Computational Photography

Computer Vision
CSE 576, Spring 2008

Richard Szeliski
Microsoft Research

Today's lecture

Computational Photography

- photometric camera calibration
- high-dynamic range imaging & tone mapping
- flash photography
- PhotoMontage
- object cutouts and matting
- Poisson blending
- inpainting and texture synthesis

Readings

- Debevec and Malik, Recovering High Dynamic Range Radiance Maps from Photographs. In *SIGGRAPH 97*.
- S. B. Kang et al. High dynamic range video. *SIGGRAPH 2003*.
- D. Lischinski. Interactive local adjustment of tonal values. *SIGGRAPH 2006.*
- G. Petschnigg *et al*. Digital photography with flash and no-flash image pairs. *SIGGRAPH 2004.*
- P. Pérez et al. Poisson image editing. *SIGGRAPH 2003*

Sources

Some of my slides are from:

6.098 Digital and Computational Photography
6.882 Advanced Computational Photography

Spring 2006

home | syllabus | problem sets and solutions | handouts | links

Bill Freeman and Frédo Durand

http://groups.csail.mit.edu/graphics/classes/CompPhoto06/

Sources

Some of my slides are from:

15-463 (15-862): Computational Photography
Computer Science Department
Carnegie Mellon University

INSTRUCTOR: Alexei (Alyosha) Efros (Office hours: Thursdays 2:30-3:30, NSH 4207)
TA: Jim McCann (Office hours: Tuesdays 5-6, NSH 4228)
UNIVERSITY UNITS: 12
SEMESTER: Fall 2007
NEWSGROUP: cmu.cs.class.cs463 (read this for important information!)
WEB PAGE: http://graphics.cs.cmu.edu/courses/15-463/
LOCATION: WeH 5312
TIME: T R 12:00--1:20 PM

COURSE OVERVIEW:
Computational Photography is an emerging new field created by the convergence of computer graphics, computer vision and photography. Its role is to overcome the limitations of the traditional camera by using computational techniques to produce a richer, more vivid, perhaps more perceptually meaningful representation of our visual world.

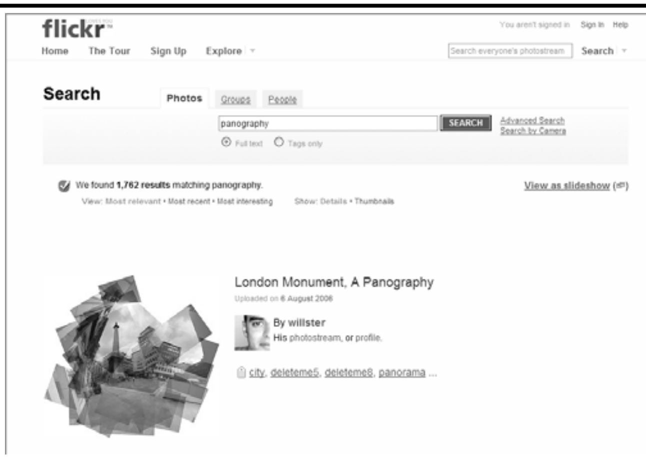
Alexei (Alyosha) Efros

http://graphics.cs.cmu.edu/courses/15-463/

But first, …

… for something (a little) different …

Panography - http://www.flickr.com/search/?q=panography



Panography - http://www.flickr.com/search/?q=panograph

Panography

What kind of motion model?

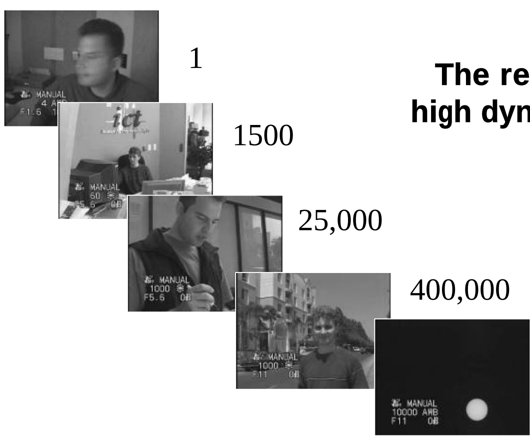
What kind of compositing?

Can you do “global alignment”?

High Dynamic Range Imaging (HDR)

slides borrowed from
15-463: Computational Photography
Alexei Efros, CMU, Fall 2007,
Paul Debevec, and my talks

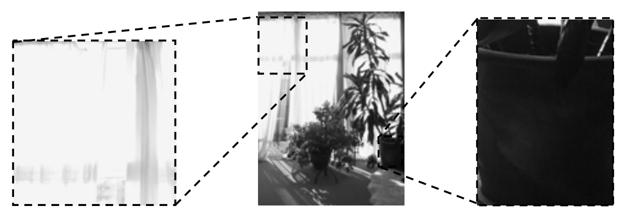
Problem: Dynamic Range



The real world is high dynamic range.

Problem: Dynamic Range

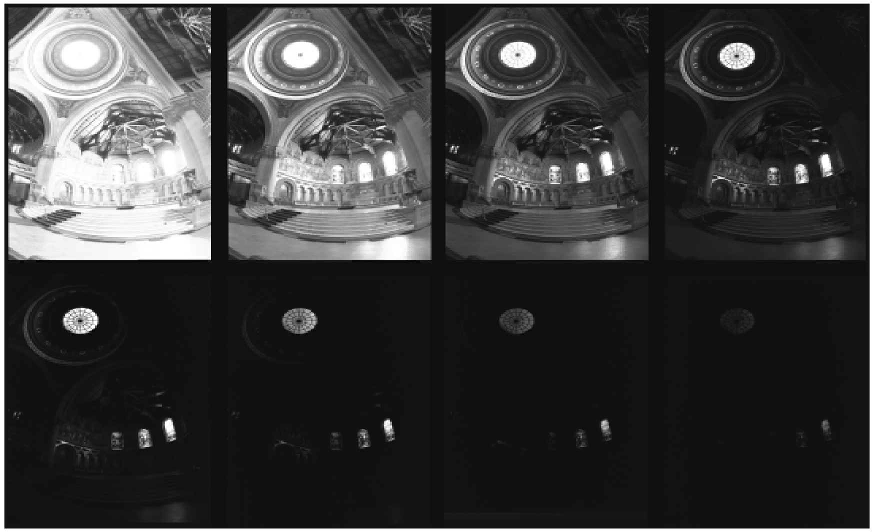
Typical cameras have limited dynamic range



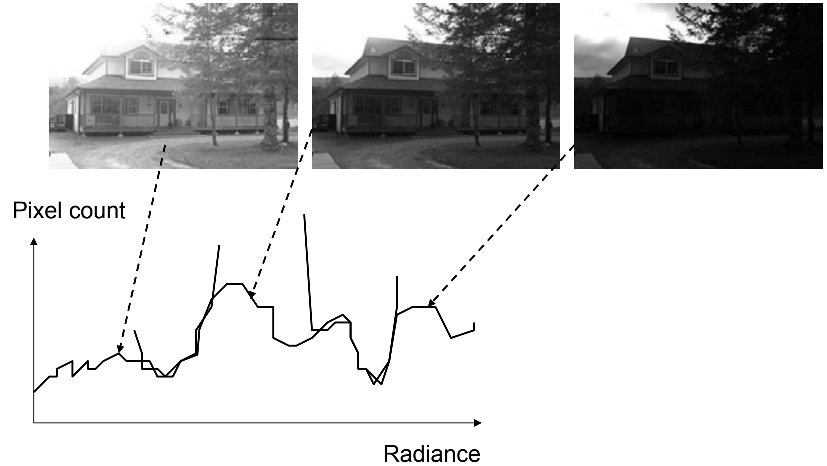
What can we do?

Solution: merge multiple exposures

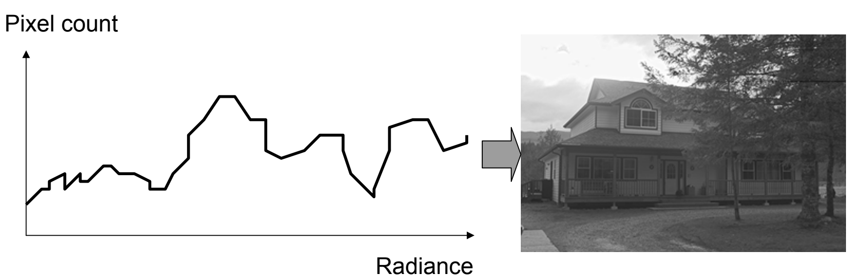
Varying Exposure



HDR images — multiple inputs



HDR images — merged



Camera is not a photometer!

Limited dynamic range

⇒Use multiple exposures?

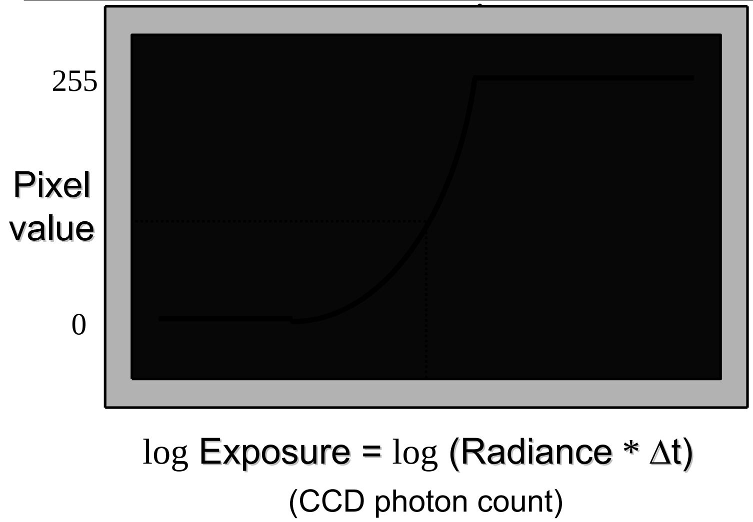
Unknown, nonlinear response

⇒ Not possible to convert pixel values to radiance

Solution:

- Recover response curve from multiple exposures, then reconstruct the ***radiance map***

