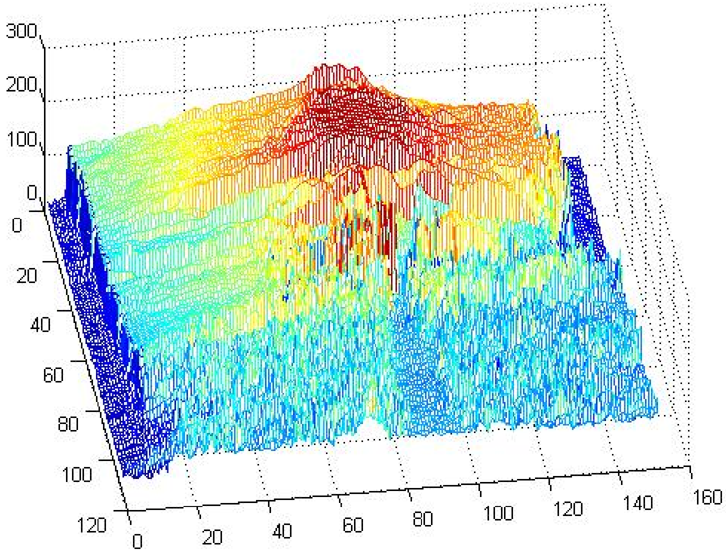
Gradient Domain HDR Compression

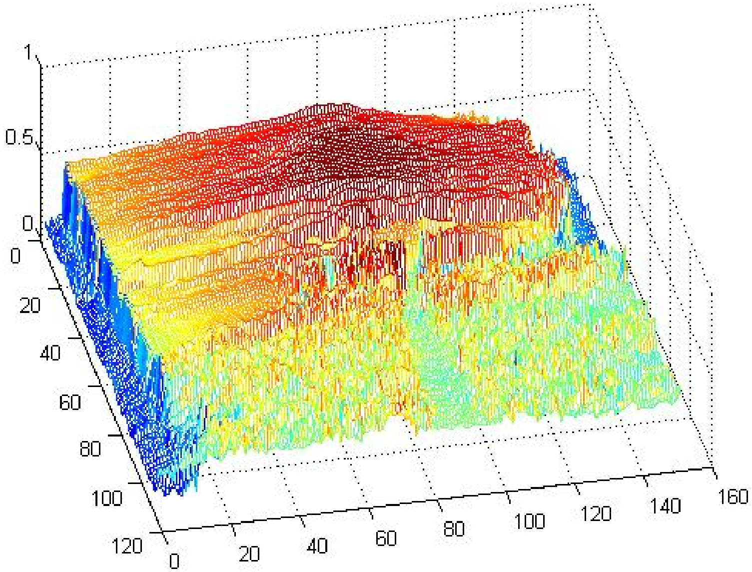
- *Gradient Domain High Dynamic Range Compression*
  - Fattal et al.
  - SIGGRAPH 2002
  - *Image Smoothing via L0 Gradient Minimization*
    - Xu et al.
    - SIGGRAPH Asia 2011

Problem

- Developing a technique for high dynamic range (HDR) compression that enables HDR images to be displayed on low dynamic range devices



you might lose the details after applying direct compression



Preserving Details Using Decomposition

- Decompose luminance into reflectance and illumination

$$I(x,y) = R(x,y)\,L(x,y)$$

- Compress the illumination, which is of high dynamic range, and re-multiply the reflectance and the compressed illumination to get a displayable image

$$\tilde{I}(x,y) = R(x,y)\,\tilde{L}(x,y)$$

Observation

- Drastic changes in the luminance across a high dynamic range image must give rise to large magnitude luminance gradients at some scale

- Fine details, such as texture, correspond to gradients of much smaller magnitude

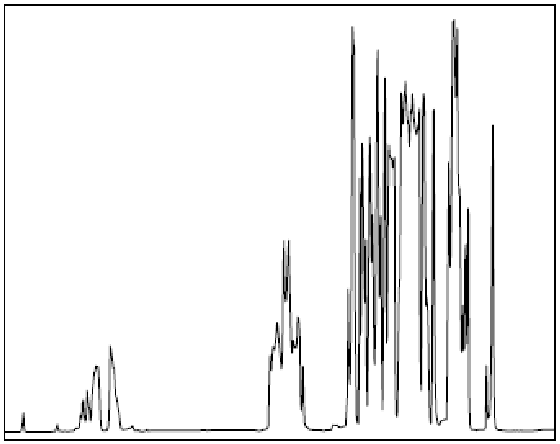
The Idea of Gradient Attenuation

- Identify large gradients at various scales, and attenuate their magnitudes while keeping their direction unaltered

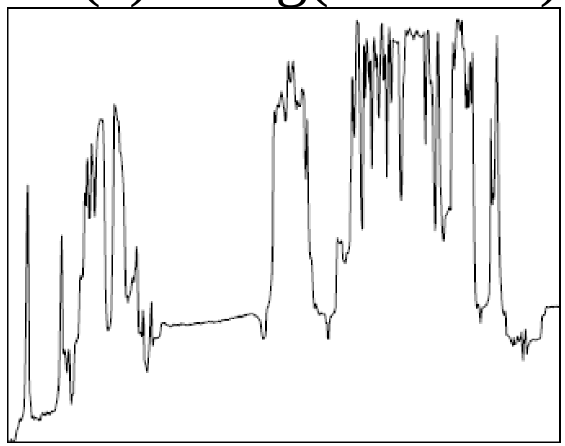
- The attenuation is progressive, penalizing larger gradients more heavily than smaller ones, thus compressing drastic luminance changes, while preserving fine details.

