

# Image-based rendering (IBR)

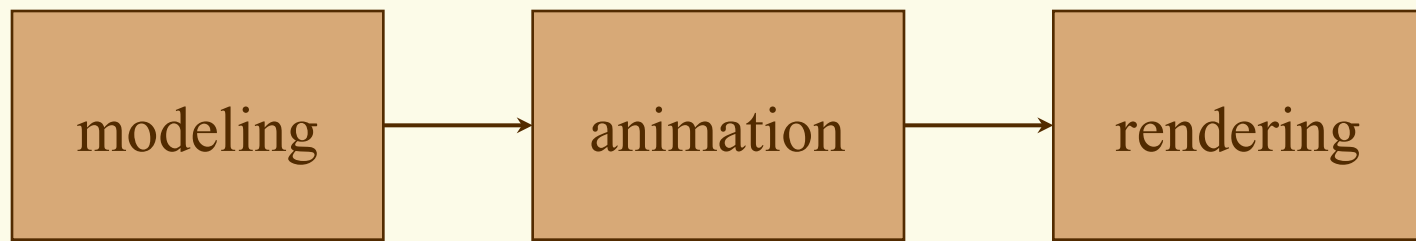
CS 248 - Introduction to Computer Graphics

Autumn quarter, 2008

Slides for December 4 lecture

# The graphics pipeline

---



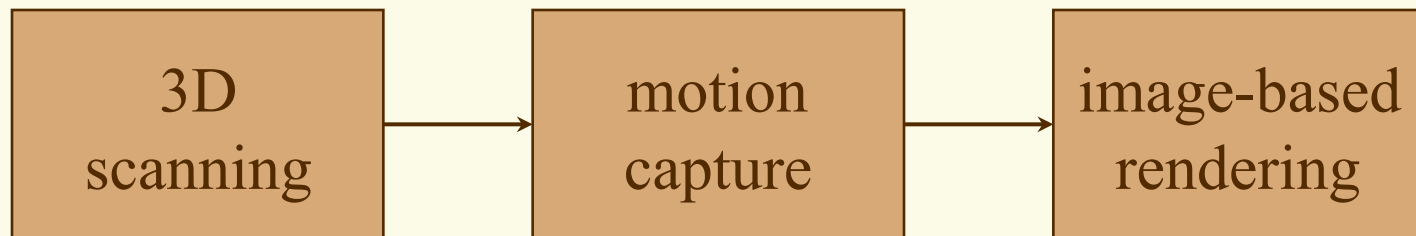
# The graphics pipeline

---

model-based



image-based



# Geometry-based versus image-based rendering

---

**conceptual world**

model  
construction

**geometry**

geometry-based  
rendering

**flythrough of scene**

**real world**

image  
acquisition

**images**

image-based  
rendering

**flythrough of scene**

rendering

computer  
vision

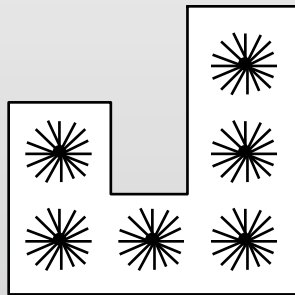


# Apple QuickTime VR

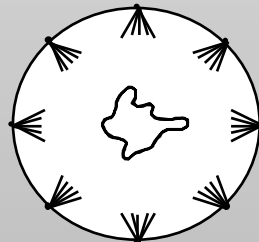
[Chen, Siggraph '95]

---

- outward-looking
  - panoramic views taken at regularly spaced points



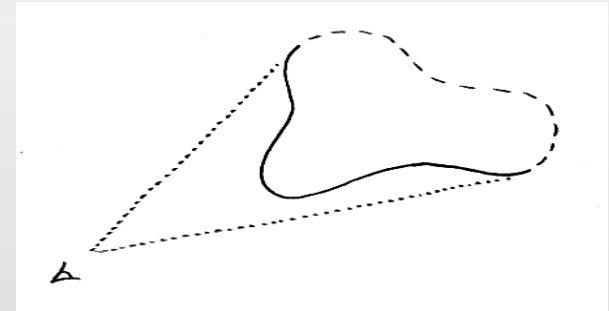
- inward-looking
  - views taken at points on the surface of a sphere



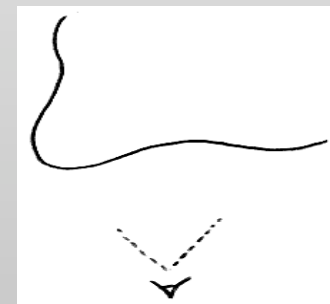
# View interpolation from a single view

---

1. Render object
2. Convert Z-buffer to range image
3. Tessellate to create polygon mesh



4. Re-render from new viewpoint
5. Use depths to resolve overlaps

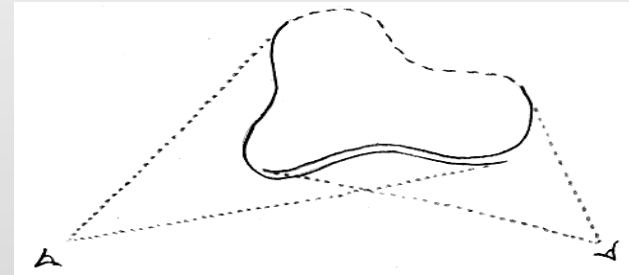


Q. How to fill in holes?

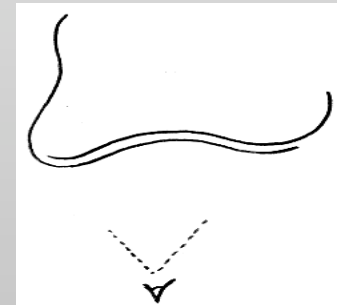
# View interpolation from multiple views

---

1. Render object from multiple viewpoints
2. Convert Z-buffers to range images
3. Tessellate to create multiple meshes

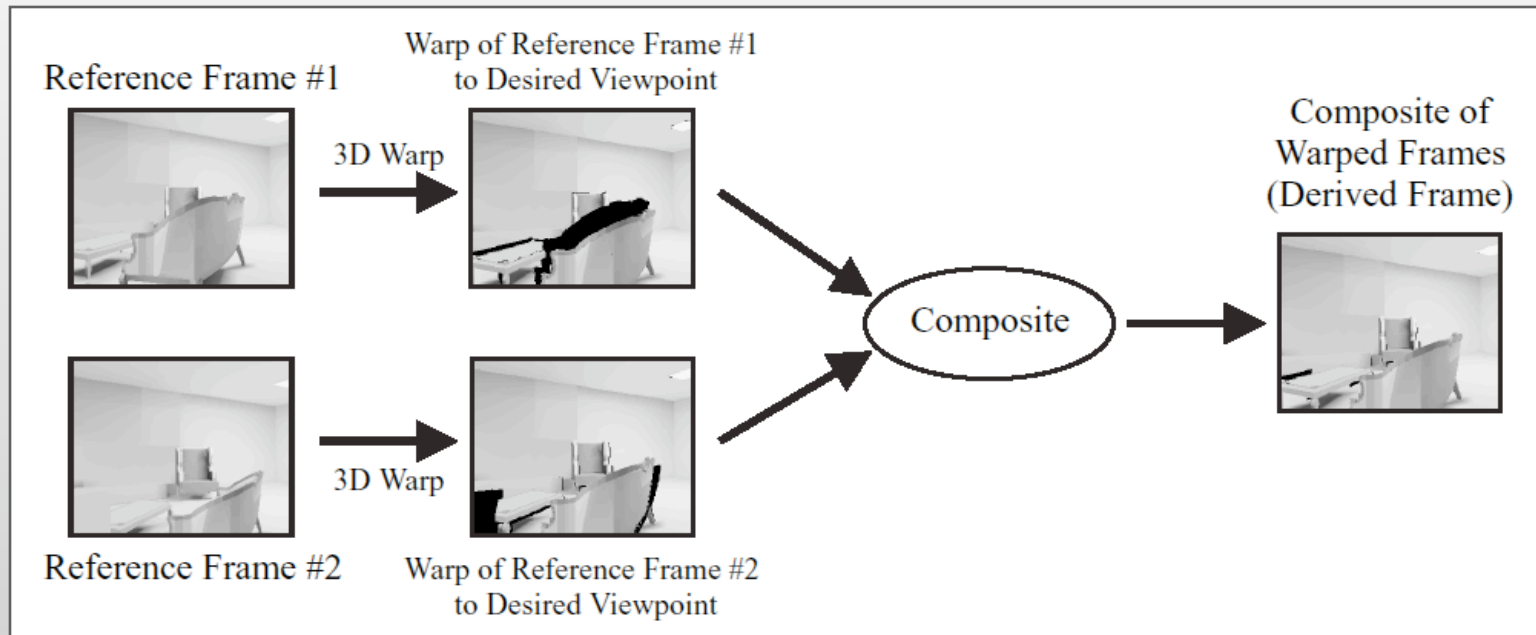


4. Re-render from new viewpoint
5. Use depths to resolve overlaps
6. Use multiple views to fill in holes



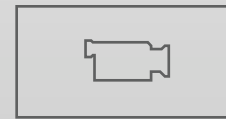
# Post-rendering 3D warping

[Mark et al., I3D97]



- render at low frame rate
- interpolate to real-time frame rate
  - interpolate observer viewpoint using B-Spline
  - convert reference images to polygon meshes
  - warp meshes to interpolated viewpoint
  - composite by Z-buffer comparison and conditional write

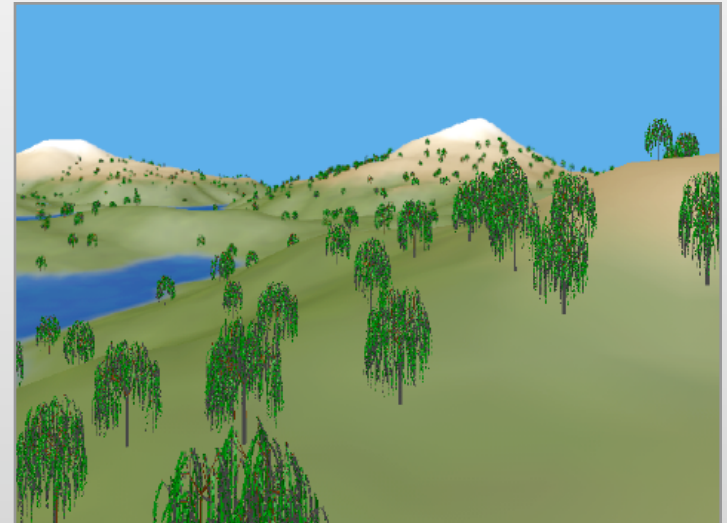
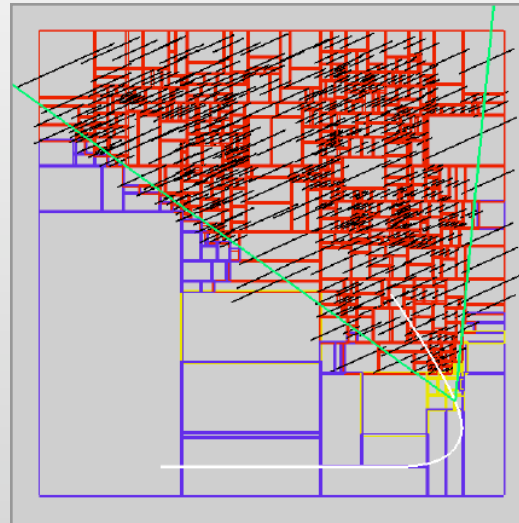
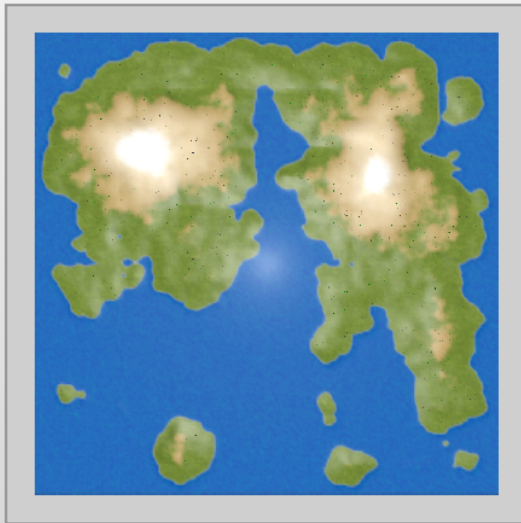
# Results



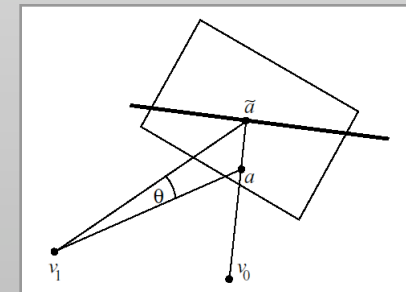
- rendered at 5 fps, interpolated to 30 fps
- live system requires reliable motion prediction
  - tradeoff between accuracy and latency
- fails on specular objects

