Computational Photography
CSE 590

Tamara Berg
Today

- About me
- About you
- Introduction to Computational Photography
- Course info
- Basics of images, pixels, & pyramids
About me

1997-2001
Undergrad at U.W. Madison
CS and Math

2001-2007
Grad at U.C. Berkeley
Ph.D. in CS

2007-2008
Postdoc at Yahoo! Research

2008-
Assistant Prof in CS at SBU
Tags: canon, eos, macro, japan, vacation, frog, animal, toad, amphibian, pet, eye, feet, mouth, finger, hand, prince, photo, art, light, photo, flickr, blurry, favorite, nice.

My Research

It’s the perfect party dress. With distinctly feminine details such as a wide sash bow around an empire waist and a deep scoopneck, this linen dress will keep you comfortable and feeling elegant all evening long.
Visually Descriptive Text

“It was an arresting face, pointed of chin, square of jaw. Her eyes were pale green without a touch of hazel, starred with bristly black lashes and slightly tilted at the ends. Above them, her thick black brows slanted upward, cutting a startling oblique line in her magnolia-white skin—that skin so prized by Southern women and so carefully guarded with bonnets, veils and mittens against hot Georgia suns” – Gone with the Wind

Visually descriptive language provides:

• information about how people construct natural language for imagery.
• information about the world, especially the visual world.
• guidance for computational visual recognition.

How do people describe the world?

How does the world work?

What should we recognize?
President George W. Bush makes a statement in the Rose Garden while Secretary of Defense Donald Rumsfeld looks on, July 23, 2003. Rumsfeld said the United States would release graphic photographs of the dead sons of Saddam Hussein to prove they were killed by American troops. Photo by Larry Downing/Reuters
Fully beaded with megawatt crystals, this Christian Louboutin suede pump matches the gleam in your eye.

Pump's linear heel plays up the alluring curves of its dipped sides.

Round toe frames low-cut vamp. Tonally topstitched collar.

4" straight, covered heel shows off signature red sole.

Creamy leather lining with padded insole. "Fifi" is made in Italy.
Given Web Images + Noisy Text Descriptions:
1) Discover visual attribute terms in text descriptions - likely domain dependent
2) Learn appearance models for attributes without labeled data
3) Characterize attributes by: type, localizability
BabyTalk: Understanding and Generating Simple Image Descriptions

Kulkarni, Premraj, Dhar, Li, Choi, AC Berg, TL Berg, CVPR 2011

Attributes

- green green grass by the lake
- a very shiny car in the car museum in my hometown of upstate NY.

Relationships

- very little person in a big rocking chair
- Our cat Tusik sleeping on the sofa near a hot radiator.
This is a photograph of one person and one brown sofa and one dog. The person is against the brown sofa. And the dog is near the person, and beside the brown sofa.
This is a picture of one sky, one road and one sheep. The gray sky is over the gray road. The gray sheep is by the gray road.

Here we see one road, one sky and one bicycle. The road is near the blue sky, and near the colorful bicycle. The colorful bicycle is within the blue sky.

This is a picture of two dogs. The first dog is near the second furry dog.
About you?
What is Computational Photography?

- “computational techniques that enhance or extend photography”

- Another medium for visual expression

Many of the following slides from Alyosha Efros & Derek Hoiem
Depicting Our World: The Beginning

Prehistoric Painting, Lascaux Cave, France
~ 15,000 -- 13,000 B.C.

Slide credit: Alyosha Efros
The Empress Theodora with her court.
Ravenna, St. Vitale 6th c.
Nuns in Procession. French ms. ca. 1300.
Depicting Our World: Renaissance

North Doors (1424)

Lorenzo Ghiberti (1378-1455)

East Doors (1452)

Slide credit: Alyosha Efros
Depicting Our World: Renaissance

Paolo Uccello, 
*Miracle of the Profaned Host (c.1467-9)*

Slide credit: Alyosha Efros
Depicting Our World: Toward Perfection

Jan van Eyck, *The Arnolfini Portrait* (1426-1434)
Depicting Our World: Toward Perfection

Lens Based Camera Obscura, 1568

Slide credit: Alyosha Efros
Depicting Our World: Toward Perfection

Camera Obscura: View of Times Square in Hotel Room, 2010
ABELARDO MORELL
Depicting Our World: Perfection!

