Computational Approaches to Cameras

Magritte, *The False Mirror* (1935)

Computational Photography
Derek Hoiem, University of Illinois
Announcements

• Vote for Project 5 favorites by Sun night

• Only three classes left
  – Today: computational cameras
  – Thurs: Kinect Sensor and applications
  – Tues: wrap-up, feedback/ICES
Final projects

**Image Editing and Synthesis**
1. Image Analogies: Grace, Sidd, Suraj
2. Interactive Photo Montage: Jiqin and Zhongbo
3. Animating Pictures with Stochastic Motion Textures: Shengjie and Brian
4. LazySnapping segmentation: Roeland
5. Image Crystalization: Micheal W.
6. Seam Carving for Content-aware Image Resizing: Nan
7. Seam Carving or Miniature Faking: Abdel
8. Seam Carving: Joanne and Mike
9. Seam Carving: Brian Goerlitz

**Geometry**
1. Interface for zooming into panoramic stitches: Nemo
2. 3D Reconstruction from Image(s): Ramy

**Other Topics**
1. FocusTest: JunYoung
2. Light fields: Xinqi, Siying
3. Tracking of Cells in Microscopic Video: Xing
Conventional cameras

- Conventional cameras are designed to capture light in a medium that is directly viewable.
Computational cameras

• With a computational approach, we can capture light and then figure out what to do with it
Questions for today

• How can we represent all of the information contained in light?

• What are the fundamental limitations of cameras?

• What sacrifices have we made in conventional cameras? For what benefits?

• How else can we design cameras for better focus, deblurring, multiple views, depth, etc.?
Q: What is the set of all things that we can ever see?
A: The Plenoptic Function (Adelson & Bergen)

Let’s start with a stationary person and try to parameterize everything that he can see…
Grayscale snapshot

\[ P(\theta,\phi) \]

is intensity of light

- Seen from a single view point
- At a single time
- Averaged over the wavelengths of the visible spectrum

(can also do \( P(x,y) \), but spherical coordinate are nice)
Color snapshot

$P(\theta, \phi, \lambda)$

is intensity of light

- Seen from a single view point
- At a single time
- As a function of wavelength
A movie

$P(\theta, \phi, \lambda, t)$

is intensity of light

- Seen from a single view point
- Over time
- As a function of wavelength
Holographic movie

\[ P(\theta, \phi, \lambda, t, V_X, V_Y, V_Z) \]

is intensity of light

- Seen from ANY viewpoint
- Over time
- As a function of wavelength
The Plenoptic Function

\[ P(\theta, \phi, \lambda, t, V_x, V_y, V_z) \]

- Can reconstruct every possible view, at every moment, from every position, at every wavelength
- Contains every photograph, every movie, everything that anyone has ever seen!
Representing light

The atomic element of light: a pixel a ray
Fundamental limitations and trade-offs

• Only so much light in a given area to capture
• Basic sensor accumulates light at a set of positions from all orientations, over all time
• We want **intensity** of light at a **given time** at one **position** for a set of orientations
• Solutions:
  – funnel, constrain, redirect light
  – change the sensor

CCD inside camera
Trade-offs of conventional camera

• Add a pinhole
  ✓ Pixels correspond to small range of orientations at the camera center, instead of all gathered light at one position
  ✗ Much less light hits sensor

• Add a lens
  ✓ More light hits sensor
  ✗ Limited depth of field
  ✗ Chromatic aberration

• Add a shutter
  • Capture average intensity at a particular range of times

• Increase sensor resolution
  ✓ Each pixel represents a smaller range of orientations
  ✗ Less light per pixel

• Controls: aperture size, focal length, shutter time
How else can we design cameras?

