中国科学技术大学
University of Science and Technology of China

数学建模
Mathematical Modeling

陈仁杰
中国科学技术大学

高阶拟合模型

数据拟合/插值再思考

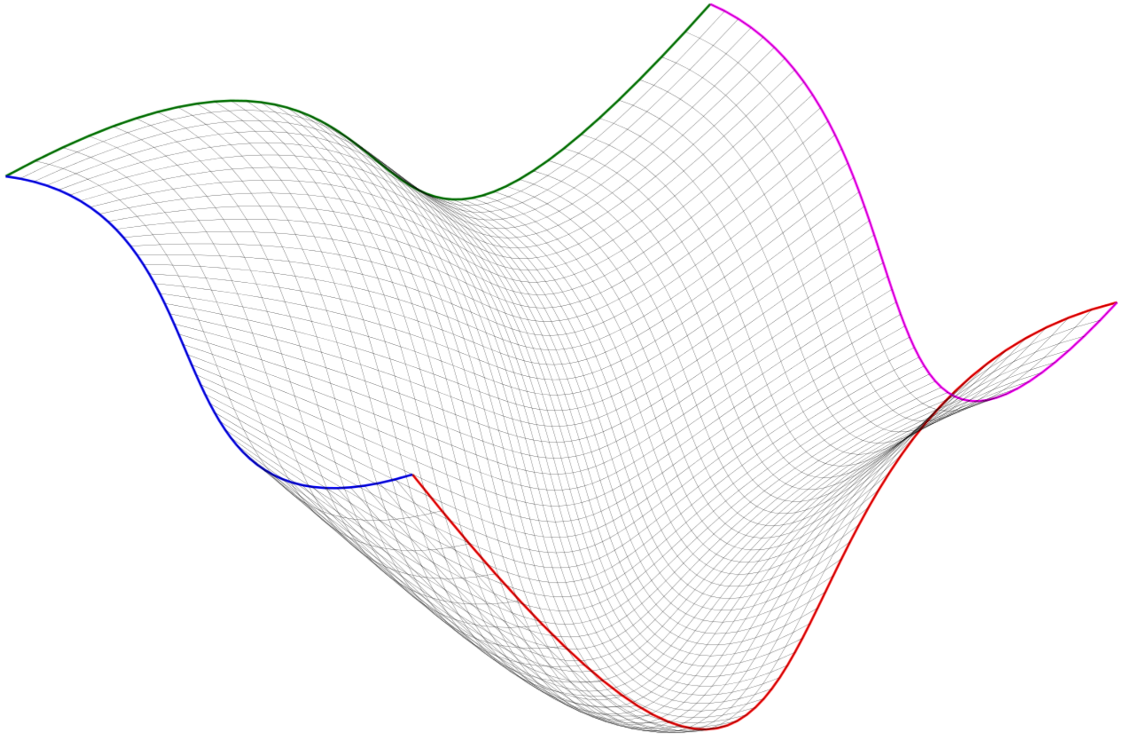
- 插值数据（函数值）
- 还有哪些属性可以插值?

