Denoising & Bilateral Filtering

Lecture 5

Rob Fergus

Many slides from Fredo Durand & Sylvain Paris
Admin stuff

• Start homework 2 early!!!!

• Have grader for course (Jiali Huang jiali.huang@nyu.edu)

• Come and see me about projects!!!!
Overview of today

• Denoising
  – Averaging
  – Wiener denoising
  – Median filtering

• Bilateral filtering
  – Cross-bilateral filter
  – Flash applications
Noisy image

- Usually for dark conditions
Noise

- Fluctuation when taking multiple shots
Canon 1D mark II N at ISO 3200
Histogram of grey patch

- Should be single values for RGB (constant color)
Where Does Noise come from?
Digital pipeline

• Photosites transform photons into charge (electrons)
  – The sensor itself is linear
• Gets amplified (depending on ISO setting)
• Then goes through analog to digital converter
  – up to 14 bits/channel these days

• Stop here when shooting RAW

• Then demosaicing, denoising, white balance, a response curve, gamma encoding are applied
• Quantized and recorded as 8-bit JPEG

F. Durand
Pipeline & noise

- This is a conceptual diagram, don’t take it too literally
  - e.g. the AD converter is a serious source of noise, but usually electronic noise, not quantization artifacts

- Orders of magnitude:
  - # of photons per photosite: 10,000-100,000
  - Electronic noise 5-30 electrons per photosite

F. Durand
ISO amplifies

- e.g. going from ISO 100 to ISO 400 amplifies by 4
- both noise & signal
- usually use high ISO when signal is low
- => worse signal/noise ratio
Canon 1D mark II N at ISO 3200

F. Durand
Canon 1D Mark II, ISO 100

• Lot less noisy!
http://wiegaertnerfilms.com/tutorials/the-best-iso-settings-for-canon-video-dslrs/

F. Durand
Denoising by Averaging
Denoising a single image
Denoising from 1 image

- We can’t take average over multiple images

Noisy input

F. Durand
Denoising from 1 image

• We can’t take average over multiple images
• Idea 1: take a spatial average
  • Most pixels have roughly the same color as their neighbor
  • Noise looks high frequency => do a low pass
• Here: Gaussian blur

F. Durand
Gaussian blur

- Noise is mostly gone
- But image is blurry
  - duh!

F. Durand
Weiner Denoising
Wiener denoising derivation

• See http://www.cs.dartmouth.edu/farid/tutorials/fip.pdf
• Pages 57→59
Wiener denoising

- Simplest model [Lee80]: every neighborhood as a Gaussian vector sample with unknown variance

\[ \mu = \sum_{n=1}^{N} x_n \]
\[ \sigma_s^2 = \sum_{n=1}^{N} (x_n - \mu)^2 - \sigma_\omega^2 \]

\[ \hat{x} = \mu + \frac{\hat{\sigma}_s^2}{\hat{\sigma}_s^2 + \sigma_\omega^2} (x - \mu) \]

Local Wiener Estimation

White Gaussian noise power (assumed to be known)

From J. Portilla ICIP’01
Observed Sample

- Wiener2 Matlab function
- 5x5 pixels neighborhood

Local Wiener Filtered
Residual

20.1 dB
28.1 dB
Denoising salt’n’pepper noise
Median filter

Replace each pixel by the median over N pixels (5 pixels, for these examples). Generalizes to “rank order” filters.

Median([1 7 1 5 1]) = 1
Mean([1 7 1 5 1]) = 2.8

5-pixel neighborhood

Spike noise is removed
Monotonic edges remain unchanged
Median filtering results

Best for salt and pepper noise

http://homepages.inf.ed.ac.uk/rbf/HIPR2/mean.htm#guidelines
Bilateral filtering
Gaussian blur

• Noise is mostly gone
• But image is blurry
  • duh!

• Problem: not all neighbors have the same color

• Bilateral filter idea: only consider neighbors that have values similar enough

F. Durand
A Gentle Introduction to Bilateral Filtering and its Applications

“Fixing the Gaussian Blur”: the Bilateral Filter

Sylvain Paris – MIT CSAIL
Fredo- Durand – MIT CSAIL
Blur Comes from Averaging across Edges

Same Gaussian kernel everywhere.
Bilateral Filter [Aurich 95, Smith 97, Tomasi 98]

No Averaging across Edges

The kernel shape depends on the image content.
Bilateral filter

• Tomasi and Manduci 1998]
  – http://www.cse.ucsc.edu/~manduchi/Papers/ICCV98.pdf
• Developed for denoising
• 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
• Full survey:
  Paris_09_Bilateral_filtering.pdf

F. Durand
Bilateral Filter Definition: an Additional Edge Term

Same idea: weighted average of pixels.

\[
BF[I]_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s}(\| p - q \|) G_{\sigma_r}(\| I_p - I_q \|) I_q
\]

- **new** normalization factor
- **not new** space weight
- **new** range weight
Illustration a 1D Image

