6.098 Digital and Computational Photography
6.882 Advanced Computational Photography

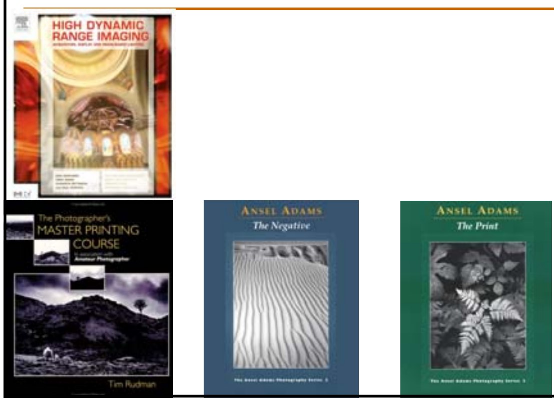
HDR imaging and the Bilateral Filter

Bill Freeman
Frédo Durand
MIT - EECS

Announcement

- **Why Matting Matters**
- **Rick Szeliski**
- **Monday at 2pm in Kiva/Patil**
- Image matting (e.g., blue-screen matting) has been a mainstay of Hollywood and the visual effects industry for decades, but its relevance to computer vision is not yet fully appreciated. In this talk, I argue that the mixing of pixel color values at the boundaries of objects (or even albedo changes) if a fundamental process that must be correctly modeled to make meaningful signal-level inferences about the visual world, as well as to support high-quality imaging transformations such as de-noising and de-blurring. Starting with Ted Adelson et al.'s seminal work on layered motion models, I review early stereo matching algorithms with transparency and matting (with Polina Golland), work on layered representations with matting (with Simon Baker and Anandan), through Larry Zitnick's 2-layer representation for 3D video. I then present our recent work (with Ce Liu et al.) on image de-noising using a segmented description of the image and Eric Bennett's et al.'s work on multi-image de-mosaicing, again using a local two-color model.

References



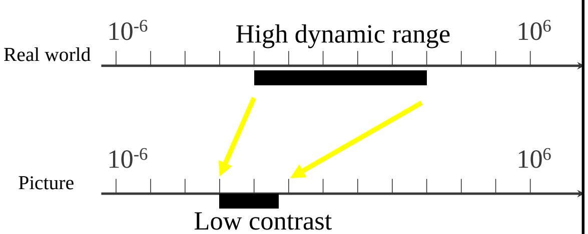
Refs

http://www.hdrsoft.com/resources/dri.html
http://www.clarkvision.com/imagedetail/dynamicrange2/
http://www.debevec.org/HDRI2004/
http://www.luminous-landscape.com/tutorials/hdr.shtml
http://www.anyhere.com/gward/hdrenc/
http://www.debevec.org/IBL2001/NOTES/42-gward-cic98.pdf
http://www.openexr.com/
http://gl.ict.usc.edu/HDRShop/
http://www.dpreview.com/learn/?/Glossary/Digital_Imaging/Dynamic_Range_01.htm
http://www.normankoren.com/digital_tonality.html
http://www.anyhere.com/

http://www.cybergrain.com/tech/hdr/

Contrast reduction

- **Match limited contrast of the medium**
- **Preserve details**



Histogram

- **See** http://www.luminous-landscape.com/tutorials/understanding-series/understanding-histograms.shtml http://www.luminous-landscape.com/tutorials/expose-right.shtml
- **Horizontal axis is pixel value**
- **Vertical axis is number of pixels**

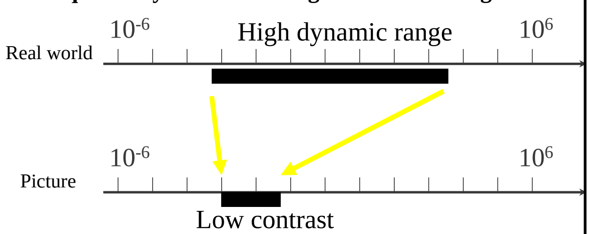
Highlights

- **Clipped pixels (value >255)**
- **Pro and semi-pro digital cameras allow you to make them blink.**

Questions?