*Still Life*, Louis Jaques Mande Daguerre, 1837

Slide credit: Alyosha Efros
• When was photography invented? 1826
• By whom? Niepce
  – Exposure time? 8 hours

• William Henry Fox Talbot invents the *calotype* in 1834 which pretty much invents the negative
First production camera?

- 1839. Daguerrotype
Beginning of hobby photography?

- 1900 Kodak Brownie
• Who did the first color photography?
• Some of the oldest color photos still preserved: Prokudin-Gorskii [http://www.loc.gov/exhibits/empire/](http://www.loc.gov/exhibits/empire/)
Prokudin-Gorskii
Instant photography?

- 1947, Edwin Land (Polaroid founder)
Electronic photography?

• A. A. CAMPBELL SWINTON AND ELECTRONIC PHOTOGRAPHY - 1908

• 25 images per second
First digital camera?

- 1975, Steve Sasson, Kodak
- Uses ccd from Fairchild semiconductor, A/D from Motorola, .01 megapixels, 23 second exposure, recorded on digital cassette
Autofocus

- 1978, Konica
- 1981 Pentax ME-F.
  - Canon AL1 had focus assist but no actuator
- Minolta Maxxum 1985 (AF in body)
First megapixel sensor

- Of reasonable size?
- (Kodak) Videk 1987, 1.4MPixels
Digital SLR?

- 1992 Kodak DCS 200, 1.5 Mpixels, based on Nikon body
Consumer digital SLR?

- Canon D30, 2000 3MPixels
But is a photo really realistic?
Depicting Our World: Realism?

Slide credit: Alyosha Efros
Paris, according to Flickr

Slide credit: Alyosha Efros
Paris, according to Google StreetView
Paris, according to me
Is reality what we want?

Newlyweds

Slide credit: Derek Hoiem

http://salavon.com
Better than realism?

City (westward)

Slide credit: Derek Hoiem

http://salavon.com
Enter Computer Graphics...
Traditional Computer Graphics

3D geometry

physics

Simulation

projection

Slide credit: Alyosha Efros
Computer graphics

Why so lifeless and sterile?

Slide credit: Alyosha Efros
The richness of our everyday world

Photo by Svetlana Lazebnik

Slide credit: Alyosha Efros

Photo by Svetlana Lazebnik
Beauty in complexity
Which parts are hard to model?
People

From “Final Fantasy”
Faces / Hair

From “Final Fantasy”

Slide credit: Alyosha Efros

Photo by Joaquín Rosales Gomez
Urban Scenes

Virtual LA (SGI)

Photo of LA

Slide credit: Alyosha Efros
The Realism Spectrum

Computer Graphics

+ easy to create new worlds
+ easy to manipulate objects/viewpoint
- very hard to look realistic

Photography

+ instantly realistic
+ easy to acquire
- very hard to manipulate objects/viewpoint

Computational Photography
Computational Photography

How can I use computational techniques to capture light in new ways?

How can I use computational techniques to breathe new life into the photograph?

How can I use computational techniques to synthesize and organize photo collections?

Slide credit: Alyosha Efros
Virtual Real World

http://www.debevec.org/Campanile/

Campanile Movie (1997)
Going beyond reality...

Benjamin Button (2008)

http://www.ted.com/talks/ed_ulbrich_shows_how_benjamin_button_got_his_face.html
The unfinished digital photography revolution

🔹 Traditional photography:
  - optics focuses optical array onto sensor
  - chemistry records final image

🔹 Digital photography
  - optics focuses optical array onto sensor
  - digital sensor records final image
Computational Photography

- Arbitrary computation between the optical array and the final image
- Data recorded by sensor is not the final image

[Images: Generalized imaging, Lots of computation, Final image]
Computational Photography

Arbitrary computation between optical array and final image (or final product)

- Post-process after traditional imaging
  - a.k.a. image processing (maybe more interactive)
  - But also combine multiple images to overcome limits of traditional imaging (HDR, panorama)

- Design imaging architecture together with computation
  - Computational cameras, computational illumination, coded imaging, data-rich imaging

- Extract more than just 2D images

- New media (panorama, photo tourism)
Examples of Computational Photography in action...
Toward better photos

Sometimes B&W photos are disappointing

Two-scale Tone Management for Photographic Look
Soonmin Bae  Sylvain Paris  Frédo Durand
Siggraph 2006
Toward better photos

Can you “transfer” some low-level qualities?