超限插值
Transfinite Interpolation

问题：插值曲线？

Transfinite Interpolation

- 问题：给定4条边界曲线，构造插值这4条曲线的一张曲面



边界插值问题

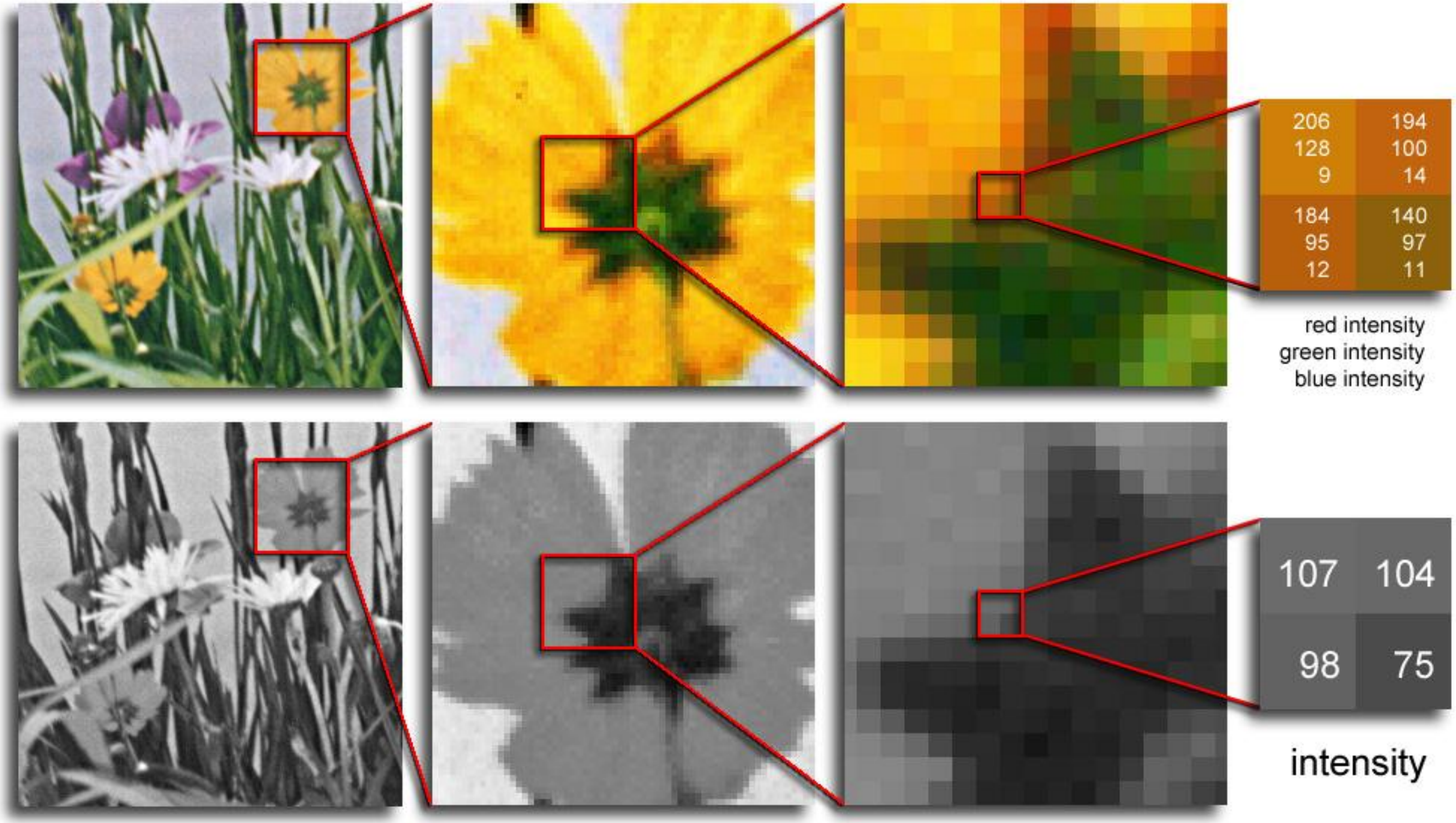
- 曲线
  - Hermite插值（3次、5次）
  - Lagrange插值
  - 几何样条
- 曲面
  - 线性混合曲面：插值2条曲线
  - Coons曲面（双线性混合曲面）：插值4条曲线
  - Bezier曲面
  - B样条曲面
  - Bernstein-Bezier三角曲面