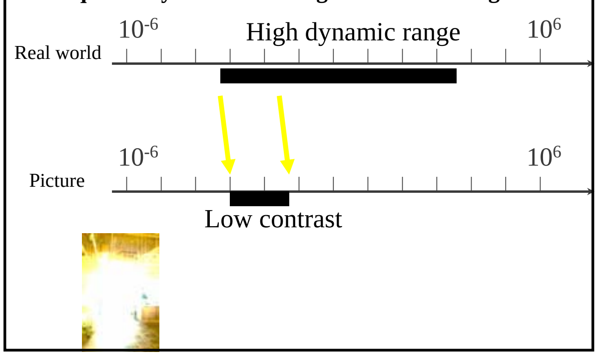
Multiple exposure photography

- **Sequentially measure all segments of the range**



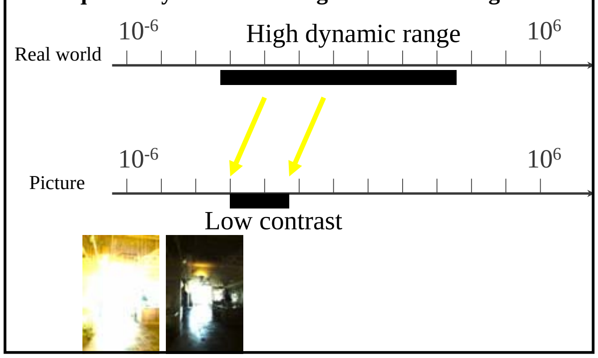
Multiple exposure photography

- **Sequentially measure all segments of the range**



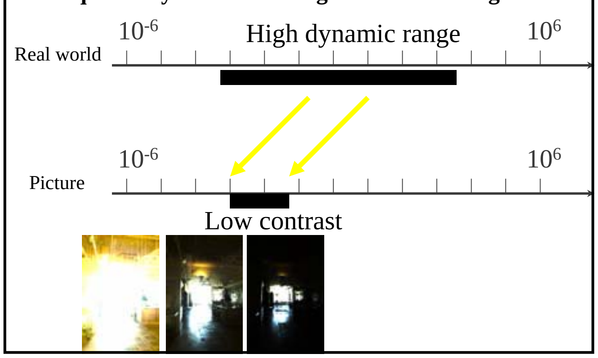
Multiple exposure photography

- **Sequentially measure all segments of the range**



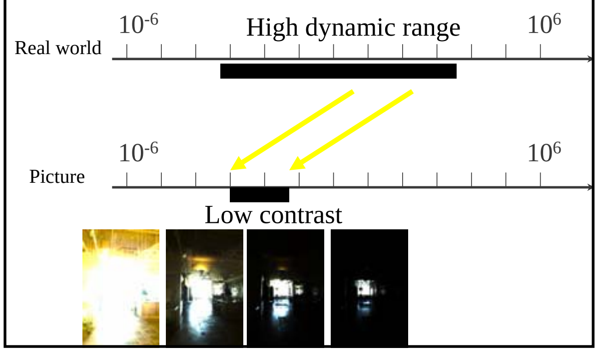
Multiple exposure photography

- **Sequentially measure all segments of the range**



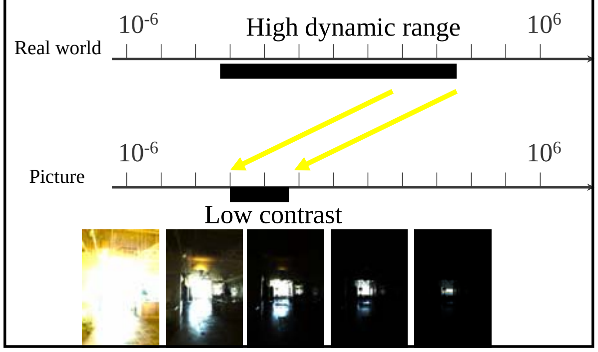
Multiple exposure photography

- **Sequentially measure all segments of the range**



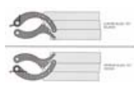
Multiple exposure photography

- **Sequentially measure all segments of the range**



How do we vary exposure?

- **Options:**
  - Shutter speed



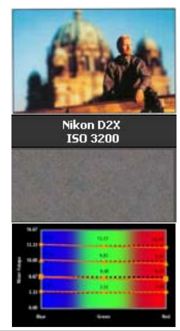
  - Aperture

  - ISO
  - Neutral density filter

Slide inspired by Siggraph 2005 course on HDR

Tradeoffs

- **Shutter speed**
  - Range: ~30 sec to 1/4000sec (6 orders of magnitude)
  - Pros: reliable, linear
  - Cons: sometimes noise for long exposure
- **Aperture**
  - Range: ~f/1.4 to f/22 (2.5 orders of magnitude)
  - Cons: changes depth of field
  - Useful when desperate
- **ISO**
  - Range: ~100 to 1600 (1.5 orders of magnitude)
  - Cons: noise
  - Useful when desperate
- **Neutral density filter**
  - Range: up to 4 densities (4 orders of magnitude) & can be stacked
  - Cons: not perfectly neutral (color shift), not very precise, need to touch camera (shake)
  - Pros: works with strobe/flash, good complement when desperate

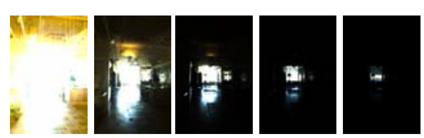


Slide after Siggraph 2005 course on HDR

Questions?

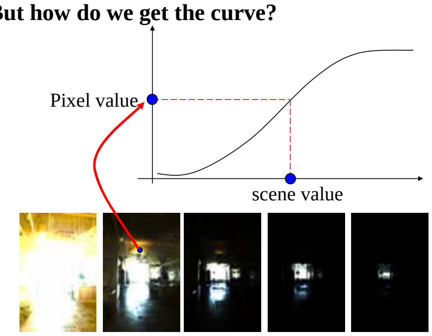
HDR image using multiple exposure

- **Given N photos at different exposure**
- **Recover a HDR color for each pixel**



If we know the response curve

- **Just look up the inverse of the response curve**
- **But how do we get the curve?**

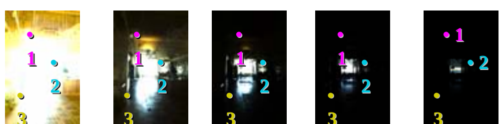


Calibrating the response curve

- **Two basic solutions**
  - Vary scene luminance and see pixel values
    - Assumes we control and know scene luminance
  - Vary exposure and see pixel value for one scene luminance
    - But note that we can usually not vary exposure more finely than by 1/3 stop
- **Best of both:**
  - Vary exposure
  - Exploit the large number of pixels

The Algorithm

Image series



Δt = 10 sec	**Δt = 1 sec**	**Δt = 1/10 sec**	**Δt = 1/100 sec**	**Δt = 1/1000 sec**

Pixel Value Z = f(Exposure)

Exposure = Radiance × Δt

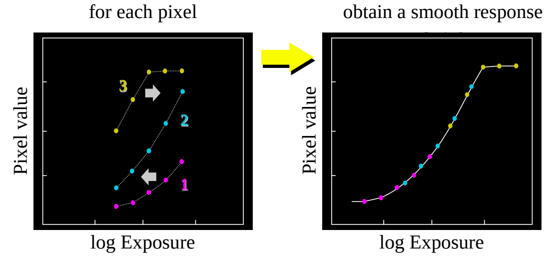
log Exposure = log Radiance + log Δt

Slide adapted from Alyosha Efros who borrowed it from Paul Debevec
Δ t don't really correspond to pictures. Oh well.

