Gradient domain operations

Digital Visual Effects, Spring 2009

Yung-Yu Chuang

2009/6/4

with slides by Fredo Durand, Ramesh Raskar, Amit Agrawal

Gradient domain operators

sources/destinations cloning seamless cloning

Gradient Domain Manipulations

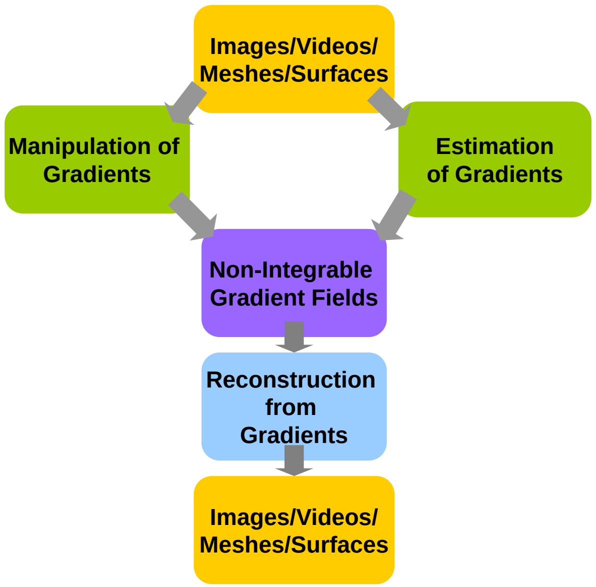
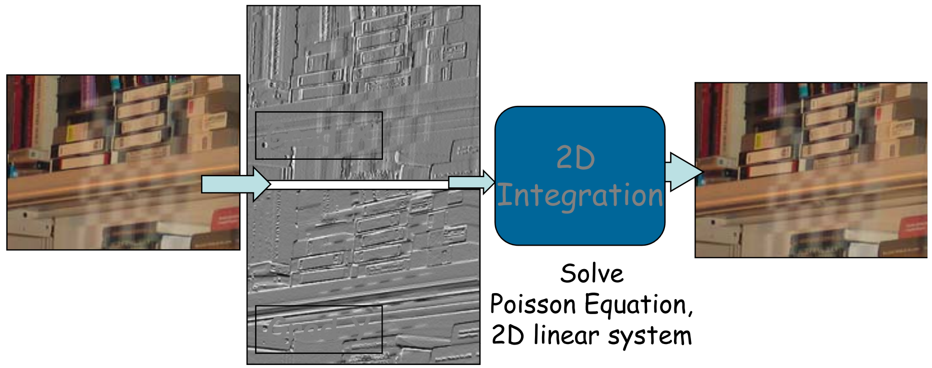
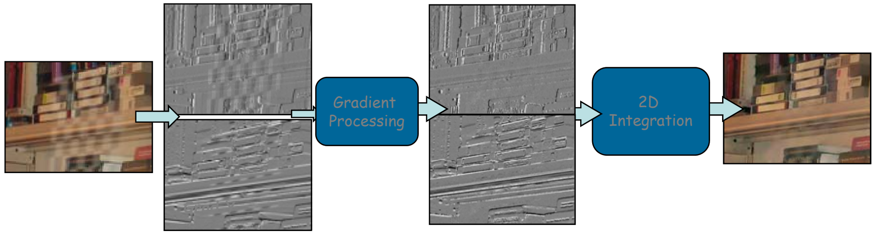


Image Intensity Gradients in 2D



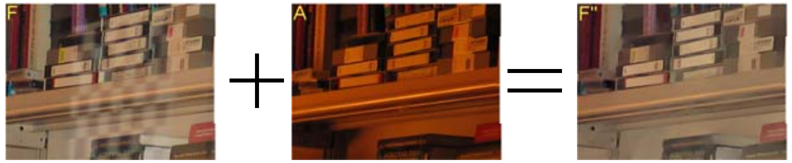
Intensity Gradient Manipulation

A Common Pipeline



1. Gradient manipulation
2. Reconstruction from gradients

Example Applications



Removing Glass Reflections

Seamless Image Stitching

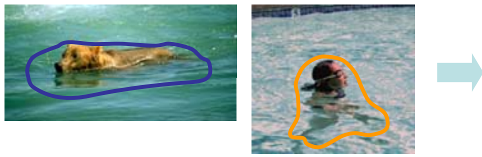
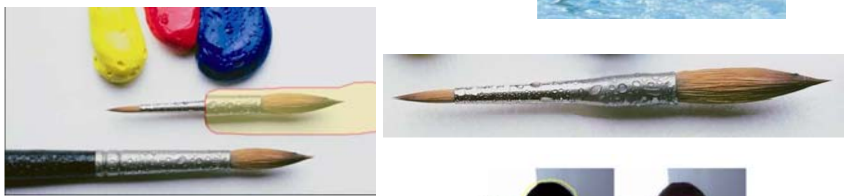
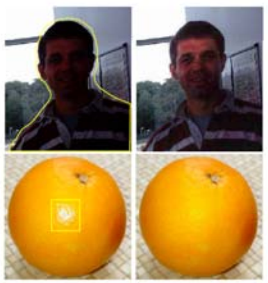
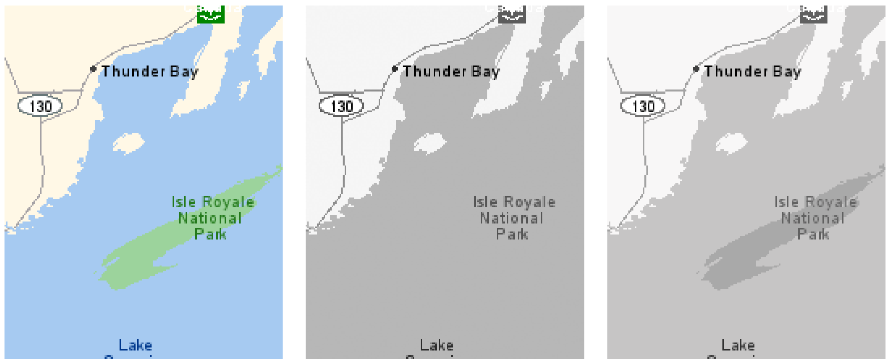


Image Editing





Changing Local Illumination



Original | PhotoshopGrey | Color2Gray

Color to Gray Conversion

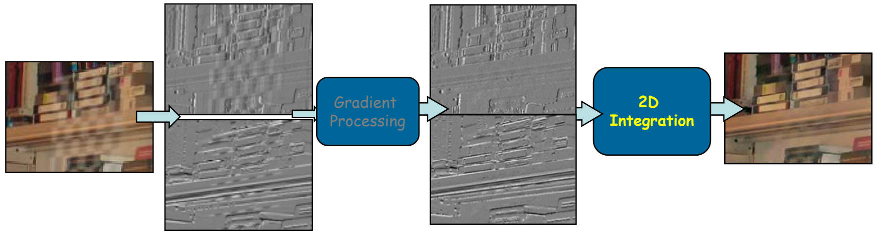
High Dynamic Range Compression

Edge Suppression under Significant Illumination Variations

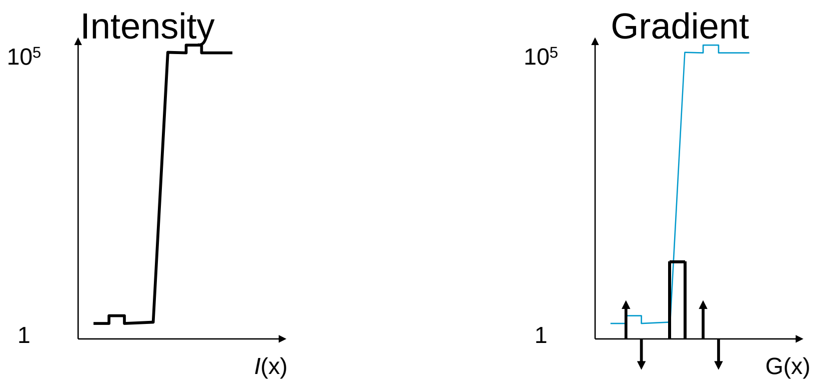

Fusion of day and night images

Intensity Gradient Manipulation

A Common Pipeline



Intensity Gradient in 1D

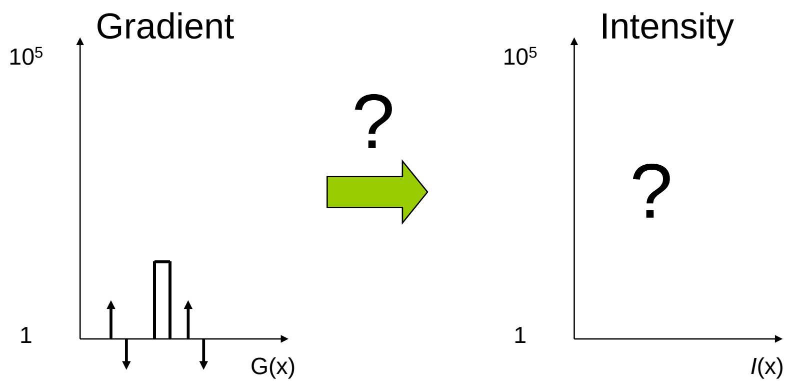


Gradient at x,

$G(x) = I(x+1) - I(x)$

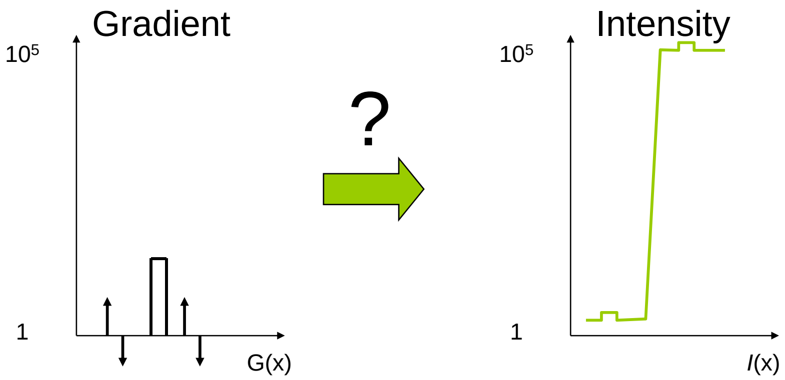
Forward Difference

Reconstruction from Gradients



For n intensity values, about n gradients

Reconstruction from Gradients

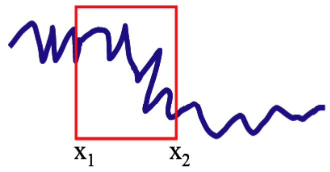


1D Integration

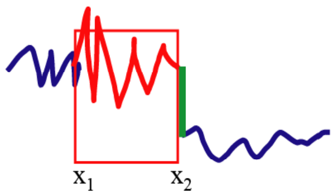
$$I(x) = I(x-1) + G(x)$$

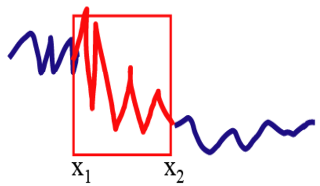
Cumulative sum

1D case with constraints

Seamlessly paste  onto 

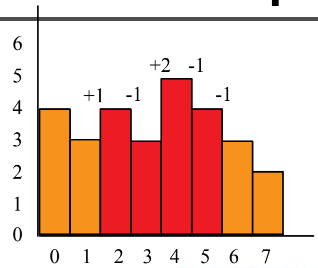
Just add a linear function so that the boundary condition is respected