A One-Dimensional Example

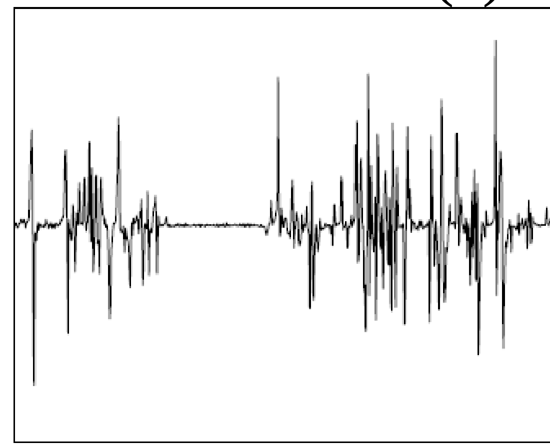
an HDR scanline



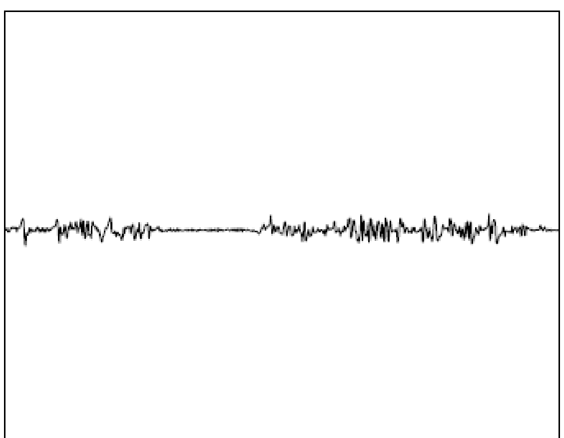
$H(x)$ = log(scanline)



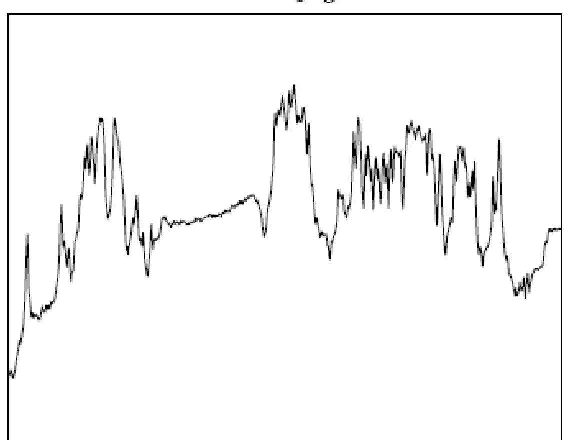
derivatives $H'(x)$



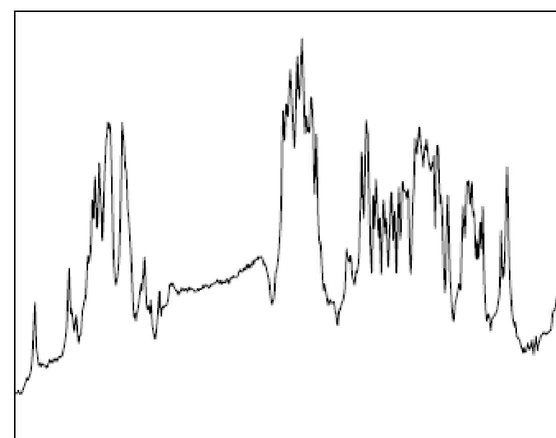
$$G(x) = H'(x)\,\Phi(x)$$



$$I(x) = C + \int_0^x G(t)\,dt$$



$\exp(I(x))$



Extension to HDR Images

In 2D:

$$G(x,y) = \nabla H(x,y)\ \Phi(x,y)$$

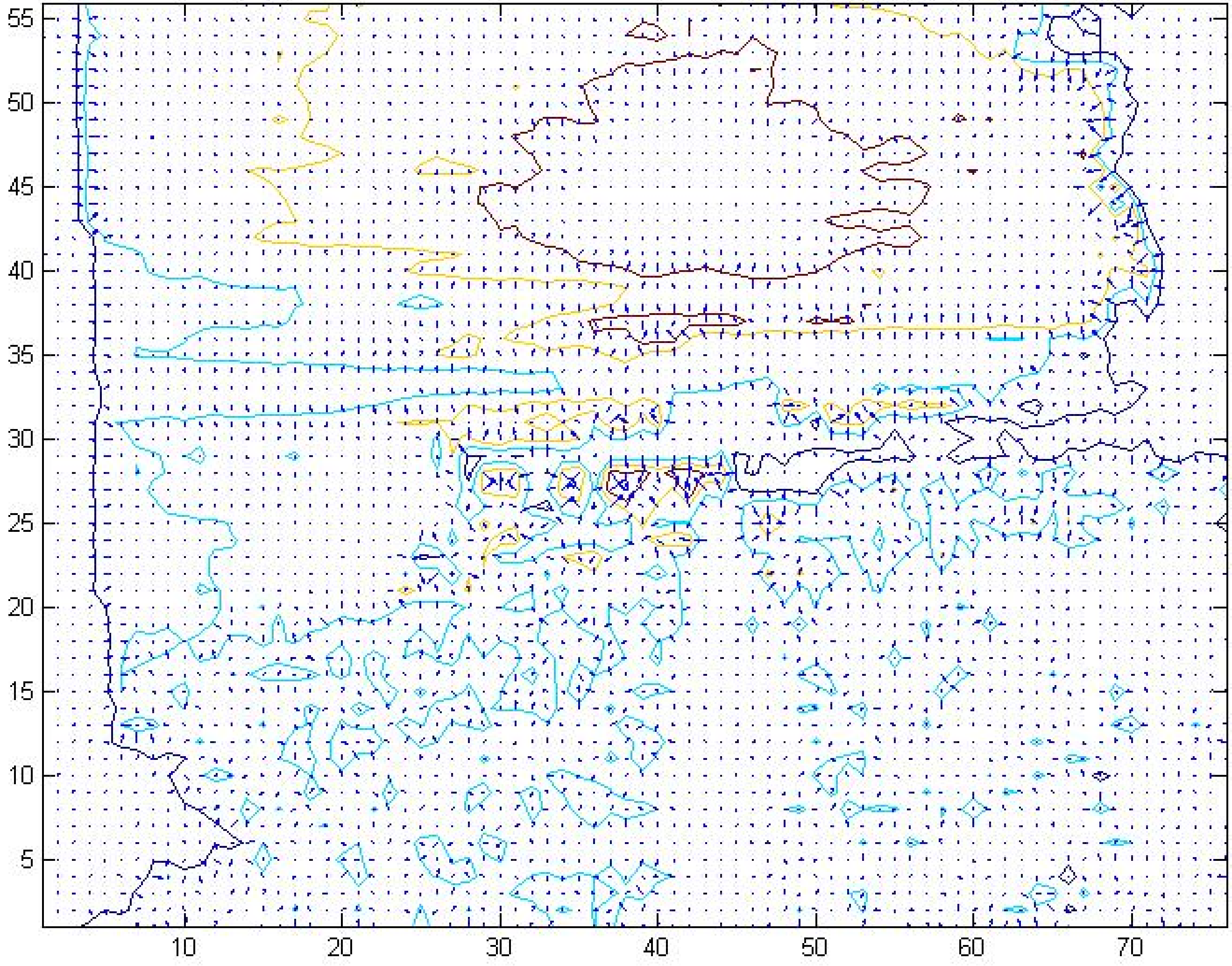
There might not exist an image I such that $G = \nabla I$