Two-scale Tone Management for Photographic Look
Soonmin Bae  Sylvain Paris  Frédo Durand
Siggraph 2006
Toward better photos

Output result

Two-scale Tone Management for Photographic Look
Soonmin Bae  Sylvain Paris  Frédo Durand
Siggraph 2006
Toward harnessing the internet
Toward harnessing the internet
Toward harnessing the internet
Toward harnessing the internet
Toward harnessing the internet
Toward better cameras

1. Light Field Photography with “Plenoptic Camera”

Adelson and Wang 1992

Ng et al. Stanford TR, 2005
Toward better cameras

• Like replacing the human retina with an insect compound eye
• Records where light ray hits the lens
Toward better cameras

Digital Refocusing
Course Info

Instructor: Tamara Berg
Office: 1411 Computer Science

Lectures: Mon 11:30-2:30, 2129 CS
Office hours: Mon 3-5pm (and by appointment)

Course webpage: tamaraberg.com/teaching/Fall_11
bring your lunch!
(and maybe sometimes we’ll get pizza)
Syllabus (subject to change)

Image manipulation & editing
pixels & pyramids
compositing & blending
cutting & growing
warping & morphing
image stitching for panoramas

What can we do with a billion images?
recognition/retrieval
modeling places on the internet
modeling people from lots of photos

Cameras & more
intro to cameras, color & light
computational approaches to cameras
3d and the Kinect
Syllabus (subject to change)

Image manipulation & editing
- pixels & pyramids
- compositing & blending
- cutting & growing
- warping & morphing
- image stitching for panoramas

What can we do with a billion images?
- recognition/retrieval
- modeling places on the internet
- modeling people from lots of photos

Cameras & more
- intro to cameras, color & light
- computational approaches to cameras
- 3d and the Kinect
Syllabus (subject to change)

Image manipulation & editing
- pixels & pyramids
- compositing & blending
- cutting & growing
- warping & morphing
- image stitching for panoramas

What can we do with a billion images?
- recognition/retrieval
- modeling places on the internet
- modeling people from lots of photos

Cameras & more
- intro to cameras, color & light
- computational approaches to cameras
- 3d and the Kinect

Interactive Digital Photomontage
Aseem Agarwala, Mira Dontcheva
Maneesh Agrawala, Steven Drucker, Alex Colburn
Brian Curless, David Salesin, Michael Cohen
Syllabus (subject to change)

Image manipulation & editing
- pixels & pyramids
- compositing & blending
- cutting & growing
- warping & morphing
- image stitching for panoramas

What can we do with a billion images?
- recognition/retrieval
- modeling places on the internet
- modeling people from lots of photos

Cameras & more
- intro to cameras, color & light
- computational approaches to cameras
- 3d and the Kinect
Syllabus (subject to change)

Image manipulation & editing
- pixels & pyramids
- compositing & blending
- cutting & growing
- warping & morphing
- image stitching for panoramas

What can we do with a billion images?
- recognition/retrieval
- modeling places on the internet
- modeling people from lots of photos

Cameras & more
- intro to cameras, color & light
- computational approaches to cameras
- 3d and the Kinect
Syllabus (subject to change)

Image manipulation & editing
- pixels & pyramids
- compositing & blending
- cutting & growing
- warping & morphing
- image stitching for panoramas

What can we do with a billion images?
- recognition/retrieval
- modeling places on the internet
- modeling people from lots of photos