# Image caching

[Shade et al., SIGGRAPH 1996]



- precompute BSP tree of scene (2D in this case)
- for first observer position
  - draw nearby nodes (yellow) as geometry
  - render distant nodes (red) to RGB $\alpha$  images (black)
  - composite images together
- as observer moves
  - if disparity exceeds a threshold, rerender image



# Why didn't IBR take over the world?

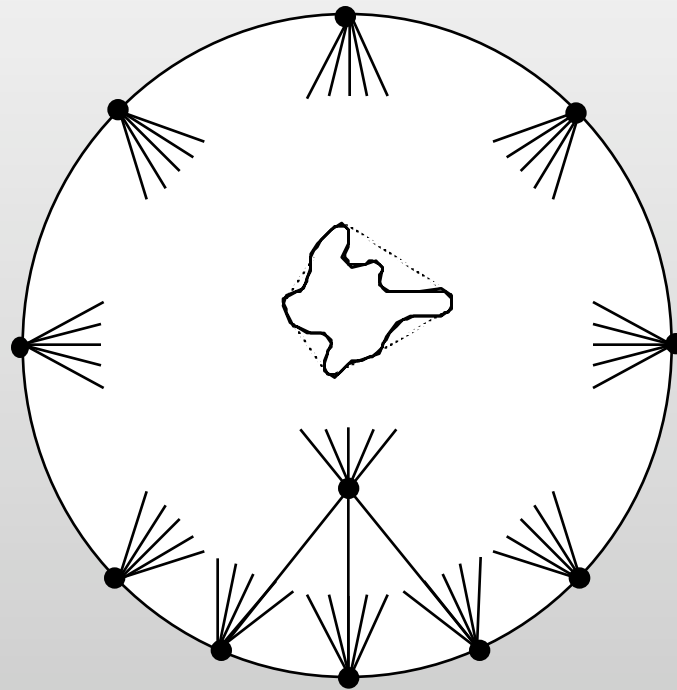
---

- warping and rendering range images is slow
  - pixel-sized triangles are inefficient
  - just as many pixels need to be touched as in normal rendering
- arms race against improvements in 3D rendering
  - level of detail (LOD)
  - culling techniques
  - hierarchical Z-buffer
  - etc.
- visual artifacts are objectionable
  - not small and homogeneous like 3D rendering artifacts

# Light field rendering

[Levoy & Hanrahan, SIGGRAPH 1996]

---



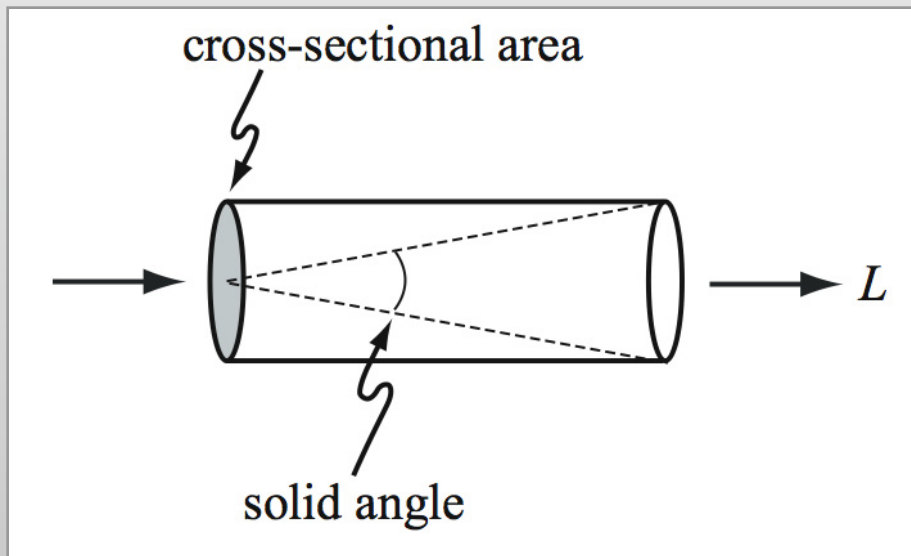
- must stay outside convex hull of the object
- like rebinning in computed tomography



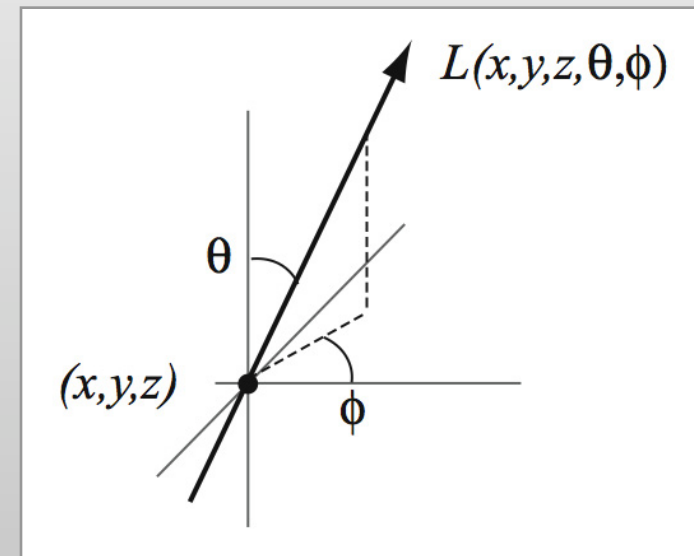
# The scalar light field (in geometrical optics)

---

*Radiance as a function of position and direction  
in a static scene with fixed illumination*



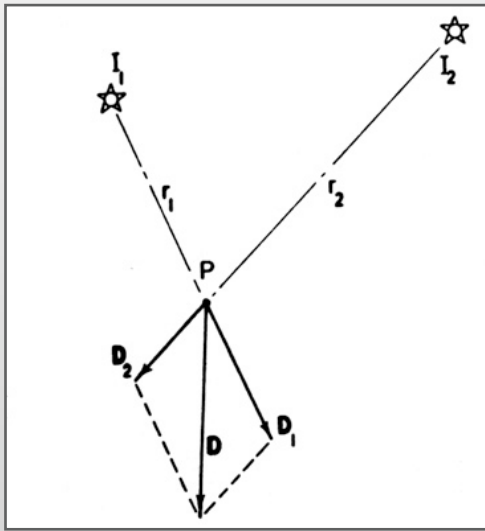
$L$  is radiance in watts / ( $\text{m}^2$  steradians)



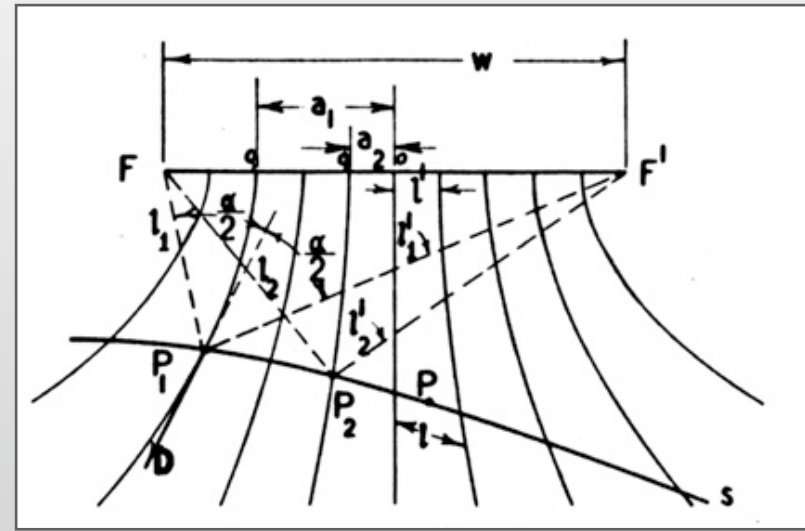
5-dimensional function

# The vector light field

[Gershun 1936]



adding two light vectors

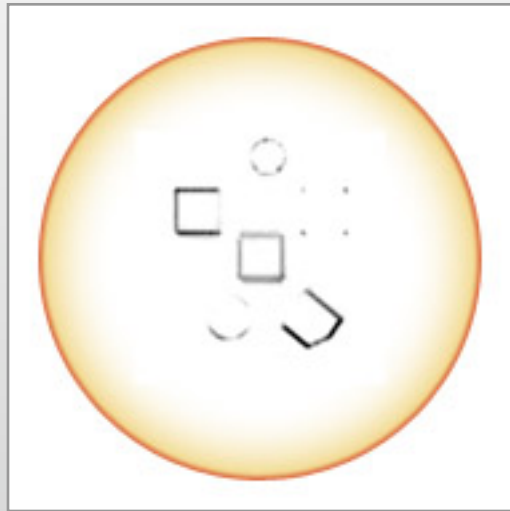


the vector light field  
produced by a luminous strip

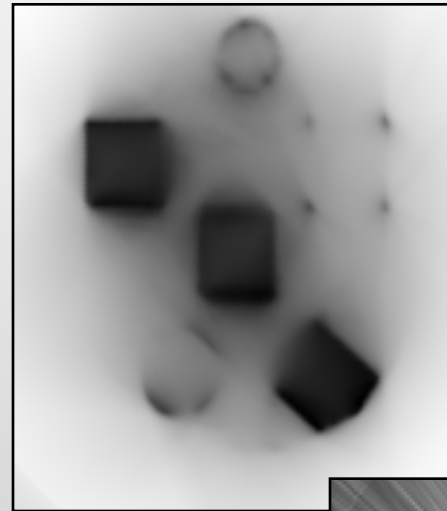
- amplitude gives irradiance at that point
- direction tells which way to orient a surface for maximum brightness under uniform illumination

# Visualizing the vector irradiance field

---

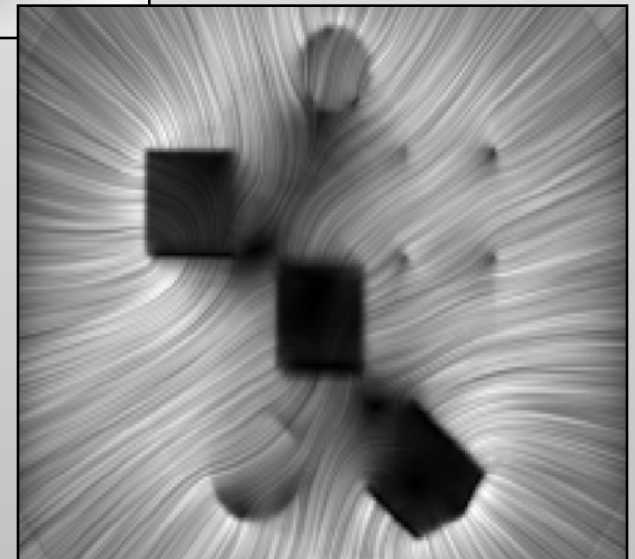


flatland scene with  
partially opaque blockers  
under uniform illumination



scalar irradiance  
at each point

vector directions, visualized using  
line integral convolution (LIC) [Cabral 1993]



# Dimensionality of the scalar light field

---

- for general scenes

⇒ 5D function

“plenoptic function”

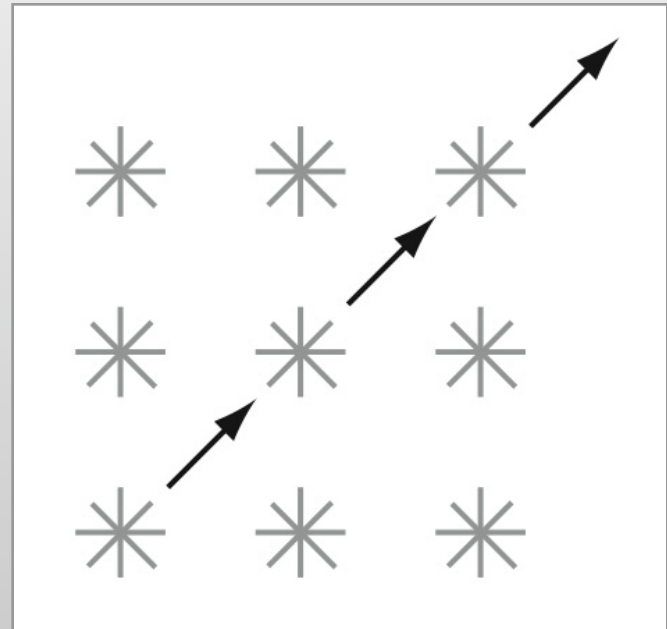
$L(x, y, z, \theta, \phi)$

- in free space

⇒ 4D function

“the (scalar) light field”