$\nabla I = (\partial I/\partial x, \partial I/\partial y)$ must satisfy

$$\frac{\partial^2 I}{\partial x \partial y} = \frac{\partial^2 I}{\partial y \partial x}$$

which is rarely the case for the attenuated gradient G

```
[px,py]=gradient(I);
contour(I), hold on, quiver(px,py), hold off
```



How to find an I whose gradient looks like the attenuated gradient G?

There might not exist an image I such that $G = \nabla I$

Least Squares

- Minimize the functional

$$\iint F(\nabla I, G)\, dx\, dy$$

where $F(\nabla I, G) = \|\nabla I - G\|^2 = \left(\frac{\partial I}{\partial x} - G_x\right)^2 + \left(\frac{\partial I}{\partial y} - G_y\right)^2$

Euler-Lagrange equation (Calculus of Variations)

$$\frac{\partial F}{\partial I} - \frac{d}{dx}\frac{\partial F}{\partial I_x} - \frac{d}{dy}\frac{\partial F}{\partial I_y} = 0$$

$$\frac{\partial F}{\partial I} - \frac{d}{dx}\frac{\partial F}{\partial I_x} - \frac{d}{dy}\frac{\partial F}{\partial I_y} = 0$$

$$F(\nabla I, G) = \|\nabla I - G\|^2 = \left(\tfrac{\partial I}{\partial x} - G_x\right)^2 + \left(\tfrac{\partial I}{\partial y} - G_y\right)^2$$

$$2\left(\frac{\partial^2 I}{\partial x^2} - \frac{\partial G_x}{\partial x}\right) + 2\left(\frac{\partial^2 I}{\partial y^2} - \frac{\partial G_y}{\partial y}\right) = 0.$$

We Get a Poisson Equation

$$\nabla^2 I = \operatorname{div} G$$

$$2\left(\frac{\partial^2 I}{\partial x^2} - \frac{\partial G_x}{\partial x}\right) + 2\left(\frac{\partial^2 I}{\partial y^2} - \frac{\partial G_y}{\partial y}\right) = 0.$$

Laplacian operator $\nabla^2 I = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$

divergence $\operatorname{div} G = \frac{\partial G_x}{\partial x} + \frac{\partial G_y}{\partial y}$

Solving the Poisson Equation for I

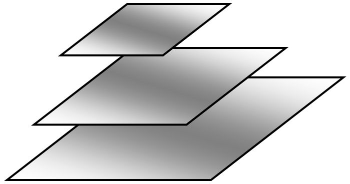
- We need to solve the Poisson equation

$$\nabla^2 I = \operatorname{div} G$$

to obtain I

Gradient Attenuation Function

$$G(x,y) = \nabla H(x,y)\,\boxed{\Phi(x,y)}$$



- Gaussian pyramid $H_0, H_1, \ldots, H_d$

$$\nabla H_k = \left(\frac{H_k(x+1,y) - H_k(x-1,y)}{2^{k+1}}, \frac{H_k(x,y+1) - H_k(x,y-1)}{2^{k+1}} \right)$$

$$\varphi_k(x,y) = \frac{\alpha}{\|\nabla H_k(x,y)\|} \left(\frac{\|\nabla H_k(x,y)\|}{\alpha} \right)^{\beta} \qquad \beta < 1$$

α is set to 0.1 times the average gradient magnitude

The Full Gradient Attenuation Function

- Coarse to fine

$$
\begin{aligned}
\Phi_d(x,y) &= \varphi_d(x,y) \\
\Phi_k(x,y) &= L\left(\Phi_{k+1}\right)(x,y)\ \varphi_k(x,y) \\
\Phi(x,y) &= \Phi_0(x,y)
\end{aligned}
$$

upsampling

- Attenuate the large gradients in different scales darker shade → stronger attenuation