Imaging system response function



Camera Calibration

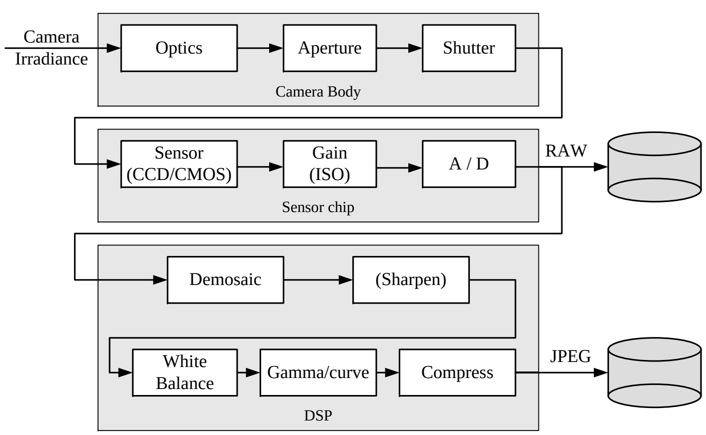
Geometric

- How pixel **coordinates** relate to **directions** in the world

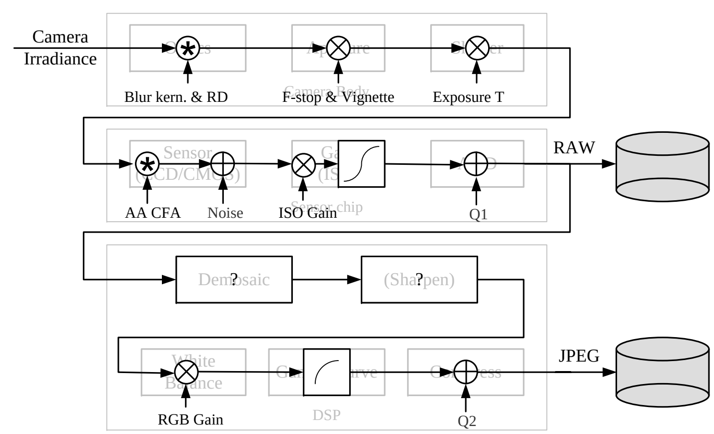
Photometric

- How pixel **values** relate to **radiance** amounts in the world
- Per-pixel **transfer** and **blur**

Camera sensing pipeline



Camera sensing pipeline



Recovering High Dynamic Range Radiance Maps from Photographs

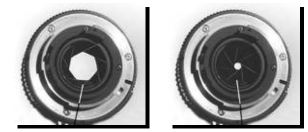
Paul Debevec
Jitendra Malik

Computer Science Division
University of California at Berkeley

SIGGRAPH'97, August 1997

Ways to vary exposure

- Shutter Speed (*)
- F/stop (aperture, iris)
- Neutral Density (ND) Filters



Shutter Speed

Ranges: Canon D30: 30 to 1/4,000 sec.
(1997) Sony VX2000: ¼ to 1/10,000 sec.

Pros:

Directly varies the exposure

Usually accurate and repeatable

Issues:

Noise in long exposures

Shutter Speed

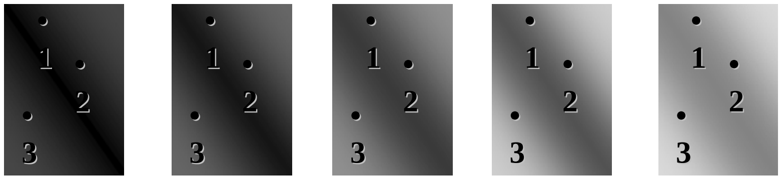
Note: shutter times usually obey a power series – each "stop" is a factor of 2

¼, 1/8, 1/15, 1/30, 1/60, 1/125, 1/250, 1/500, 1/1000 sec

Usually really is:

¼, 1/8, 1/16, 1/32, 1/64, 1/128, 1/256, 1/512, 1/1024 sec

The Algorithm



Δt = 1/64 sec	**Δt = 1/16 sec**	**Δt = 1/4 sec**	**Δt = 1 sec**	**Δt = 4 sec**

Pixel Value Z = f(Exposure)

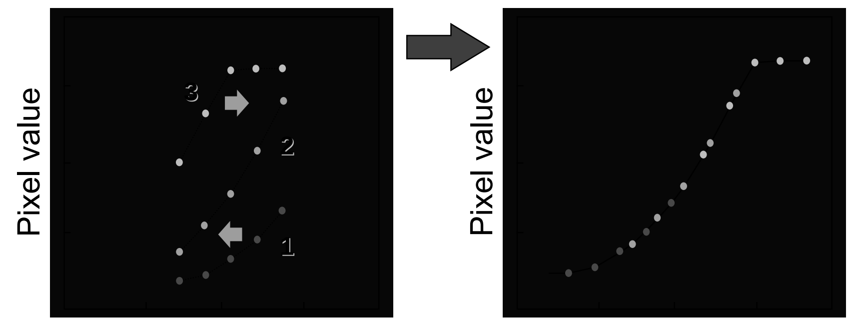
Exposure = Radiance × Δt

log Exposure = log Radiance + log Δt

Response Curve

Assuming unit radiance for each pixel

After adjusting radiances to obtain a smooth response curve



The Math

Let *g(z)* be the *discrete* inverse response function

For each pixel site *i* in each image *j*, want:

$$\ln Radiance_i + \ln \Delta t_j = g(Z_{ij})$$

Solve the over-determined linear system:

$$\underbrace{\sum_{i=1}^{N}\sum_{j=1}^{P}\left[\ln Radiance_i + \ln \Delta t_j - g(Z_{ij})\right]^2}_{\text{fitting term}} + \underbrace{\lambda \sum_{z=Z_{min}}^{Z_{max}} g''(z)^2}_{\text{smoothness term}}$$

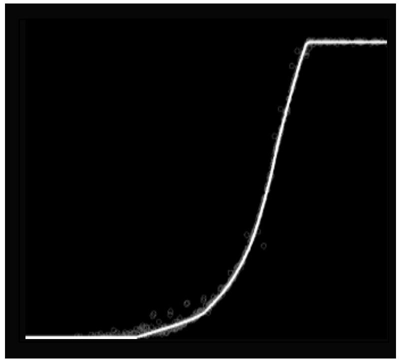
MatLab code

```
function [g,lE]=gsolve(Z,B,l,w)

n = 256;
A = zeros(size(Z,1)*size(Z,2)+n+1,n+size(Z,1));
b = zeros(size(A,1),1);

k = 1;                  %% Include the data-fitting equations
for i=1:size(Z,1)
  for j=1:size(Z,2)
    wij = w(Z(i,j)+1);
    A(k,Z(i,j)+1) = wij; A(k,n+i) = -wij; b(k,1) = wij * B(i,j);
    k=k+1;
  end
end

A(k,129) = 1;           %% Fix the curve by setting its middle value to 0
k=k+1;

for i=1:n-2             %% Include the smoothness equations
  A(k,i)=l*w(i+1); A(k,i+1)=-2*l*w(i+1); A(k,i+2)=l*w(i+1);
  k=k+1;
end

x = A\b;                %% Solve the system using SVD

g = x(1:n);
lE = x(n+1:size(x,1));
```

Results: digital camera

Kodak DCS460
1/30 to 30 sec

Recovered response curve



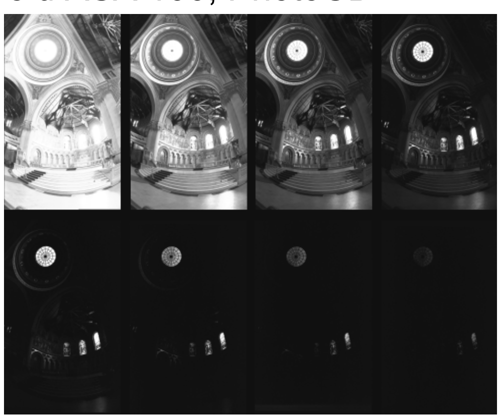
Pixel value

log Exposure

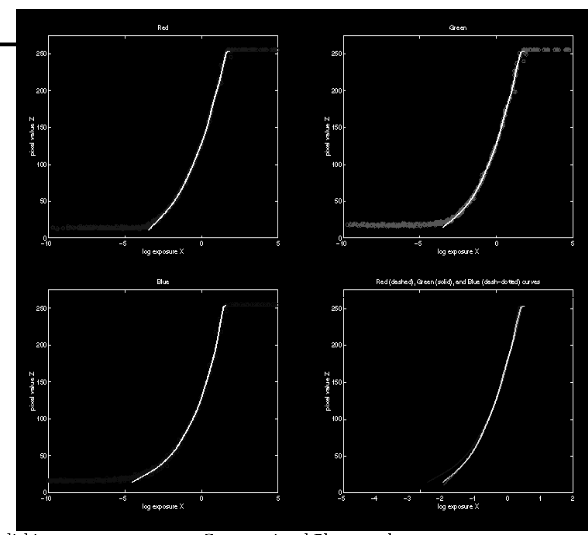
Reconstructed Radiance Map

Results: Color Film

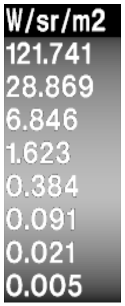
Kodak Gold ASA 100, PhotoCD



Recovered Response Curves



The Radiance Map



The Radiance Map

Linearly scaled to
display device

Portable FloatMap (.pfm)

12 bytes per pixel, 4 for each channel

sign exponent mantissa

Text header similar to Jeff Poskanzer's .ppm image format:

```
PF
768 512
1
<binary image data>
```

Floating Point TIFF similar

Radiance Format (.pic, .hdr)

32 bits / pixel

Red Green Blue Exponent

(145, 215, 87, 149) =	(145, 215, 87, 103) =
(145, 215, 87) * 2^(149-128) =	(145, 215, 87) * 2^(103-128) =
(1190000, 1760000, 713000)	(0.00000432, 0.00000641, 0.00000259)

Ward, Greg. "Real Pixels," in Graphics Gems IV, edited by James Arvo, Academic Press, 1994

ILM's OpenEXR (.exr)

6 bytes per pixel, 2 for each channel, compressed

sign exponent mantissa

- Several lossless compression options, 2:1 typical
- Compatible with the "half" datatype in NVidia's Cg
- Supported natively on GeForce FX and Quadro FX
- Available at http://www.openexr.net/

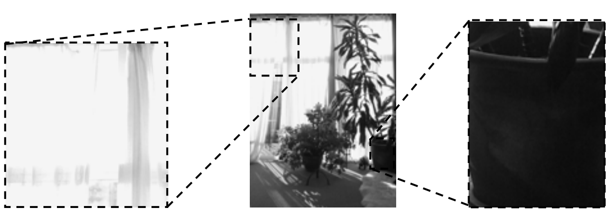
High Dynamic Range Video

Sing Bing Kang, Matt Uyttendaele,
Simon Winder, Rick Szeliski

[SIGGRAPH'2003]