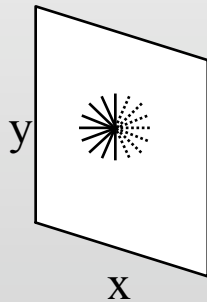
$L(?)$



# Some candidate parameterizations for the 4D light field

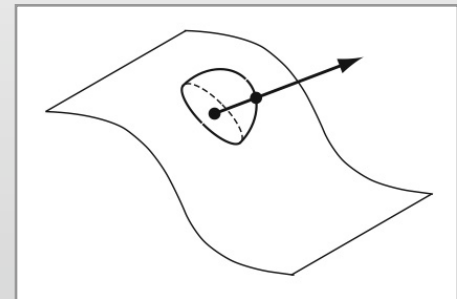
---

Point-on-plane + direction  
(or point-on-surface + direction)



$$L(x, y, \theta, \phi)$$

or



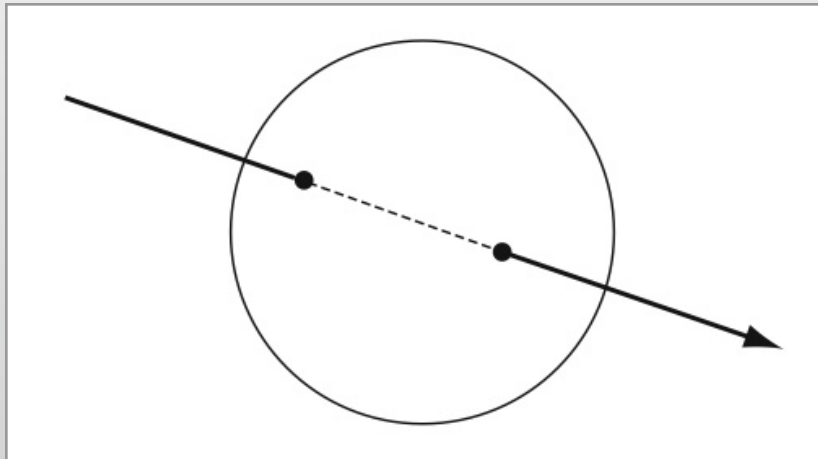
$$L(u, v, \theta, \phi)$$

- convenient for measuring BSSRDFs
- photon buffer for global illumination

# More parameterizations

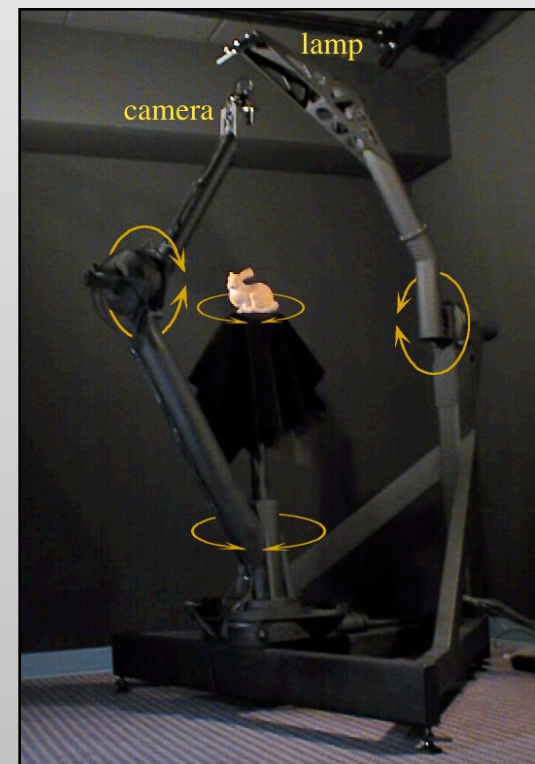
---

## Chords of a sphere



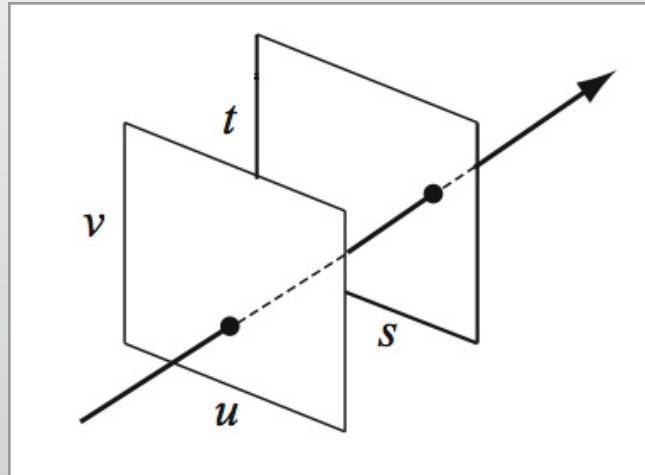
$$L(\theta_1, \phi_1, \theta_2, \phi_2)$$

- convenient for spherical gantry
- facilitates uniform sampling



---

## Two planes (“light slab”)

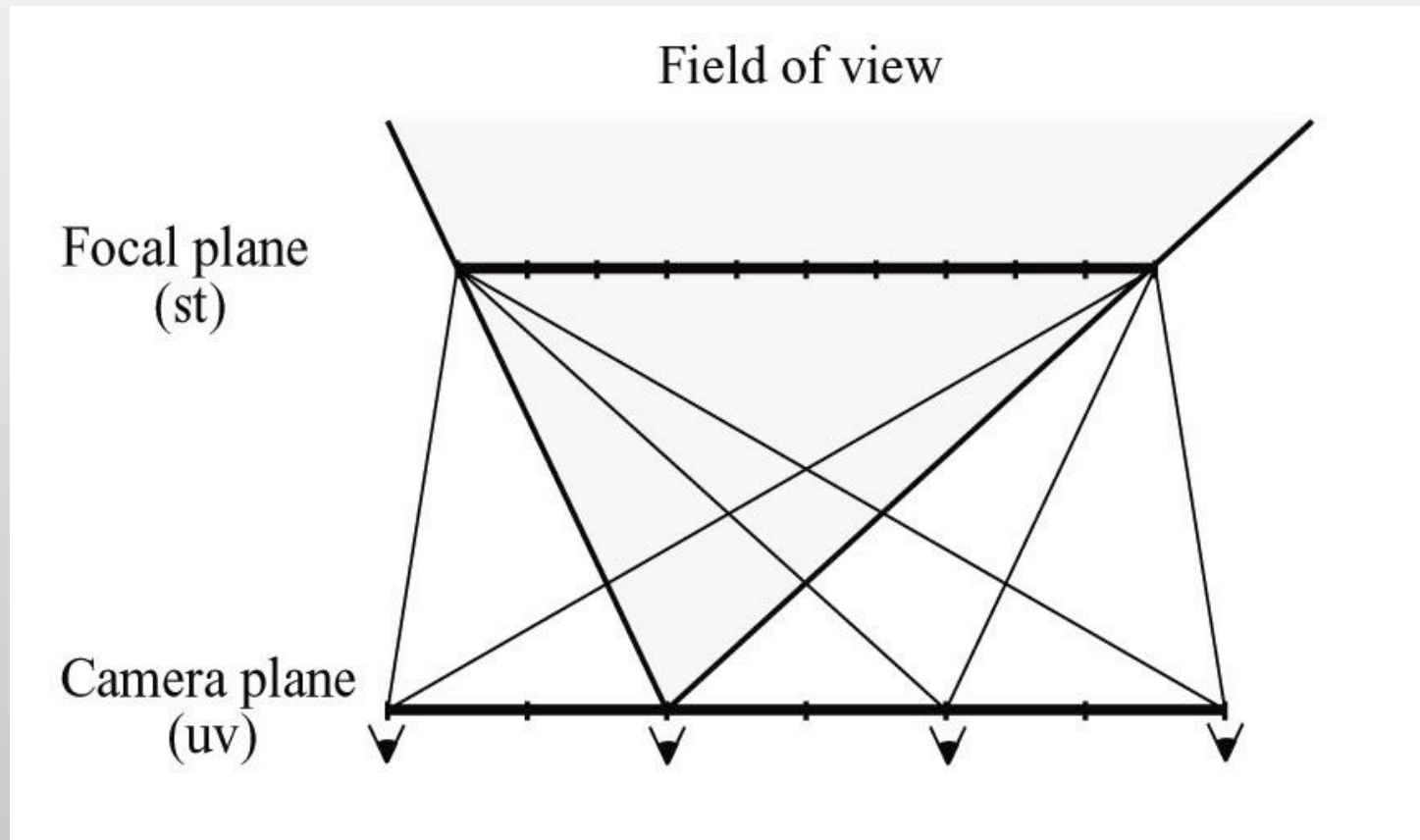


$$L(u, v, s, t)$$

- uses projective geometry
  - one plane at infinity  $\Rightarrow$  array of orthographic images
  - fast incremental display algorithms