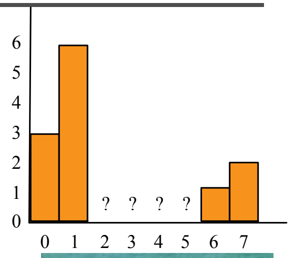


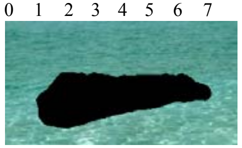
Discrete 1D example: minimization

- Copy



to



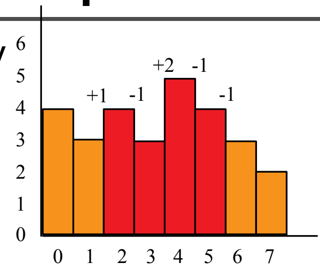


- Min $((f_2-f_1)-1)^2$
- Min $((f_3-f_2)-(-1))^2$
- Min $((f_4-f_3)-2)^2$
- Min $((f_5-f_4)-(-1))^2$
- Min $((f_6-f_5)-(-1))^2$

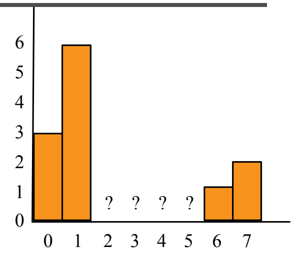
With
$f_1=6$
$f_6=1$

1D example: minimization

- Copy



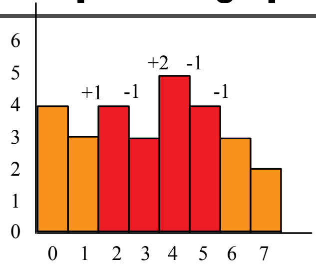
to



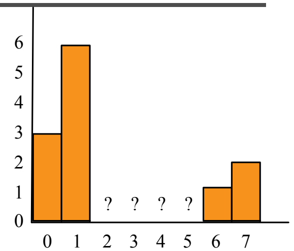
- Min $((f_2-6)-1)^2$ ==> $f_2^2+49-14f_2$
- Min $((f_3-f_2)-(-1))^2$ ==> $f_3^2+f_2^2+1-2f_3f_2+2f_3-2f_2$
- Min $((f_4-f_3)-2)^2$ ==> $f_4^2+f_3^2+4-2f_3f_4-4f_4+4f_3$
- Min $((f_5-f_4)-(-1))^2$ ==> $f_5^2+f_4^2+1-2f_5f_4+2f_5-2f_4$
- Min $((1-f_5)-(-1))^2$ ==> $f_5^2+4-4f_5$

1D example: big quadratic

- Copy



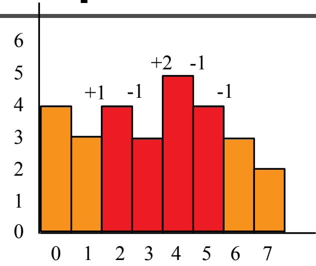
to



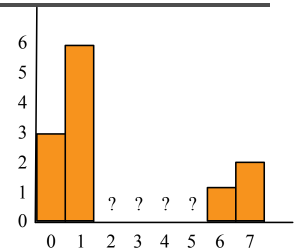
- Min (f_2^2+49-14f_2
  + f_3^2+f_2^2+1-2f_3f_2 +2f_3-2f_2
  + f_4^2+f_3^2+4-2f_3f_4 -4f_4+4f_3
  + f_5^2+f_4^2+1-2f_5f_4 +2f_5-2f_4
  + f_5^2+4-4f_5)
  Denote it Q

1D example: derivatives

- Copy



to

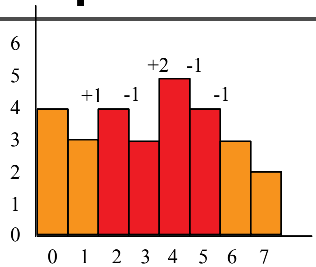


Min (f_2^2+49-14f_2
+ f_3^2+f_2^2+1-2f_3f_2 +2f_3-2f_2
+ f_4^2+f_3^2+4-2f_3f_4 -4f_4+4f_3
+ f_5^2+f_4^2+1-2f_5f_4 +2f_5-2f_4
+ f_5^2+4-4f_5)
Denote it Q

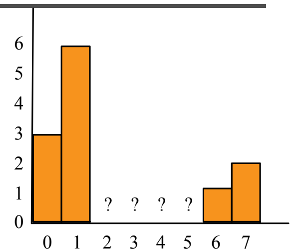
$$\frac{dQ}{df_2} = 2f_2 + 2f_2 - 2f_3 - 16$$
$$\frac{dQ}{df_3} = 2f_3 - 2f_2 + 2 + 2f_3 - 2f_4 + 4$$
$$\frac{dQ}{df_4} = 2f_4 - 2f_3 - 4 + 2f_4 - 2f_5 - 2$$
$$\frac{dQ}{df_5} = 2f_5 - 2f_4 + 2 + 2f_5 - 4$$

1D example: set derivatives to zero

- Copy



to

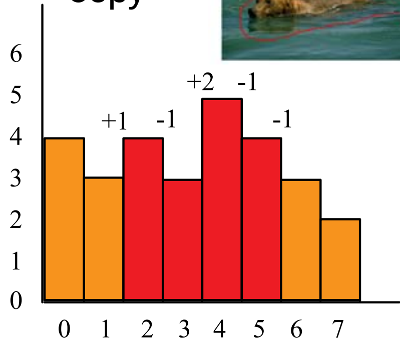


$$\frac{dQ}{df_2} = 2f_2 + 2f_2 - 2f_3 - 16$$
$$\frac{dQ}{df_3} = 2f_3 - 2f_2 + 2 + 2f_3 - 2f_4 + 4$$
$$\frac{dQ}{df_4} = 2f_4 - 2f_3 - 4 + 2f_4 - 2f_5 - 2$$
$$\frac{dQ}{df_5} = 2f_5 - 2f_4 + 2 + 2f_5 - 4$$

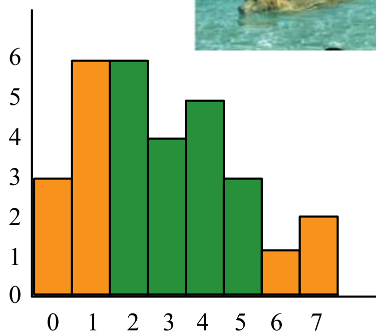
$$\Longrightarrow \begin{pmatrix} 4 & -2 & 0 & 0 \\ -2 & 4 & -2 & 0 \\ 0 & -2 & 4 & -2 \\ 0 & 0 & -2 & 4 \end{pmatrix} \begin{pmatrix} f_2 \\ f_3 \\ f_4 \\ f_5 \end{pmatrix} = \begin{pmatrix} 16 \\ -6 \\ 6 \\ 2 \end{pmatrix}$$

1D example

- Copy



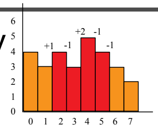
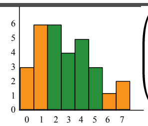
to



$$\begin{pmatrix} 4 & -2 & 0 & 0 \\ -2 & 4 & -2 & 0 \\ 0 & -2 & 4 & -2 \\ 0 & 0 & -2 & 4 \end{pmatrix} \begin{pmatrix} f_2 \\ f_3 \\ f_4 \\ f_5 \end{pmatrix} = \begin{pmatrix} 16 \\ -6 \\ 6 \\ 2 \end{pmatrix}$$

$$\boxed{\begin{pmatrix} f_2 \\ f_3 \\ f_4 \\ f_5 \end{pmatrix} = \begin{pmatrix} 6 \\ 4 \\ 5 \\ 3 \end{pmatrix}}$$

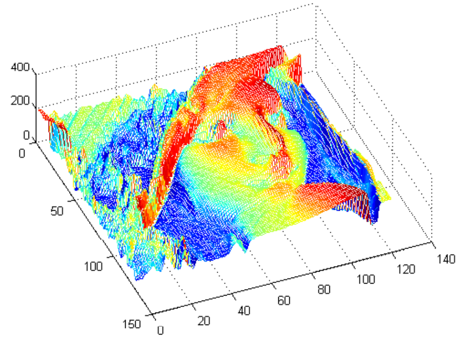
1D example: remarks

- Copy  to  $\begin{pmatrix} 4 & -2 & 0 & 0 \\ -2 & 4 & -2 & 0 \\ 0 & -2 & 4 & -2 \\ 0 & 0 & -2 & 4 \end{pmatrix} \begin{pmatrix} f_2 \\ f_3 \\ f_4 \\ f_5 \end{pmatrix} = \begin{pmatrix} 16 \\ -6 \\ 6 \\ 2 \end{pmatrix}$

- Matrix is sparse
- Matrix is symmetric
- Everything is a multiple of 2
  - because square and derivative of square
- Matrix is a convolution (kernel -2 4 -2)
- Matrix is independent of gradient field. Only RHS is
- Matrix is a second derivative