Response curve

- **Exposure is unknown, fit to find a smooth curve**

Assuming unit radiance for each pixel | After adjusting radiances to obtain a smooth response



Slide stolen from Alyosha Efros who stole it from Paul Debevec

The Math

- **Let *g(z)* be the *discrete* inverse response function**
- **For each pixel site *i* in each image *j*, want:**

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

- **Solve the overdetermined linear system:**

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

Slide stolen from Alyosha Efros who stole it from Paul Debevec

Matlab code

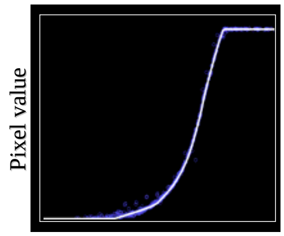
```
function [g,lE]=gsolve(Z,B,l,w)

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));
```

Slide stolen from Alyosha Efros who stole it from Paul Debevec

Result: digital camera

Kodak DCS460
1/30 to 30 sec

Recovered response curve



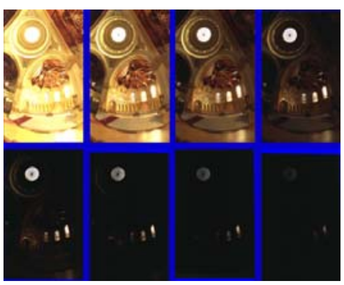
Slide stolen from Alyosha Efros who stole it from Paul Debevec

Reconstructed radiance map

Slide stolen from Alyosha Efros who stole it from Paul Debevec

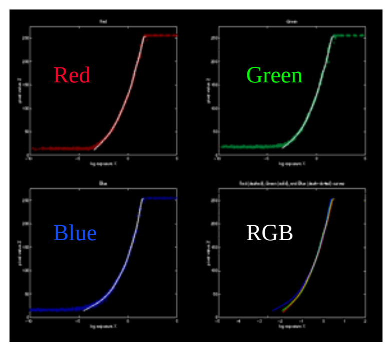
Result: color film

- **Kodak Gold ASA 100, PhotoCD**



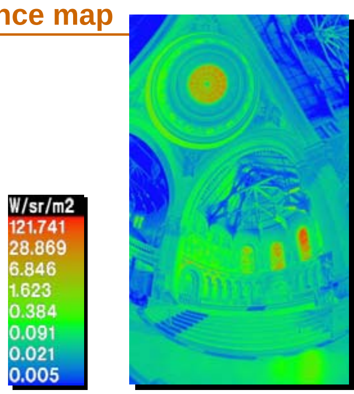
Slide stolen from Alyosha Efros who stole it from Paul Debevec

Recovered response curves



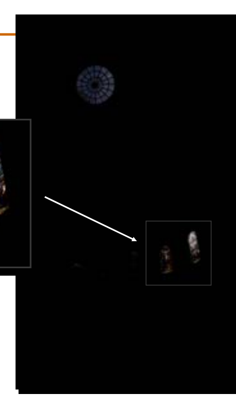
Slide stolen from Alyosha Efros who stole it from Paul Debevec

The Radiance map



Slide stolen from Alyosha Efros who stole it from Paul Debevec

The Radiance map

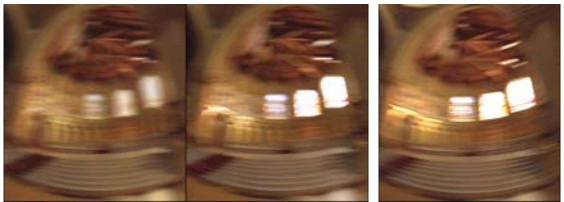


Linearly scaled to
display device

Slide stolen from Alyosha Efros who stole it from Paul Debevec

HDR image processing

Images from Debevec & Malik 1997



Motion blur applied to **low**-dynamic-range picture | Motion blur applied to **high**-dynamic-range picture | Real motion-blurred picture

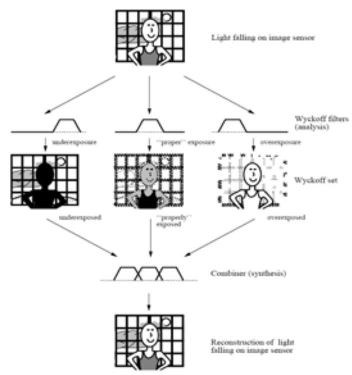
- **Important also for depth of field post-process**

Available in HDRShop

Slide from Siggraph 2005 course on HDR

HDR combination papers

- **Steve Mann**
  http://genesis.eecg.toronto.edu/wyckoff/index.html
- **Paul Debevec**
  http://www.debevec.org/Research/HDR/
- **Mitsunaga, Nayar , Grossberg**
  http://www1.cs.columbia.edu/CAVE/projects/rad_cal/rad_cal.php



From Being Undigital by Mann & Picard

Questions?

Smarter HDR capture

Ward, Journal of Graphics Tools, 2003
http://www.anyhere.com/gward/papers/jgtpap2.pdf

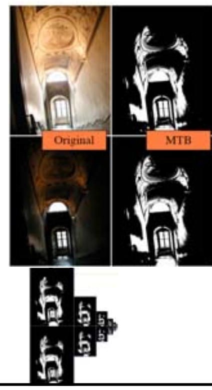
Implemented in Photosphere **http://www.anyhere.com/**

- **Image registration (no need for tripod)**
- **Lens flare removal**
- **Ghost removal**

Images Greg Ward

Image registration

- **How to robustly compare images of different exposure?**
- **Use a black and white version of the image thresholded at the median**
  - Median-Threshold Bitmap (MTB)
- **Find the translation that minimizes difference**
- **Accelerate using pyramid**