# Creating a light field

---

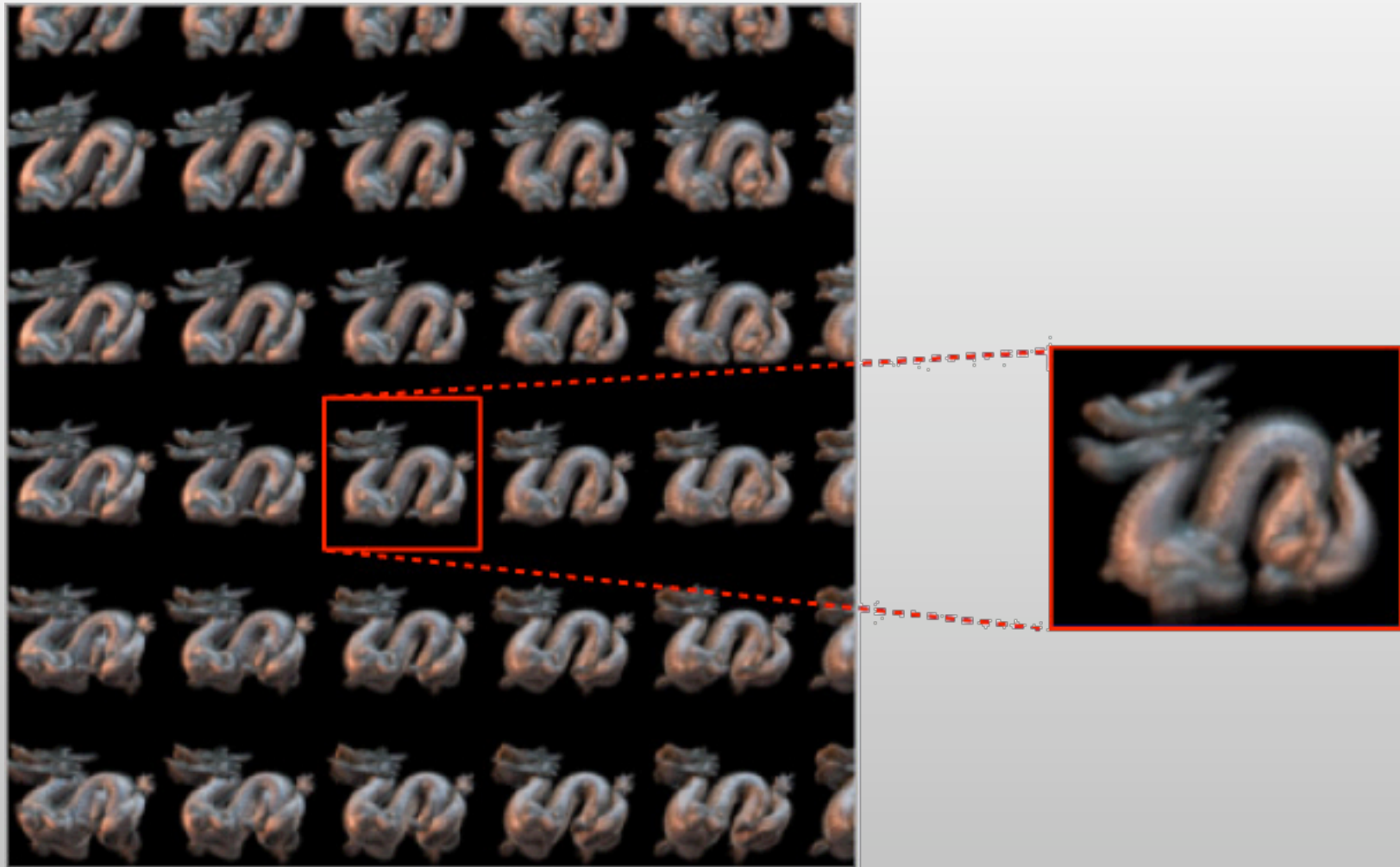


- off-axis (sheared) perspective views

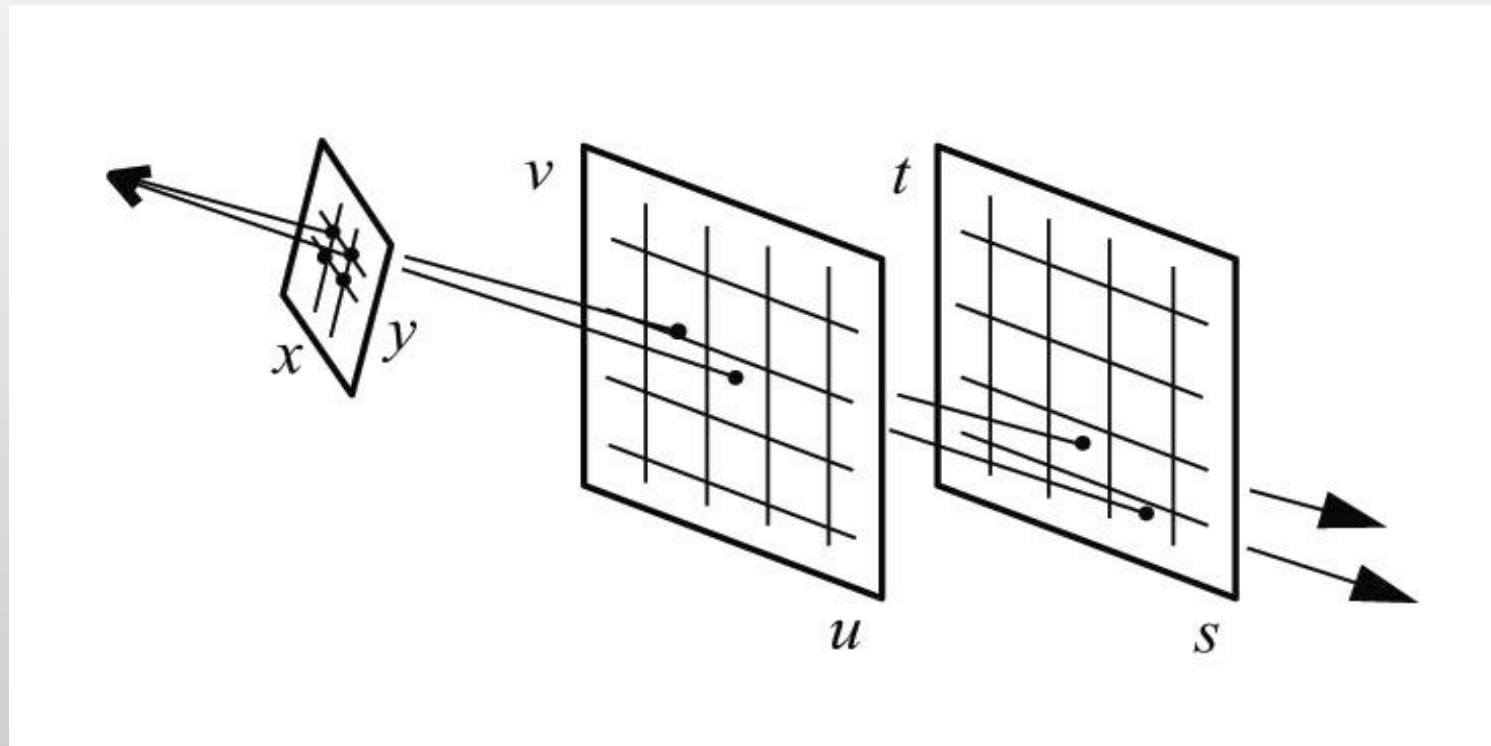


# A light field is an array of images

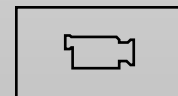
---



# Displaying a light field



```
foreach  $x, y$   
  compute  $u, v, s, t$   
   $I(x, y) = L(u, v, s, t)$ 
```

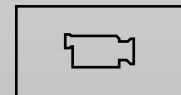


# Capturing light fields using an array of cameras

---



Manex's bullet time array



A spiral-bound notebook with a light brown, textured cover and a dark brown spine on the left. The notebook is open to a blank page with a light beige, textured background. The text is centered on the page.

# Computational photography

CS 248 - Introduction to Computer Graphics

Autumn quarter, 2008

Slides for December 4 lecture

# Some (tentative) definitions

---

- *computational imaging*
  - any image formation method that requires a digital computer
  - first used in medical imaging and remote sensing
- *computational photography*
  - computational imaging techniques that enhance or extend the capabilities of digital photography
  - output is an ordinary photograph, but one that could not have been taken by a traditional camera

# 19 papers on computational photography at SIGGRAPH 2007

---

- Image Analysis & Enhancement
  - Image Deblurring with Blurred/Noisy Image Pairs
  - Photo Clip Art
  - Scene Completion Using Millions of Photographs
- Image Slicing & Stretching
  - Soft Scissors: An Interactive Tool for Realtime High Quality Matting
  - Seam Carving for Content-Aware Image Resizing
- Light Field & High-Dynamic-Range Imaging
  - Veiling Glare in High-Dynamic-Range Imaging
  - Ldr2Hdr: On-the-Fly Reverse Tone Mapping of Legacy Video and Photographs
- Appearance Capture & Editing
  - Multiscale Shape and Detail Enhancement from Multi-light Image Collections
- Computational Cameras
  - Active Refocusing of Images and Videos
  - Multi-Aperture Photography
  - Dappled Photography: Mask-Enhanced Cameras for Heterodyned Light Fields and Coded Aperture Refocusing
  - Image and Depth from a Conventional Camera with a Coded Aperture
- Big Images
  - Capturing and Viewing Gigapixel Images
  - Efficient Gradient-Domain Compositing Using Quadrees
  - Image Upsampling via Imposed Edge Statistics
  - Joint Bilateral Upsampling
- Video Processing
  - Factored Time-Lapse Video
  - Computational Time-Lapse Video
  - Real-Time Edge-Aware Image Processing With the Bilateral Grid



# Computational Photography

Film-like  
Photography  
with bits

Digital  
Photography

Image processing applied to captured images to produce better images.

Examples:  
Interpolation, Filtering, Enhancement, Dynamic Range Compression, Color Management, Morphing, Hole Filling, Artistic Image Effects, Image Compression, Watermarking.



Computational Camera

Computational Processing

Processing of a set of captured images to create new images.

Examples:  
Mosaicing, Matting, Super-Resolution, Multi-Exposure HDR, Light Field from Multiple View, Structure from Motion, Shape from X.

Computational Imaging/Optics

Capture of optically coded images and computational decoding to produce new images.

Examples:  
Coded Aperture, Optical Tomography, Diaphanography, SA Microscopy, Integral Imaging, Assorted Pixels, Catadioptric Imaging, Holographic Imaging.

Computational Sensor

Detectors that combine sensing and processing to create smart pixels.

Examples:  
Artificial Retina, Retinex Sensors, Adaptive Dynamic Range Sensors, Edge Detect Chips, Focus of Expansion Chips, Motion Sensors.

Smart Light

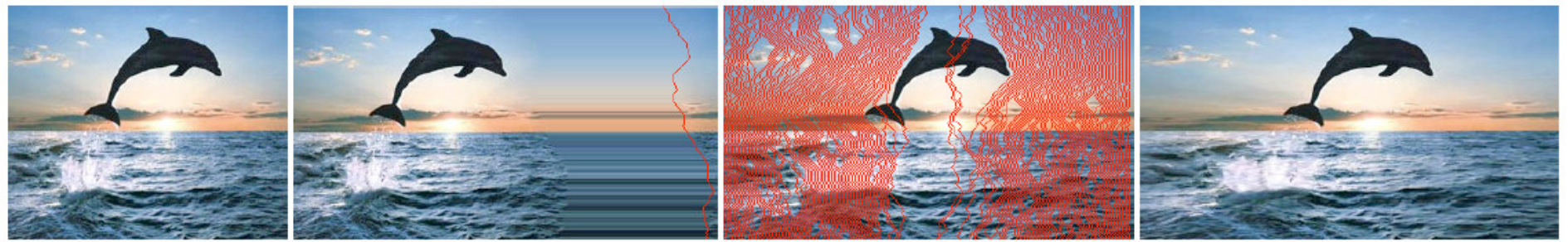
Computational Illumination

Adapting and Controlling Illumination to Create revealing image

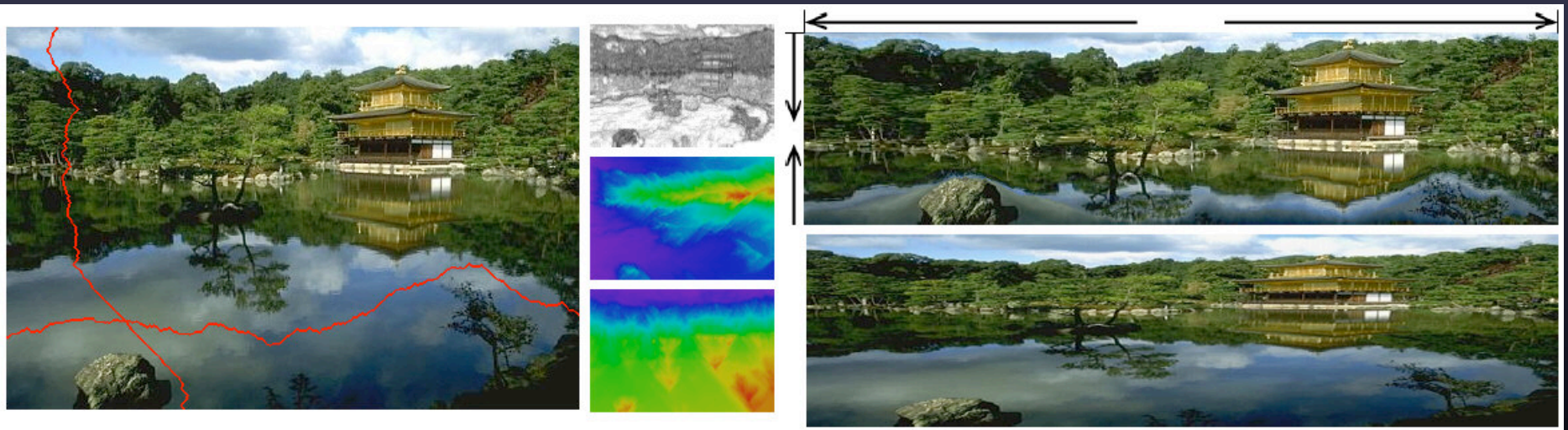
Examples:  
Flash/no flash, Lighting domes, Multi-flash for depth edges, Dual Photos, Polynomial texture Maps, 4D light source

# Content-aware image resizing

[Avidan SIGGRAPH 2007]



- to expand: insert pixels along seams that, if removed in order, would yield the original image





# Content-aware image resizing

[Avidan SIGGRAPH 2007]

- to compress: remove pixels along lowest-energy seams, ordered using dynamic programming
- to expand: insert pixels in order
- application to object removal



**NOW AVAILABLE IN  
PHOTOSHOP CS4 !!**

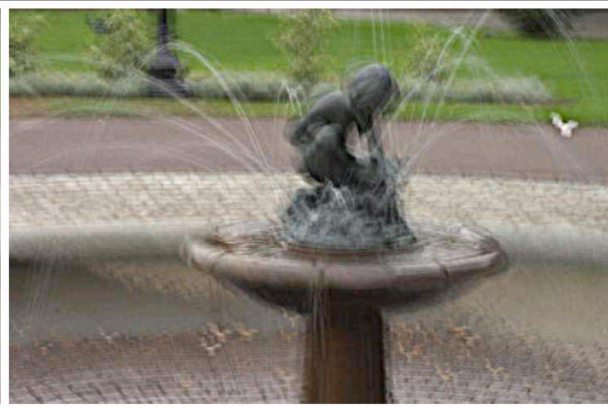
# Removing camera shake

[Fergus SIGGRAPH 2006]

---



image with camera shake



Photoshop Unsharp Mask



deconvolution



blur kernel

# Computational Photography

Film-like  
Photography  
with bits

Digital  
Photography

Image processing applied to captured images to produce better images.

Examples:  
Interpolation, Filtering, Enhancement, Dynamic Range Compression, Color Management, Morphing, Hole Filling, Artistic Image Effects, Image Compression, Watermarking.

Computational Camera

Computational Processing

Processing of a set of captured images to create new images.

Examples:  
Mosaicing, Matting, Super-Resolution, Multi-Exposure HDR, Light Field from Multiple View, Structure from Motion, Shape from X.

Computational Imaging/Optics

Capture of optically coded images and computational decoding to produce new images.