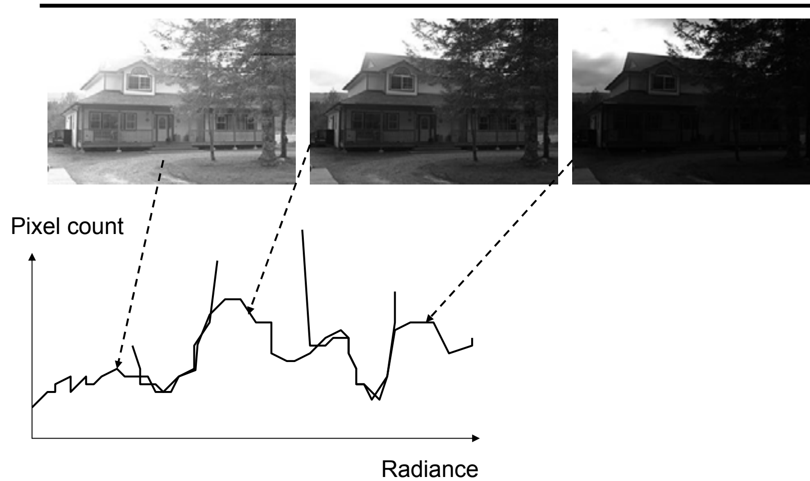
High dynamic range photography

Typical cameras have limited dynamic range

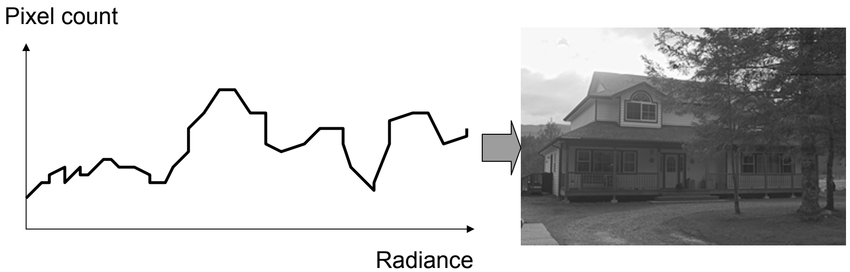


Solution: merge multiple exposures

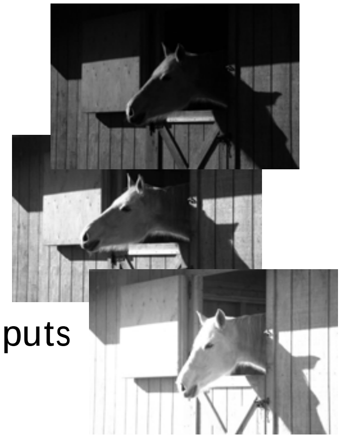
HDR images — multiple inputs



HDR images — merged



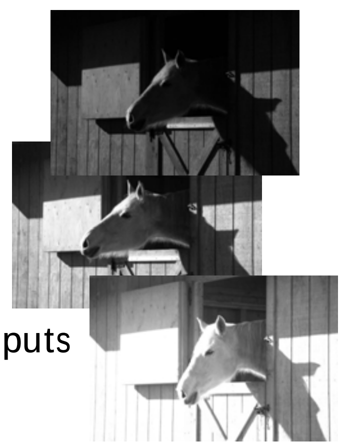
What about scene motion?



Inputs

Tonemapped output
(no compensation or
consistency check)

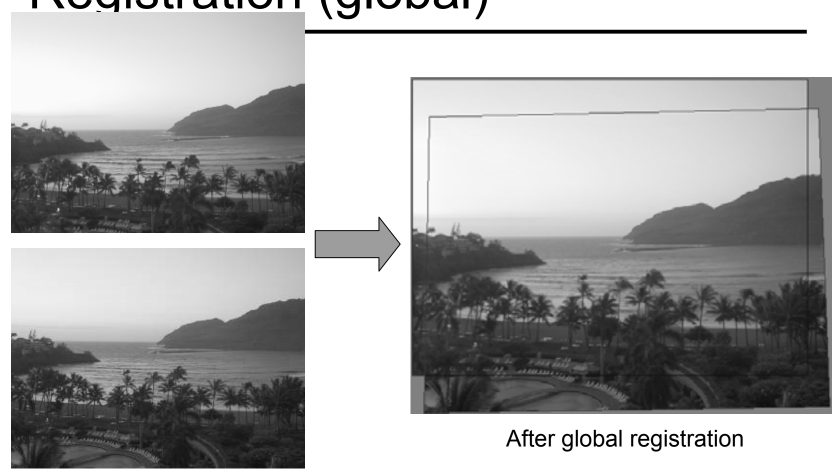
With motion compensation



Inputs

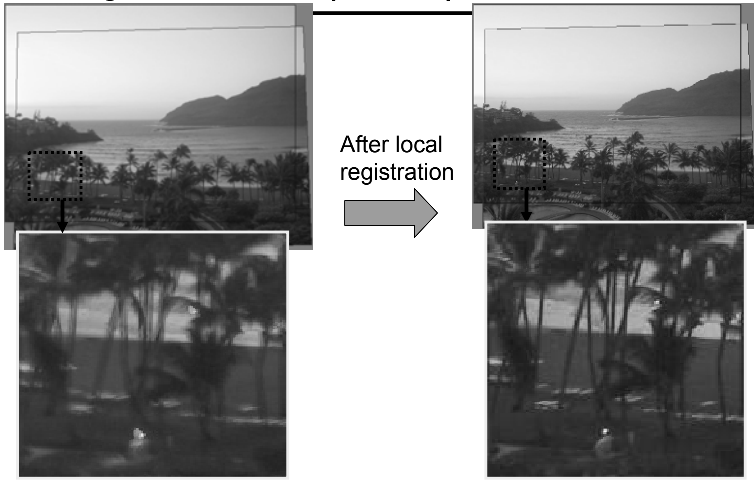
Tonemapped output
(global+local compensation)

Registration (global)



After global registration

Registration (local)

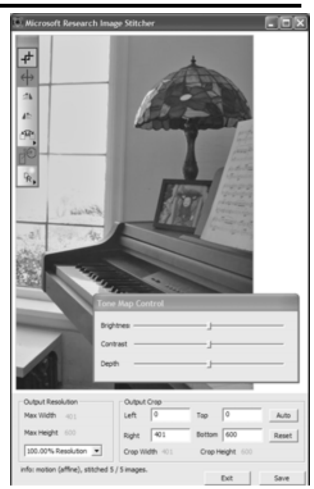


HDR image viewing

Interactively adjust exposure in window

HDR Merge Application

Launch from
MSR Batch Stitcher



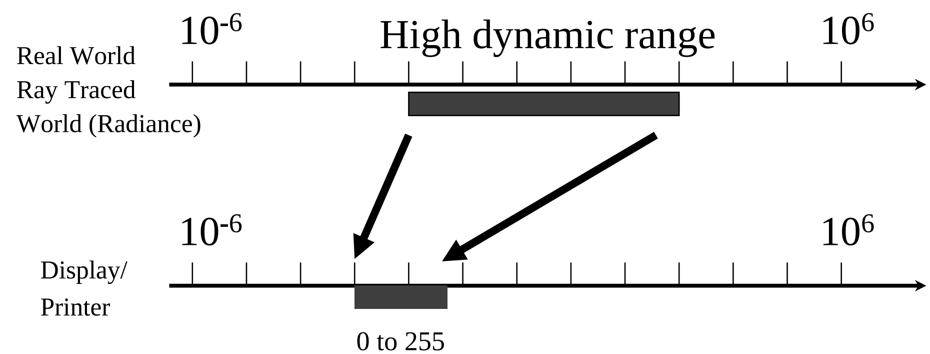
Now What?

Tone Mapping

Tone Mapping

How can we do this?

Linear scaling?, thresholding? Suggestions?



Simple Global Operator

Compression curve needs to

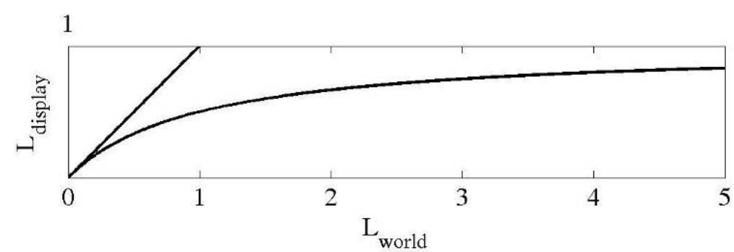
- Bring everything within range
- Leave dark areas alone

In other words

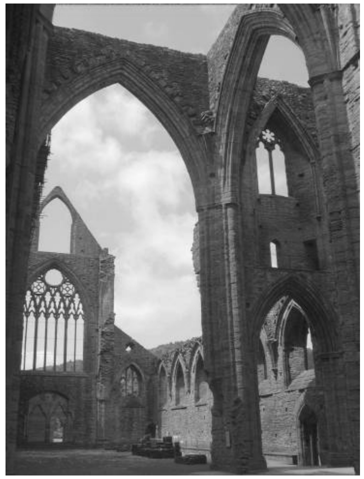
- Asymptote at 255
- Derivative of 1 at 0

Global Operator (Reinhart et al)

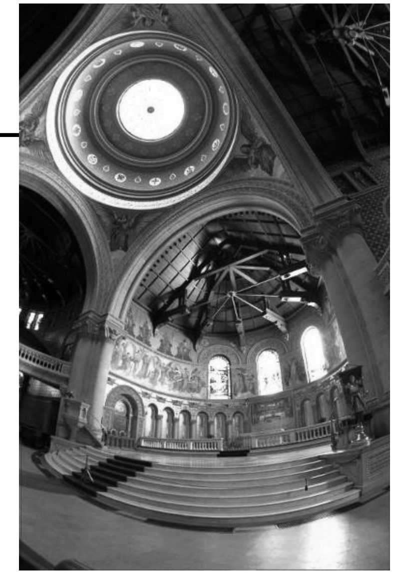
$$L_{display} = \frac{L_{world}}{1 + L_{world}}$$



Global Operator Results



Reinhart Operator



Darkest 0.1% scaled to display device

What do *we* see?

Vs.

What does the eye sees?

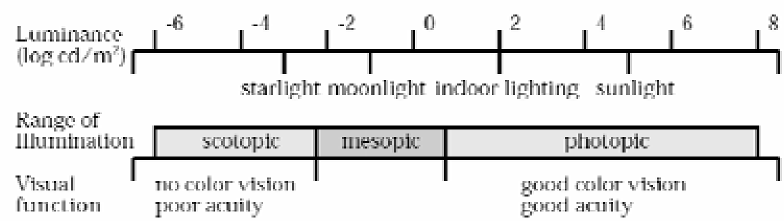
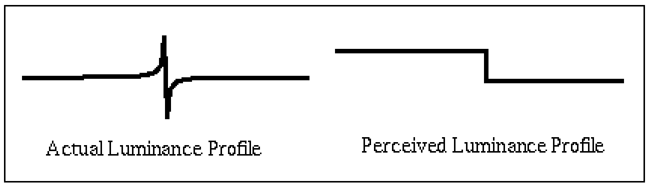


Figure 1: The range of luminances in the natural environment and associated visual parameters. After Hood (1986).

The eye has a huge dynamic range
Do we see a true radiance map?