How to Solve a Poisson Equation $\nabla^2 I = \operatorname{div} G$

- Boundary condition $\nabla I \cdot \mathbf{n} = 0$
- The solution is defined up to a single additive term

central difference

$$\nabla^2 I(x,y) \approx I(x+1,y) + I(x-1,y) + I(x,y+1) + I(x,y-1) - 4I(x,y)$$

forward difference

$$\nabla H(x,y) \approx (H(x+1,y) - H(x,y), H(x,y+1) - H(x,y))$$

backward difference

$$\boxed{G(x,y) = \nabla H(x,y)\ \Phi(x,y)}$$

$$\operatorname{div} G \approx G_x(x,y) - G_x(x-1,y) + G_y(x,y) - G_y(x,y-1)$$

Finite Difference

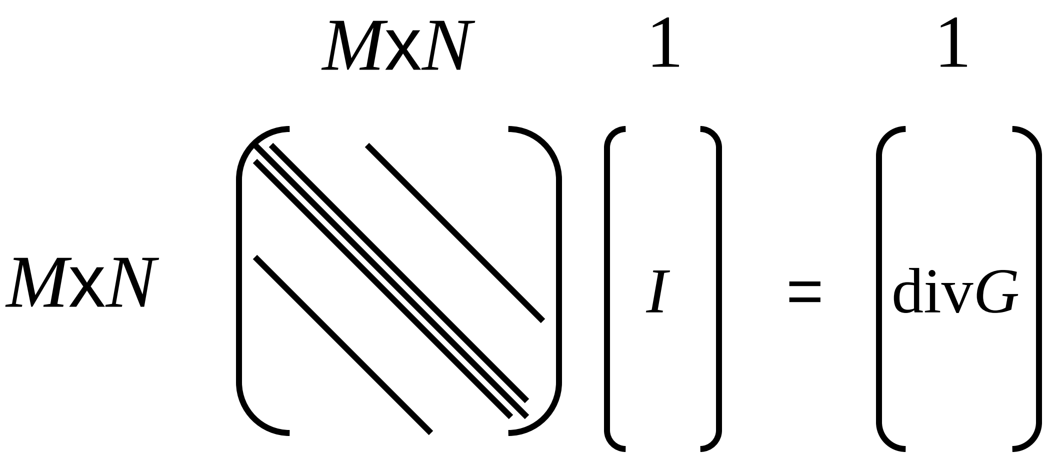
How to Solve a Poisson Equation $\nabla^2 I = \operatorname{div} G$

$$\nabla I \cdot \mathbf{n} = 0$$

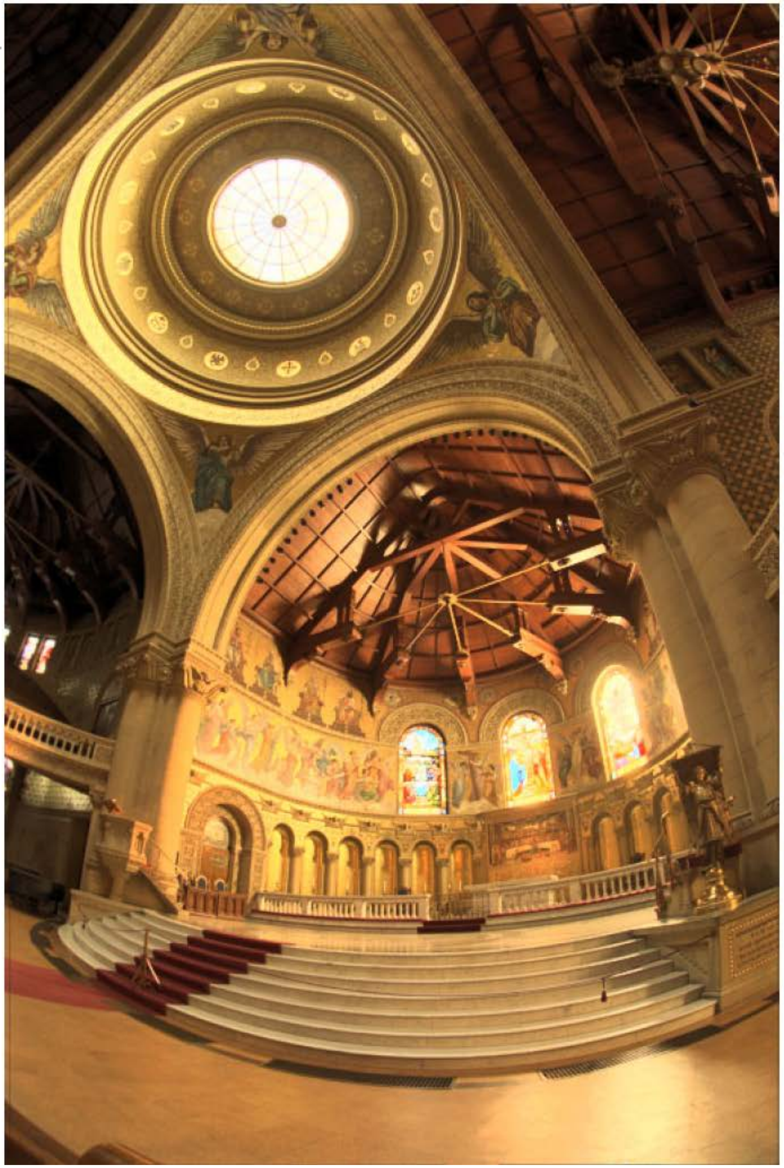
$$\nabla^2 I(x,y) \approx I(x+1,y) + I(x-1,y) + I(x,y+1) + I(x,y-1) - 4I(x,y)$$