Alignment Results

5 unaligned exposures | Close-up detail | MTB alignment

Time: About .2 second/exposure for 3 MPixel image

Slide from Siggraph 2005 course on HDR

Automatic "Ghost" Removal

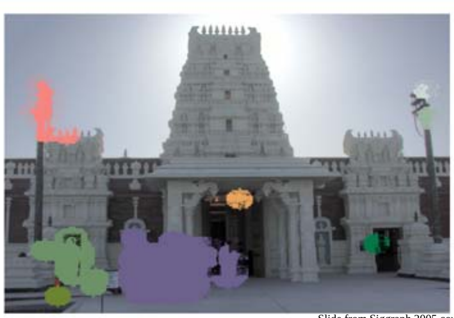
Before | After

Slide from Siggraph 2005 course on HDR

Variance-based Detection

Slide from Siggraph 2005 course on HDR

Region Masking



Slide from Siggraph 2005 course on HDR

Best Exposure in Each Region

Slide from Siggraph 2005 course on HDR

Lens Flare Removal

SIGGRAPH2005

Slide from Siggraph 2005 course on HDR

Extension: HDR video

- **Kang et al. Siggraph 2003 http://portal.acm.org/citation.cfm?id=882262.882270**

Figure 1: High dynamic range video of a driving scene. *Top row: Input video with alternating short and long exposures. Bottom row: High dynamic range video (tonemapped).*

Extension: HDR video

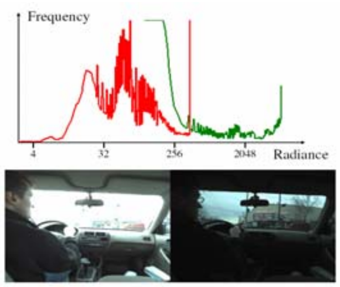


Figure 3: Two input exposures from the driving video. *The radiance histogram is shown on top. The red graph goes with the long exposure frame (bottom left), while the green graph goes with the short exposure frame (bottom right). Notice that the combination of these graphs spans a radiance range greater than a single exposure can capture.*

Questions?

HDR encoding

- **Most formats are lossless**
- **Adobe DNG (digital negative)**
  - Specific for RAW files, avoid proprietary formats
- **RGBE**
  - 24 bits/pixels as usual, plus 8 bit of common exponent
  - Introduced by Greg Ward for Radiance (light simulation)
  - Enormous dynamic range
- **OpenEXR**
  - By Industrial Light + Magic, also standard in graphics hardware
  - 16bit per channel (48 bits per pixel) 10 mantissa, sign, 5 exponent
  - Fine quantization (because 10 bit mantissa), only 9.6 orders of magnitude
- **JPEG 2000**
  - Has a 16 bit mode, lossy

HDR formats

- **Summary of all HDR encoding formats (Greg Ward): http://www.anyhere.com/gward/hdrenc/hdr_encodings.html**
- **Greg's notes: http://www.anyhere.com/gward/pickup/CIC13course.pdf**
- **http://www.openexr.com/**
- **High Dynamic Range Video Encoding**

(MPI) http://www.mpi-sb.mpg.de/resources/hdrvideo/

HDR code

- HDRShop http://gl.ict.usc.edu/HDRShop/ (v1 is free)
- Columbia's camera calibration and HDR combination with source code Mitsunaga, Nayar , Grossberg http://www1.cs.columbia.edu/CAVE/projects/rad_cal/rad_cal.php
- Greg Ward Phososphere HDR browser and image combination with regsitration (Macintosh, command-line version under Linux) with source code http://www.anyhere.com/
- Photoshop CS2
- Idruna http://www.idruna.com/photogenicshdr.html
- MPI PFScalibration (includes source code) http://www.mpii.mpg.de/resources/hdr/calibration/pfs.html
- EXR tools http://scanline.ca/exrtools/
- HDR Image Editor http://www.acm.uiuc.edu/siggraph/HDRIE/
- CinePaint http://www.cinepaint.org/
- Photomatix http://www.hdrsoft.com/
- EasyHDR http://www.astro.leszno.net/easyHDR.php
- Artizen HDR http://www.supportingcomputers.net/Applications/Artizen/Artizen.htm
- *Automated High Dynamic Range Imaging Software & Images* http://www2.cs.uh.edu/~somalley/hdri_images.html
- Optipix http://www.imaging-resource.com/SOFT/OPT/OPT.HTM

HDR images

- http://www.debevec.org/Research/HDR/
- http://www.mpi-sb.mpg.de/resources/hdr/gallery.html
- http://people.csail.mit.edu/fredo/PUBLI/Siggraph2002/
- http://www.openexr.com/samples.html
- http://www.flickr.com/groups/hdr/
- http://www2.cs.uh.edu/~somalley/hdri_images.html#hdr_others
- http://www.anyhere.com/gward/hdrenc/pages/originals.html
- http://www.cis.rit.edu/mcsl/icam/hdr/rit_hdr/
- http://www.cs.utah.edu/%7Ereinhard/cdrom/hdr.html
- http://www.sachform.de/download_EN.html
- http://lcavwww.epfl.ch/%7Elmeylan/HdrImages/February06/February06.html
- http://lcavwww.epfl.ch/%7Elmeylan/HdrImages/April04/april04.html
- http://books.elsevier.com/companions/0125852630/hdri/html/images.html

HDR Cameras

- **HDR sensors using CMOS**
  - Use a log response curve
  - e.g. SMaL,
- **Assorted pixels**
  - Fuji
  - Nayar et al.
- **Per-pixel exposure**
  - Filter
  - Integration time
- **Multiple cameras using beam splitters**
- **Other computational photography tricks**