Metamores

Craik-O'Brien Cornsweet Effect



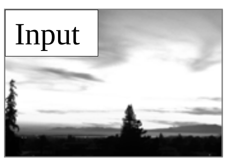
Can we use this for range compression?

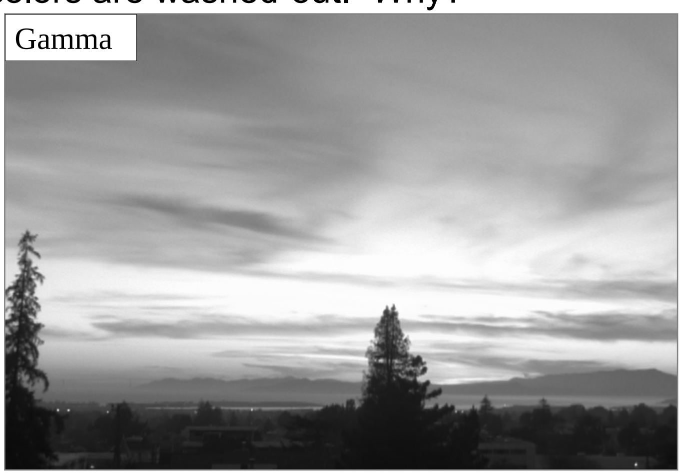
Fast bilateral filtering for the display of high-dynamic-range images

Frédo Durand and Julie Dorsey
SIGGRAPH 2002.

Naïve: Gamma compression

$X \rightarrow X^{\gamma}$, colors are washed-out. Why?



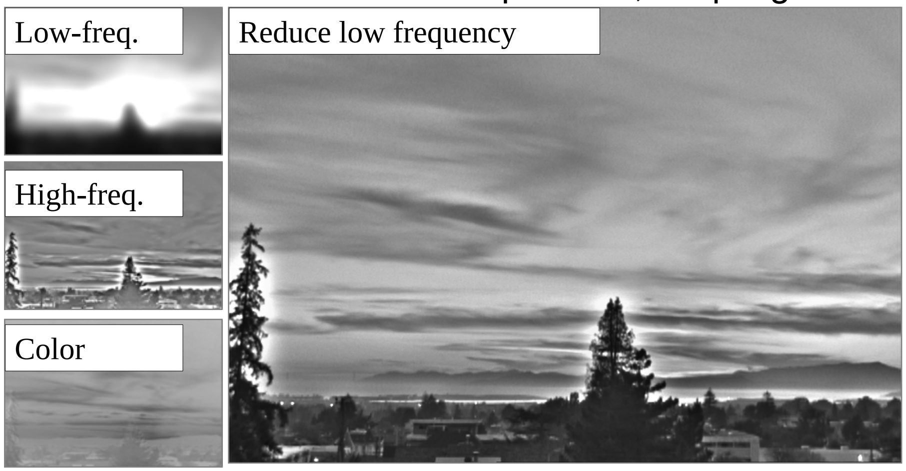


Gamma compression on intensity

Colors are OK, details are blurred

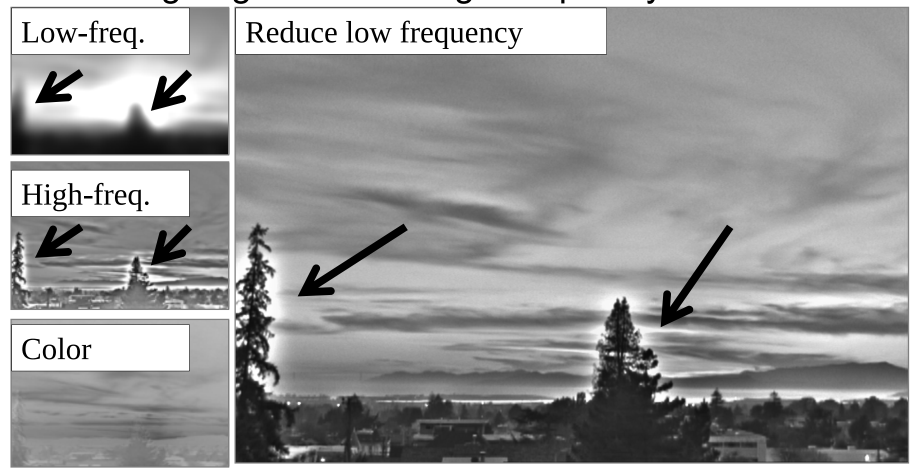
Oppenheim 1968, Chiu et al. 1993

Reduce contrast of low-frequencies, keep high



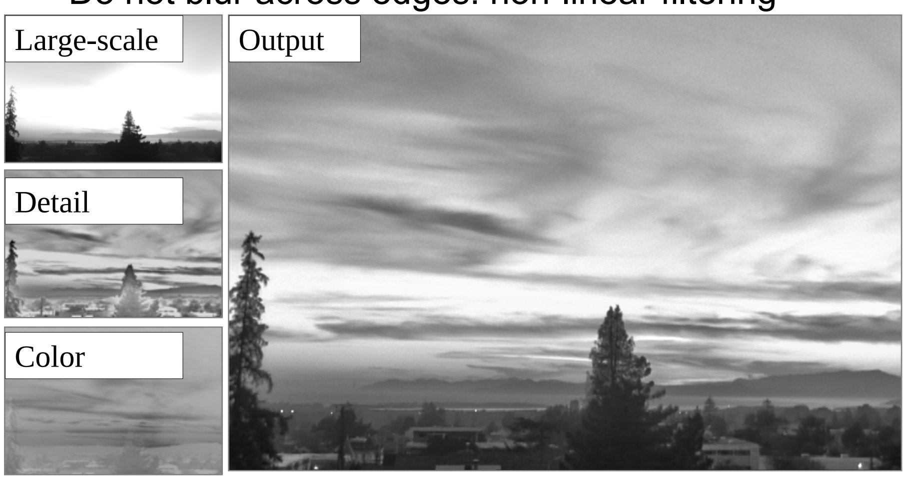
Halos

Strong edges contain high frequency



Our approach

Do not blur across edges: non-linear filtering



Bilateral filter

Tomasi and Manduci 1998

http://www.cse.ucsc.edu/~manduchi/Papers/ICCV98.pdf

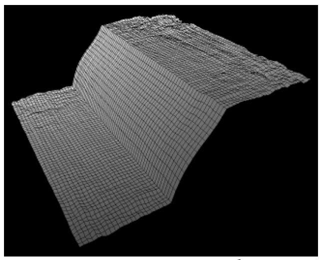
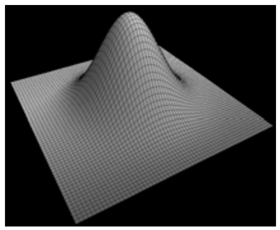
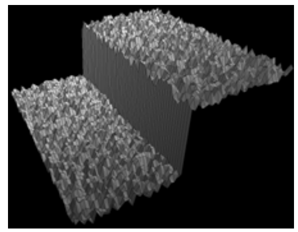
Related to

- SUSAN filter [Smith and Brady 95]
  http://citeseer.ist.psu.edu/smith95susan.html
- Digital-TV [Chan, Osher and Chen 2001]
  http://citeseer.ist.psu.edu/chan01digital.html
- sigma filter
  http://www.geogr.ku.dk/CHIPS/Manual/f187.htm

Start with Gaussian filtering

Output is blurred

$$J \quad = \quad f \quad \otimes \quad I$$

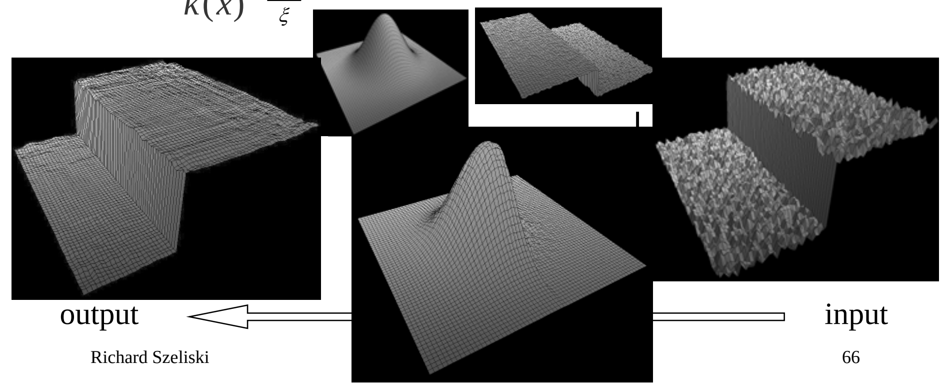




output input

Bilateral filtering is non-linear

The weights are different for each output pixel

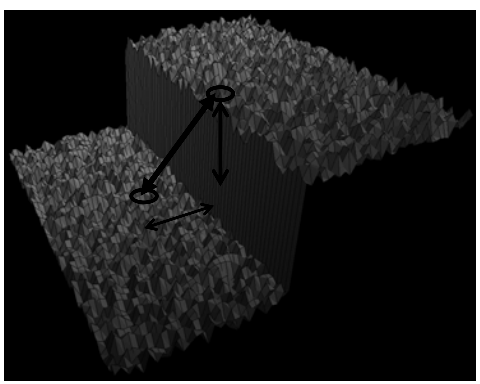
$$J(x) = \frac{1}{k(x)} \sum_{\xi} \; f(x,\xi) \quad g(I(\xi)-I(x)) \quad I(\xi)$$