Examples:  
Coded Aperture, Optical Tomography, Diaphanography, SA Microscopy, Integral Imaging, Assorted Pixels, Catadioptric Imaging, Holographic Imaging.

Computational Sensor

Detectors that combine sensing and processing to create smart pixels.

Examples:  
Artificial Retina, Retinex Sensors, Adaptive Dynamic Range Sensors, Edge Detect Chips, Focus of Expansion Chips, Motion Sensors.

Smart Light

Computational Illumination

Adapting and Controlling Illumination to Create revealing image

Examples:  
Flash/no flash, Lighting domes, Multi-flash for depth edges, Dual Photos, Polynomial texture Maps, 4D light source





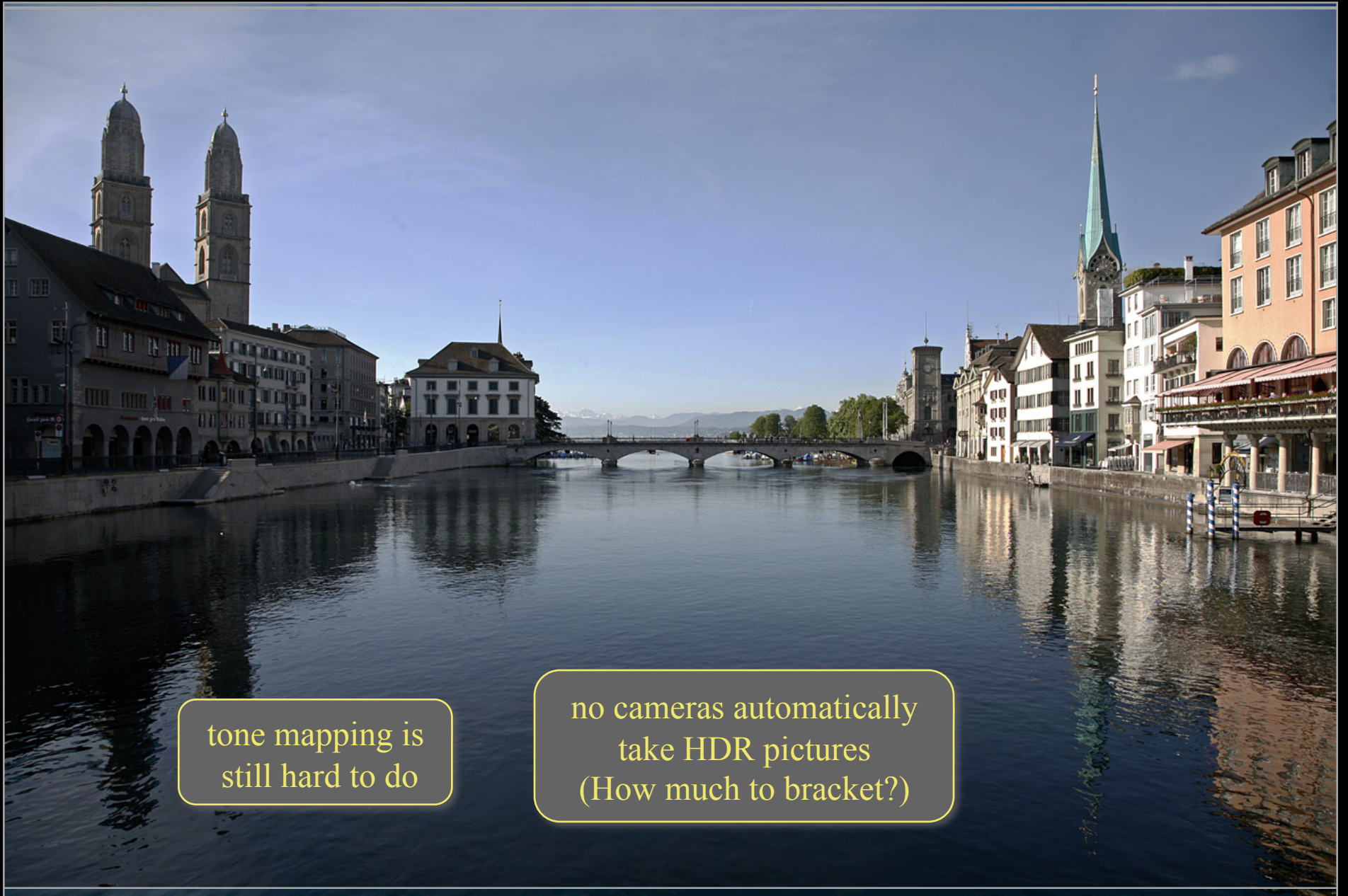
# Gigapixel mosaicing

[xyrez.com]

---





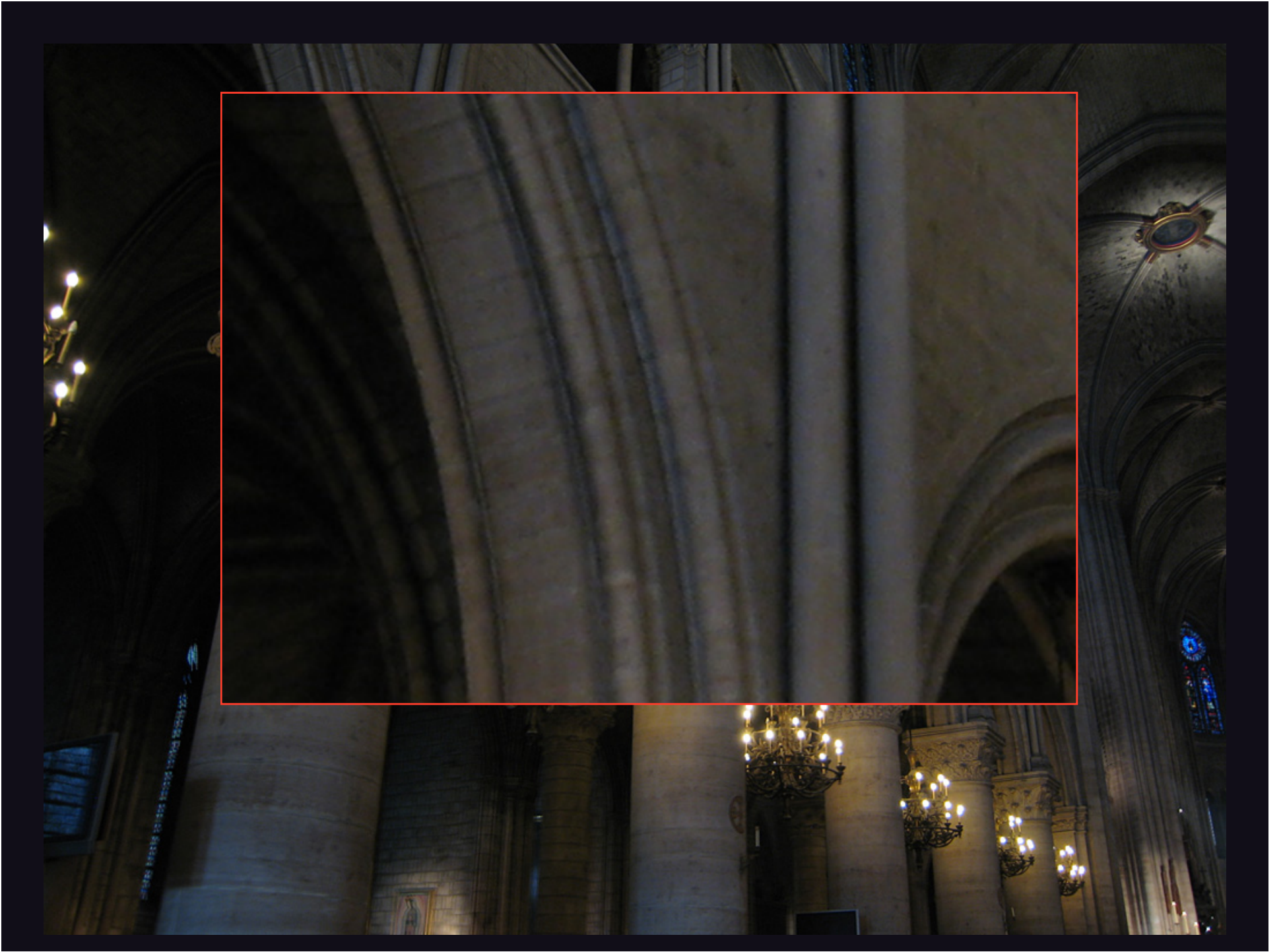


tone mapping is  
still hard to do

no cameras automatically  
take HDR pictures  
(How much to bracket?)









# Aligning a burst of short-exposure, high-ISO shots using the Casio EX-F1

1/3 sec

burst  
at 60fps





# Aligning on a foreground object using the Casio EX-F1

---



# All-focus algorithms

[Agarwala 2004]

---



**NOW AVAILABLE IN  
PHOTOSHOP CS4 !!**



# Digital Photomontage

[Agarwala SIGGRAPH 2004]

---



- multi-shot images
  - shoot until everybody has smiled at least once

# Digital Photomontage

[Agarwala SIGGRAPH 2004]

---



– segment



# Digital Photomontage

[Agarwala SIGGRAPH 2004]

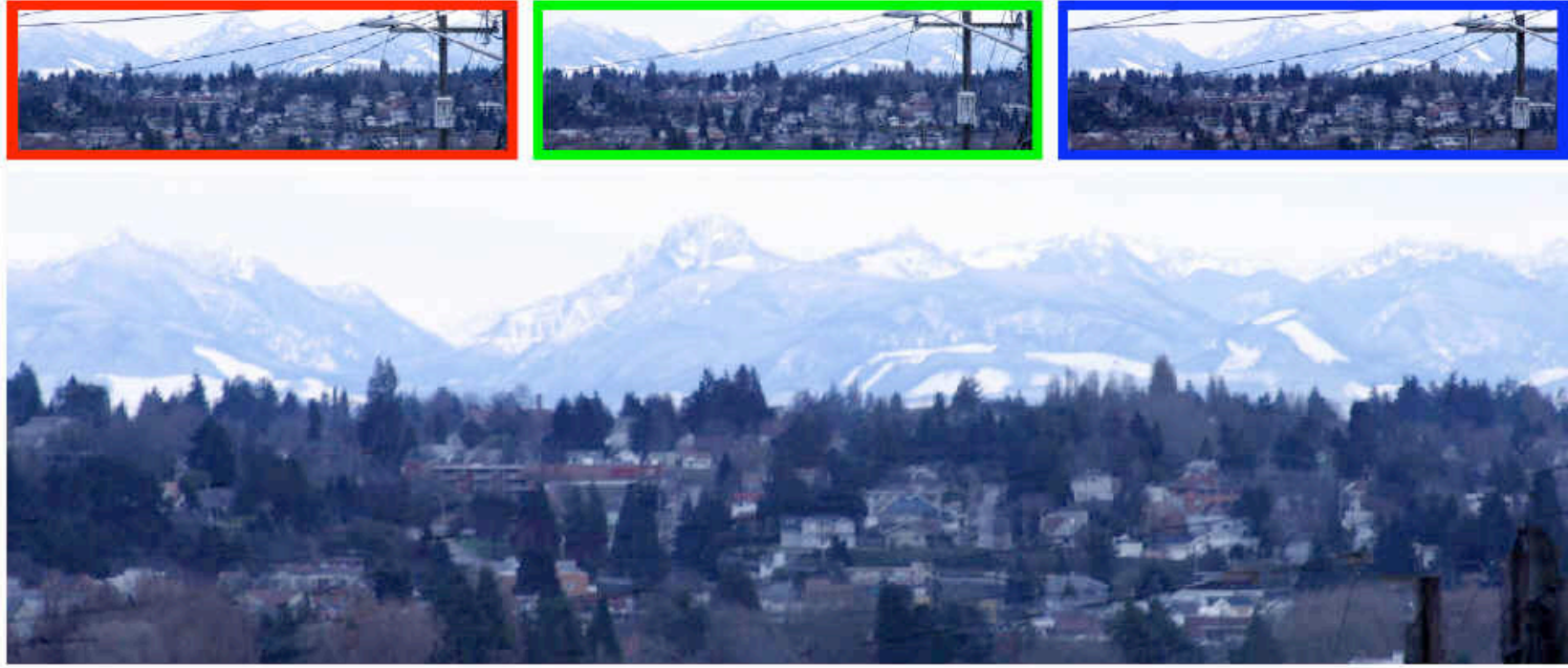
---



– assemble

# Digital Photomontage

[Agarwala SIGGRAPH 2004]



- remove foreground objects that don't appear in all shots as the camera translates (based on median filter)



# Removing camera shake (again)

---

- deconvolve long-exposure (blurred) image, using short-exposure (noisy) image as prior [Yuan SIGGRAPH 2007]



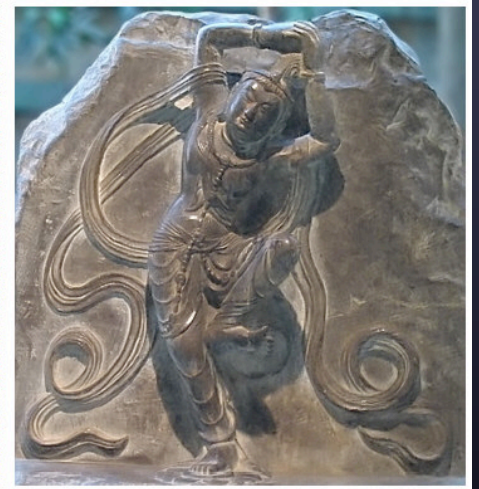
long exposure  
(blurry)



short exposure  
(dark)



same, scaled up  
(noisy)



joint deconvolution

# Computational Photography

Film-like  
Photography  
with bits

Digital  
Photography

Image processing applied to captured images to produce better images.

