learner (encoder) and regressor/classifier (decoder) guided by same objective

- **Flexible objective** design: can use any differentiable function to guide learning

- **Convolutional features** make sense for images because they are shift invariant and have relatively few parameters

- **High capacity** – can encode lots of data

Key factors in network performance

- **Objective function**: defines what the network is trying to do
- **Architecture:**
  - CNN vs Transformer
  - Size: Number of layers, filters, width of fully connected layers
  - Normalizations, skip connections, bottlenecks
- Amount of **training data**: more is better
- **Optimization**: gradient descent tools and parameters

Summary

- Many questions have been asked before, photos have been taken before

- Sometimes, we can shortcut hard problems by looking up the answer

- Deep networks learn features that make the lookup more effective

Next class (Thursday)

- Generating and detecting fakes
  - Pix2pix
  - CycleGAN
  - StyleNet
  - Diffusion