Fuji SuperCCD

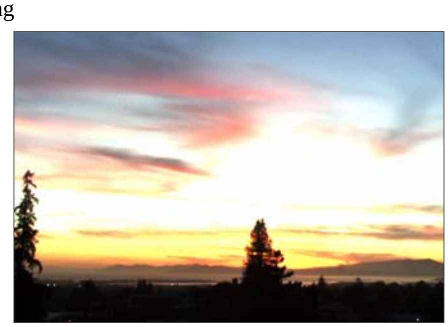
HDR cameras

- http://www.hdrc.com/home.htm
- http://www.smalcamera.com/technology.html
- http://www.cfar.umd.edu/~aagrawal/gradcam/gradcam.html
- http://www.spheron.com/spheron/public/en/home/home.php
- http://www.ims-chips.com/home.php3?id=e0841
- http://www.thomsongrassvalley.com/products/cameras/viper/
- http://www.pixim.com/
- http://www.ptgrey.com/
- http://www.siliconimaging.com/
- http://www-mtl.mit.edu/researchgroups/sodini/PABLOACO.pdf
- http://www1.cs.columbia.edu/CAVE/projects/adr_lcd/adr_lcd.php
- http://www1.cs.columbia.edu/CAVE/projects/gen_mos/gen_mos.php
- http://www1.cs.columbia.edu/CAVE/projects/pi_micro/pi_micro.php
- http://www.cs.cmu.edu/afs/cs/usr/brajovic/www/labweb/index.html

Questions?

The second half: contrast reduction

- **Input: high-dynamic-range image**
  - (floating point per pixel)

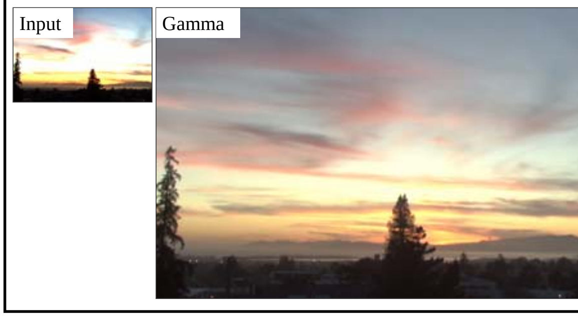


Naïve technique

- **Scene has *1:10,000* contrast, display has *1:100***
- **Simplest contrast reduction?**

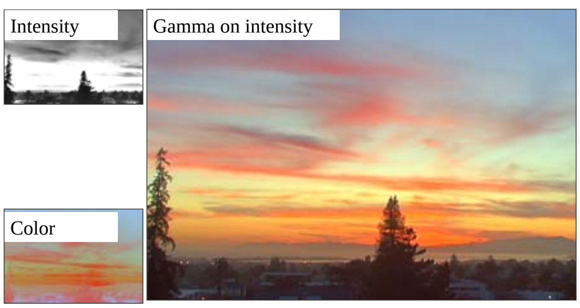
Naïve: Gamma compression

- **X –> X^{γ} (where γ=0.5 in our case)**
- **But… colors are washed-out. Why?**