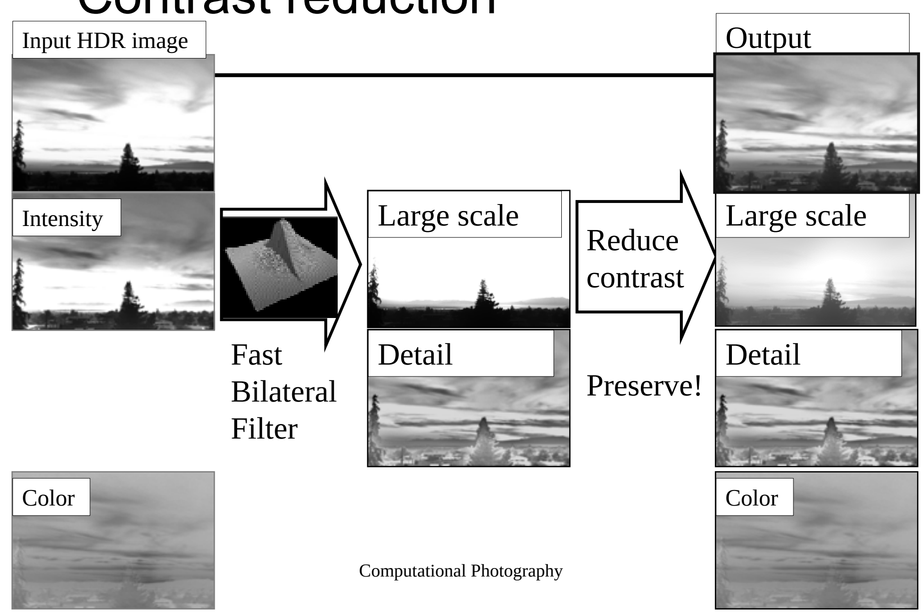
output input

Other view

The bilateral filter uses the 3D distance



Contrast reduction

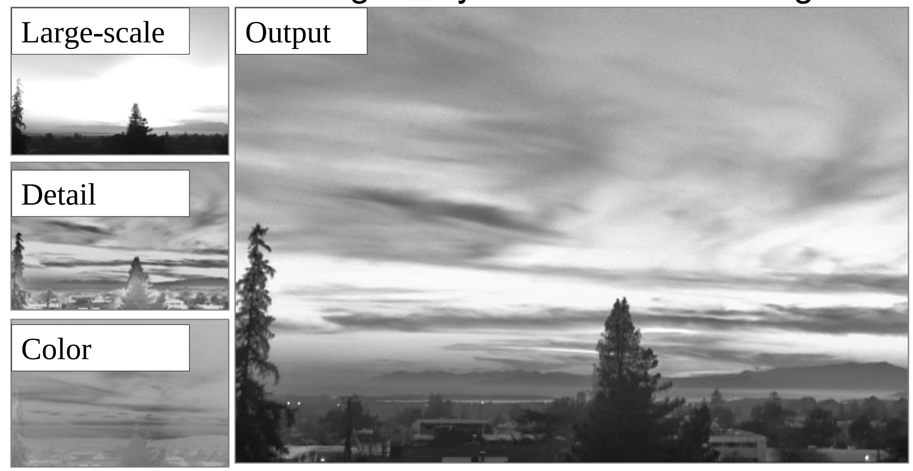


Dynamic range reduction

- To reduce contrast of base layer
  - scale in the log domain ➔ γ exponent in linear
- Set a target range: $\log_{10}(5)$
- Compute range in the log layer: (max-min)
- Deduce γ using *division*
- Normalize so that the biggest value in the (linear) base is 1 (0 in log):
  - offset the compressed based by its max

Summary of approach

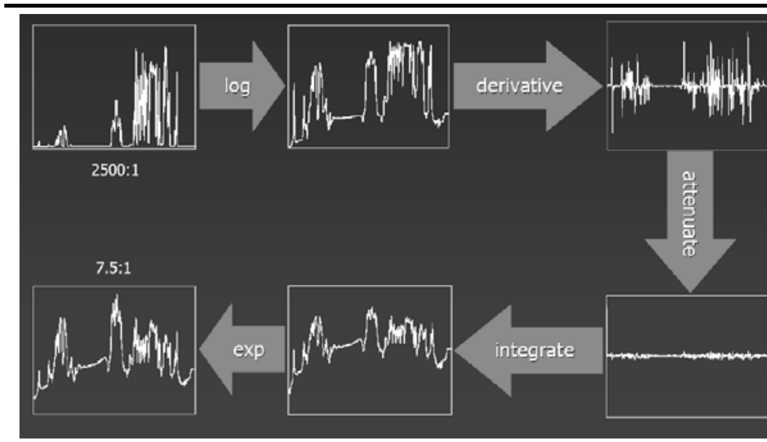
Do not blur base/gain layer: non-linear filtering



Gradient domain high dynamic range compression

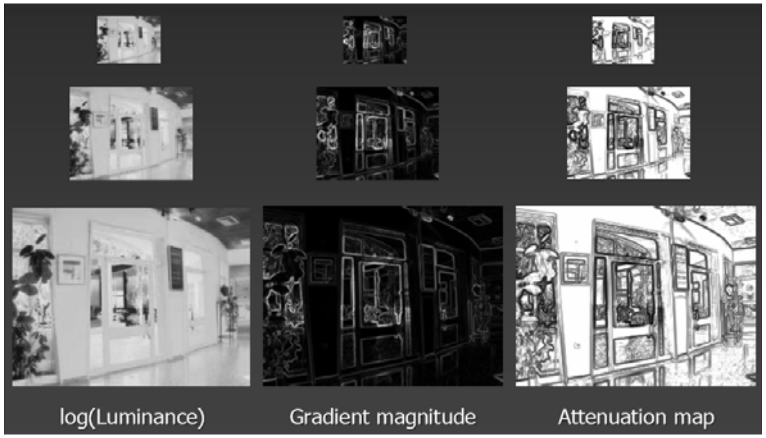
Raanan Fattal, Dani Lischinski, and
Michael Werman
SIGGRAPH 2002.

Gradient Tone Mapping



Slide from Siggraph 2005 by Raskar (Graphs by Fattal et al.)

Gradient attenuation



From Fattal et al.

Interactive Local Adjustment of Tonal Values

Dani Lischinski
Zeev Farbman
The Hebrew University

Matt Uyttendaele
Rick Szeliski
Microsoft Research

SIGGRAPH 2006

Tonal Manipulation

- brightness
- exposure
- contrast
- saturation
- color temperature
- …

Interpretation 1:

Interpretation 2:

Interpretation 3:

This Work is About:

New tool for interactive tonal manipulation: developing negatives in the digital darkroom.

Target material:

- HDR images: the ultimate digital negative.
- Camera RAW images: the most common digital negative.
- Ordinary snapshots.

Existing Tools

Automatic tone mapping algorithms

- Why do we need yet another tone mapping approach?
- Why interactive rather than automatic?

Image manipulation and editing packages, e.g., Adobe Photoshop.

Tone Reproduction Operators

Bilateral Filtering

Durand & Dorsey 2002

Gradient Domain

Fattal et al. 2002

Photographic

Reinhard et al. 2002

Automatic vs. Interactive

Bilateral Filtering

Durand & Dorsey 2002

Interactive Tone Mapping

Photographic

Reinhard et al. 2002

Automatic vs. Interactive

Existing automatic TM operators are "black boxes"

- No direct control over the outcome
- No local adjustment
- Not suitable for creative/artistic work
- Results do not always look "photographic"
- Most operators not *really* automatic

But What About Photoshop?

You can do just about everything …

Adjustment Layers

Layer Masks

- Select regions
- Paint blending weights

… but you need a lot of experience, patience, and time!

Example

15 minutes in Photoshop: 3 minutes:

Approach

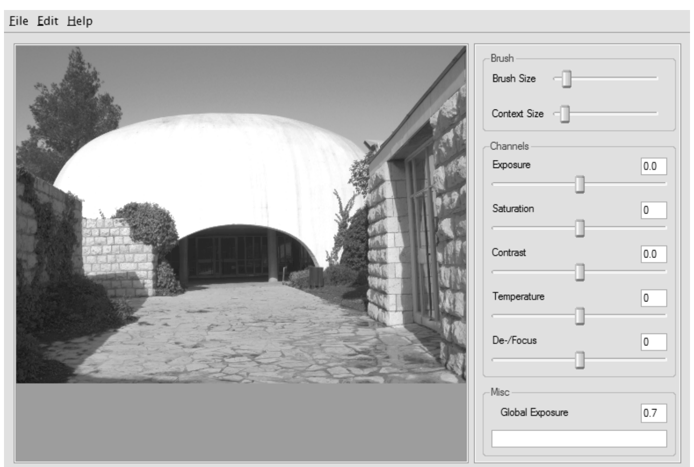
User indicates regions using scribbles.

User adjusts tonal values using sliders.

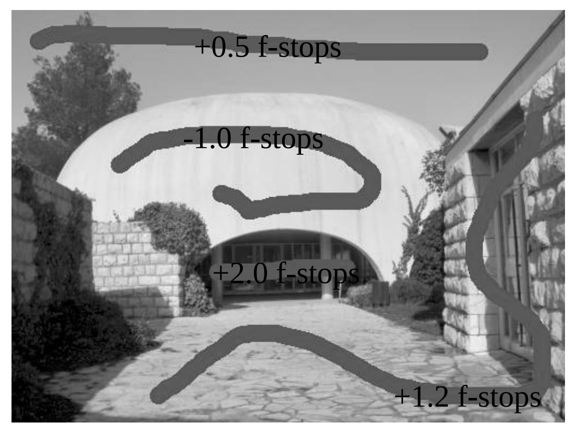
Scribbles + tonal values are interpreted as soft constraints.

Optimization framework "propagates" the constraints to the entire image.

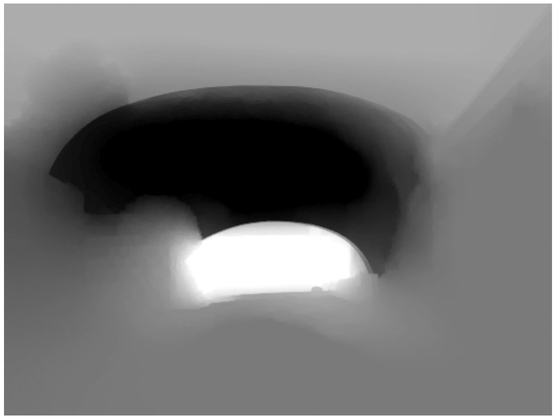
User interface



Input: constraints



Result: adjustment map



Constraint Propagation

Approximate constraints with a function whose smoothness is determined by underlying image:

$$f = \arg\min_{f} \left\{ \underbrace{\sum_{\mathbf{x}} w(\mathbf{x})\, (f(\mathbf{x}) - g(\mathbf{x}))^2}_{\text{data term}} + \lambda \underbrace{\sum_{\mathbf{x}} h(\nabla f, \nabla L)}_{\text{smoothness term}} \right\}$$

Our smoothness term:

$$h(\nabla f, \nabla L) = \frac{|f_x|^2}{|L_x|^{\alpha} + \varepsilon} + \frac{|f_y|^2}{|L_y|^{\alpha} + \varepsilon}$$

Linear System

$$\mathbf{A} f = b$$

$$\mathbf{A}_{ij} = \begin{cases} -\lambda \left(|L_i - L_j|^{\alpha} + \varepsilon \right)^{-1} & j \in N_4(i) \\ w_i - \sum_{k \in N_4(i)} \mathbf{A}_{ik} & i = j \\ 0 & \text{otherwise} \end{cases}$$

$$b_i = g_i\, w_i$$

Solving the System

Sparse symmetric positive definite system:

- Use preconditioned conjugate gradients (PCG)
- Which preconditioner?

$$\mathbf{A} = \mathbf{W} - \mathbf{L}$$

Matrix **L** depends on the image, only **W** depends on constraints.

Idea: use incomplete Cholesky decomposition of **I - L**.