- 1D image = line of pixels

- Better visualized as a plot
Gaussian Blur and Bilateral Filter

Gaussian blur

Bilateral filter
[Aurich 95, Smith 97, Tomasi 98]

\[ GB[I]_p = \sum_{q \in S} G_{\sigma}(\| p - q \|) I_q \] space

\[ BF[I]_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s}(\| p - q \|) G_{\sigma_r}(\| I_p - I_q \|) I_q \] space

normalization

range
Bilateral Filter on a Height Field

\[
BF[I]_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s}(\|p - q\|) G_{\sigma_r}(\|I_p - I_q\|) I_q
\]
Space and Range Parameters

\[ BF[I]_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s} (\| p - q \|) G_{\sigma_r} (\| I_p - I_q \|) I_q \]

- space \( \sigma_s \): spatial extent of the kernel, size of the considered neighborhood.

- range \( \sigma_r \): “minimum” amplitude of an edge
Influence of Pixels

Only pixels close in space and in range are considered.
Bilateral filter

Noisy input

After bilateral filter
Exploring the Parameter Space

\[ \sigma_s = 2 \]

\[ \sigma_s = 6 \]

\[ \sigma_s = 18 \]

\[ \sigma_r = 0.1 \]

\[ \sigma_r = 0.25 \]

\[ \sigma_r = \infty \]

(Gaussian blur)
Varying the Range Parameter

\[ \sigma_s = 2 \]

\[ \sigma_s = 6 \]

\[ \sigma_s = 18 \]

\[ \sigma_r = 0.1 \]

\[ \sigma_r = 0.25 \]

\[ \sigma_r = \infty \]

(Gaussian blur)
σ_r = 0.1
\( \sigma_r = 0.25 \)
\[ \sigma_r = \infty \]
(Gaussian blur)
Varying the Space Parameter

- $\sigma_s = 2$
- $\sigma_s = 6$
- $\sigma_s = 18$

- $\sigma_r = 0.1$
- $\sigma_r = 0.25$
- $\sigma_r = \infty$

(Gaussian blur)
$\sigma_s = 6$
How to Set the Parameters

Depends on the application. For instance:

• space parameter: proportional to image size
  – e.g., 2% of image diagonal

• range parameter: proportional to edge amplitude
  – e.g., mean or median of image gradients

• independent of resolution and exposure
Bilateral Filter Crosses Thin Lines

- Bilateral filter averages across features thinner than $\sim 2\sigma_s$
- Desirable for smoothing: more pixels = more robust
- Different from diffusion that stops at thin lines
Bilateral Filtering Color Images

For gray-level images

\[ BF[I]_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s}(\| p - q \|) G_{\sigma_r}(\| I_p - I_q \|) I_q \]

For color images

\[ BF[I]_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s}(\| p - q \|) G_{\sigma_r}(\| C_p - C_q \|) C_q \]
Hard to Compute

- Nonlinear
  \[ BF[I]_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s}(\|p - q\|) G_{\sigma_r}(\|I_p - I_q\|) I_q \]

- Complex, spatially varying kernels
  - Cannot be precomputed, no FFT...

- Brute-force implementation is slow > 10min
Basic denoising

Noisy input

Bilateral filter 7x7 window
Basic denoising

Bilateral filter

Median 3x3
Basic denoising

Bilateral filter vs Median 5x5
Basic denoising

Bilateral filter

Bilateral filter – lower sigma
Basic denoising

Bilateral filter

Bilateral filter – higher sigma
Denoising

- Small spatial sigma (e.g. 7x7 window)
- Adapt range sigma to noise level
- Maybe not best denoising method, but best simplicity/quality tradeoff
  - No need for acceleration (small kernel)
  - But the denoising feature in e.g. Photoshop is better
Ordinary Bilateral Filter

Bilateral $\rightarrow$ two kinds of weights, one image $A$:

$$BF[ A]_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s}(\| p - q \|) \cdot G_{\sigma_r}(\| A_p - A_q \|) \cdot A_q$$

Image $A$:
NEW: two kinds of weights, two images

\[ BF[A]_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s} (\| p - q \|) G_{\sigma_r} (| B_p - B_q |) A_q \]

A: Noisy, dim (ambient image)

B: Clean, strong (Flash image)
Image A: Warm, shadows, but too Noisy
(too dim for a good quick photo)
Image B: Cold, Shadow-free, Clean

(Flash: simple light, ALMOST no shadows)
MERGE BEST OF BOTH: apply ‘Cross Bilateral’ or ‘Joint Bilateral’
(it really is much better!)
Dark Flash Photography

Dilip Krishnan
Rob Fergus

Dept. of Computer Science,
Courant Institute,
New York University
Our Camera & Dark Flash

Dark Flash

Emits Ultraviolet (UV) and Infrared (IR) light just outside visible wavelength range
Dark Flash Photography