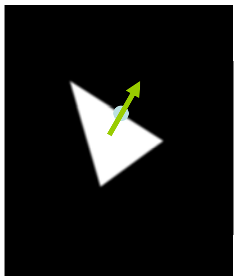
2D example: images

- Images as scalar fields
  - R^2 -> R



Gradients

- Vector field (gradient field)
  - Derivative of a scalar field
- Direction
  - Maximum rate of change of scalar field
- Magnitude
  - Rate of change



Gradient Field

- Components of gradient
  - Partial derivatives of scalar field

$$I(x,y) \qquad \nabla I = \{\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\}$$

$$I(x,y,t) \qquad \nabla I = \{\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}, \frac{\partial I}{\partial t}\}$$

Example

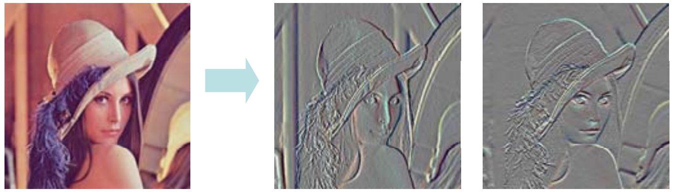


Image
I(x,y)

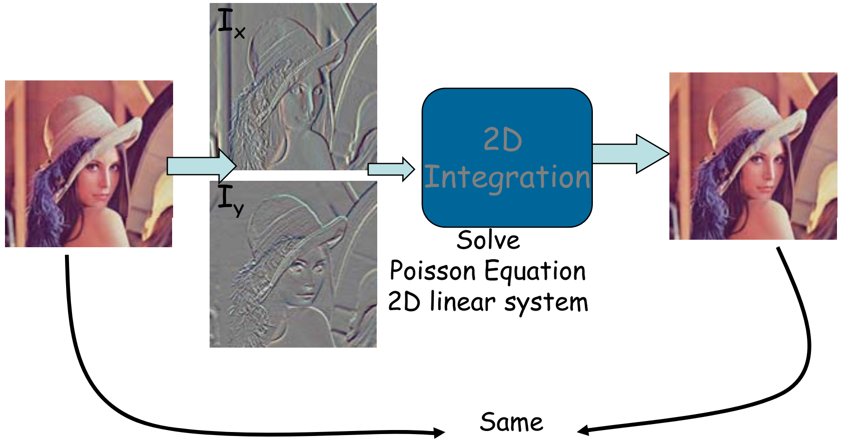
I_x I_y

Gradient at x,y as Forward Differences

$$G_x(x,y) = I(x+1\ ,\ y) - I(x,y)$$
$$G_y(x,y) = I(x\ ,\ y+1) - I(x,y)$$
$$G(x,y) = (G_x\ ,\ G_y)$$

Reconstruction from Gradients

Sanity Check:
Recovering Original Image

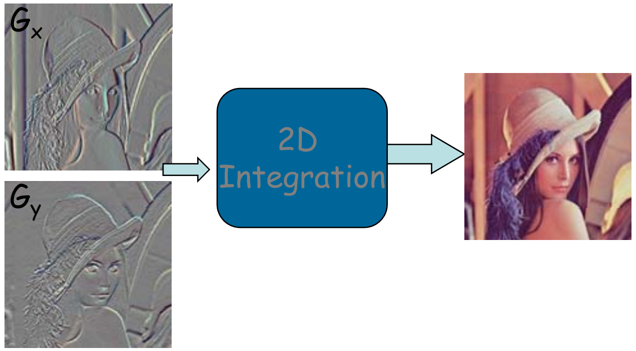


Reconstruction from Gradients

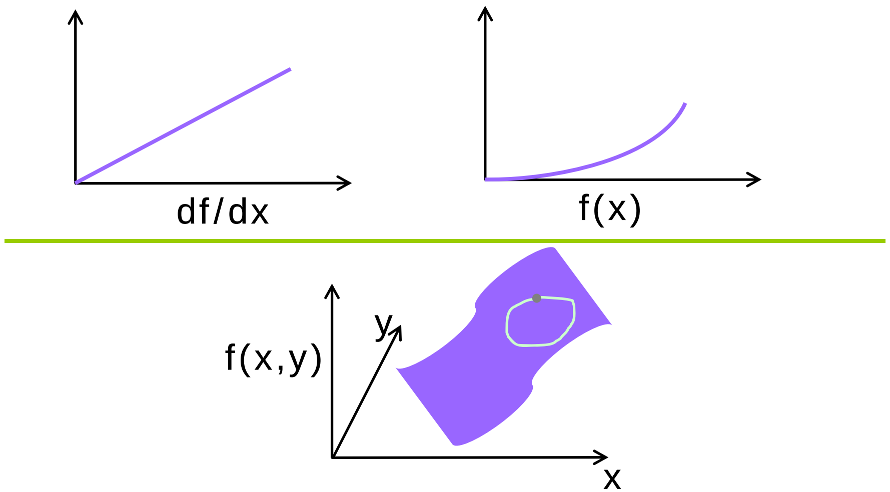
Given $G(x,y) = (G_x\ ,\ G_y)$

How to compute $I(x,y)$ for the image ?

For n^2 image pixels, $2\,n^2$ gradients !



2D Integration is non-trivial



Reconstruction depends on chosen path

Reconstruction from Gradient Field G

- Look for image $\boldsymbol{I}$ with gradient closest to $\boldsymbol{G}$ in the least squares sense.
- $\boldsymbol{I}$ minimizes the integral: $\iint F(\nabla I, G)\,dxdy$

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

$$\Rightarrow \quad \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} = \frac{\partial G_x}{\partial x} + \frac{\partial G_y}{\partial y}$$

Solve $\dfrac{\partial^2 I}{\partial x^2} + \dfrac{\partial^2 I}{\partial y^2} = \dfrac{\partial G_x}{\partial x} + \dfrac{\partial G_y}{\partial y}$

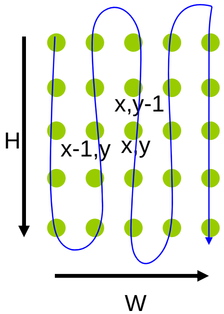
$$G_x(x,y) - G_x(x-1,y) + G_y(x,y) - G_y(x,y-1)$$

$$I(x+1,y) + I(x-1,y) + I(x,y+1) + I(x,y-1) - 4I(x,y)$$

$$\begin{bmatrix} \\ ..\ 1 \ldots \quad 1\ {-4}\ 1 \ldots \quad 1\ .. \\ \\ \end{bmatrix} \begin{bmatrix} \\ \mathbf{I} \\ \\ \end{bmatrix} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}$$

Linear System

$$-4I(x,y) + I(x,y+1) + I(x,y-1) + I(x+1,y) + I(x-1,y) = u(x,y)$$



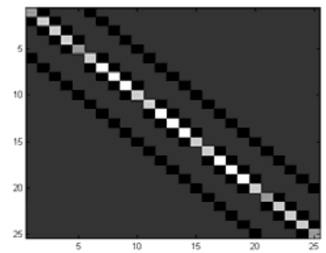
$$\underbrace{[.\ 1\ .\ .\ .\ 1}_{H}\ {-4}\ 1\ .\ .\ .\ 1\ .] \begin{bmatrix} . \\ . \\ I(x-1,y) \\ . \\ . \\ . \\ I(x,y-1) \\ I(x,y) \\ I(x,y+1) \\ . \\ . \\ . \\ I(x+1,y) \\ . \end{bmatrix} = \begin{bmatrix} . \\ . \\ . \\ . \\ . \\ . \\ . \\ u(x,y) \\ . \\ . \\ . \\ . \\ . \\ . \end{bmatrix}$$

A x b

Sparse Linear system

$$\begin{bmatrix}
 & & 1 & -4 & 1 & & & & 1 & & & & \\
 & & & 1 & -4 & 1 & & & & 1 & & & \\
1 & & & & 1 & -4 & 1 & & & & 1 & & \\
 & 1 & & & & 1 & -4 & 1 & & & & 1 & \\
 & & 1 & & & & 1 & -4 & 1 & & & & 1 \\
 & & & 1 & & & & 1 & -4 & 1 & & & \\
 & & & & 1 & & & & 1 & -4 & 1 & &
\end{bmatrix}$$

A matrix



Solving Linear System

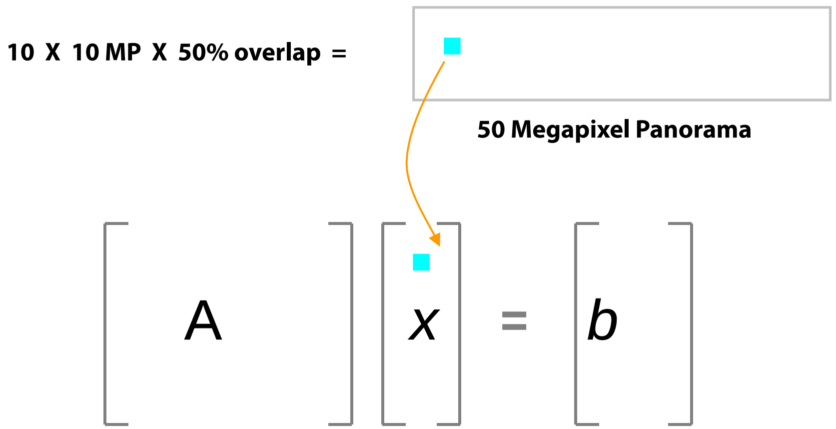
- Image size N*N
- Size of A ~ N^2 by N^2
- Impractical to form and store A

- Direct Solvers
- Basis Functions
- Multigrid
- Conjugate Gradients

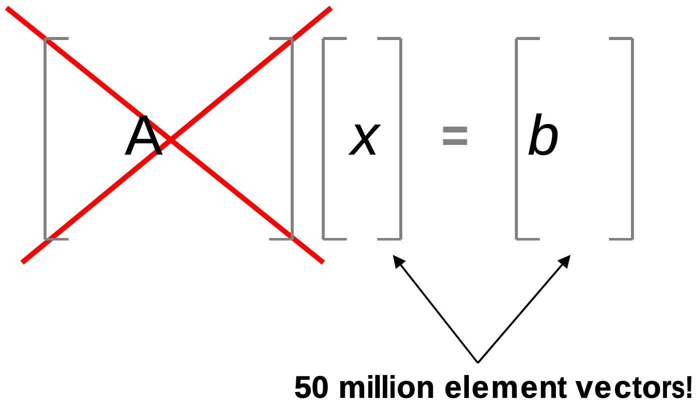
Approximate Solution for Large Scale Problems