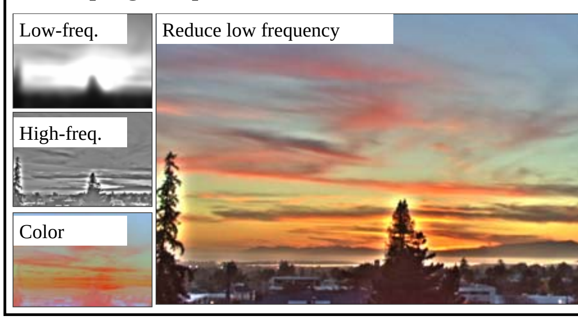
Gamma compression on intensity

- **Colors are OK,
but details (intensity high-frequency) are blurred**



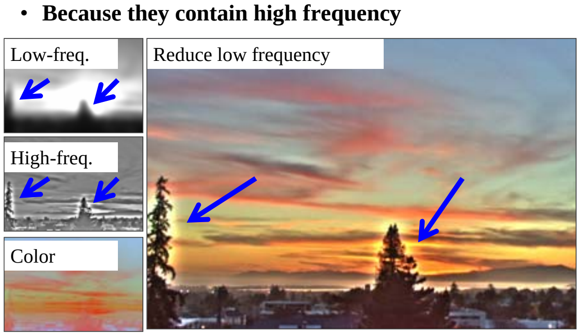
Oppenheim 1968, Chiu et al. 1993

- **Reduce contrast of low-frequencies**
- **Keep high frequencies**



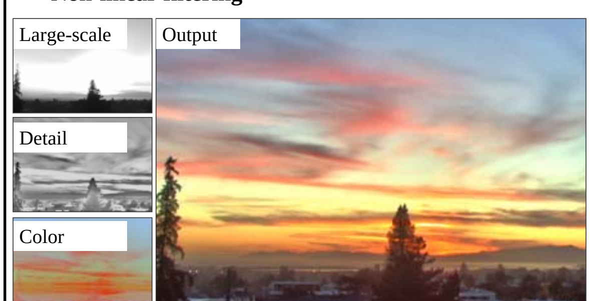
The halo nightmare

- **For strong edges**
- **Because they contain high frequency**



Our approach

- **Do not blur across edges**
- **Non-linear filtering**



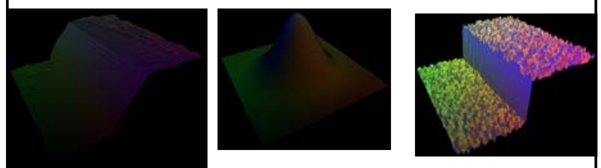
Bilateral filter

- **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

- **Here, input is a step function + noise**

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

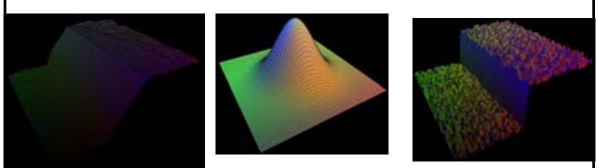


output ⟵ input

Start with Gaussian filtering

- **Spatial Gaussian f**

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

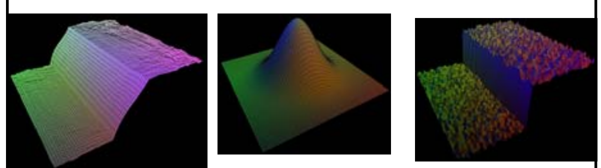


output ⟵ input

Start with Gaussian filtering

- **Output is blurred**

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

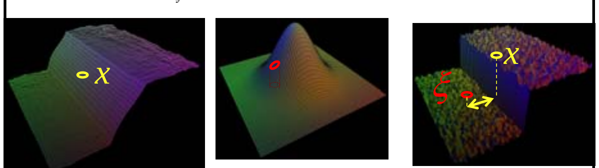


output ⟵ input

Gaussian filter as weighted average

- **Weight of ξ depends on distance to x**

$$J(x) = \sum_{\xi} \quad f(x,\xi) \quad I(\xi)$$

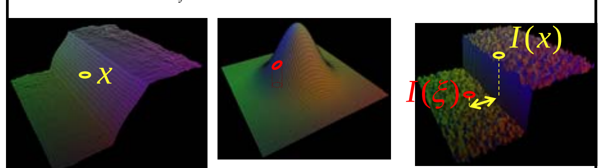


output ⟵ input

The problem of edges

- **Here, $I(\xi)$ "pollutes" our estimate J(x)**
- **It is too different**

$$J(x) = \sum_{\xi} \quad f(x,\xi) \quad I(\xi)$$



output ⟵ input

Principle of Bilateral filtering

[Tomasi and Manduchi 1998]

- **Penalty g on the intensity difference**

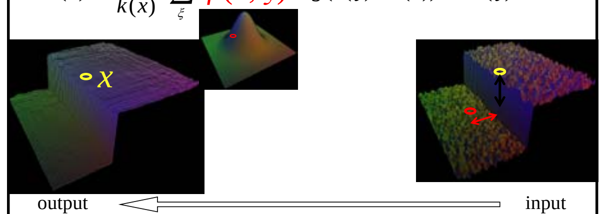
$$J(x) = \frac{1}{k(x)} \sum_{\xi} f(x,\xi) \quad g(I(\xi)-I(x)) \quad I(\xi)$$

Bilateral filtering

[Tomasi and Manduchi 1998]

- **Spatial Gaussian f**

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

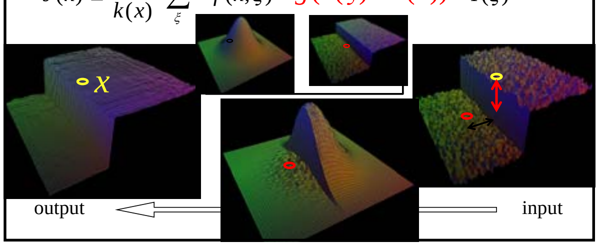


Bilateral filtering

[Tomasi and Manduchi 1998]

- **Spatial Gaussian f**
- **Gaussian g on the intensity difference**

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

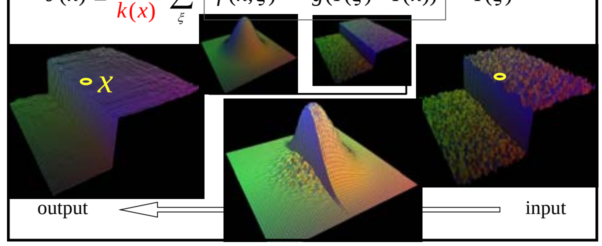


Normalization factor

[Tomasi and Manduchi 1998]

- **k(x)=** $\sum_{\xi} f(x,\xi) \quad g(I(\xi)-I(x))$

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

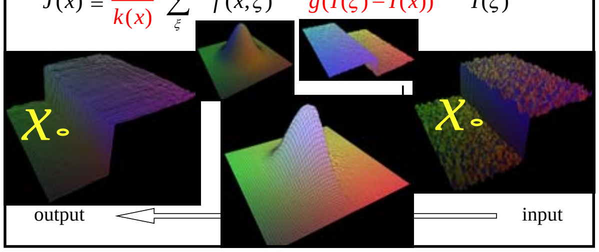


Bilateral filtering is non-linear

[Tomasi and Manduchi 1998]

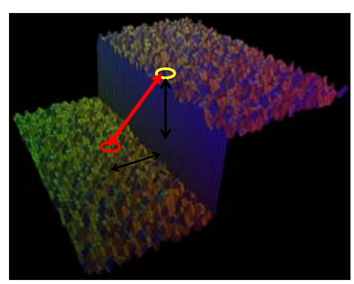
- **The weights are different for each output pixel**

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



Other view

- **The bilateral filter uses the 3D distance**

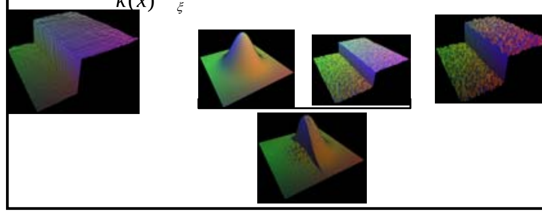


Questions?

