CS448f: Image Processing For Photography and Vision

Deconvolution
Assignment 3

• Competition
• Lessons Learned
Project

- Proposals due Thursday
- Everyone should have a pretty good idea of what they plan to do at this stage
- Presentations begin next Tuesday
- Schedule?
## Problems in Photography

<table>
<thead>
<tr>
<th></th>
<th>Linear Filters</th>
<th>Non-Linear Filters</th>
<th>Alignment</th>
<th>Wavelets</th>
<th>Gradient Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Misfocus or Lens Blur</td>
<td>Sharpening</td>
<td><strong>Sharpening</strong></td>
<td>Focal Stacks</td>
<td>Sharpening</td>
<td>?</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Panoramas</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Motion Blur</td>
<td>Sharpening</td>
<td>Sharpening</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Noise</td>
<td>Blurring</td>
<td><strong>Bilateral Nonlocal Means</strong></td>
<td>Aligned Averaging</td>
<td>Wavelet Shrinkage</td>
<td>?</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dynamic Range</td>
<td>?</td>
<td><strong>Tone-Mapping</strong></td>
<td>HDR Acquisition</td>
<td>?</td>
<td>Tone-Mapping</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Composition</td>
<td>Multi-Band Blending</td>
<td>?</td>
<td>Panoramas</td>
<td>?</td>
<td>Poisson Blending</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Problems in Photography

<table>
<thead>
<tr>
<th></th>
<th>Linear Filters</th>
<th>Non-Linear Filters</th>
<th>Alignment</th>
<th>Wavelets</th>
<th>Gradient Domain</th>
<th>Deconvolution</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Misfocus or Lens Blur</strong></td>
<td>Sharpening</td>
<td><strong>Sharpening</strong></td>
<td>Focal Stacks Panoramas</td>
<td>Sharpening</td>
<td>?</td>
<td>✓</td>
</tr>
<tr>
<td><strong>Motion Blur</strong></td>
<td>Sharpening</td>
<td>Sharpening</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>✓</td>
</tr>
<tr>
<td><strong>Noise</strong></td>
<td>Blurring</td>
<td><strong>Bilateral Nonlocal Means</strong></td>
<td>Aligned Averaging</td>
<td>Wavelet Shrinkage</td>
<td>?</td>
<td>×</td>
</tr>
<tr>
<td><strong>Dynamic Range</strong></td>
<td>?</td>
<td>Tone-Mapping</td>
<td>HDR Acquisition</td>
<td>?</td>
<td>Tone-Mapping</td>
<td>?</td>
</tr>
<tr>
<td><strong>Composition</strong></td>
<td>Multi-Band Blending</td>
<td>?</td>
<td>Panoramas</td>
<td>?</td>
<td>Poisson Blending</td>
<td>?</td>
</tr>
</tbody>
</table>
Motion Blur (Handheld 200mm 1/50 s)
Motion Blur (Handheld 200mm, 1/50s)
Less Motion Blur (1/640s)
Motion Blur (Rolling the camera)
Motion Blur = Convolution
Convolution = Linear Operator

- $\text{Image} * \text{kernel} = \text{blurry}$
- $K m = b$
  - $K = \text{the blur (may or may not be known)}$
  - $m = \text{the unknown good image}$
  - $b = \text{the known blurry image}$
- $K$ is known = nonblind deconvolution
- $K$ is unknown = blind deconvolution
Estimating K

• Include an accelerometer
• Look for the path traced by bright points
• Bounce back and forth between estimating $K$ and estimating $m$
  – Deconvolution using Natural Image Priors
    • Levin et al. 2007
Deconvolution = Least Squares

• Assuming we know K
• Find m such that Km = b
• Alternatively, minimize \((Km-b)^2\)
Solution Methods: Input
Solution Methods: Gradient Descent
Solution Methods: Richardson-Lucy

- $m \ast= K^T(b/(Km))$
- Like a multiplicative gradient descent
- Each step conserves average brightness in each region
- ImageStack -load blurry.tmp -loop --dup --load kernel.tmp --pull 1 --convolve --pull 1 --pop --load blurry.tmp --divide --load kernel.tmp --flip x --flip y --pull 1 --convolve --pull 1 --pop --multiply --save rl.tmp --display
High-Frequency Junk
Priors

• The result image above satisfies the equation:
  – $Km = b$

• Why does it look bad?
Priors

• The result image above satisfies the equation:
  – $K_m = b$

• Why does it look bad?

• There’s extra high-frequency junk
Gradient Magnitude

Original

Richardson Lucy Result
Gradient Magnitude

Original

Richardson Lucy Result
Let’s also minimize gradients

- $K_m = b$
- $D_x m = 0$
- $D_y m = 0$

- Solving this least-squares minimizes:
  
  $|K_m - b|^2 + |D_x m|^2 + |D_y m|^2$
  
  = L2-norm of error + L2-norm of gradient field
Let $m =$ correct answer

$|Km - b|^2 \quad |D_x m|^2 + |D_y m|^2$
Let $m = \text{Richardson Lucy}$

| $|K_m - b|^2$ | $|D_x m|^2 + |D_y m|^2$ |
Let $m = \text{blurry input}$

\[ |Km - b|^2 \]

\[ |D_x m|^2 + |D_y m|^2 \]
Gradient Magnitude is a Bad Prior

• It strongly prefers blurry output if at all possible
• The prior and the error fight each other
• What’s a better prior?
Strong Gradients are Sparse
Strong Gradients are Sparse
Our old prior:

- Original Grad $^2$
- Motion-Blurred Grad $^2$
Slightly better to count the number of large edges, and minimize that.
Given a black-white transition...

Sum of gradients raised to power $< 1$ prefers sharp edges:

Sum of gradients raised to power $> 1$ prefers smooth edges:
Optimization

• Solving this least-squares minimizes:
  – $|Km - b|^2 + |D_xm|^2 + |D_ym|^2$

• We want to minimize something like this:
  – $|Km - b|^2 + |D_xm|^{1/2} + |D_ym|^{1/2}$

• No longer a convex optimization problem...

• Can still use gradient descent to find a local minima
  – it picks a sensible looking place for each edge
Some results

- http://graphics.ucsd.edu/~neel/dissertation/chapter5results/
More Fun in the Gradient Domain

• So if gradients should be sparse, and we see a gradient that looks like this:

```
0 0 1 3 5 6 4 1 0 0 0
```

• Why not convert it to this:

```
0 0 1 3 5 6 4 1 0 0 0
0 0 0 0 1 3 15 1 0 0 0
```
More Fun in the Gradient Domain

• If it works: call it deblurring

• If it doesn’t: call it a “painterly effect”