Cameras & more
- intro to cameras, color & light
- computational approaches to cameras
- 3d and the Kinect
Syllabus (subject to change)

Image manipulation & editing
pixels & pyramids
compositing & blending
cutting & growing
warping & morphing
image stitching for panoramas

What can we do with a billion images?
recognition/retrieval
modeling places on the internet
modeling people from lots of photos

Cameras & more
intro to cameras, color & light
computational approaches to cameras
3d and the Kinect
Syllabus (subject to change)

Image manipulation & editing
- pixels & pyramids
- compositing & blending
- cutting & growing
- warping & morphing
- image stitching for panoramas

What can we do with a billion images?
- recognition/retrieval
- modeling places on the internet
- modeling people from lots of photos

Cameras & more
- intro to cameras, color & light
- computational approaches to cameras
- 3d and the Kinect
Syllabus (subject to change)

Image manipulation & editing
- pixels & pyramids
- compositing & blending
- cutting & growing
- warping & morphing
- image stitching for panoramas

What can we do with a billion images?
- recognition/retrieval
- modeling places on the internet
- modeling people from lots of photos

Cameras & more
- intro to cameras, color & light
- computational approaches to cameras
- 3d and the Kinect

Photo Tourism
Exploring photo collections in 3D
Noah Snavely  Steven M. Seitz  Richard Szeliski
University of Washington  Microsoft Research
SIGGRAPH 2006
Syllabus (subject to change)

Image manipulation & editing
- pixels & pyramids
- compositing & blending
- cutting & growing
- warping & morphing
- image stitching for panoramas

What can we do with a billion images?
- recognition/retrieval
- modeling places on the internet
- modeling people from lots of photos

Cameras & more
- intro to cameras, color & light
- computational approaches to cameras
- 3d and the Kinect
Syllabus (subject to change)

Image manipulation & editing
- pixels & pyramids
- compositing & blending
- cutting & growing
- warping & morphing
- image stitching for panoramas

What can we do with a billion images?
- recognition/retrieval
- modeling places on the internet
- modeling people from lots of photos

Cameras & more
- intro to cameras, color & light
- computational approaches to cameras
- 3d and the Kinect
Syllabus (subject to change)

Image manipulation & editing
- pixels & pyramids
- compositing & blending
- cutting & growing
- warping & morphing
- image stitching for panoramas

What can we do with a billion images?
- recognition/retrieval
- modeling places on the internet
- modeling people from lots of photos

Cameras & more
- intro to cameras, color & light
- computational approaches to cameras
- 3d and the Kinect
Syllabus (subject to change)

Image manipulation & editing
- pixels & pyramids
- compositing & blending
- cutting & growing
- warping & morphing
- image stitching for panoramas

What can we do with a billion images?
- recognition/retrieval
- modeling places on the internet
- modeling people from lots of photos

Cameras & more
- intro to cameras, color & light
- computational approaches to cameras
- 3d and the Kinect
Nitty gritty details (also on website)

- 4 assignments
- Weekly readings – Szeliski’s online book or research papers
- ~4 in class quizzes
- Final project

Participation expected

Grading: Assignments (40%), Project (30%), Quizzes (20%), Participation (10%)
Assignment 1
(aka getting started with matlab and images)

Colorize images of the Russian Empire!

You will need to:
1) extract the three color channel images
2) place them on top of each other
3) align them so that they form a single RGB color image.

Assume that a simple x,y translation model is sufficient for proper alignment. However, the full-size glass plate images are very large, so your alignment procedure will need to be relatively fast and efficient (e.g. coarse to fine pyramid search).

Details on course website
Due Sept 18 by email to: sbu590@gmail.com
(with links to your results)
Break
Images, pixels & pyramids
A digital camera replaces film with a sensor array

- Each cell in the array is a light-sensitive diode that converts photons to electrons
- Two common types: Charge Coupled Device (CCD) and CMOS
Images

Images are sampled and quantized measurements of light hitting a sensor.