Acceleration

- **Non-linear because of g**

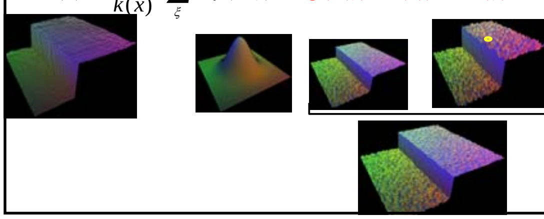
$$J(x) = \frac{1}{k(x)} \sum_{\xi} f(x,\xi) \quad g(I(\xi)-I(x)) \quad I(\xi)$$



Acceleration

- **Linear for a given value of I(x)**
- **Convolution of g I by Gaussian f**

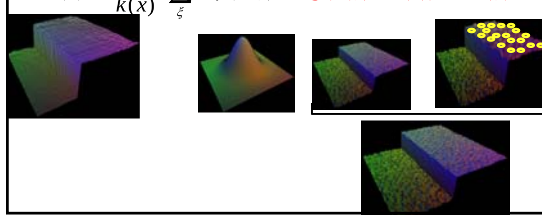
$$J(x) = \frac{1}{k(x)} \sum_{\xi} f(x,\xi) \quad g(I(\xi)-I(x)) \quad I(\xi)$$



Acceleration

- **Linear for a given value of I(x)**
- **Convolution of g I by Gaussian f**
- **Valid for all x with same value I(x)**

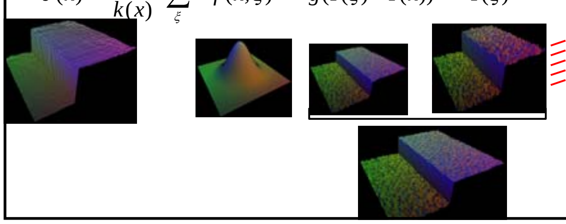
$$J(x) = \frac{1}{k(x)} \sum_{\xi} f(x,\xi) \quad g(I(\xi)-I(x)) \quad I(\xi)$$



Acceleration

- **Discretize the set of possible I(x)**
- **Perform linear Gaussian blur (FFT)**
- **Linear interpolation in between**

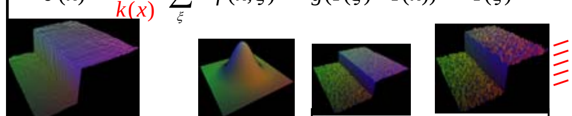
$$J(x) = \frac{1}{k(x)} \sum_{\xi} f(x,\xi) \quad g(I(\xi)-I(x)) \quad I(\xi)$$



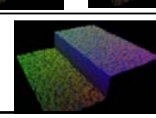
Acceleration

- **Discretize the set of possible I(x)**
- **Perform linear Gaussian blur (FFT)**
- **Linear interpolation in between**

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



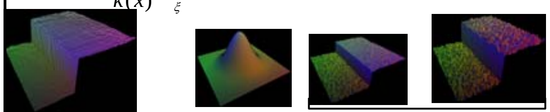
- **k(x) treated similarly**



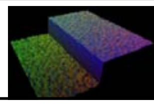
More acceleration

- **Discretize the set of possible I(x)**
- **Perform linear Gaussian blur (FFT)**
- **Linear interpolation in between**
- **Subsample in space**

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

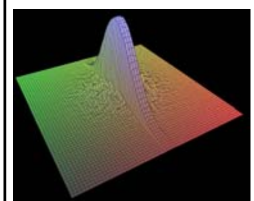


- **k(x) treated similarly**

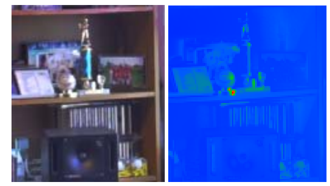


Handling uncertainty

- **Sometimes, not enough "similar" pixels**
- **Happens for specular highlights**
- **Can be detected using normalization k(x)**
- **Simple fix (average with output of neighbors)**



Weights with high uncertainty



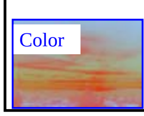
Uncertainty

Questions?

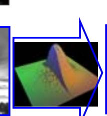
Contrast reduction

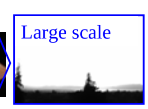
Contrast
too high!