Examples:  
Interpolation, Filtering, Enhancement, Dynamic Range Compression, Color Management, Morphing, Hole Filling, Artistic Image Effects, Image Compression, Watermarking.

Computational Camera

Computational Processing

Processing of a set of captured images to create new images.

Examples:  
Mosaicing, Matting, Super-Resolution, Multi-Exposure HDR, Light Field from Multiple View, Structure from Motion, Shape from X.

Computational Imaging/Optics

Capture of optically coded images and computational decoding to produce new images.

Examples:  
Coded Aperture, Optical Tomography, Diaphanography, SA Microscopy, Integral Imaging, Assorted Pixels, Catadioptric Imaging, Holographic Imaging.

Computational Sensor

Detectors that combine sensing and processing to create smart pixels.

Examples:  
Artificial Retina, Retinex Sensors, Adaptive Dynamic Range Sensors, Edge Detect Chips, Focus of Expansion Chips, Motion Sensors.

Smart Light

Computational Illumination

Adapting and Controlling Illumination to Create revealing image

Examples:  
Flash/no flash, Lighting domes, Multi-flash for depth edges, Dual Photos, Polynomial texture Maps, 4D light source

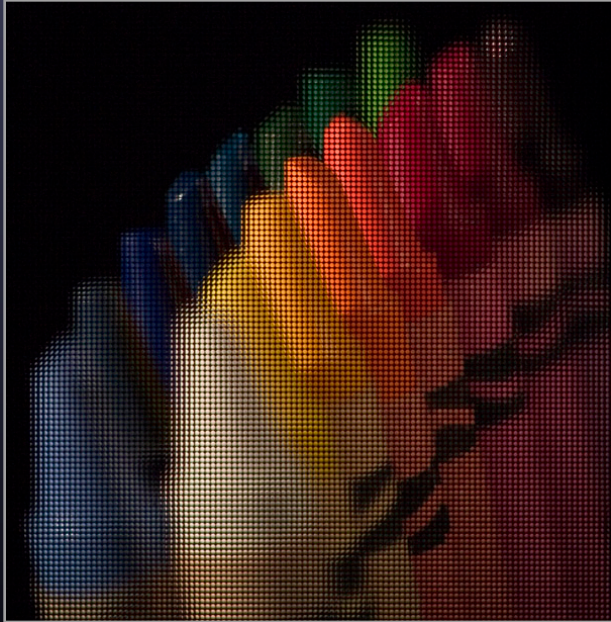




# Light field photography using a handheld plenoptic camera

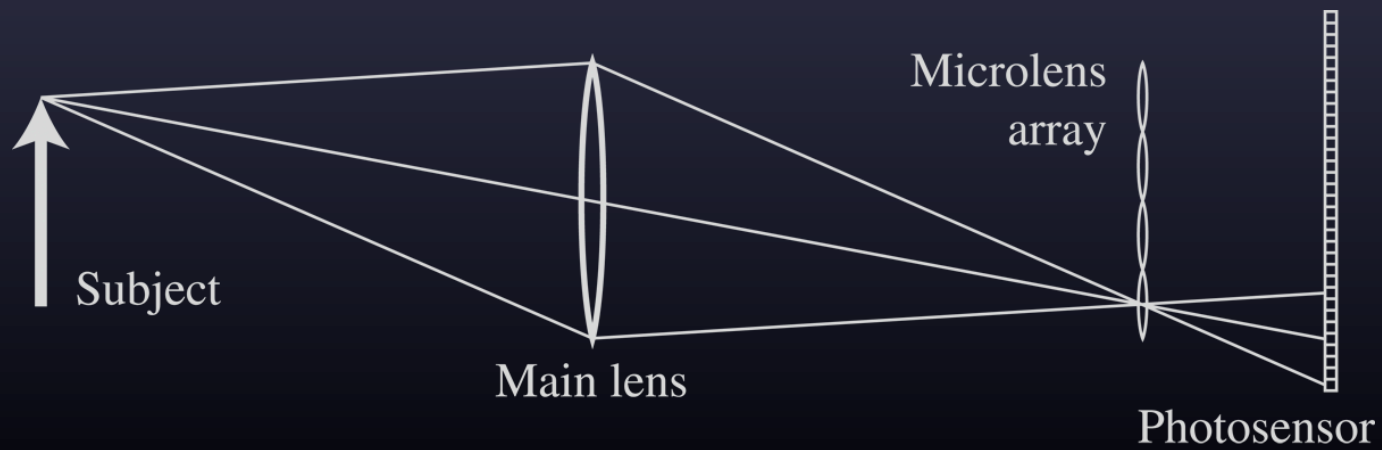
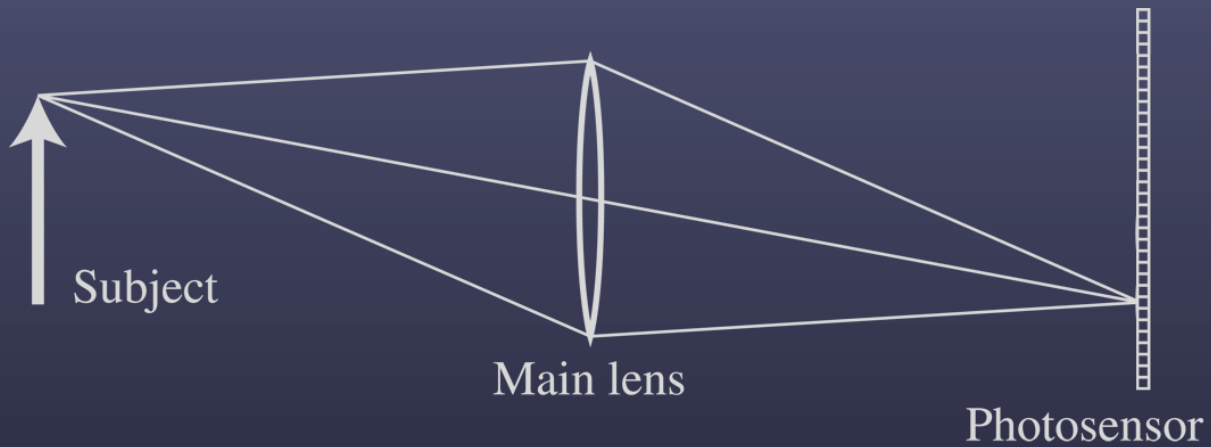
*Ren Ng, Marc Levoy, Mathieu Brédif,  
Gene Duval, Mark Horowitz and Pat Hanrahan*

*(Proc. SIGGRAPH 2005  
and TR 2005-02)*



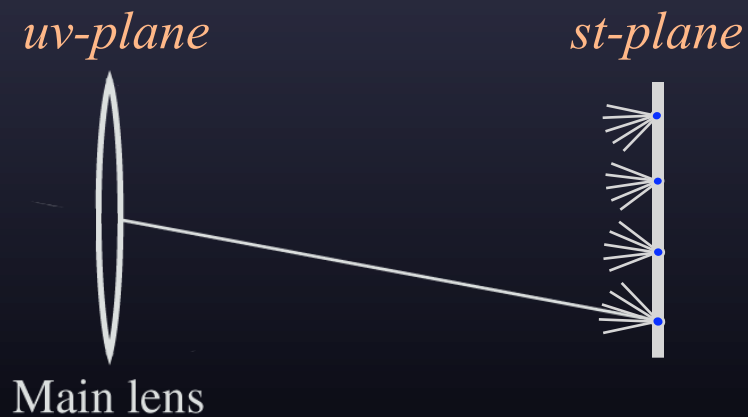
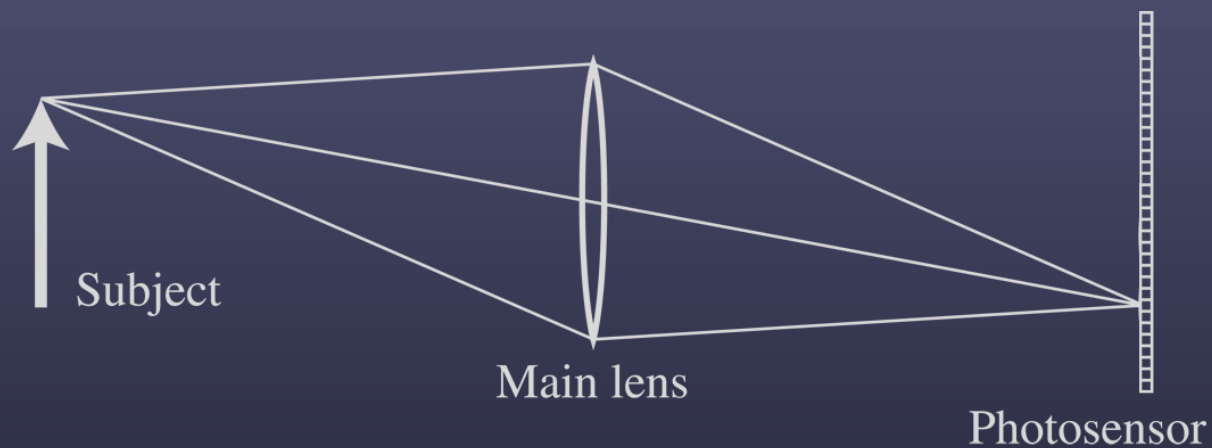
# Conventional versus light field camera

---



# Conventional versus light field camera

---



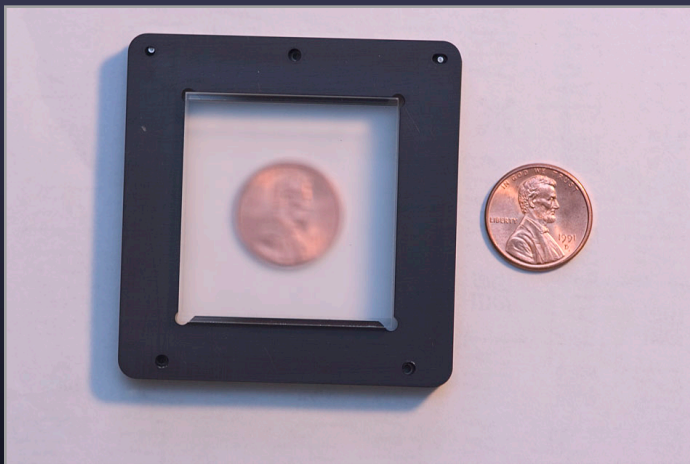
# Prototype camera



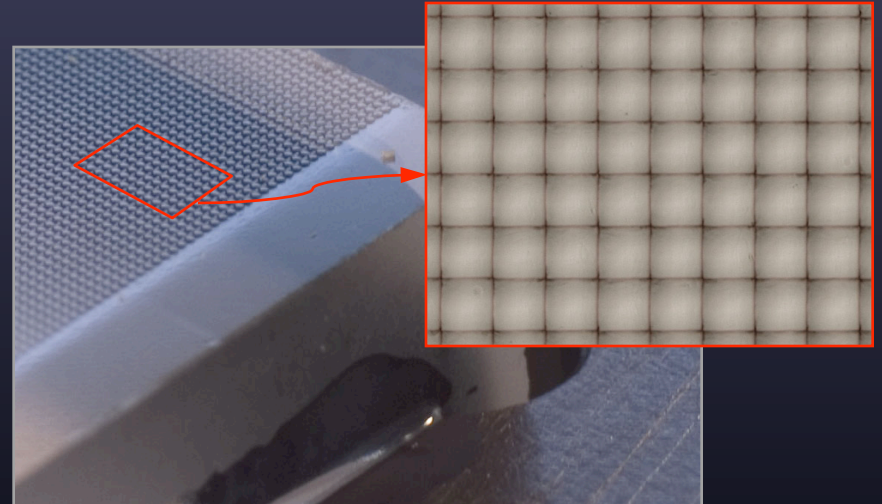
Contax medium format camera



Kodak 16-megapixel sensor



Adaptive Optics microlens array



125 $\mu$  square-sided microlenses

$$4000 \times 4000 \text{ pixels} \div 292 \times 292 \text{ lenses} = 14 \times 14 \text{ pixels per lens}$$