- Resolution is increasing in digital cameras
- Stitching, Alignment requires solving large linear system

Scalability problem



Scalability problem



Approximate Solution

- Reduce size of linear system
- Handle high resolution images
- Part of Photoshop CS3

Aseem Agarwala. "Efficient gradient-domain compositing using quadtrees," ACM Transactions on Graphics (Proceedings of SIGGRAPH 2007)

The key insight

Desired solution x

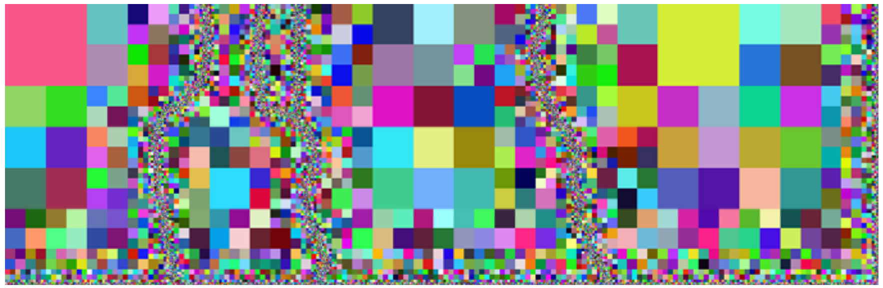
–

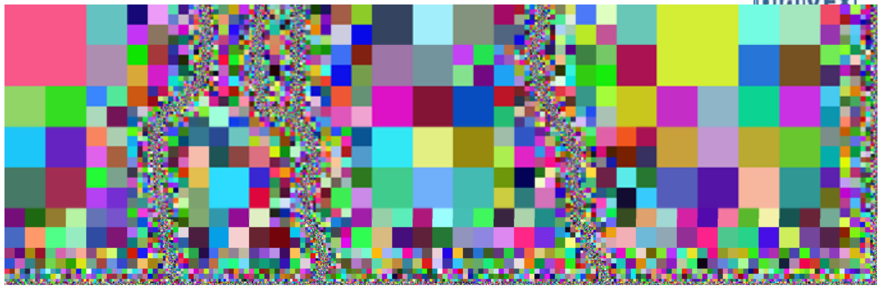
Initial Solution x_0

=

Difference x_δ

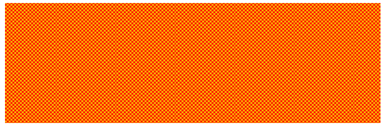
Quadtree decomposition

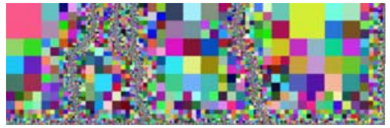




- Maximally subdivide quadtree along seams
- Variables placed at node corners
- Restricted quadtree
- Bi-linear interpolation reconstructs full solution
- Square nodes

Reduced space



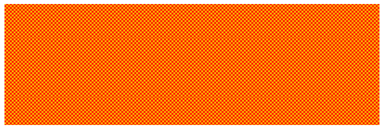


x
n variables

y
m variables

$$m << n$$

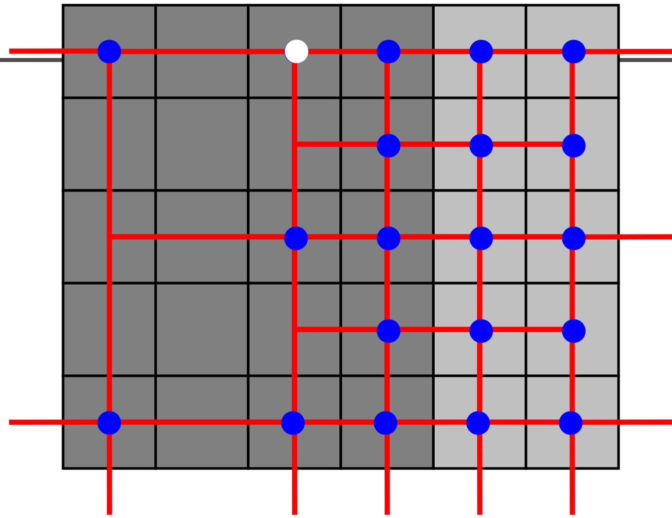
Reduced space



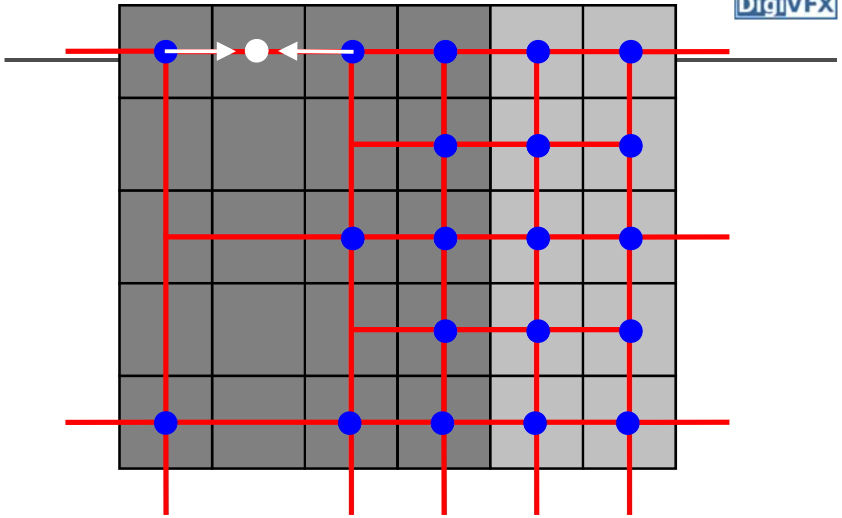
x
n variables

y
m variables

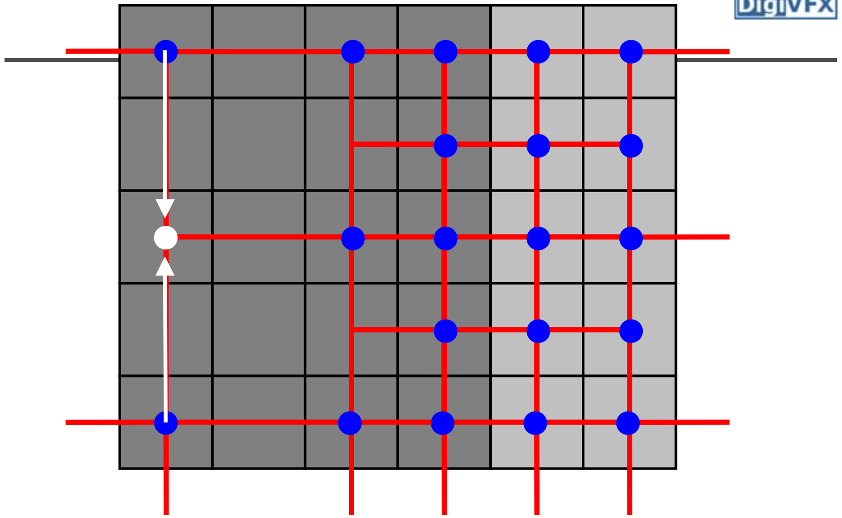
$$x = Sy$$



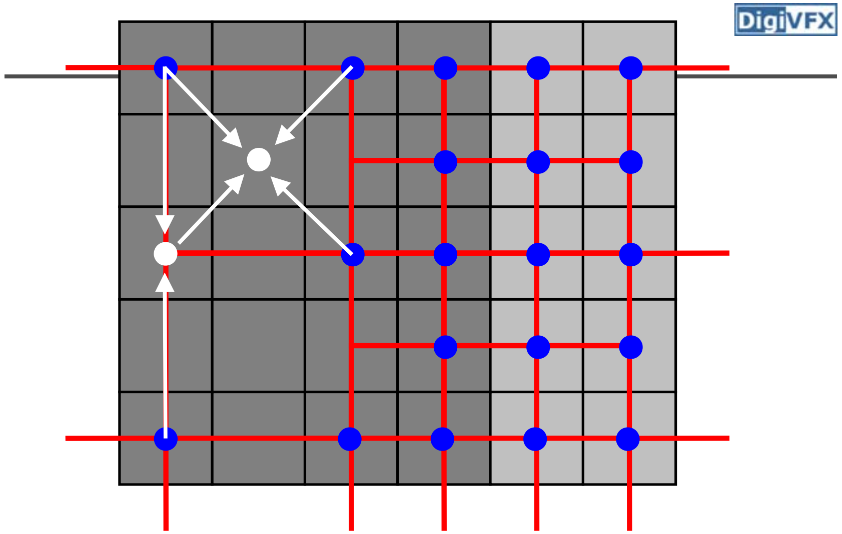
$$x = Sy$$



$$x = Sy$$

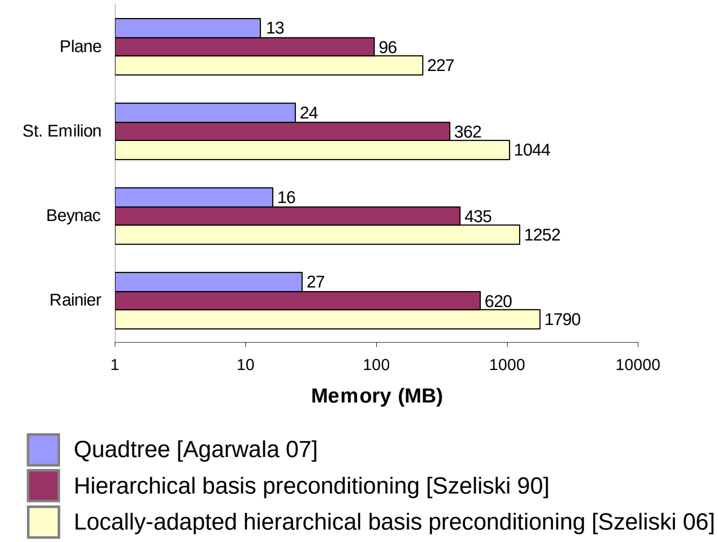


$$x = Sy$$



$$x = Sy$$

Performance

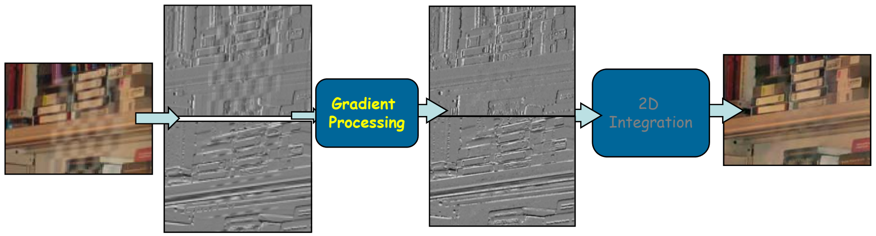


Cut-and-paste

Cut-and-paste

Intensity Gradient Manipulation

A Common Pipeline



Gradient Domain Manipulations: Overview

(A) Per pixel

(B) Corresponding gradients in <u>two images</u>

(C) Corresponding gradients in <u>multiple images</u>

(D) Combining gradients <u>along seams</u>

Gradient Domain Manipulations: Overview

(A) Per pixel
- Non-linear operations (HDR compression, local illumination change)
- Set to zero (shadow removal, intrinsic images, texture de-emphasis)
- Poisson Matting