What do they sacrifice/gain?
1. Light Field Photography with “Plenoptic Camera”

Adelson and Wang 1992

Ng et al. Stanford TR, 2005
Light field photography

- Like replacing the human retina with an insect compound eye
- Records where light ray hits the lens
Stanford Plenoptic Camera [Ng et al 2005]

Contax medium format camera

Kodak 16-megapixel sensor

Adaptive Optics microlens array

125μ square-sided microlenses

\[
4000 \times 4000 \text{ pixels} \div 292 \times 292 \text{ lenses} = 14 \times 14 \text{ pixels per lens}
\]
Light field photography: applications
Light field photography: applications

Change in viewpoint
Light field photography: applications

Change in viewpoint

Lateral

Along Optical Axis
Digital Refocusing
Light field photography w/ microlenses

• We gain
  – Ability to refocus or increase depth of field
  – Ability for small viewpoint shifts

• What do we lose (vs. conventional camera)?
2. Coded apertures
Image and Depth from a Conventional Camera with a Coded Aperture

Anat Levin, Rob Fergus, Frédo Durand, William Freeman

MIT CSAIL
Single input image:

Output #1: Depth map
Single input image:

Output #1: Depth map

Output #2: All-focused image
Lens and defocus

Lens’ aperture

Image of a point light source

Focal plane

Lens

Camera sensor

Point spread function
Lens and defocus

Lens’ aperture → Image of a defocused point light source

Object → Lens → Camera sensor

Focal plane
Lens and defocus

Lens’ aperture

Image of a defocused point light source

Object

Lens’ aperture

Focal plane

Lens

Camera sensor

Point spread function
Lens and defocus

Lens’ aperture

Image of a defocused point light source

Object

Lens

Camera sensor

Focal plane

Point spread function
Lens and defocus

Lens’ aperture

Image of a defocused point light source

Object

Lens

Camera sensor

Focal plane

Point spread function
Depth and defocus

Depth from defocus:
Infer depth by analyzing local scale of defocus blur
Challenges

• Hard to discriminate a smooth scene from defocus blur

• Hard to undo defocus blur

Ringing with conventional deblurring algorithm
Key ideas

• Exploit prior on natural images
  - Improve deconvolution
  - Improve depth discrimination

• Coded aperture (mask inside lens)
  - make defocus patterns different from natural images and easier to discriminate
Defocus as local convolution

\[ y = f_k \otimes x \]

- **Input defocused image**
- **Local sub-window**
- **Calibrated blur kernels at depth \( k \)**
- **Sharp sub-window**

Depth \( k=1 \):

Depth \( k=2 \):

Depth \( k=3 \):
Overview

Try deconvolving local input windows with different scaled filters:

Larger scale

Correct scale

Smaller scale

Somehow: select best scale.
Challenges

- Hard to deconvolve even when kernel is known
- Hard to identify correct scale:
  - Larger scale
  - Correct scale
  - Smaller scale

Input

Ringing with the traditional Richardson-Lucy deconvolution algorithm
Deconvolution is ill posed

\[ f \otimes x = y \]
Deconvolution is ill posed

\[ f \otimes x = y \]

Solution 1:

Solution 2:
Idea 1: Natural images prior

What makes images special?

Natural images have sparse gradients

\[\Rightarrow\] put a penalty on gradients
Deconvolution with prior

\[ x = \arg \min \left( \| f \otimes x - y \|^2 + \lambda \sum_i \rho(\nabla x_i) \right) \]

- Convolution error
- Derivatives prior

Equal convolution error: Low

High convolution error: High
Recall: Overview

Try deconvolving local input windows with different scaled filters:

- Larger scale
- Correct scale
- Smaller scale

Somehow: select best scale.

Challenge: smaller scale not so different than correct
Idea 2: Coded Aperture

- Mask (code) in aperture plane
  - make defocus patterns different from natural images and easier to discriminate