Multi-resolution Solver

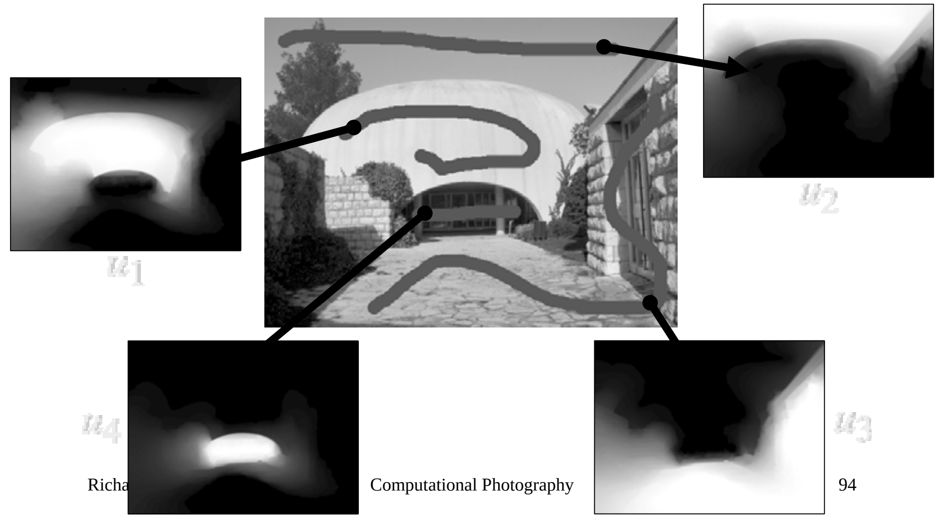
Solve a coarse version of the problem using a direct solver.

Repeat:

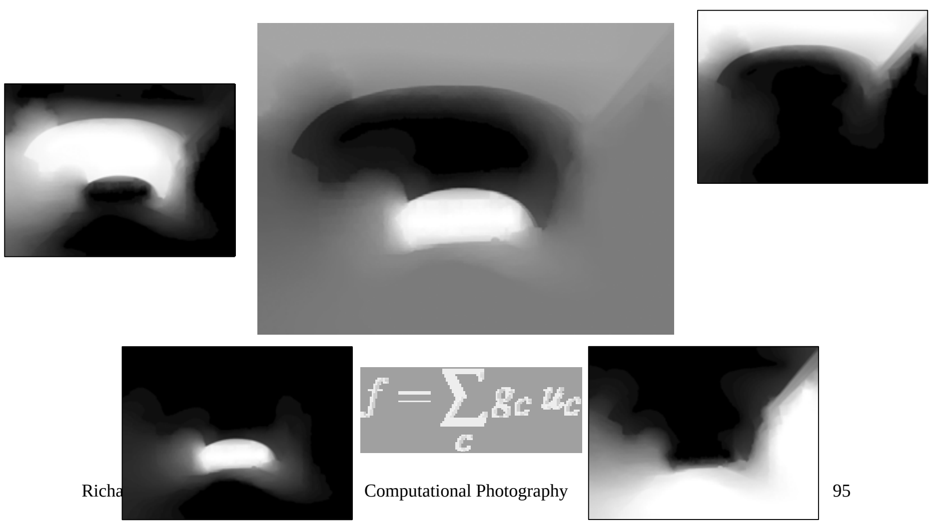
- Upsample solution to next level, perform a few PCG iterations.

Stop once desired preview resolution has been reached.

Influence Functions



Influence Functions



$$f = \sum_c g_c \, u_c$$

Rapid Solution Update

When a single constraint's value is modified:

$$g'_c = g_c + \Delta g_c$$

The new solution *f'* is given as a linear combination:

$$f' = f + \Delta g_c \, u_c$$

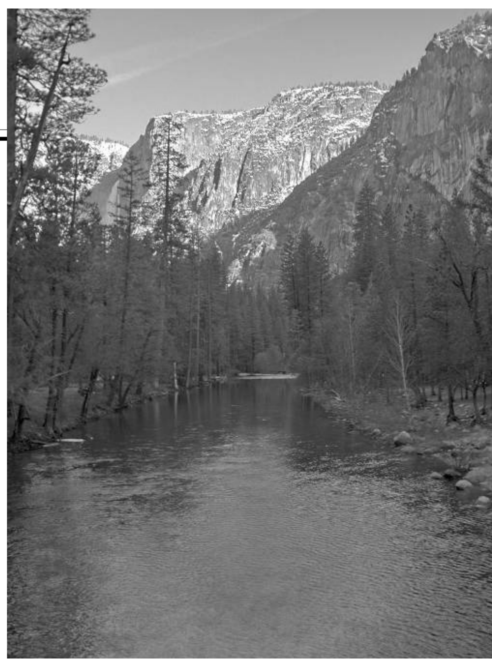
Automatic Initialization

Inspired by Ansel Adams' "Zone System".

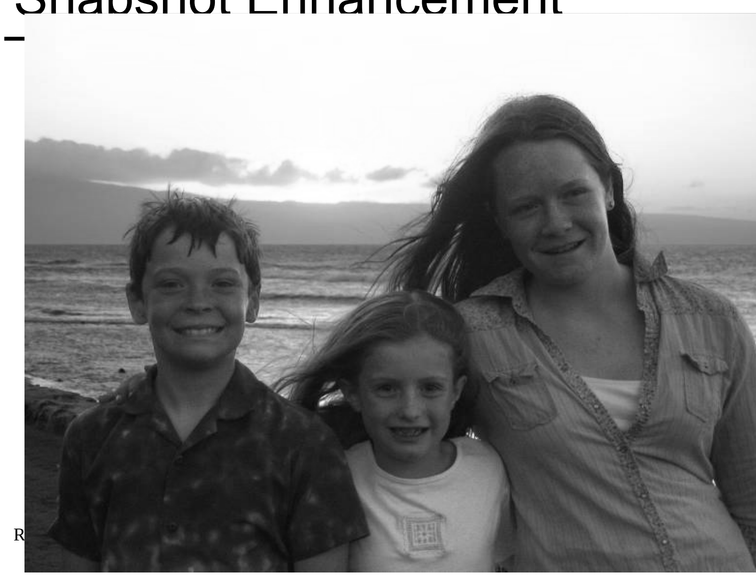
- Segment image (very crudely) into brightness "zones"
- Determine the desired exposure for each zone
- Let the image-guided optimization produce a piecewise smooth exposure map

Results – Automatic mode

Results – Automatic mode



Snapshot Enhancement



Snapshot Enhancement

Spatially Variant White Balance

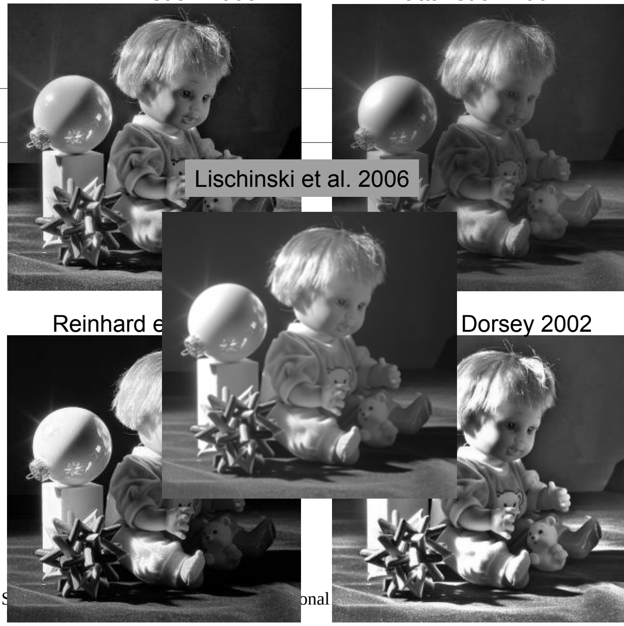
Fake Depth of Field

Comparison of tone mappers

Durand and Dorsey. *Fast bilateral filtering for the display of high-dynamic-range images*. SIGGRAPH 2002.

Fattal, Lischinski, and Werman. *Gradient domain high dynamic range compression*. SIGGRAPH 2002.

Li, Sharan, and Adelson. *Compressing and Companding High Dynamic Range Images with Subband Architectures*. SIGGRAPH 2005.



Merging flash and non-flash images

Georg Petschnigg, Maneesh Agrawala,
Hugues Hoppe, Rick Szeliski,
Michael Cohen, Kentaro Toyama
[SIGGRAPH'2004]

Flash + non-flash images

Flash photos have less noise, more detail

Non-flash photos have better color

Idea: merge them together

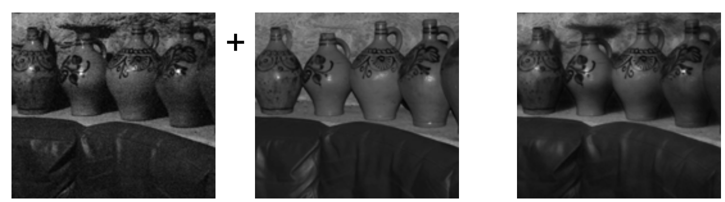
- But how?

non-flash flash merged

Flash + non-flash images

Smooth non-flash photo using flash photo's edge information

Add high-frequency details from flash image



non-flash flash merged

Joint bilateral filter

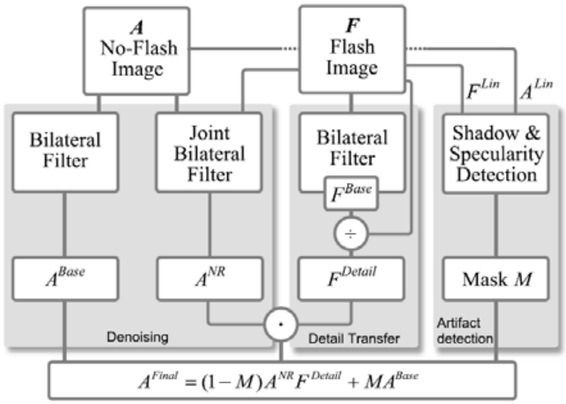


Figure 3: Overview of our algorithms for denoising, detail transfer, and flash artifact detection.

Bilateral detail filter

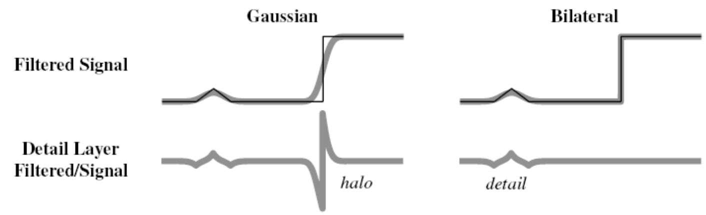


Figure 5: (left) A Gaussian low-pass filter blurs across all edges and will therefore create strong peaks and valleys in the detail image that cause halos. (right) The bilateral filter does not smooth across strong edges and thereby reduces halos, while still capturing detail.