(B) Corresponding gradients in two images
- Vector operations (gradient projection)
  - Combining flash/no-flash images, Reflection removal
- Projection Tensors
  - Reflection removal, Shadow removal
- Max operator
  - Day/Night fusion, Visible/IR fusion, Extending DoF
- Binary, choose from first or second, copying
  - Image editing, seamless cloning

Gradient Domain Manipulations

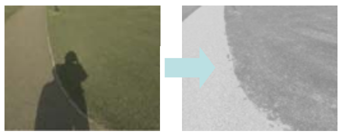
(C) Corresponding gradients in multiple images
- Median operator
  - Specularity reduction
  - Intrinsic images
- Max operation
  - Extended DOF

(D) Combining gradients along seams
- Weighted averaging
- Optimal seam using graph cut
  - Image stitching, Mosaics, Panoramas, Image fusion
  - A usual pipeline: Graph cut to find seams + gradient domain fusion

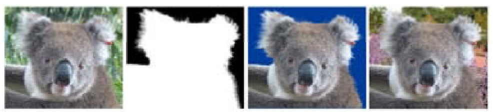
A. Per Pixel Manipulations

- Non-linear operations
  - HDR compression, local illumination change

- Set to zero
  - Shadow removal, intrinsic images, texture de-emphasis



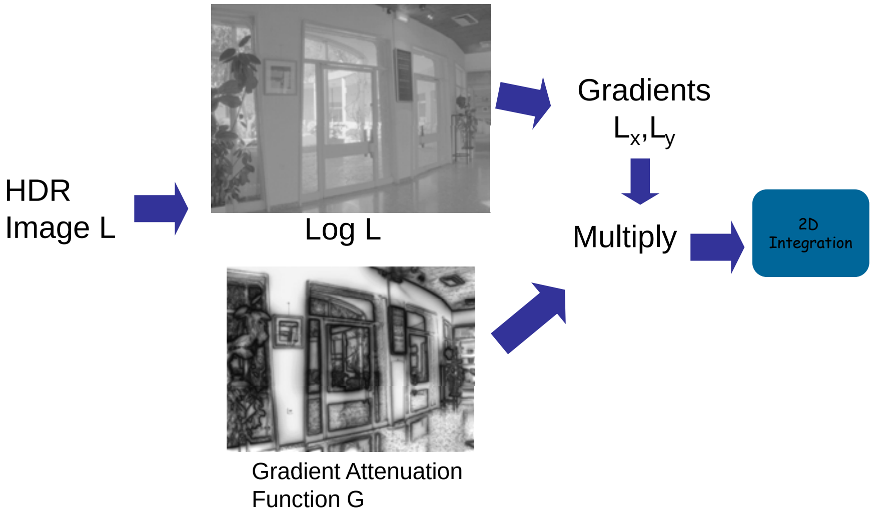
- Poisson Matting



High Dynamic Range Imaging

Images from Raanan Fattal

Gradient Domain Compression



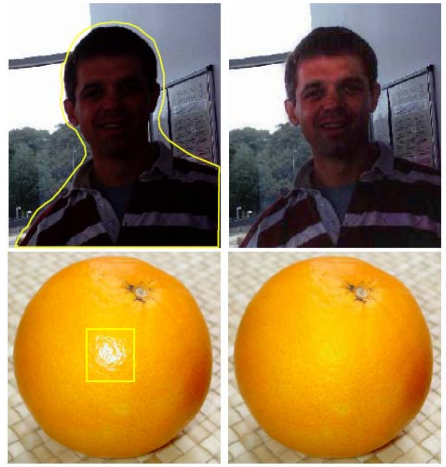
Local Illumination Change

Original Image: f

$$\mathbf{v} = \alpha^{\beta} |\nabla f^*|^{-\beta} \nabla f^*,$$

Original gradient field: ∇f^*

Modified gradient field: v



Perez et al. Poisson Image editing, SIGGRAPH 2003

Illumination Invariant Image

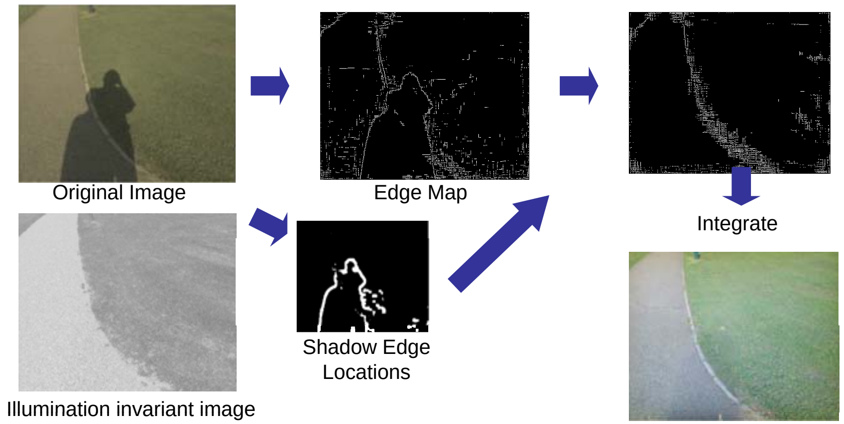
Original Image

Illumination invariant image

- Assumptions
  - Sensor response = delta functions R, G, B in wavelength spectrum
  - Illumination restricted to Outdoor Illumination

G. D. Finlayson, S.D. Hordley & M.S. Drew, Removing Shadows From Images, ECCV 2002

Shadow Removal Using Illumination Invariant Image

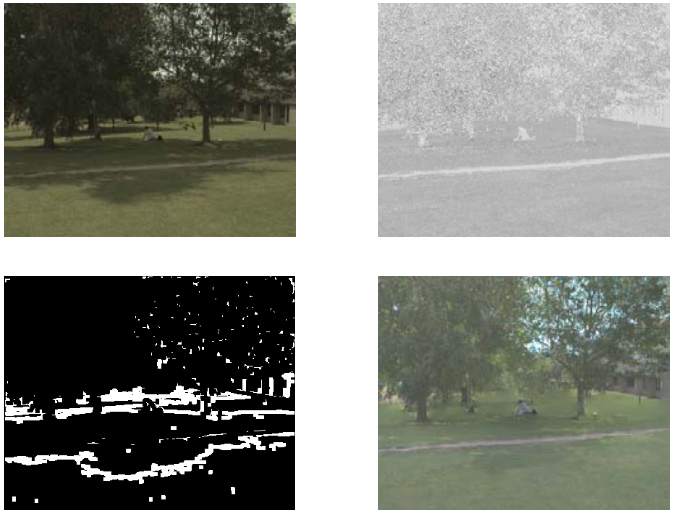


G. D. Finlayson, S.D. Hordley & M.S. Drew, Removing Shadows From Images, ECCV 2002

Illumination invariant image

Original Image

Invariant Image



Detected Shadow Edges

Shadow Removed

G. D. Finlayson, S.D. Hordley & M.S. Drew, Removing Shadows From Images, ECCV 2002

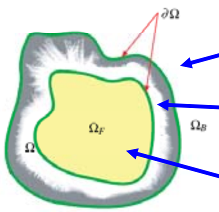
Intrinsic Image

- Photo = Illumination Image * Intrinsic Image
- Retinex [Land & McCann 1971, Horn 1974]
  - Illumination is smoothly varying
  - Reflectance, piece-wise constant, has strong edges
  - Keep strong image gradients, integrate to obtain reflectance

low-frequency attenuate more

high-frequency attenuate less

Poisson Matting



Background B

Alpha

Foreground F

Trimap: User specified

Jian Sun, Jiaya Jia, Chi-Keung Tang, Heung-Yeung Shum, Poisson Matting, SIGGRAPH 2004

Poisson Matting

$$I = \alpha F + (1-\alpha)B$$

$$\nabla I = (F-B)\nabla\alpha + \alpha\nabla F + (1-\alpha)\nabla B$$

Approximate: Assume F and B are smooth

$$\nabla I = (F-B)\nabla\alpha$$

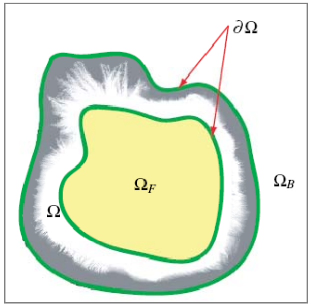
$$\nabla\alpha \approx \frac{1}{F-B}\nabla I \quad \Rightarrow \quad \Delta\alpha = div\left(\frac{\nabla I}{F-B}\right)$$

F and B in tri-map using nearest pixels

Poisson Equation

Poisson Matting

- Steps
  - Approximate F and B in trimap Ω.
  - Solve for α $\Delta\alpha = div(\frac{\nabla I}{F - B})$
  - Refine F and B using α
  - Iterate