Contrast reduction



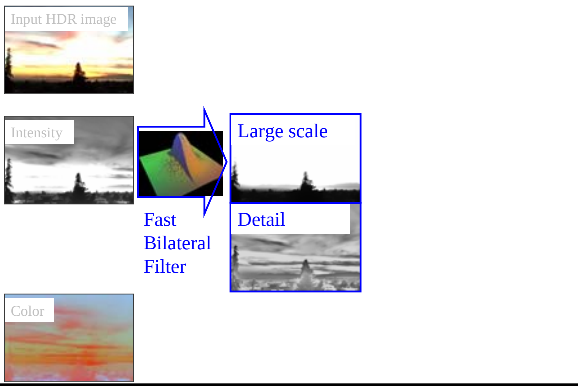
Contrast reduction



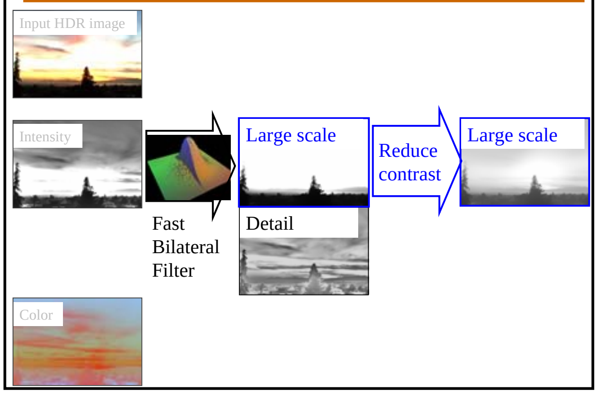


Fast
Bilateral
Filter

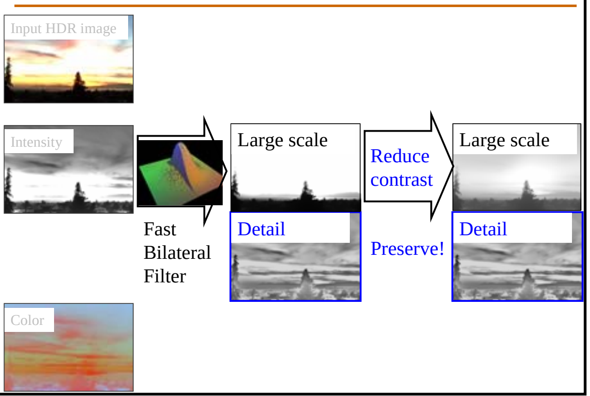
Contrast reduction



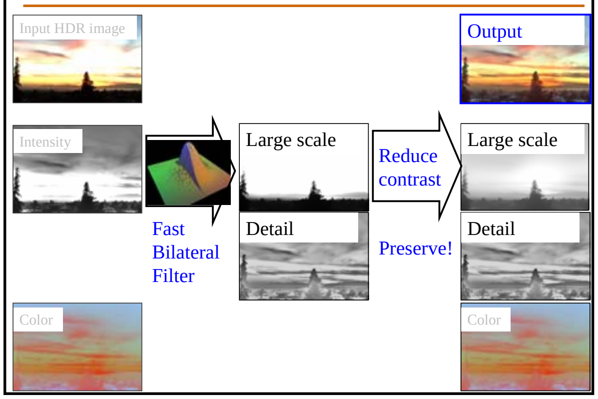
Contrast reduction



Contrast reduction



Contrast reduction

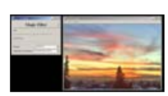


Reduction

- **To reduce contrast of base layer**
  - scale in the log domain
  - ➔ γ exponent in linear space
- **Set a target range: $\log_{10}$ (5)**
- **Compute range in the base (log) layer: (max-min)**
- **Deduce γ using an elaborate operation known as *division***
- **You finally need to normalize so that the biggest value in the (linear) base is 1 (0 in log):**
  - Offset the compressed based by its max

Live demo

- **Xx GHz Pentium Whatever PC**



Questions?

Cleaner version of the acceleration

- **Paris & Durand, ECCV 06** http://people.csail.mit.edu/sparis/#publications
- **Signal processing foundation**
- **Better accuracy**

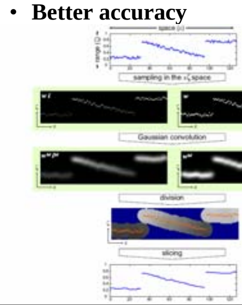
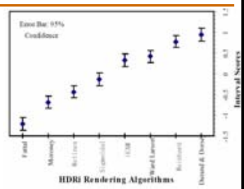


Fig. 1. Our computation pipeline applied to a 1D signal. The original data (top row) are represented by a two-dimensional function (wi, w) (second row). This function is convolved with a Gaussian kernel to form $(w^{bf} i^{bf}, w^{bf})$ (third row). The first component is then divided by the second (fourth row, blue area is undefined because of numerical limitation, $w^{bf} \approx 0$). Then the final result (last row) is extracted by sampling the former result at the location of the original data (shown in red on the fourth row).

Tone mapping evaluation

- **Recent work has performed user experiments to evaluate competing tone mapping operators**
  - Ledda et al. 2005
    http://www.cs.bris.ac.uk/Publications/Papers/2000255.pdf
  - Kuang et al. 2004
    http://www.cis.rit.edu/fairchild/PDFs/PRO22.pdf
- **Interestingly, the former concludes my method is the worst, the latter that my method is the best!**
  - They choose to test a different criterion: fidelity vs. preference
- **More importantly, they focus on algorithm and ignore parameters**



From Kuang et al.

	1st	2nd	3rd	4th	5th	6th
Scene 1	P	B	A	H	I	L
Scene 2	I	P	H	A	B	L
Scene 3	P	I	A	H	L	B
Scene 4	P	L	I	A	H	B
Scene 5	I	H	A	P	L	B
Scene 6	I	H	A	P	L	B
Scene 7	I	A	P	H	B	L
Scene 8	I	P	A	H	L	B
Scene 9	P	A	L	H	B	I

Adapted from Ledda et al.

Other tone mapping references

- **J. DiCarlo and B. Wandell, Rendering High Dynamic Range Images** http://www-isl.stanford.edu/%7Eabbas/group/papers_and_pub/spie00_jeff.pdf
- **Choudhury, P., Tumblin, J., "The Trilateral Filter for High Contrast Images and Meshes".** http://www.cs.northwestern.edu/~jet/publications.html
- **Tumblin, J., Turk, G., "Low Curvature Image Simplifiers (LCIS): A Boundary Hierarchy for Detail-Preserving Contrast Reduction."** http://www.cs.northwestern.edu/~jet/publications.html
- **Tumblin, J., "Three Methods For Detail-Preserving Contrast Reduction For Displayed Images"** http://www.cs.northwestern.edu/~jet/publications.html
- **Photographic Tone Reproduction for Digital Images**
  Erik Reinhard, Mike Stark, Peter Shirley and Jim Ferwerda http://www.cs.utah.edu/%7Ereinhard/cdrom/
- **Ashikhmin, M. ``A Tone Mapping Algorithm for High Contrast Images''** http://www.cs.sunysb.edu/~ash/tm.pdf
- **Retinex at Nasa** http://dragon.larc.nasa.gov/retinex/background/retpubs.html
- **Gradient Domain High Dynamic Range Compression Raanan Fattal, Dani Lischinski, Michael Werman** http://www.cs.huji.ac.il/~danix/hdr/
- **Li et al. : Wavelets and activity maps** http://web.mit.edu/yzli/www/hdr_companding.htm

Tone mapping code

- **http://www.mpi-sb.mpg.de/resources/pfstools/**
- **http://scanline.ca/exrtools/**
- **http://www.cs.utah.edu/~reinhard/cdrom/source.html**
- **http://www.cis.rit.edu/mcsl/icam/hdr/**

Next Time: Gradient Manipulation