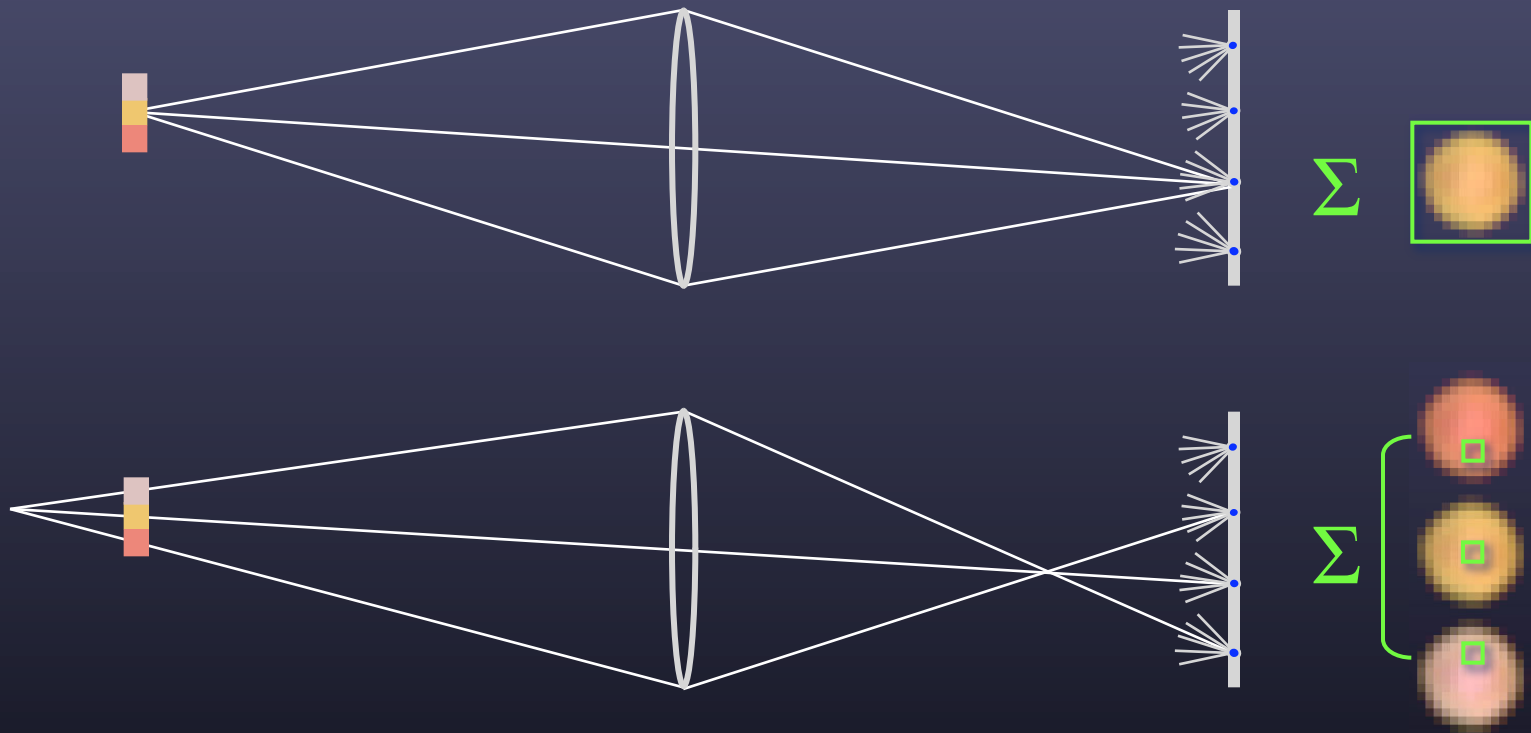






# Digital refocusing

---



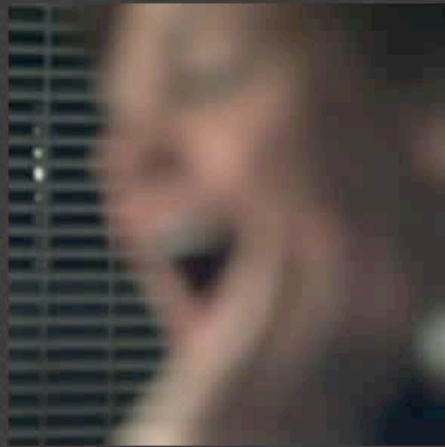
# Example of digital refocusing

---



# Refocusing portraits

---



# Computational Photography

Film-like  
Photography  
with bits

Digital  
Photography

Image processing applied to captured images to produce better images.

Examples:  
Interpolation, Filtering, Enhancement, Dynamic Range Compression, Color Management, Morphing, Hole Filling, Artistic Image Effects, Image Compression, Watermarking.

Computational Camera

Computational Processing

Processing of a set of captured images to create new images.

Examples:  
Mosaicing, Matting, Super-Resolution, Multi-Exposure HDR, Light Field from Multiple View, Structure from Motion, Shape from X.

Computational Imaging/Optics

Capture of optically coded images and computational decoding to produce new images.

Examples:  
Coded Aperture, Optical Tomography, Diaphanography, SA Microscopy, Integral Imaging, Assorted Pixels, Catadioptric Imaging, Holographic Imaging.

Computational Sensor

Detectors that combine sensing and processing to create smart pixels.

Examples:  
Artificial Retina, Retinex Sensors, Adaptive Dynamic Range Sensors, Edge Detect Chips, Focus of Expansion Chips, Motion Sensors.



Smart Light

Computational Illumination

Adapting and Controlling Illumination to Create revealing image

Examples:  
Flash/no flash, Lighting domes, Multi-flash for depth edges, Dual Photos, Polynomial texture Maps, 4D light source

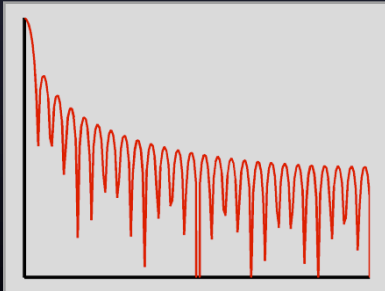
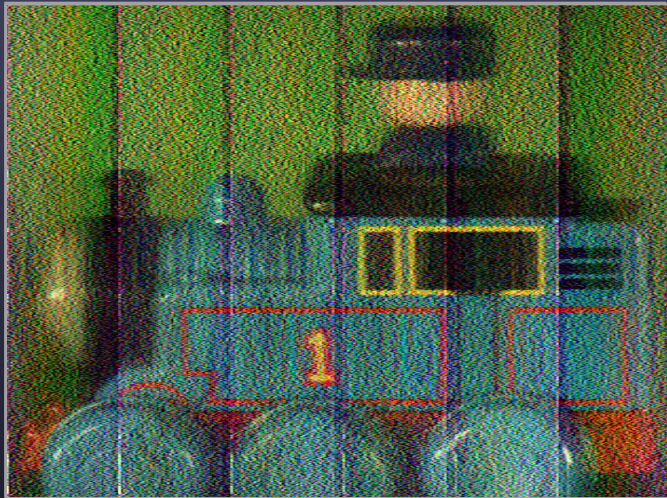


# Coded-exposure photography

[Raskar SIGGRAPH 2006]

---

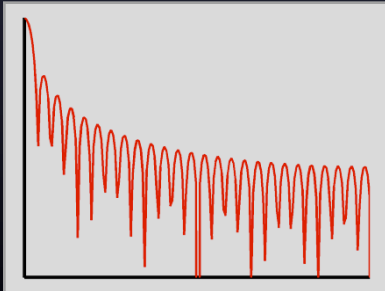
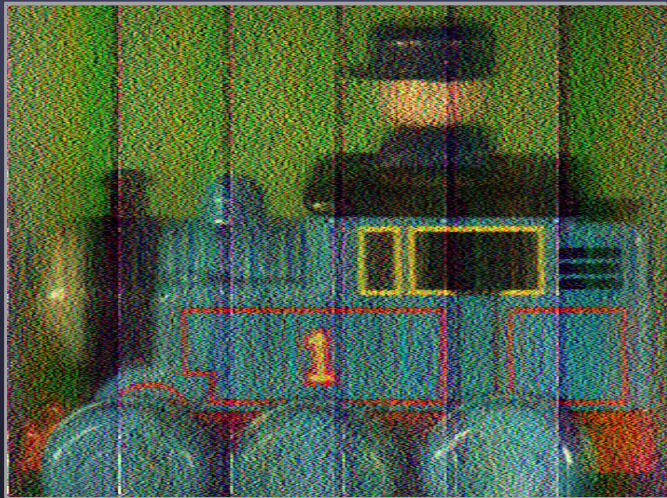
continuous shutter



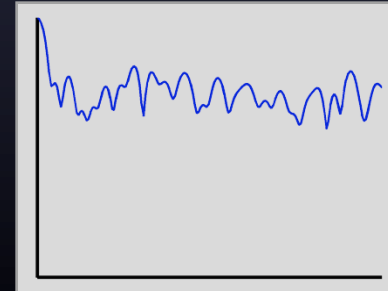
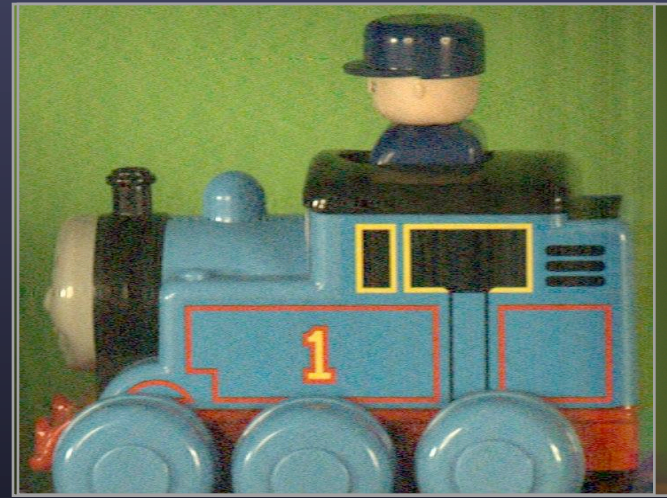
# Coded-exposure photography

[Raskar SIGGRAPH 2006]

continuous shutter



fluttered shutter











# Computational Photography

Film-like  
Photography  
with bits

Digital  
Photography

Image processing applied to captured images to produce better images.

Examples:  
Interpolation, Filtering, Enhancement, Dynamic Range Compression, Color Management, Morphing, Hole Filling, Artistic Image Effects, Image Compression, Watermarking.

Computational Camera

Computational Processing

Processing of a set of captured images to create new images.

Examples:  
Mosaicing, Matting, Super-Resolution, Multi-Exposure HDR, Light Field from Multiple View, Structure from Motion, Shape from X.

Computational Imaging/Optics

Capture of optically coded images and computational decoding to produce new images.

Examples:  
Coded Aperture, Optical Tomography, Diaphanography, SA Microscopy, Integral Imaging, Assorted Pixels, Catadioptric Imaging, Holographic Imaging.

Computational Sensor

Detectors that combine sensing and processing to create smart pixels.

Examples:  
Artificial Retina, Retinex Sensors, Adaptive Dynamic Range Sensors, Edge Detect Chips, Focus of Expansion Chips, Motion Sensors.