Gradient Domain Manipulations: Overview

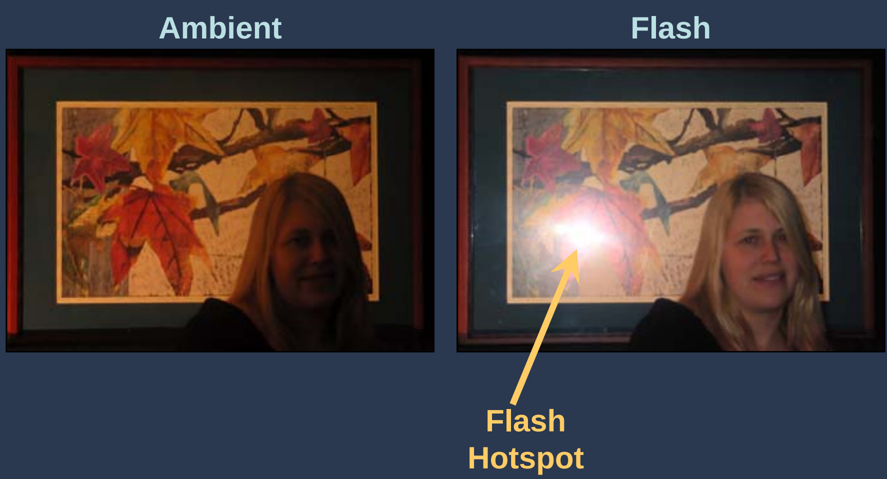
(A) Per pixel

(B) Corresponding gradients in <u>two images</u>

(C) Corresponding gradients in <u>multiple images</u>

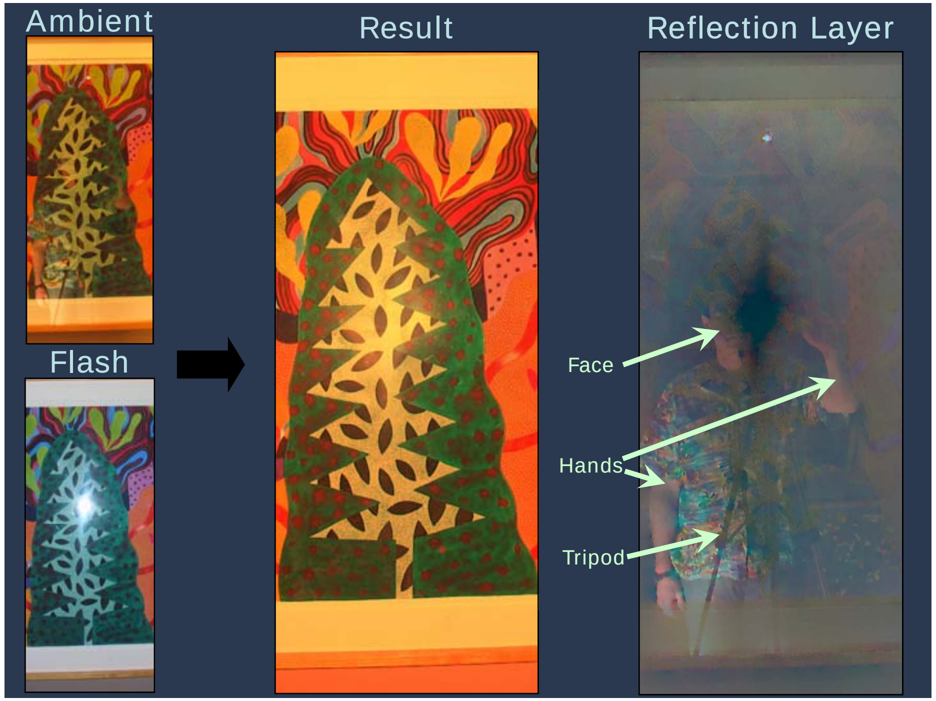
(D) Combining gradients <u>along seams</u>