Final result

Today's lecture

Computational Photography

- photometric camera calibration
- high-dynamic range imaging & tone mapping
- flash photography
- PhotoMontage
- object cutouts and matting
- Poisson blending
- inpainting and texture synthesis

Interactive Digital Photomontage

Aseem Agarwala, Mira Dontcheva,
Maneesh Agrawala, Steven Drucker,
Alex Colburn, Brian Curless,
David Salesin, Michael Cohen
(U. Washington & Microsoft Research)
[SIGGRAPH'2004]

PhotoMontage

Goal: select pieces to form the "best" composite

Q: How can we formulate this?

Mathematical "Diversion"

A Comparative Study of Energy Minimization Methods for Markov Random Fields

Richard Szeliski, Ramin Zabih, Daniel Scharstein,
Olga Veksler, Vladimir Kolmogorov, Aseem Agarwala,
Marshall Tappen, and Carsten Rother

ECCV 2006

Markov Random Fields

Used a lot in computer vision and graphics:

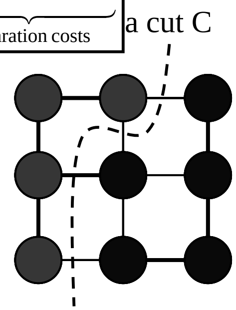
- stereo matching
- image segmentation
- image blending
- texture synthesis
- image restoration

Markov Random Fields

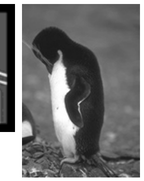
We want to minimize the energy function $E(f)$

$$\arg\min_f \underbrace{\sum_p D(p, f_p)}_{\text{assignment costs}} + \underbrace{\sum_{p,q\in N} V(f_p, f_q)}_{\text{separation costs}}$$

with $\boxed{\text{spatially varying}}$ smoothness/interaction potentials $\boxed{V_{pq}(f_p,f_q)}$



MRF labels



- Ordered labels
  - depth map (stereo)
  - gray levels (image restoration)
- Unordered labels
  - image id (quilting / PhotoMontage)
- Binary labels
  - segmentation (GrabCut)

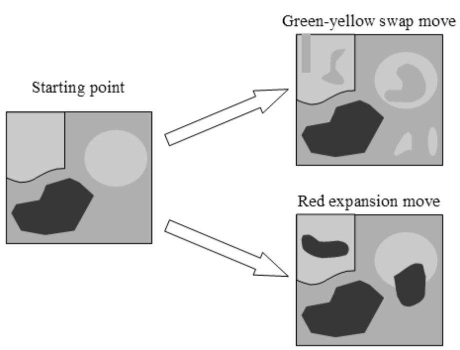
Graph cuts

Sequence of *binary* sub-problems solved by max-flow algorithm:

[Boykov, Veksler & Zabih, PAMI 2001]

•α-β swap: optimize pair of labels

•α expansion: change all pixels to *one* value

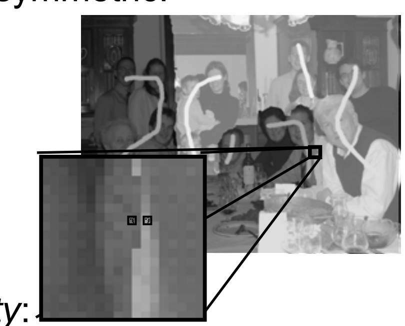


PhotoMontage

Interaction potentials *not* symmetric:
  measures similarity between neighboring pixels in two *different* images

Interaction potential may *not* be symmetric

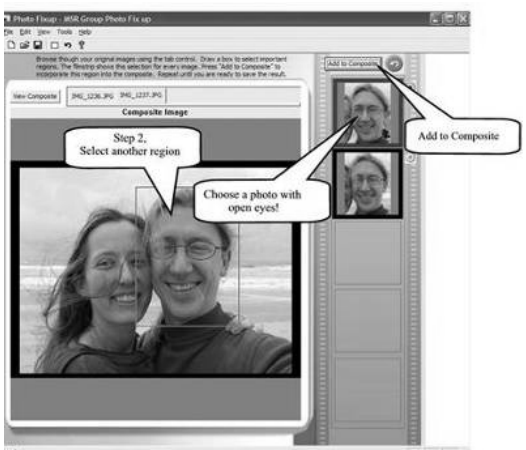
May violate *sub-modularity*:
  some algorithms don't work



Photomontage: input

Photomontage: output

GroupShot application



Cutout-based compositing

Interactively blend *different* images:
focus settings

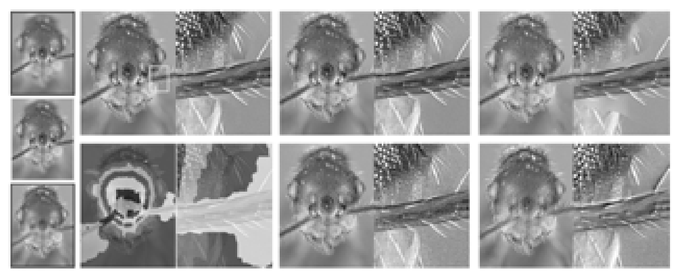


Figure 2 A set of macro photographs of an ant (three of eleven used shown on the left) taken at different focal lengths. We use a global maximum contrast image objective to compute the graph-cut composite automatically (top left, with an inset to show detail, and the labeling shown directly below). A small number of remaining artifacts disappear after gradient-domain fusion (top, middle). For comparison we show composites made by Auto-Montage (top, right), by Haeberli's method (bottom, middle), and by Laplacian pyramids (bottom, right). All of these other approaches have artifacts; Haeberli's method creates excessive noise, Auto-Montage fails to attach some hairs to the body, and Laplacian pyramids create halos around some of the hairs.

Cutout-based compositing

Interactively blend *different* images:
people's faces

Figure 6 We use a set of portraits (first row) to mix and match facial features, to either improve a portrait, or create entirely new people. The faces are first hand-aligned, for example, to place all the noses in the same location. In the first two images in the second row, we replace the closed eyes of a portrait with the open eyes of another. The user paints strokes with the *designated source* objective to specify desired features. Next, we create a fictional person by combining three source portraits. Gradient-domain fusion is used to smooth out skin tone differences. Finally, we show two additional mixed portraits.

Cutout-based de-ghosting

- Select only one image per output pixel, using spatial continuity
- Blend across seams using gradient continuity ("Poisson blending")

GrabCut:
Cut & Paste Images Easily

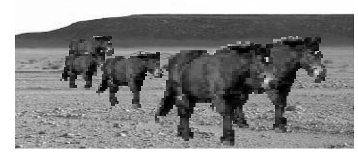
Carsten Rother
Andrew Blake
Vladimir Kolmogorov
[SIGGRAPH'2004]

GrabCut

User draws a rectangle or lasso around an object
Object edges are detected and feathered
Approach: binary graph cut w/ color statistics