**FIGURE 2.17** (a) Continuous image projected onto a sensor array. (b) Result of image sampling and quantization.
Images in the computer
Images in the computer
The goal of computer vision

- To perceive the story behind the picture

What we see

What a computer sees

Source: S. Narasimhan
“I stand at the window and see a house, trees, sky. Theoretically I might say there were 327 brightnesses and nuances of color. Do I have 327? No. I have sky, house, and trees.”

Laws of Organization in Perceptual Forms
Max Wertheimer (1923)
Perception of Intensity

from Ted Adelson
Perception of Intensity from Ted Adelson
Color Sensing in Camera (RGB)

- 3-chip vs. 1-chip: quality vs. cost

Why 3 colors?

http://www.cooldictionary.com/words/Bayer-filter.wikipedia
Images in Matlab

img = Ellsworth Kelly
Red Blue Green, 1963
Images in Matlab

Ellsworth Kelly
Red Blue Green, 1963

\[ \text{img} = \]
Images in Matlab

Ellsworth Kelly
Red Blue Green, 1963

\[
\text{img} = \begin{bmatrix}
\text{Red} & \text{Green} & \text{Blue}
\end{bmatrix}
\]
Images in Matlab

- Stored as 3d matrix of r, g, and b values at each pixel
  - Image matrix would be size N\times M\times 3
  - \( ? = \text{img}(::,1) \)  \( ? = \text{img}(::,2) \)  \( ? = \text{img}(::,3) \)

Ellsworth Kelly
Red Blue Green, 1963
Images in Matlab

- Stored as 3d matrix of r, g, and b values at each pixel
  - Image matrix would be size N x M x 3
  - $R = \text{img}(:,:,1)$  $G = \text{img}(:,:,2)$  $B = \text{img}(:,:,3)$
Red component

```matlab
imshow(img(:,:,1) =
```

```matlab
imshow(img(:,:,1) =
```
Green component

`imshow(img(:,:,2) =`
Blue component

```matlab
imshow(img(:,:,3))
```
Images in Matlab

Stored as 3d matrix of r, g, and b values at each pixel
- How could you get the r,g,b values for pixel (i,j)?
Images in Matlab

Stored as 3d matrix of r, g, and b values at each pixel

\[ \text{vals} = \text{img}(i,j,:) \]

Then \( r = \text{vals}(1) \); \( g = \text{vals}(2) \); \( b = \text{vals}(3) \);
Images in Matlab

Stored as 3d matrix of r, g, and b values at each pixel
- So r,g,b values are stored in img(i,j,:).
- In the case above these values might be [1 0 0].
Comparing Images

\[
\begin{array}{ccc}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9 \\
\end{array}
\]

img = \[
\begin{array}{ccc}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9 \\
\end{array}
\]

in matlab -- [1 2 3; 4 5 6; 7 8 9]

matrix to vector -- img2 = img(:);
Comparing Images

Sum of squared distances (SSD): 
\[
\sum_{i=1}^{n} (X_i - Y_i)^2
\]

Angle between vectors (NormCorr):
\[
\frac{a \cdot b}{||a|| \cdot ||b||}
\]

Dot Product: 
\[
a \cdot b = \sum_{i=1}^{n} a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n
\]

Length (Euclidean norm): 
\[
||a|| = \sqrt{a_1^2 + a_2^2 + \cdots + a_n^2}
\]
Matlab demo
Image filtering

• Image filtering: compute function of local neighborhood at each position

• Really important!
  – Enhance images
    • Denoise, resize, increase contrast, etc.
  – Extract information from images
    • Texture, edges, distinctive points, etc.
  – Detect patterns
    • Template matching
Example: box filter

\[ g[\cdot, \cdot] \]

\[
\begin{array}{ccc}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
\end{array}
\]

\[
\frac{1}{9}
\]
Image filtering

\[ f[\ldots] \]

\[ h[\ldots] \]

\[ h[m,n] = \sum_{k,l} g[k,l] f[m+k,n+l] \]
Image filtering

\[ f[\ldots] \]

\[ h[\ldots] \]

\[ h[m,n] = \sum_{k,l} g[k,l] f[m+k,n+l] \]