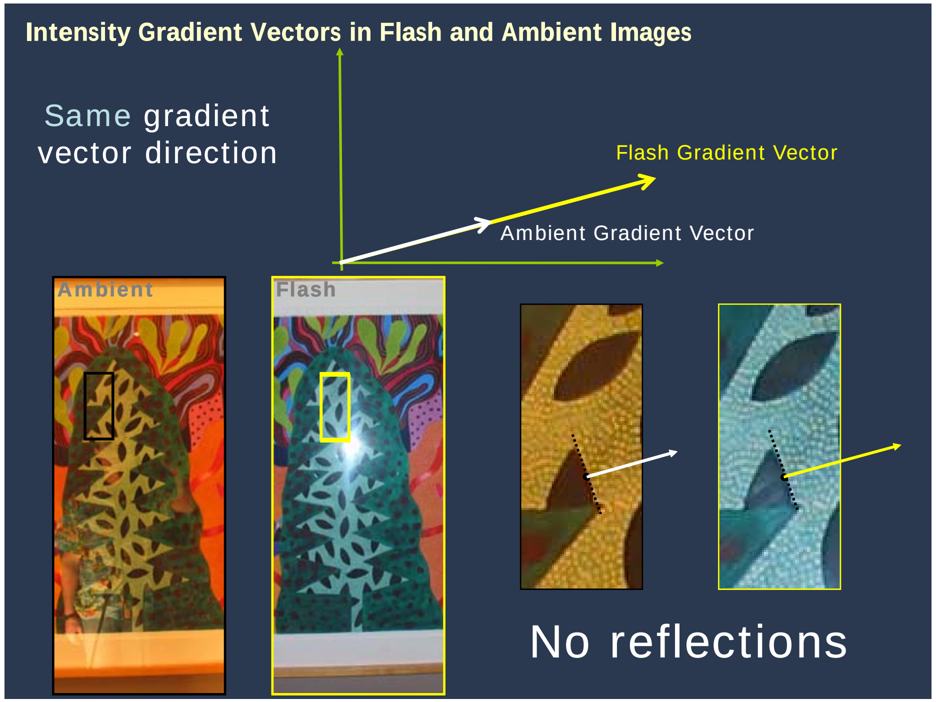
Photography Artifacts: Flash Hotspot

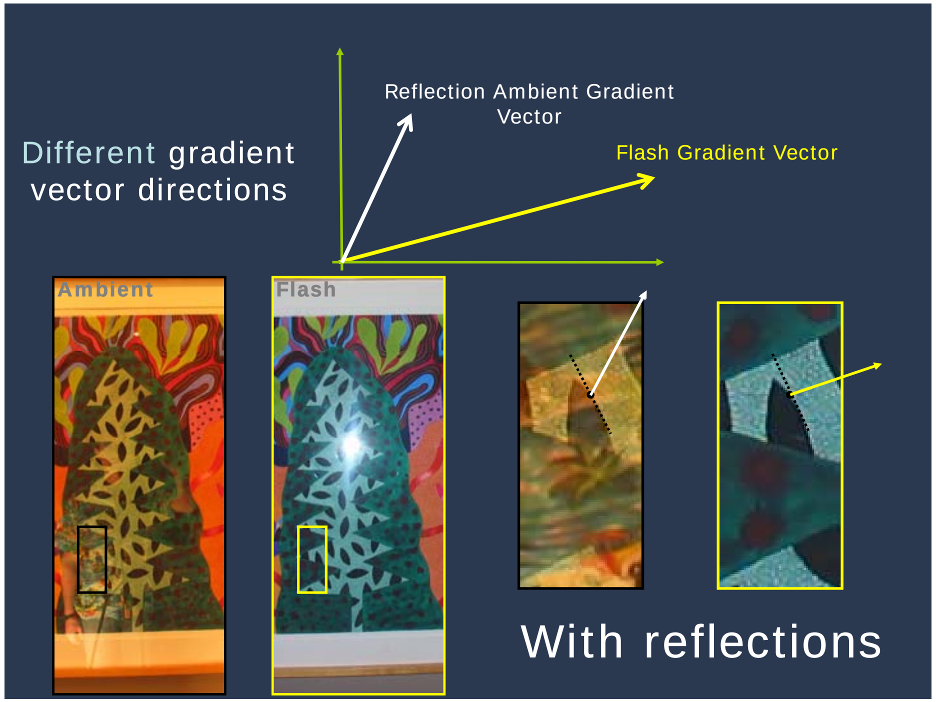


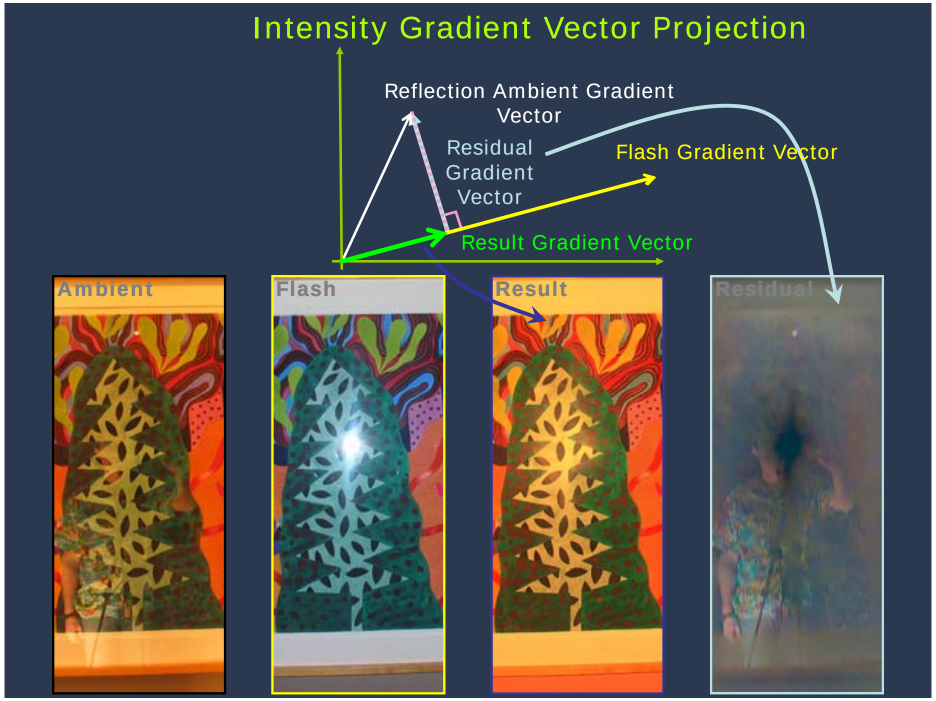
Reflections due to Flash

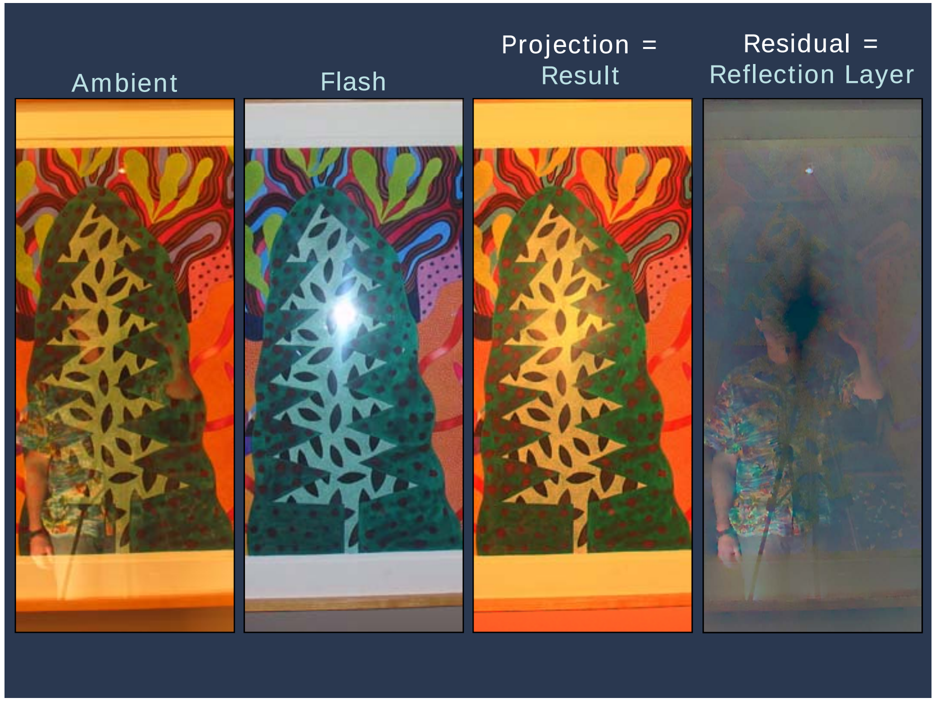

Self-Reflections and Flash Hotspot
Ambient
Flash
Face
Hands
Tripod


Ambient
Flash
Result
Reflection Layer
Face
Hands
Tripod


Intensity Gradient Vectors in Flash and Ambient Images
Same gradient vector direction
Flash Gradient Vector
Ambient Gradient Vector
Ambient
Flash
No reflections


Different gradient vector directions
Reflection Ambient Gradient Vector
Flash Gradient Vector
Ambient
Flash
With reflections


Intensity Gradient Vector Projection
Reflection Ambient Gradient Vector
Residual Gradient Vector
Flash Gradient Vector
Result Gradient Vector
Ambient
Flash
Result
Residual


Ambient
Flash
Projection = Result
Residual = Reflection Layer


Flash
Ambient
Checkerboard outside glass window
Reflections on glass window

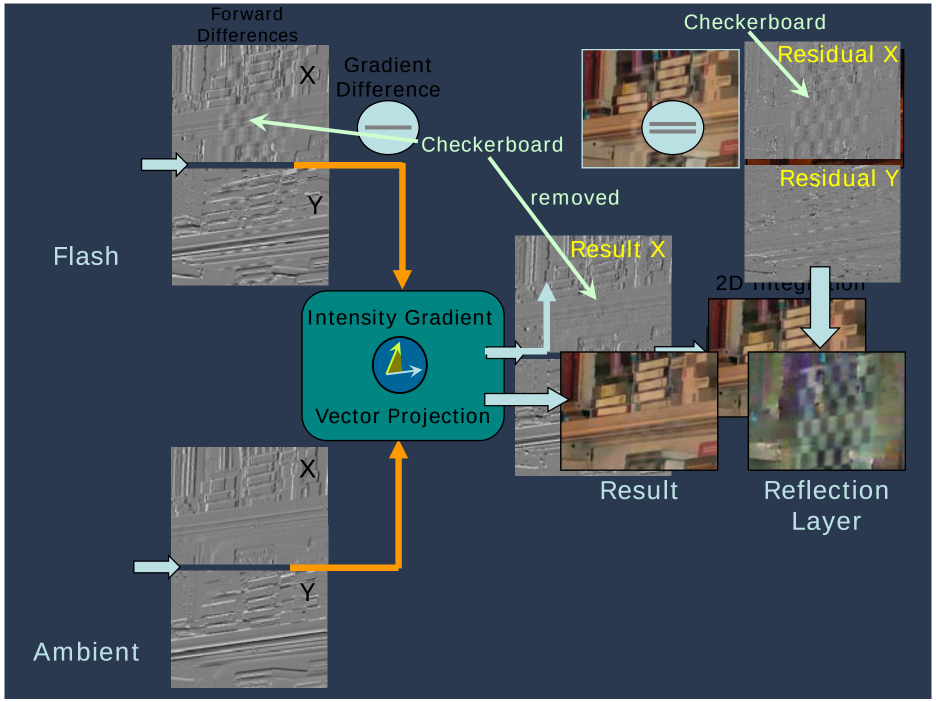
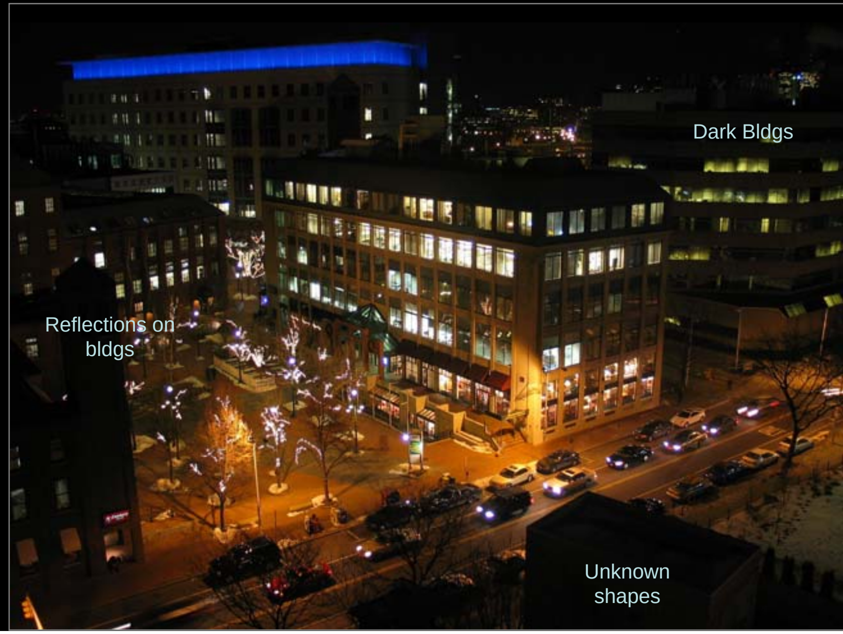

Forward Differences
X
Y
Gradient Difference
Checkerboard
removed
Checkerboard
Residual X
Residual Y
Result X
Flash
Intensity Gradient
Vector Projection
Ambient
Result
Reflection Layer

Image Fusion for Context Enhancement and Video Surrealism

Ramesh Raskar
Mitsubishi Electric Research Labs, (MERL)

Adrian Ilie
UNC Chapel Hill

Jingyi Yu
MIT



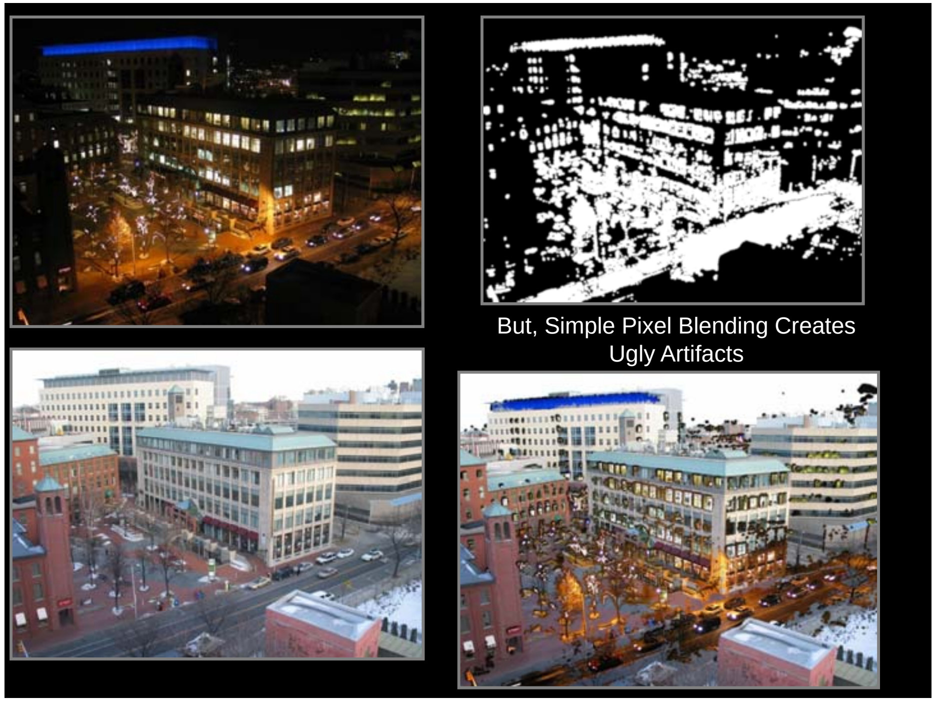
Night Image

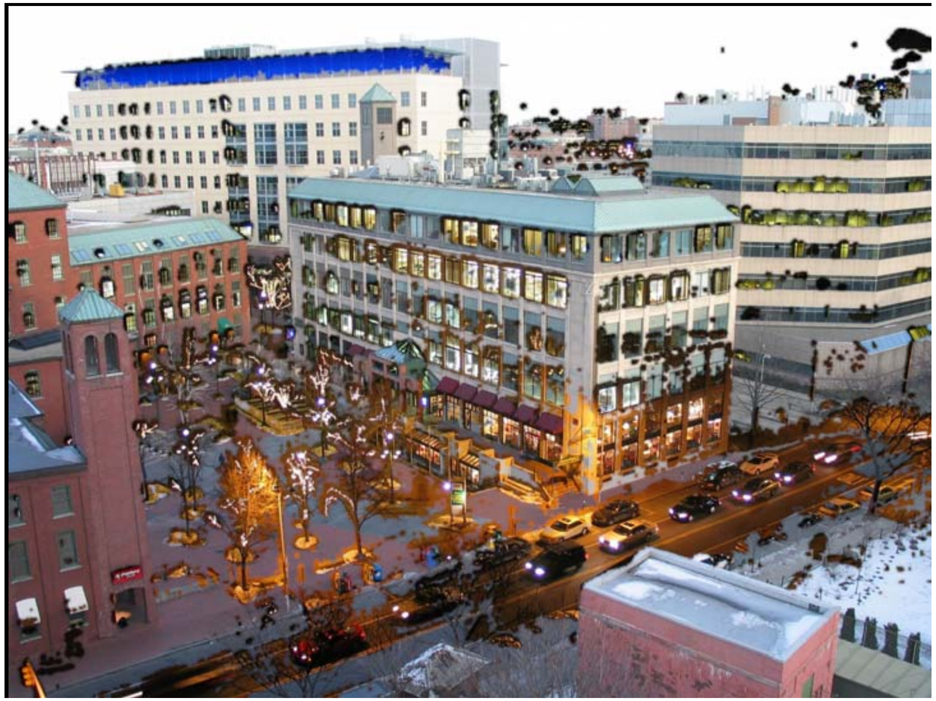
Background is captured from day-time scene using the same fixed camera