Sparse Linear System

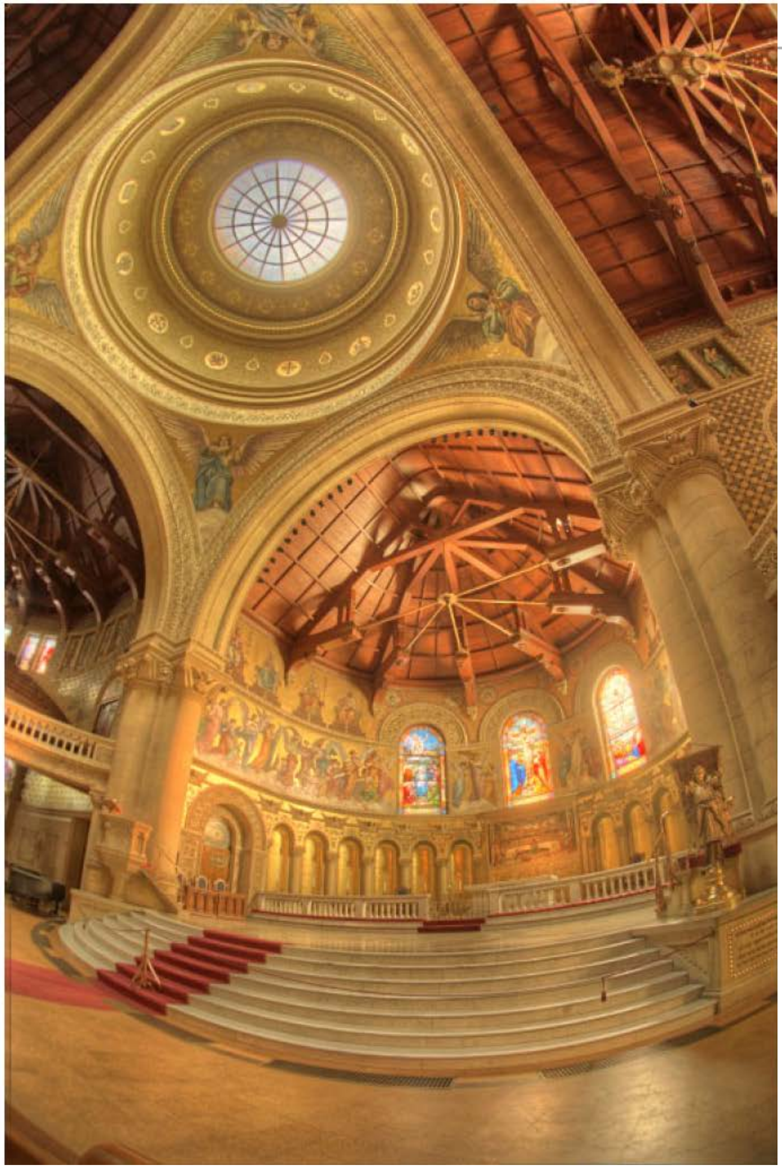
- Using multigrid methods
  - Complexity $O(M*N)$



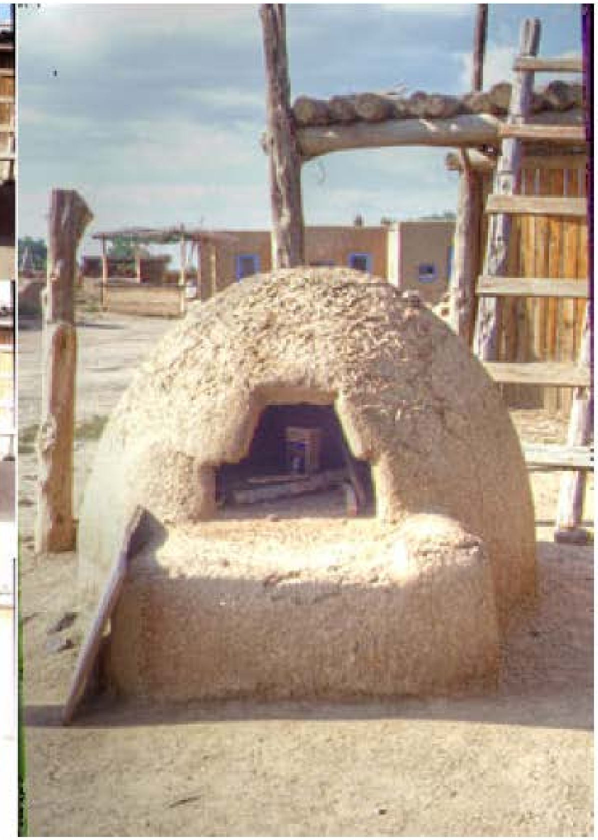
[Ward, 97]



gradient attenuation





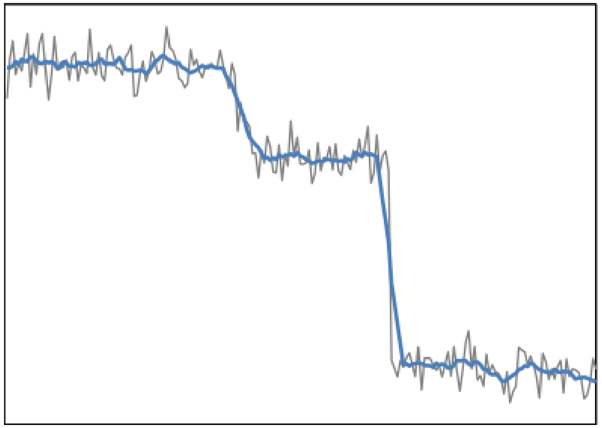



LDR Image Enhancement

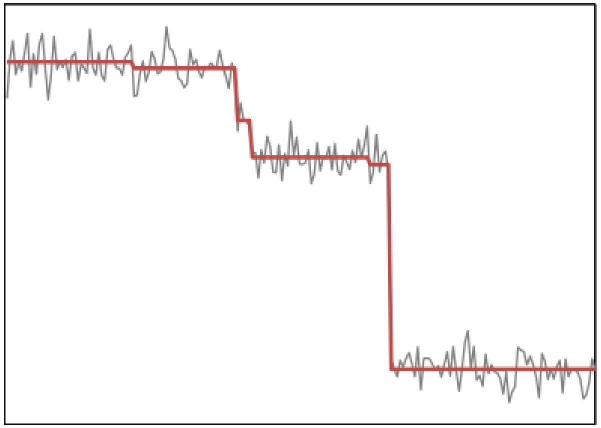
- *Image Smoothing via L0 Gradient Minimization*
  - Xu et al.
  - SIGGRAPH Asia 2011