Smart Light

Computational Illumination

Adapting and Controlling Illumination to Create revealing image

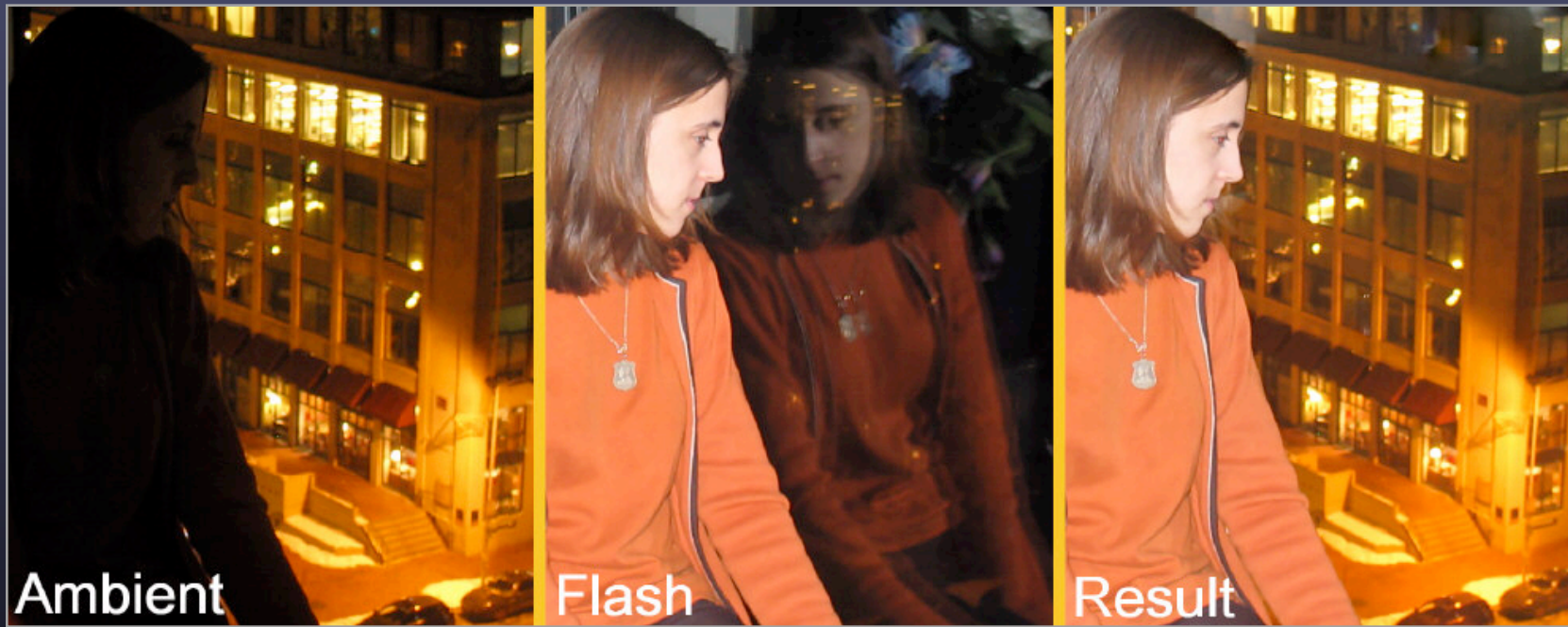
Examples:  
Flash/no flash, Lighting domes, Multi-flash for depth edges, Dual Photos, Polynomial texture Maps, 4D light source

[Nayar, Tumblin]



# Flash-noflash photography

[Agrawal SIGGRAPH 2005]



- compute ambient + flash – features in sum that don't appear in ambient alone (as determined from image gradients) (except where ambient image is nearly black)

# Large online photo collections

---

- Facebook
  - 3 billion photos
- Flickr
  - 9 billion photos
- Google Library Project
  - 50 million books  $\times$  300 pages = 15 billion images
- Google Earth
- Google StreetView Project
  - formerly the Stanford CityBlock Project

# Computational photography using online photograph collections

---

- scene completion
- texture synthesis
- image-based image search



# Scene completion using millions of photographs

[Hays & Efros SIGGRAPH 2007]

---

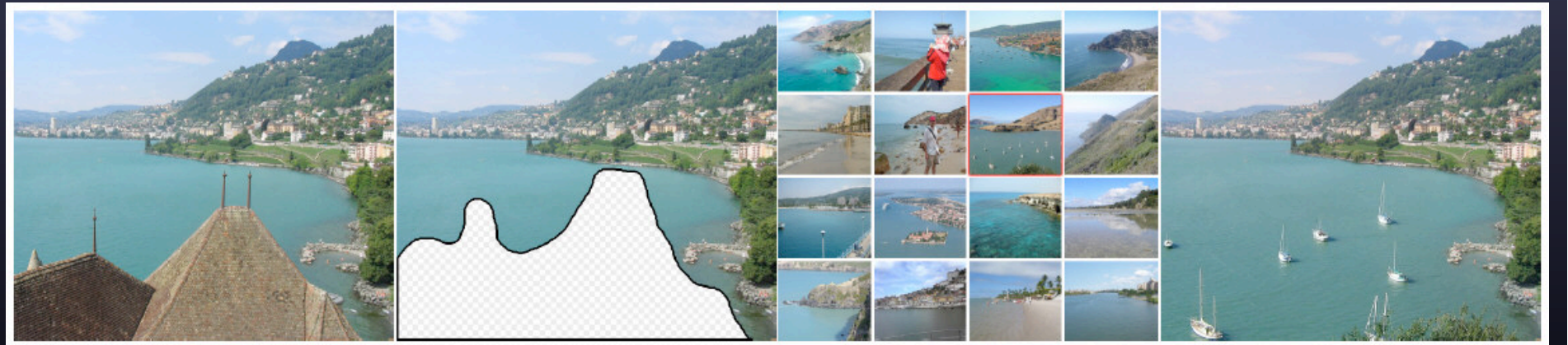
- search for matches from a large database
- Find least visible seams using graph-cut algorithm
- blend gradients & integrate to create image



# Scene completion using millions of photographs

[Hays & Efros SIGGRAPH 2007]

---





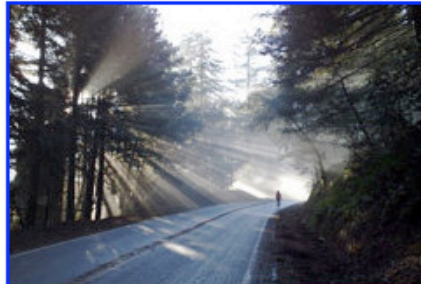
# CS 448 – Computational photography on mobile computing platforms

## CS 448A - Topics in Computer Graphics

### Computational photography on mobile computing platforms (cell phones)



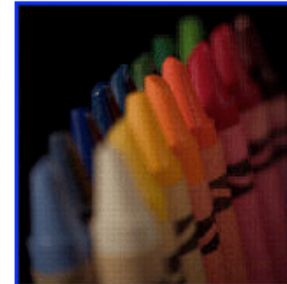
This [Nokia N95](#) smartphone has a 5-megapixel camera with Carl Zeiss optics. The phone also contains a 2nd video camera and 5 radios: cell, WiFi, Bluetooth, GPS, and FM!



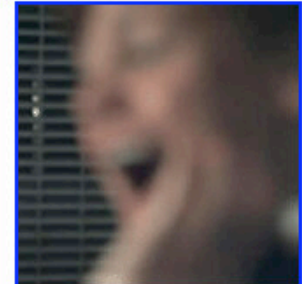
The cell phone at left took this picture, shown here at reduced resolution. Can your point-and-shoot camera do any better? Here is an album of [nature pictures](#) taken by the N95, and one of the [Stanford campus](#).



The boat harbor doesn't belong in this picture. It was found on the Internet and inserted into this [photograph](#). For details, see this [SIGGRAPH 2007 paper](#).



By inserting a microlens array into a handheld camera, one can create a [plenoptic camera](#), which can record a [light field](#) in a single snapshot.



The photographs produced by this camera can be refocused after they are captured. Click above for an example of digital refocusing.

#### *Quarter*

Spring, 2008

#### *Units*

3-4 (+/NC or letter grade)

#### *Time*

Tue/Thu 2:45 - 4:00

#### *Place*

392 Gates Hall (graphics lab conference room)

#### *Instructors*

[Marc Levoy](#), [Andrew Adams](#), [Kari Pulli](#) (Nokia)

#### *Office hours*

Marc Levoy: Tue/Thu, 11:00 - 12:15  
Andrew Adams: TBA

#### *Prerequisite*

Any introductory course in computer graphics or computer vision

#### *Televised?*

No

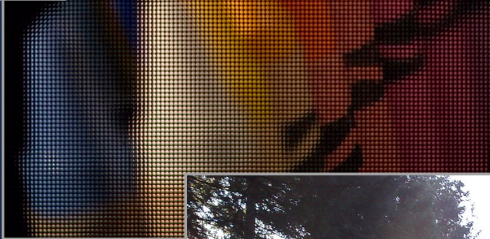
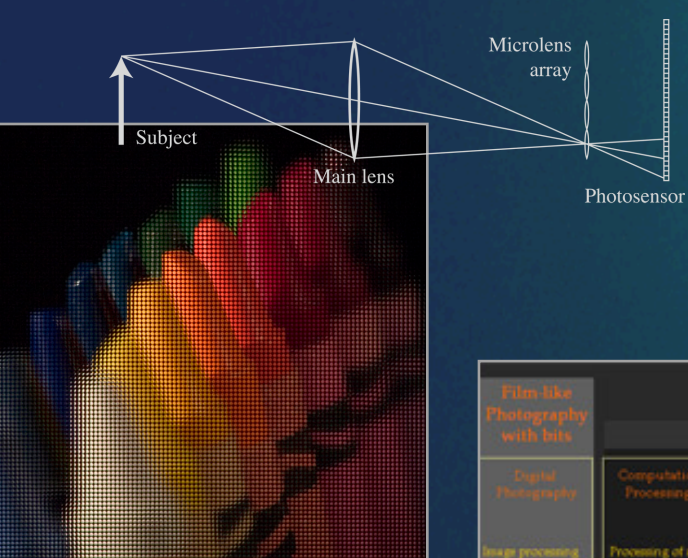
# CS178 – Digital Photography

---

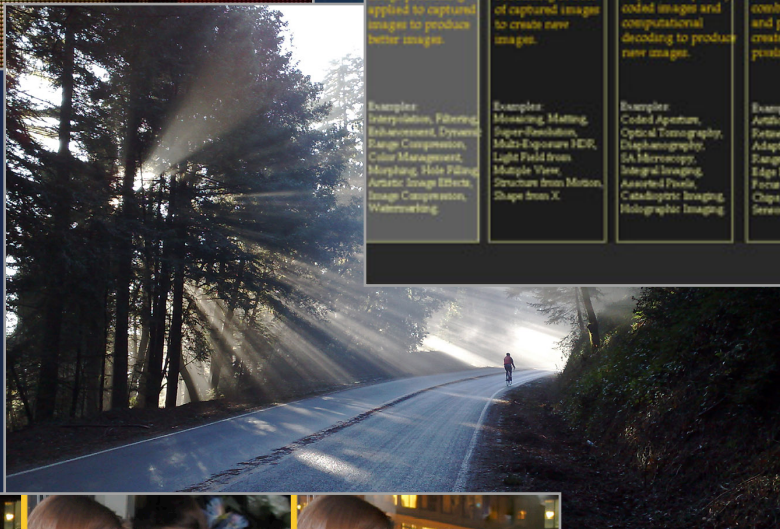
- university-wide, undergraduate course

CS 178. Digital Photography - An introduction to the scientific, artistic, and computing aspects of digital photography, with the goal of understanding how digital cameras work, how to take good pictures using them, and how to manipulate these pictures afterwards. Scientific topics include lenses and optics, light and sensors, color theory, optical effects in nature, the laws of perspective and depth of field, and a bit about sampling and noise. Computational topics include the camera as a computing platform, image processing and editing, and computational photography. Coursework is written assignments, exams, and photography assignments. Prerequisites: calculus at the level of MATH 41/42. Programming experience is not required, but by substituting programming projects for 3 other assignments, students may count the course towards the CS Major and a Specialization in Graphics. Also satisfies GER;2b. Students must have access to a digital camera with manual control over shutter and aperture. Students will be given a free license to Adobe Photoshop for the duration of the course. 3-4 credits. Spring (Levoy)





Computational Photography				
Film-like Photography with bits	Computational Camera			Smart Light
Digital Photography	Computational Processing	Computational Imaging/Optics	Computational Sensor	Computational Illumination
Image processing applied to captured images to produce better images.	Processing of a set of captured images to create new images.	Capture of optically coded images and computational decoding to produce new images.	Detectors that combine sensing and processing to create smart pixels.	Adapting and Controlling Illumination to create revealing images.
Examples: Interpolation, Filtering, Enhancement, Dynamic Range Compression, Color Management, Morphing, Hole Filling, Artistic Image Effects, Image Compression, Watermarking	Examples: Mosaicing, Mating, Super-Resolution, Multi-Exposure HDR, Light Field from Multiple Views, Structure from Motion, Shape from X	Examples: Coded Aperture, Optical Tomography, Depth Photography, SA Microscopy, Integral Imaging, Assorted Pixels, Catadioptric Imaging, Holographic Imaging	Examples: Artificial Retina, Retina Sensors, Adaptive Dynamic Range Sensors, Edge Detect Chips, Focus of Expansion Chips, Motion Sensors	Examples: Flashless flash, Lighting domes, Multi-Flash for depth edges, Dual Photos, Polynomial texture Maps, 4D light source



Ambient

Flash

Result