Credit: S. Seitz
Image filtering

\[ f[\ldots] \]

\[ h[\ldots] \]

\[ h[m,n] = \sum_{k,l} g[k,l] f[m+k,n+l] \]
Image filtering

\[ h[m,n] = \sum_{k,l} g[k,l] f[m+k,n+l] \]
Image filtering

\[ f[\ldots] \]

\[ h[\ldots] \]

\[
h[m,n] = \sum_{k,l} g[k,l] f[m+k,n+l]\]

Credit: S. Seitz
Image filtering

\[ f[\ldots] \]

\[ h[\ldots] \]

\[
h[m,n] = \sum_{k,l} g[k,l] f[m+k,n+l]
\]

Credit: S. Seitz
Image filtering

\[ f[\ldots] \]

\[ h[\ldots] = \sum_{k,l} g[k,l] \cdot f[m+k,n+l] \]

Credit: S. Seitz
Image filtering

\[ f[\ldots] \]

\[ g[\cdot,\cdot] \]

\[ h[\ldots] \]

\[
h[m,n] = \sum_{k,l} g[k,l] f[m+k,n+l]
\]

Credit: S. Seitz
Box Filter

What does it do?

• Replaces each pixel with an average of its neighborhood

• Achieve smoothing effect (remove sharp features)

Slide credit: David Lowe (UBC)
Smoothing with box filter
Practice with linear filters

Original

Source: D. Lowe
Practice with linear filters

Original

Filtered (no change)

Source: D. Lowe
Practice with linear filters

Original

![Image of an eye with a filter matrix]

Source: D. Lowe
Practice with linear filters

Original

Shifted left
By 1 pixel

Source: D. Lowe
Practice with linear filters

Original

\[
\begin{array}{ccc}
0 & 0 & 0 \\
0 & 2 & 0 \\
0 & 0 & 0 \\
\end{array}
\quad - \quad
\begin{array}{ccc}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
\end{array}
\quad ?
\]
Practice with linear filters

Original

Sharpening filter
- Accentuates differences with local average

Source: D. Lowe
Sharpening

before

after

Source: D. Lowe
Edge detection

• **Goal:** Identify sudden changes (discontinuities) in an image
  - Intuitively, most semantic and shape information from the image can be encoded in the edges
  - More compact than pixels

• **Ideal:** artist’s line drawing (but artist is also using object-level knowledge)

Source: D. Lowe
Origin of Edges

Edges are caused by a variety of factors

Source: Steve Seitz
Characterizing edges

• An edge is a place of rapid change in the image intensity function

![Image of an image and its intensity function along a horizontal scanline, showing an edge and its corresponding first derivative with extrema.](source: Svetlana Lazebnik)
Edge filters

Approximations of derivative filters:

Prewitt: \[ M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} ; \quad M_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \]

Sobel: \[ M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} ; \quad M_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \]

Roberts: \[ M_x = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} ; \quad M_y = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \]

Convolve filter with image to get edge map

Source: K. Grauman
Edge filters

Approximations of derivative filters:

Prewitt:

\[ M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad ; \quad M_y = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \]

Sobel:

\[ M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad ; \quad M_y = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \]

Roberts:

\[ M_x = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad ; \quad M_y = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ -1 \end{bmatrix} \]

Respond highly to vertical edges

Source: K. Grauman
Edge filters

Approximations of derivative filters:

\[
\begin{align*}
\text{Prewitt:} & & M_x &= \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} ; & M_y &= \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \\
\text{Sobel:} & & M_x &= \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} ; & M_y &= \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \\
\text{Roberts:} & & M_x &= \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} ; & M_y &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}
\end{align*}
\]

Respond highly to horizontal edges

Source: K. Grauman
Edges: example

source: Svetlana Lazebnik
Gaussian Kernel

\[ G_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \]

- Constant factor at front makes volume sum to 1 (can be ignored, as we should re-normalize weights to sum to 1 in any case)

Source: C. Rasmussen
Example: Smoothing with a Gaussian

source: Svetlana Lazebnik
Gaussian filters

• Remove “high-frequency” components from the image (low-pass filter)
  – Images become more smooth