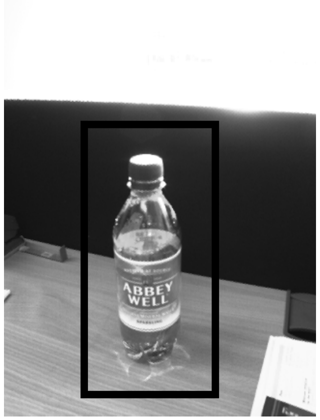
User Input

Segmentation

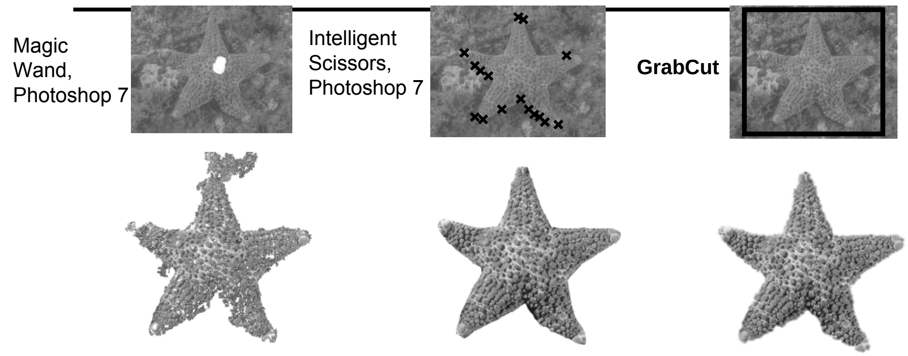


New composed Image

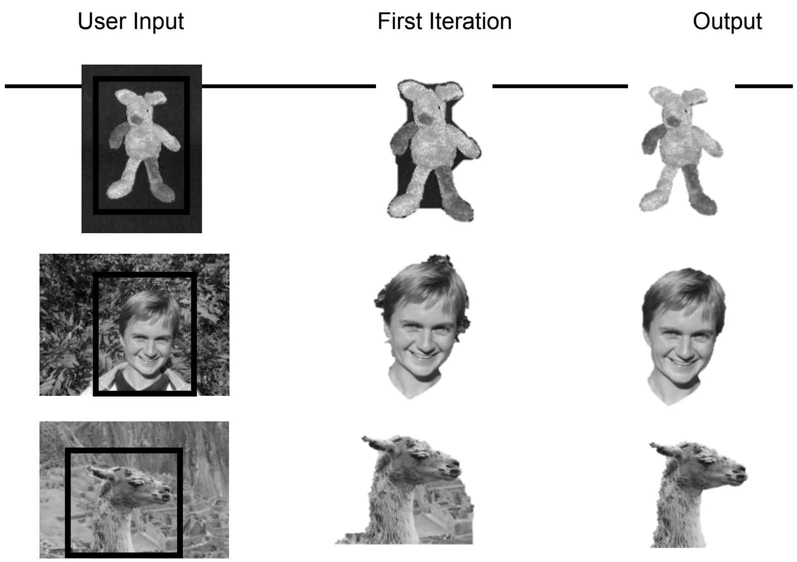
GrabCut example



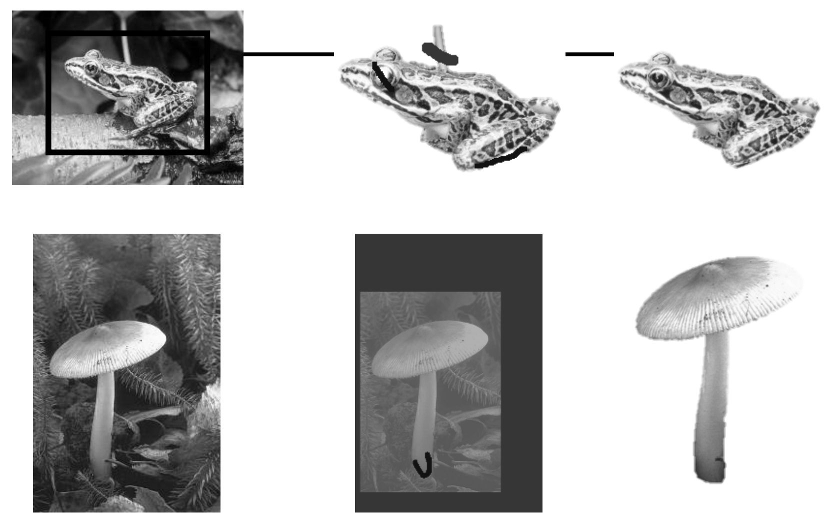
GrabCut — Comparison



Results — No User Interaction



Results — Little User Interaction



Results — More User Interaction

Today's lecture

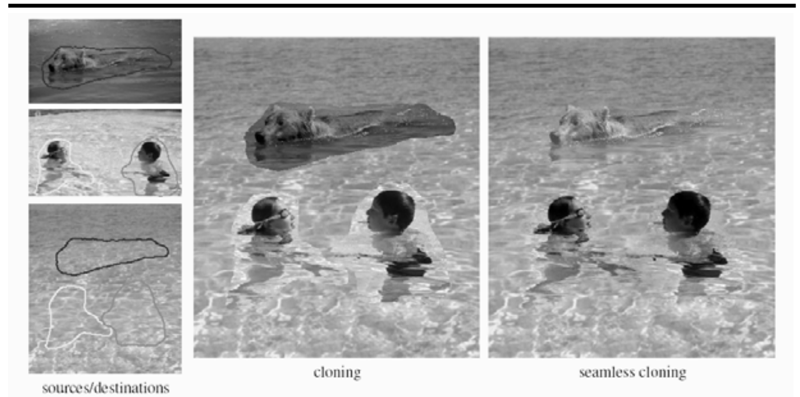
Computational Photography

- photometric camera calibration
- high-dynamic range imaging & tone mapping
- flash photography
- PhotoMontage
- object cutouts and matting
- Poisson blending
- inpainting and texture synthesis

Poisson Image Editing

Patrick Pérez, Michel Gangnet, Andrew Blake
SIGGRAPH 2003

Poisson Image Editing



Blend the gradients of the two images, then integrate

Seamless Poisson cloning

Given vector field *v* (pasted gradient), find the value of *f* in unknown region that optimizes:

$$\min_f \iint_\Omega |\nabla f - \mathbf{v}|^2 \text{ with } f|_{\partial\Omega} = f^*|_{\partial\Omega}$$

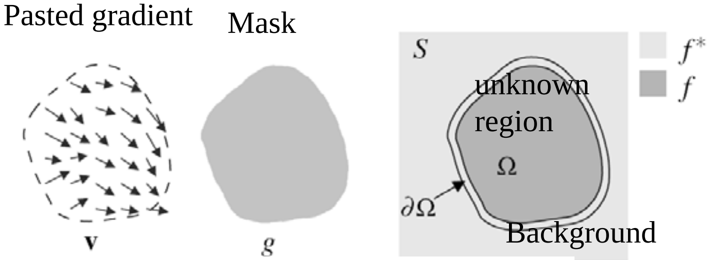


Figure 1: **Guided interpolation notations.** Unknown function f interpolates in domain Ω the destination function f^*, under guidance of vector field $\mathbf{v}$, which might be or not the gradient field of a source function g.

Discrete Poisson solver

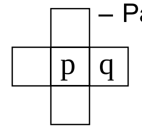
Two approaches:

- Minimize variational problem $\min_f \iint_\Omega |\nabla f - \mathbf{v}|^2 \text{ with } f|_{\partial\Omega} = f^*|_{\partial\Omega},$
- Solve Euler-Lagrange equation $\Delta f = \operatorname{div}\mathbf{v} \text{ over } \Omega, \text{ with } f|_{\partial\Omega} = f^*|_{\partial\Omega}$

In practice, variational is best

In both cases, need to discretize derivatives

- Finite differences over 4 pixel neighbors
- We are going to work using pairs
  - Partial derivatives are easy on pairs



Discrete Poisson solver

Minimize variational problem

$$\min_f \iint_\Omega |\nabla f - \mathbf{v}|^2 \text{ with } f|_{\partial\Omega} = f^*|_{\partial\Omega},$$

$$\min_{f|_\Omega} \sum_{\langle p,q\rangle \cap \Omega \neq \emptyset} \underbrace{(f_p - f_q - v_{pq})^2}_{\text{Discretized gradient}}, \text{ with } f_p = f_p^*, \text{for all } p \in \partial\Omega$$

(all pairs that are in Ω)

Discretized v: g(p)-g(q)

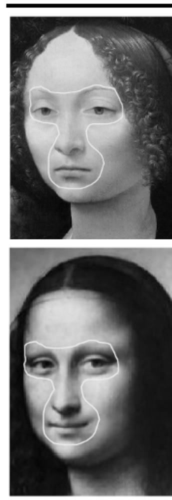
Boundary condition

Rearrange and call N_p the neighbors of p

Big yet sparse linear system

Similar to MRF, but continuous variables

Face cloning

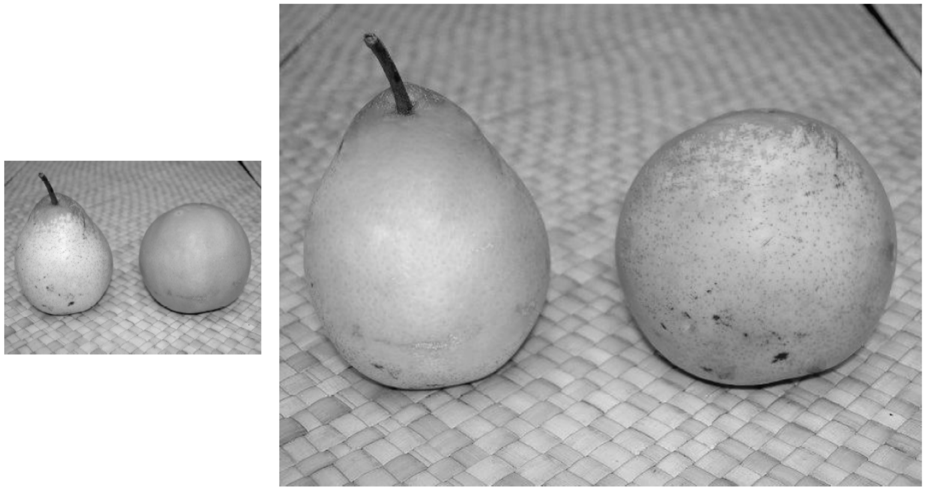


source/destination

cloning

seamless cloning

Texture swapping



swapped textures

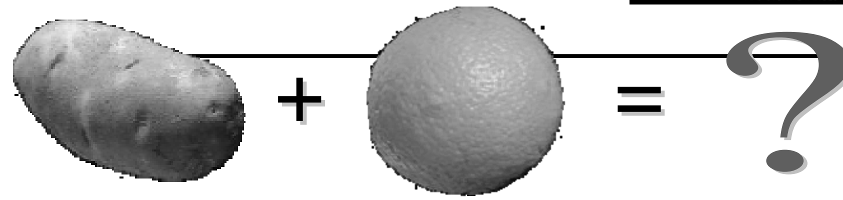


Image Quilting for Texture Synthesis & Transfer

Alexei Efros *(UC Berkeley)*

Bill Freeman *(MERL)*

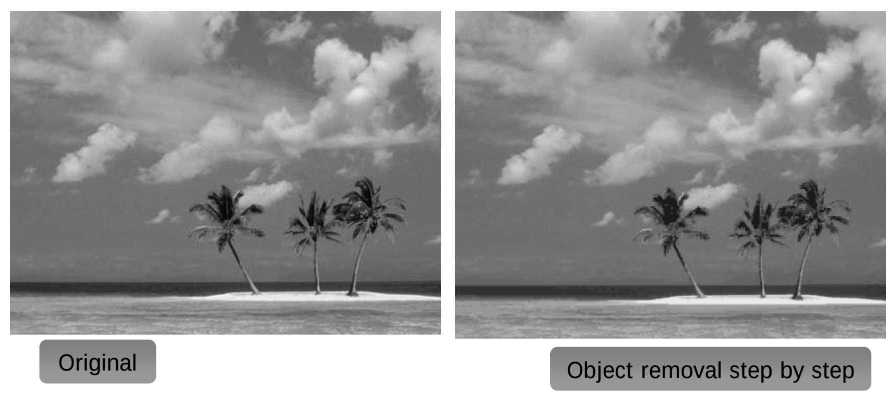
Jump to their Slide Deck

Object Removal by Exemplar-Based Inpainting

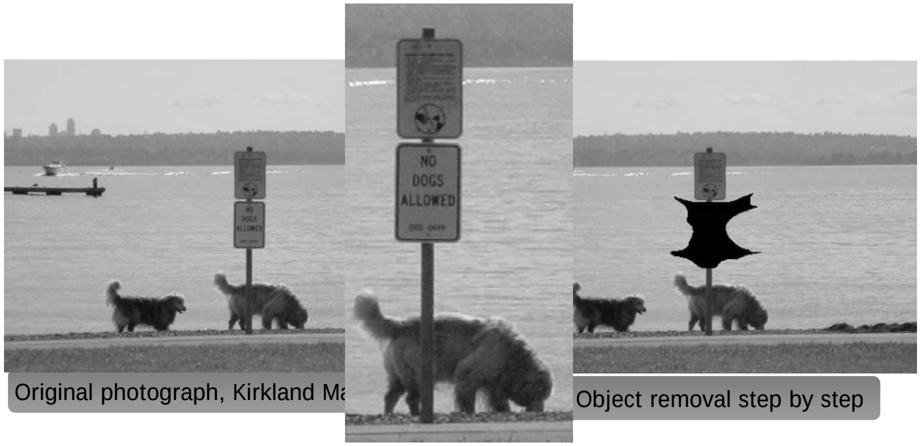
A. Criminisi, P. Pérez, K. Toyama

CVPR 2003

Object removal step by step

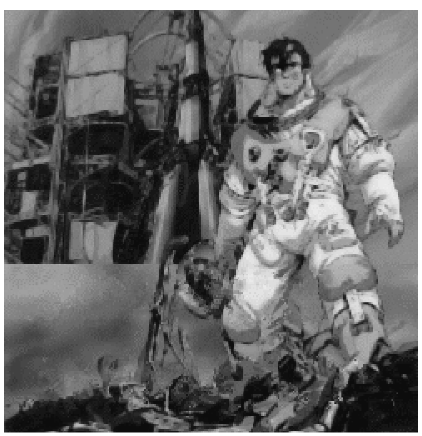


Object removal step by step



Removing people

Removing text



Original images courtesy of Bertalmio et al.

Completing panoramas

Summary of today's lecture

Computational Photography

- photometric camera calibration
- high-dynamic range imaging & tone mapping
- flash photography
- PhotoMontage
- object cutouts and matting
- Poisson blending
- inpainting and texture synthesis

Questions?