Conventional aperture

Our coded aperture
Solution: lens with occluder
Solution: lens with occluder

Aperture pattern

Image of a defocused point light source

Object

Focal plane

Lens with coded aperture

Camera sensor

Point spread function
Solution: lens with occluder

Aperture pattern

Image of a defocused point light source

Object

Focal plane

Lens with coded aperture

Camera sensor

Point spread function
Solution: lens with occluder

Aperture pattern

Image of a defocused point light source

Object

Focal plane

Lens with coded aperture

Camera sensor

Point spread function
Solution: lens with occluder

Aperture pattern

Image of a defocused point light source

Object

Lens with coded aperture

Focal plane

Camera sensor

Point spread function
Solution: lens with occluder

Aperture pattern

Image of a defocused point light source

Object

Lens with coded aperture

Camera sensor

Focal plane

Point spread function
Coded aperture reduces uncertainty in scale identification

- **Larger scale**
  - Conventional: Incorrectly identifies the correct scale
  - Coded: Correctly identifies the correct scale

- **Correct scale**
  - Conventional: Correctly identifies the correct scale
  - Coded: Correctly identifies the correct scale

- **Smaller scale**
  - Conventional: Correctly identifies the correct scale
  - Coded: Incorrectly identifies the correct scale
Convolution - frequency domain representation

Spatial convolution $\iff$ frequency multiplication

Output spectrum has zeros where filter spectrum has zeros
Coded aperture: Scale estimation and division by zero

- Estimated image
- Filter, correct scale
- Estimated image
- Filter, wrong scale

Large magnitude in image to compensate for tiny magnitude in filter

Spatial ringing

- Observed image
Division by zero with a conventional aperture?

- Estimated image
- Filter, correct scale
- Estimated image
- Filter, wrong scale

- Observed image

⇒ no spatial ringing
Filter Design

Analytically search for a pattern maximizing discrimination between images at different defocus scales \((KL\text{-divergence})\)

Account for image prior and physical constraints
Depth results
Regularizing depth estimation

Try deblurring with 10 different aperture scales

\[ x = \arg \min f \otimes x - y \|^2 + \lambda \sum_i \rho(\nabla x_i) \]

Convolution error

Derivatives prior

Keep minimal error scale in each local window + regularization
Regularizing depth estimation

Input

Local depth estimation

Regularized depth
All focused results
All-focused (deconvolved)
Close-up

Original image

All-focus image
Comparison - conventional aperture result

Ringing due to wrong scale estimation
Comparison - coded aperture result
All-focused (deconvolved)
Close-up

Original image  All-focus image  Naïve sharpening
Application: Digital refocusing from a single image
Application: Digital refocusing from a single image
Application: Digital refocusing from a single image
Application: Digital refocusing from a single image
Application: Digital refocusing from a single image
Application: Digital refocusing from a single image
Coded aperture: pros and cons

- Image AND depth at a single shot
- No loss of image resolution
- Simple modification to lens
  - Depth is coarse
    - unable to get depth at untextured areas, might need manual corrections.
- But depth is a pure bonus
- Lose some light
- But deconvolution increases depth of field
50mm f/1.8: $79.95
Cardboard: $1
Tape: $1
Depth acquisition: priceless
Some more quick examples
• Quickly move camera in a parabola when taking a picture
• A motion at any speed in the direction of the parabola will give the same blur kernel
Results

Static Camera

Parabolic Camera
Results

Static Camera

Parabolic Camera

Motion in wrong direction
Looking Around the Corner using Transient Imaging

Ahmed Kirmani *¹, Tyler Hutchison¹, James Davis †², and Ramesh Raskar ‡¹

¹MIT Media Laboratory  ² UC Santa Cruz
RGBW Sensors

- 2007: Kodak ‘Panchromatic’ Pixels
- Outperforms Bayer Grid
  - 2X-4X sensitivity (W: no filter loss)
  - May improve dynamic range (W >> RGB sensitivity)

Computational Approaches to Display

• 3D TV without glasses
  – You see different images from different angles

http://news.cnet.com/8301-13506_3-20018421-17.html
Newer version: http://www.pcmag.com/article2/0,2817,2392380,00.asp

Toshiba
Recap of questions

• How can we represent all of the information contained in light?

• What are the fundamental limitations of cameras?

• What sacrifices have we made in conventional cameras? For what benefits?

• How else can we design cameras for better focus, deblurring, multiple views, depth, etc.?
Next class

- Kinect sensor