Sparse Linear System

- Using multigrid methods



Bilateral filtering



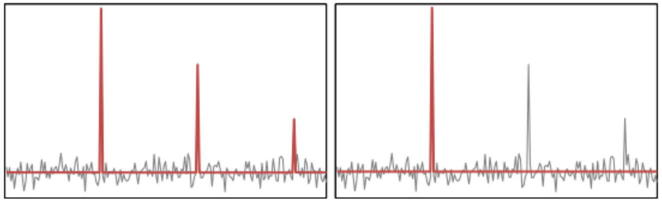
L0 smoothing

1D Smoothing

L0 norm of derivatives $c(f) = \#\{p \mid |f_p - f_{p+1}| \neq 0\}$

forward difference

input g smoothed f



Optimization problem

$$\min_f \sum_p (f_p - g_p)^2 \ \text{ s.t. } \ c(f) = k$$

$$\Rightarrow \quad \min_f \sum_p (f_p - g_p)^2 + \lambda \cdot c(f)$$

2D Formulation

gradient $\nabla S_p = (\partial_x S_p, \partial_y S_p)^T$

gradient measure $C(S) = \#\left\{ p \;\middle|\; |\partial_x S_p| + |\partial_y S_p| \neq 0 \right\}$

$|\partial S_p|$ sum of gradient magnitude in RGB

Optimization problem

$$\min_S \left\{ \sum_p (S_p - I_p)^2 + \lambda \cdot C(S) \right\}$$

Solver

auxiliary variables h_p v_p

$$\min_{S,h,v}\left\{\sum_p (S_p - I_p)^2 + \lambda C(h,v) + \beta((\partial_x S_p - h_p)^2 + (\partial_y S_p - v_p)^2)\right\}$$

$$C(h,v) = \#\{p \mid |h_p| + |v_p| \neq 0\}$$

two subproblems

alternately solve for S and (h,v)

Subproblem 1: computing S

$$\underset{S}{\text{minimize}} \quad \left\{ \sum_p (S_p - I_p)^2 + \beta((\partial_x S_p - h_p)^2 + (\partial_y S_p - v_p)^2) \right\}$$

$$S = \mathcal{F}^{-1}\left(\frac{\mathcal{F}(I) + \beta(\mathcal{F}(\partial_x)^*\mathcal{F}(h) + \mathcal{F}(\partial_y)^*\mathcal{F}(v))}{\mathcal{F}(1) + \beta(\mathcal{F}(\partial_x)^*\mathcal{F}(\partial_x) + \mathcal{F}(\partial_y)^*\mathcal{F}(\partial_y)} \right)$$

$$S - I + \beta(\partial_x^T \partial_x S - \partial_x^T h + \partial_y^T \partial_y S - \partial_y^T v) = 0$$

$$\mathcal{F}(S) - \mathcal{F}(I) + \beta\left(\mathcal{F}(\partial_x^T \partial_x S) - \mathcal{F}(\partial_x^T h) + \mathcal{F}(\partial_y^T \partial_y S) - \mathcal{F}(\partial_y^T v)\right) = 0$$

Subproblem 2: computing (h, v)

$$\min_{h,v}\left\{\sum_p(\partial_x S_p - h_p)^2 + (\partial_y S_p - v_p)^2) + \frac{\lambda}{\beta}C(h,v)\right\}$$

$$\Rightarrow \sum_p \min_{h_p,v_p}\left\{(h_p - \partial_x S_p)^2 + (v_p - \partial_y S_p)^2 + \frac{\lambda}{\beta}H(|h_p| + |v_p|)\right\}$$

binary function

$$\Downarrow$$

$$E_p = \left\{(h_p - \partial_x S_p)^2 + (v_p - \partial_y S_p)^2 + \frac{\lambda}{\beta}H(|h_p| + |v_p|)\right\}$$

minimized by $\Downarrow$

$$(h_p, v_p) = \begin{cases} (0,0) & (\partial_x S_p)^2 + (\partial_y S_p)^2 \le \lambda/\beta \\ (\partial_x S_p, \partial_y S_p) & \text{otherwise} \end{cases}$$

Excerpt from the paper

Proof.

1) When $\lambda/\beta \geq (\partial_x S_p)^2 + (\partial_y S_p)^2$, non-zero (h_p, v_p) yields

$$\begin{aligned} E_p((h_p, v_p) \neq (0,0)) &= (h_p - \partial_x S_p)^2 + (v_p - \partial_y S_p)^2 + \lambda/\beta, \\ &\geq \lambda/\beta, \\ &\geq (\partial_x S_p)^2 + (\partial_y S_p)^2. \end{aligned} \tag{13}$$

Note that $(h_p, v_p) = (0,0)$ leads to

$$E_p((h_p, v_p) = (0,0)) = (\partial_x S_p)^2 + (\partial_y S_p)^2. \tag{14}$$

Comparing Eqs. (13) and (14), the minimum energy $E_p^* = (\partial_x S_p)^2 + (\partial_y S_p)^2$ is produced when $(h_p, v_p) = (0,0)$.

2) When $(\partial_x S_p)^2 + (\partial_y S_p)^2 > \lambda/\beta$ and $(h_p, v_p) = (0,0)$, Eq. (14) still holds. But $E_p((h_p, v_p) \neq (0,0))$ has its minimum value λ/β when $(h_p, v_p) = (\partial_x S_p, \partial_y S_p)$. Comparing these two values, the minimum energy $E_p^* = \lambda/\beta$ is produced when $(h_p, v_p) = (\partial_x S_p, \partial_y S_p)$. ■

Algorithm

Algorithm 1 L_0 Gradient Minimization

Input: image I, smoothing weight λ, parameters β_0, $\beta_{\max}$, and rate κ

Initialization: $S \leftarrow I$, $\beta \leftarrow \beta_0$, $i \leftarrow 0$

repeat

  With $S^{(i)}$, solve for $h_p^{(i)}$ and $v_p^{(i)}$ in Eq. (12).