• Convolution with self is another Gaussian
  – So can smooth with small-width kernel, repeat, and get same result as larger-width kernel would have

• *Separable* kernel
  – Factors into product of two 1D Gaussians

Source: K. Grauman
Separability of the Gaussian filter

\[ G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \]

\[ = \left( \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right) \right) \left( \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{y^2}{2\sigma^2}\right) \right) \]

The 2D Gaussian can be expressed as the product of two functions, one a function of \( x \) and the other a function of \( y \)

The filter factors into a product of 1D filters:

\[
\begin{array}{ccc}
1 & 2 & 1 \\
2 & 4 & 2 \\
1 & 2 & 1 \\
\end{array}
\times
\begin{array}{c}
1 \\
2 \\
1 \\
\end{array}
= 
\begin{array}{c}
1 \\
2 \\
1 \\
\end{array}
\]

Source: D. Lowe
Practical matters

• What is the size of the output?
• MATLAB: `imfilter(g, f, shape)`
  – `shape = 'full'`: output size is sum of sizes of `f` and `g`
  – `shape = 'same'`: output size is same as `f`
  – `shape = 'valid'`: output size is difference of sizes of `f` and `g

Source: S. Lazebnik
Practical matters

• What about near the edge?
  – the filter window falls off the edge of the image
  – need to extrapolate
  – methods:
    • clip filter (black)
    • wrap around
    • copy edge
    • reflect across edge

Source: S. Marschner
Practical matters

– methods (MATLAB):
  • clip filter (black): \texttt{imfilter}(f, g, 0)
  • wrap around: \texttt{imfilter}(f, g, ‘circular’)
  • copy edge: \texttt{imfilter}(f, g, ‘replicate’)
  • reflect across edge: \texttt{imfilter}(f, g, ‘symmetric’)

Source: S. Marschner
Image half-sizing

This image is too big to fit on the screen. How can we reduce it?

How to generate a half-sized version?
Image sub-sampling

Throw away every other row and column to create a 1/2 size image - called image sub-sampling
Image sub-sampling

1/2  
1/4 (2x zoom)  
1/8 (4x zoom)

Aliasing! What do we do?
Anti-aliasing

Solutions:
• Sample more often

• Get rid of all frequencies that are greater than half the new sampling frequency
  – Will lose information
  – But it’s better than aliasing
  – Apply a smoothing filter
Algorithm for downsampling by factor of 2

1. Start with image(h, w)
2. Apply low-pass filter
   \[
   \text{im\_blur} = \text{imfilter} (\text{image}, \text{fspecial('gaussian', 7, 1)})
   \]
3. Sample every other pixel
   \[
   \text{im\_small} = \text{im\_blur}(1:2:end, 1:2:end);
   \]
Gaussian (lowpass) pre-filtering

Solution: filter the image, *then* subsample
Subsampling with Gaussian pre-filtering

Gaussian 1/2

G 1/4

G 1/8

Slide by Steve Seitz
Compare with...

1/2

1/4 (2x zoom)

1/8 (4x zoom)
A bar in the big images is a hair on the zebra’s nose; in smaller images, a stripe; in the smallest, the animal’s nose.
Assignment 1
(aka getting started with matlab and images)

Colorize images of the Russian Empire!

- Try SSD alignment
- Try normxcorr2 alignment
- Simple implementation will work for small images
- But larger images will take forever (well, many hours)

Details on course website
Due Sept 18 by email to: sbu590@gmail.com
(with links to your results)
Coarse-to-fine Image Registration

1. Compute Gaussian pyramid
2. Align with coarse pyramid
3. Successively align with finer pyramids
   - Search smaller range

Why is this faster?
What are pyramids good for?

• Improve Search
  – Search over translations
    • Like HW1
    • Classic coarse-to-fine strategy
  – Search over scale
    • Template matching
    • E.g. find a face at different scales

• Pre-computation
  – Need to access image at different blur levels
  – Useful for texture mapping at different resolutions (called mip-mapping)
  – Useful for compositing/blending – next week!
For next week

Compositing/Blending
Readings on the website