Context Enhanced Image

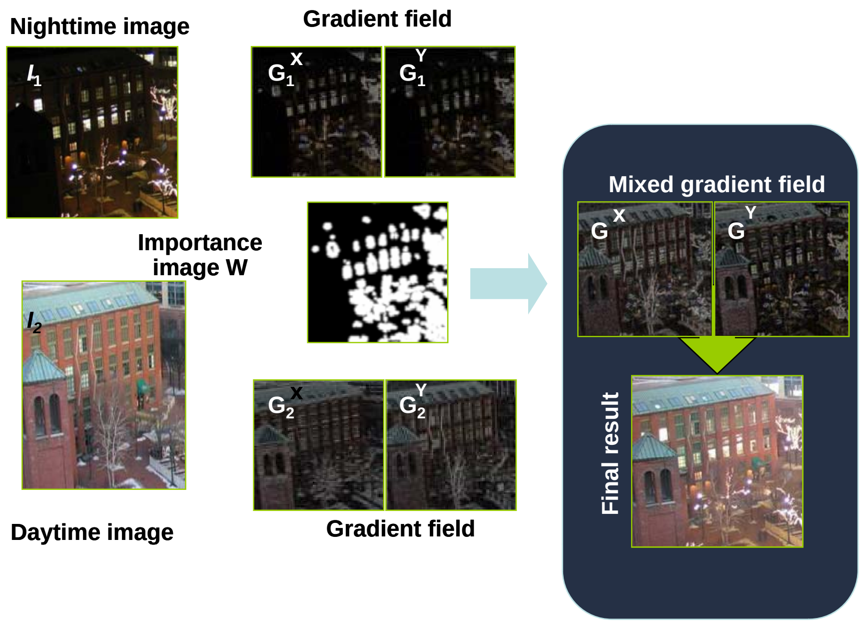
Day Image

Mask is automatically computed from scene contrast


But, Simple Pixel Blending Creates
Ugly Artifacts




Pixel Blending
solution:
Integration of
blended Gradients


Nighttime image
Gradient field
I_1
G_1^X
G_1^Y
Importance
image W
Mixed gradient field
G^X
G^Y
I_2
Final result
G_2^X
G_2^Y
Daytime image
Gradient field

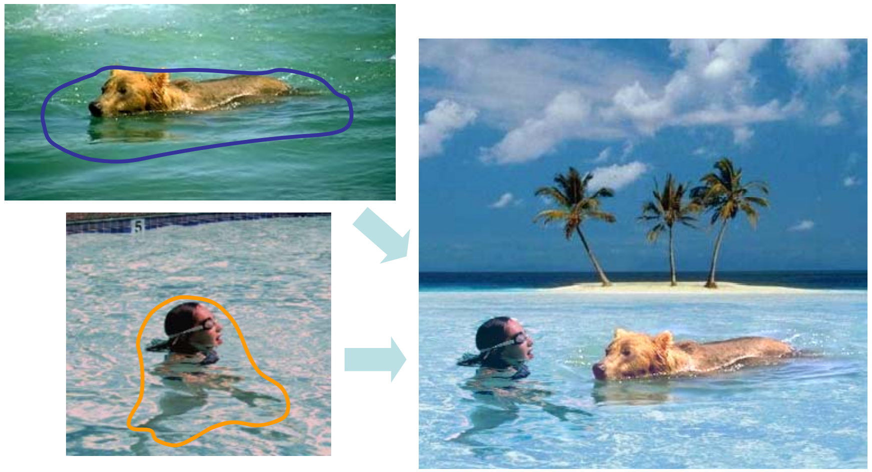
Poisson Image Editing

- Precise selection: tedious and unsatisfactory
- Alpha-Matting: powerful but involved
- Seamless cloning: loose selection but no seams?

Conceal

Copy Background gradients (user strokes)

Compose



Source Images

Target Image

Transparent Cloning

$$\mathbf{v} = \frac{\nabla f^* + \nabla g}{2}$$

Largest variation from source and destination at each point

Gradient Domain Manipulations: Overview

(A) Per pixel

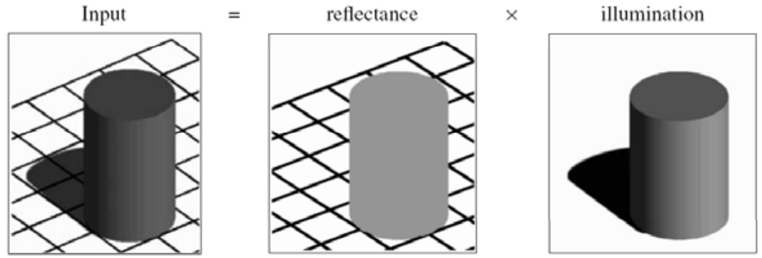
(B) Corresponding gradients in two images

(C) Corresponding gradients in multiple images

(D) Combining gradients along seams

Intrinsic images

- I = L * R
- L = illumination image
- R = reflectance image



Intrinsic images

- Use multiple images under different illumination
- Assumption
  - Illumination image gradients = Laplacian PDF
  - Under Laplacian PDF, Median = ML estimator
- At each pixel, take Median of gradients across images
- Integrate to remove shadows

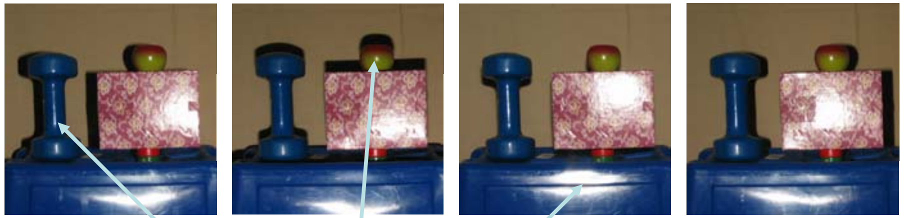
Yair Weiss, "Deriving intrinsic images from image sequences", ICCV 2001

frame 1   frame 11   ML reflectance

Shadow free
Intrinsic Image

Result = Illumination Image * (Label in Intrinsic Image)

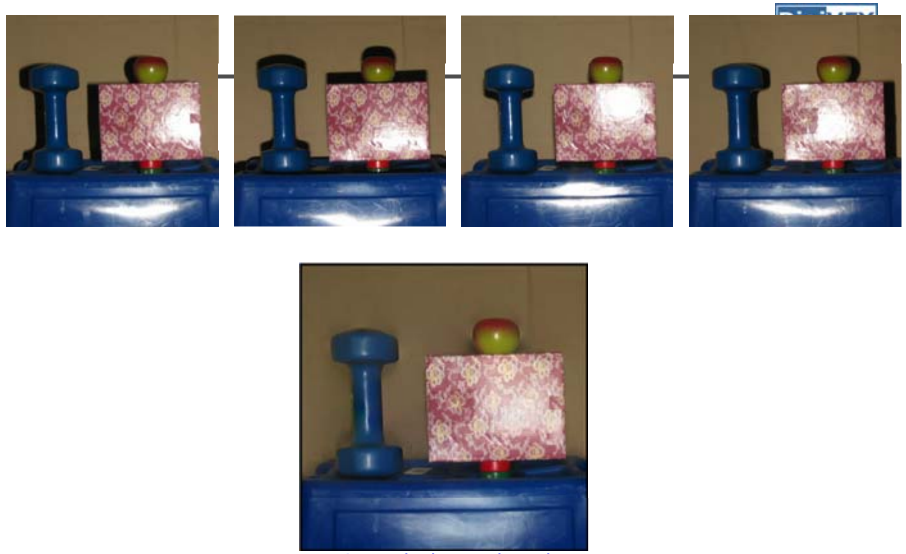
Specularity Reduction in Active Illumination



Line Specularity   Point Specularity   Area Specularity

Multiple images with same viewpoint, varying illumination

How do we remove highlights?



Specularity Reduced Image

Gradient Domain Manipulations: Overview

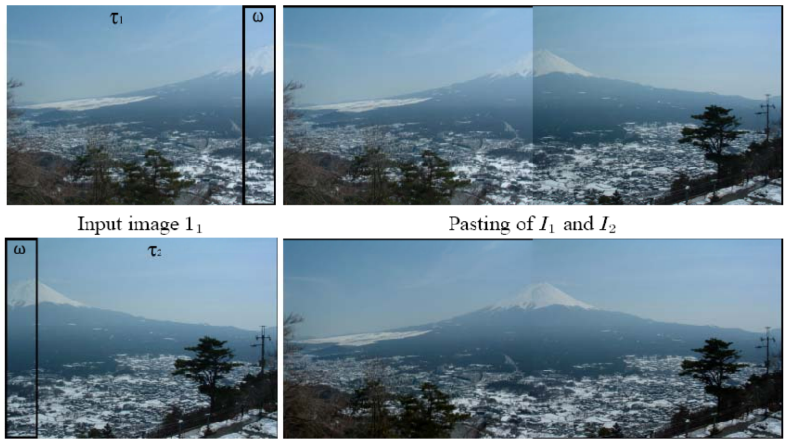
(A) Per pixel

(B) Corresponding gradients in <u>two images</u>

(C) Corresponding gradients in <u>multiple images</u>

(D) Combining gradients <u>along seams</u>

Seamless Image Stitching



Anat Levin, Assaf Zomet, Shmuel Peleg and Yair Weiss, “Seamless Image Stitching in the Gradient Domain”, ECCV 2004