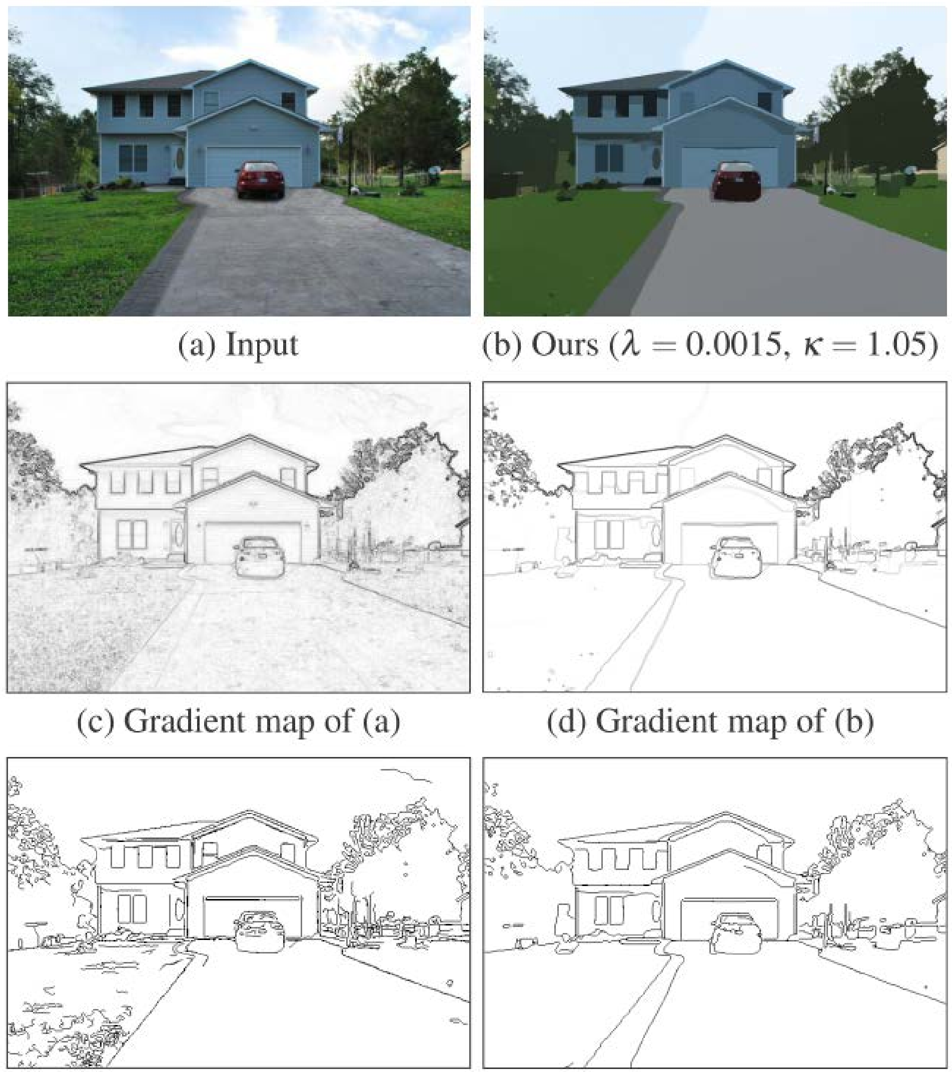
  With $h^{(i)}$ and $v^{(i)}$, solver for $S^{(i+1)}$ with Eq. (8).

  $\beta \leftarrow \kappa\beta$, $i++$.

until $\beta \geq \beta_{\max}$

Output: result image S

Applications:
Edge Enhancement



(a) Input

(b) Ours ($\lambda = 0.0015$, $\kappa = 1.05$)

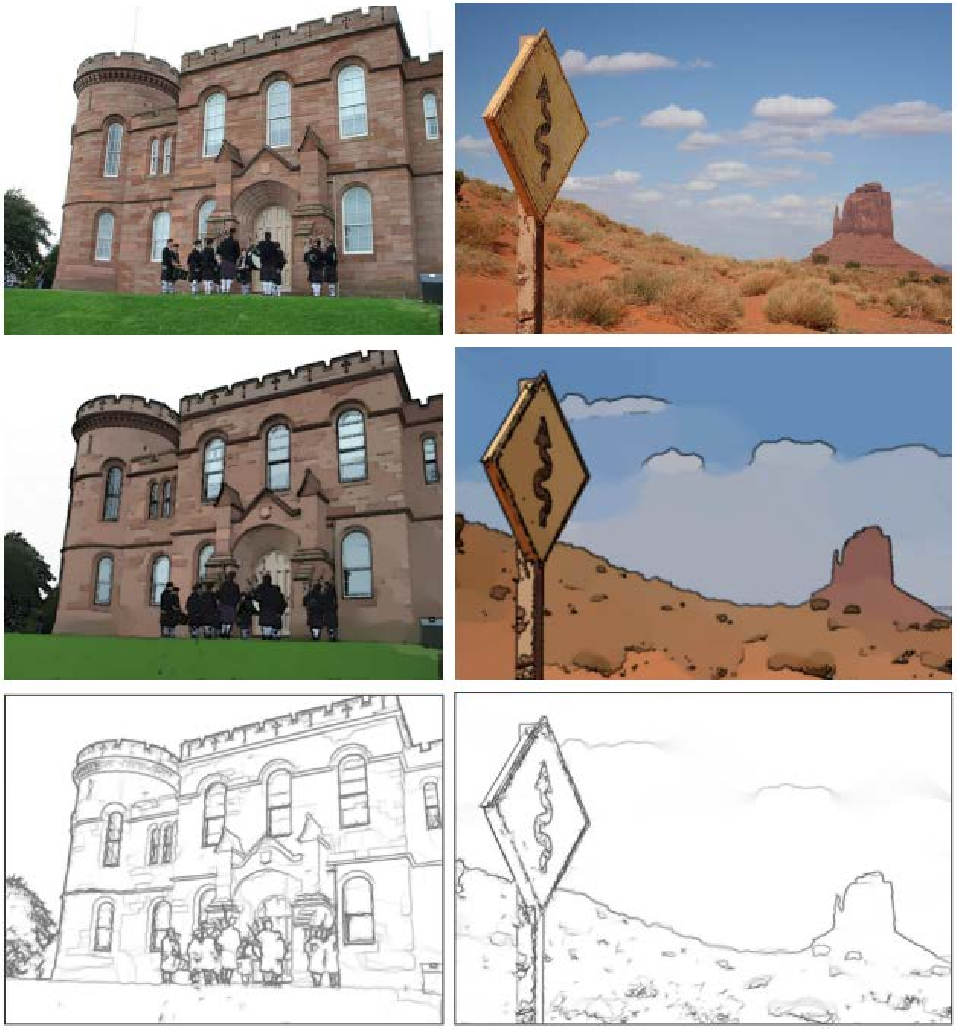
(c) Gradient map of (a)