- Dark flash is ~200 times dimmer than conventional

1. Dark Flash image
Key Challenges

1. How to add light to the scene without it being perceived by people.

2. How to obtain an image with correct colors.
Key Challenges

1. How to add light to the scene without it being perceived by people.

2. How to obtain an image with correct colors.
Dark Flash Emission Spectrum

Camera Spectral Sensitivity

Dark Flash Emission
Flash Safety

- Government tables specify safe limits of exposure to UV (< 400nm)

- Safety limit is 115,000 flashes per day @ 1m

- Each flash equivalent to being outside for 1/100th second

---

**Dark Flash**

**Hazard Factor**

Visible
Key Challenges

1. How to add light to the scene without it being perceived by people.

2. How to obtain an image with correct colors.
Two Images: Five Spectral Bands

1. Dark Flash
   - “Blue” channel records UV
   - “Red” channel records IR
2. Ambient
   - R
   - G
   - B

Assumptions

1. Little ambient UV and IR light
2. UV/IR flash dominates ambient visible light
Ambient: 1/20th sec  Reconstruction  Long exposure: 4 sec
Ambient (Fraction of Normal Illumination)

1/64th

1/90th

1/256th

Reconstruction
Ambient Illumination: 1/40th Normal Lighting
Reconstruction

Reconstruction

Zoom-in
Limitations – Lack of edges in UV/IR

Ambient - Fraction of normal illumination

Dark flash  1/45th  1/90th  1/256th

Long Exposure  Reconstruction
High Dynamic Range Imaging
Real world dynamic range

- Eye can adapt from \( \sim 10^{-6} \) to \( 10^6 \) cd/m\(^2\)
- Often 1 : 10,000 in a scene
Picture dynamic range

- Typically 1: 20 or 1:50
  - Black is ~ 50x darker than white

Real world

\[ 10^{-6} \quad \text{to} \quad 10^6 \]

Picture

\[ 10^{-6} \quad \text{to} \quad 10^6 \]
Multiple exposure photography

- Merge multiple exposure to cover full range

Real world $10^{-6}$

High dynamic range $10^6$

- We obtain one single image with floats per pixel
  - But we still can’t display it
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?

Pixel value → scene value
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

\[ \Delta t = \frac{1}{100} \text{ sec} \]

\[ \Delta t = 1 \text{ sec} \]

\[ \Delta t = \frac{1}{10} \text{ sec} \]

\[ \Delta t = \frac{1}{1000} \text{ sec} \]

\[ \Delta t = 10 \text{ sec} \]

Pixel Value \( Z = f(\text{Exposure}) \)

Exposure = Radiance \( \times \) \( \Delta t \)

\[ \log \text{Exposure} = \log \text{Radiance} + \log \Delta t \]

Slide adapted from Alyosha Efros who borrowed it from Paul Debevec.

\( \Delta t \) 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 curve

Slide stolen from Alyosha Efros who stole it from Paul Debevec
Reconstructed radiance map

Slide stolen from Fredo Durand who stole it from Alyosha Efros who stole it from Paul Debevec
Problem: Contrast reduction

- Match limited contrast of the medium
- Preserve details
Tone mapping

- 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

• $X \rightarrow X^\gamma$ (where $\gamma=0.5$ 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 (log domain)
- Keep high frequencies

Low-freq.

Reduce low frequency

High-freq.

Color
The halo nightmare

- For strong edges
- Because they contain high frequency

Low-freq. → Reduce low frequency
High-freq. →

Color
Bilateral filtering to the rescue

- Large scale = bilateral (log intensity)
- Detail = residual

[Durand & Dorsey 2002]
Contrast reduction

Input HDR image

Contrast too high!
Contrast reduction

Input HDR image

Intensity

Color
Contrast reduction

Input HDR image

Intensity

Bilateral Filter
(in log domain!)

Spatial sigma: 2% image size
Range sigma: 0.4 (in log 10)

Large scale

Color
Contrast reduction

Input HDR image

Intensity

Color

Bilateral Filter

Large scale

Detail

Detail = log intensity – large scale (residual)
Contrast reduction

Input HDR image

Intensity

Bilateral Filter

Large scale

Reduce contrast

Detail

Large scale

Color
Contrast reduction

Input HDR image

Intensity

Bilateral Filter

Large scale

Detail

Reduce contrast

Preserve!

Large scale

Detail
Contrast reduction

Input HDR image

Intensity

Bilateral Filter

Large scale

Detail

Reduce contrast

Preserve!

Large scale

Detail

Color

Output

Color
Contrast reduction in log domain

- Set target large-scale contrast (e.g. $\log_{10} 10$)
  - In linear output, we want 1:10 contrast for large scale
- Compute range of input large scale layer:
  - $\text{largeRange} = \max(\text{inLogLarge}) - \min(\text{inLogLarge})$
- Scale factor $k = \log_{10} (10) / \text{largeRange}$
- Normalize so that the biggest value is 0 in log

$$\text{outLog} = \text{inLogDetail} + \text{inLogLarge} \times k - \max(\text{inLogLarge})$$