(d) Gradient map of (b)

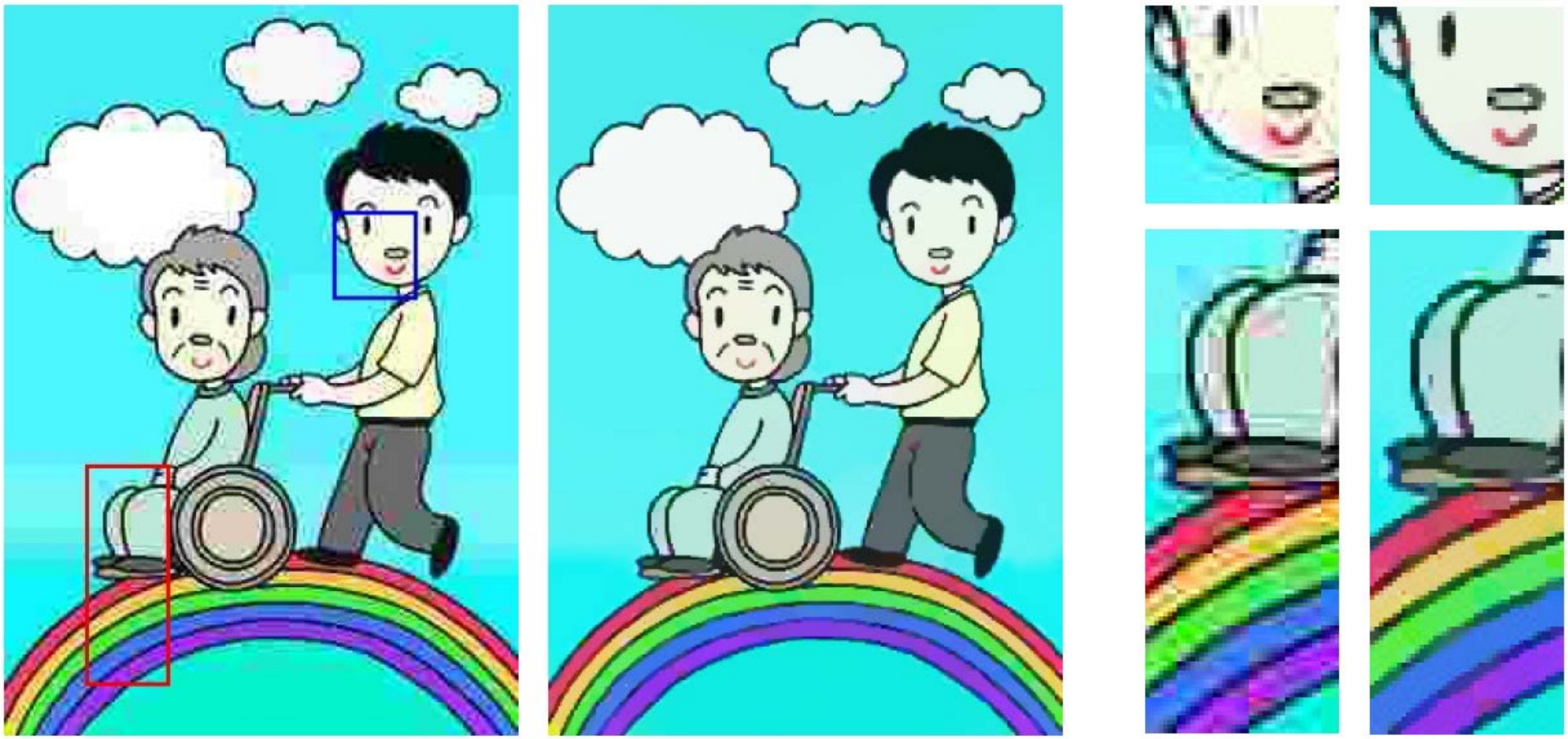
(e) Edge map of (a)

(f) Edge map of (b)

Applications:
Image Abstraction and Pencil Sketching



Applications:
Clip-Art Compression Artifact Removal



Applications: Layer-Based Contrast Manipulation

re-blur $$\min_{\sigma}\left\{\sum_{p}\left((G(\sigma_p)*S)-I_p\right)^2+\gamma\left((\partial_x\sigma_p)^2+(\partial_y\sigma_p)^2\right)\right\}$$

to find suitable Gaussian scale for each pixel

assigning discrete values to σ_p → a labeling problem

enhance gradients in the detail layer

Applications:
Tone Mapping

as an alternative to bilateral filter