高阶插值

高阶插值

- 插值高阶导数
- 多元：偏导数
- 更高阶光滑性

Color Image and Gray Image



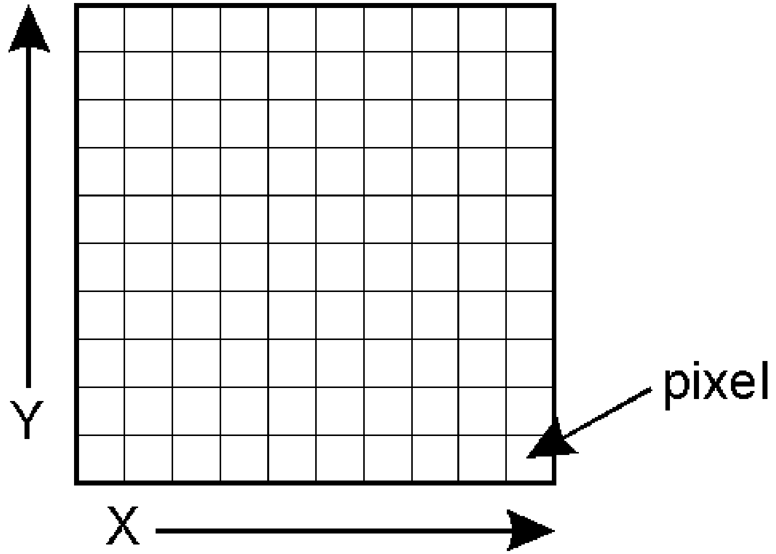
图像的另一个模型：函数

- 2D区域上的向量值函数

$$I = I(x, y)$$

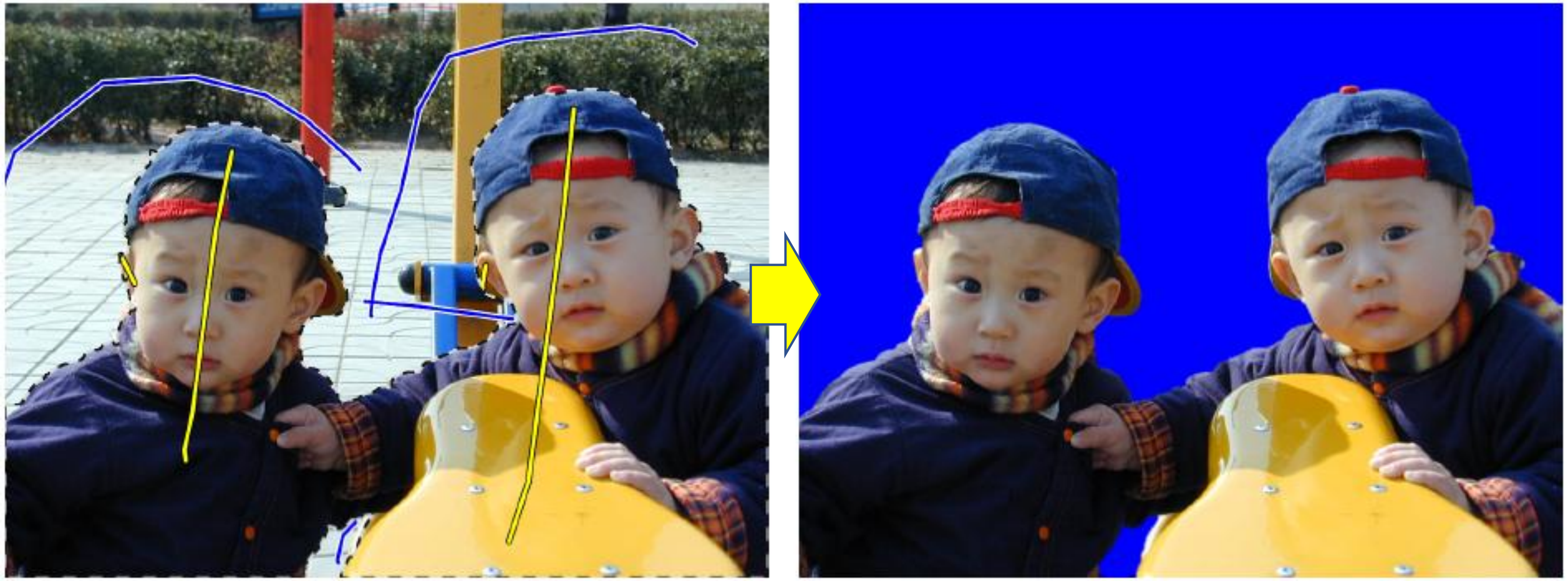
Discrete sampling

$$I = f(i, j)$$

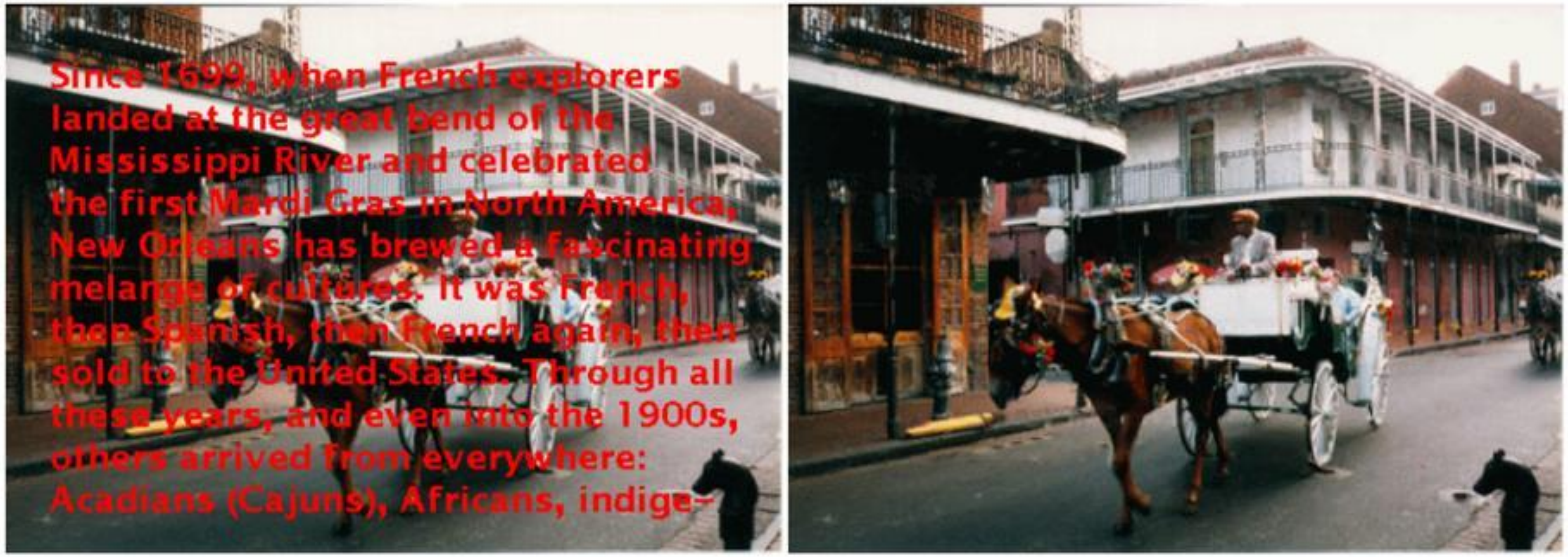


思考： 图像分割

前景/背景交互式图像分割

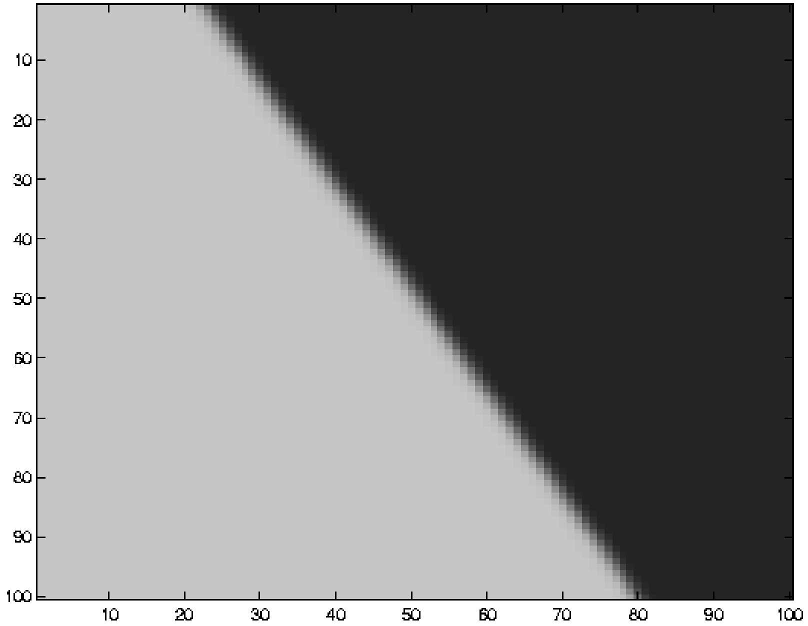


思考： 图像修复

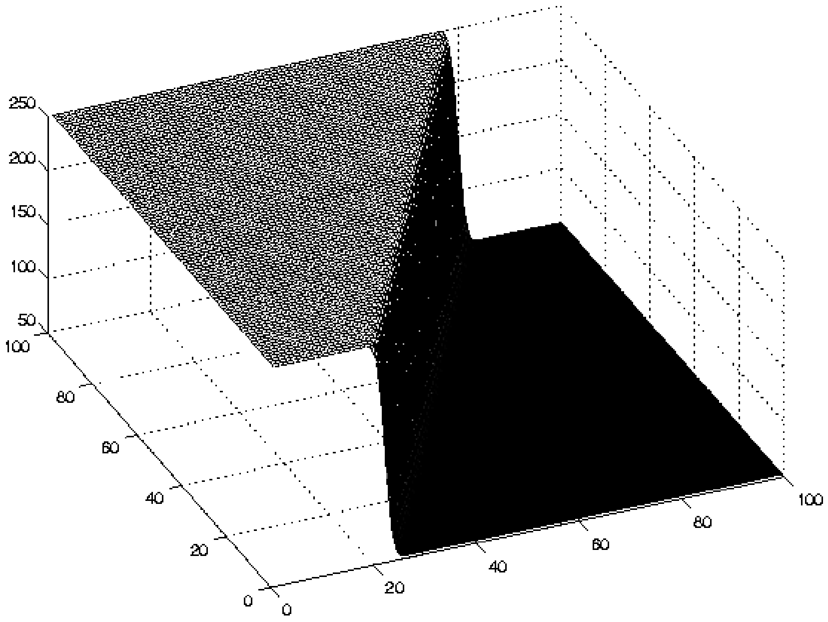

Since 1699, when French explorers
landed at the great bend of the
Mississippi River and celebrated
the first Mardi Gras in North America,
New Orleans has brewed a fascinating
melange of cultures. It was French,
then Spanish, then French again, then
sold to the United States. Through all
these years, and even into the 1900s,
others arrived from everywhere:
Acadians (Cajuns), Africans, indige-

Partial Differential Equation (PDE)

Image as a Surface



single-edge image



3D visualization

If image can be viewed as a surface, it is then natural to ask: can we apply *geometric* tools to process this surface (or its equivalent image signals)?

Courtesy of Xin Li

Geometric Formulation

- Image I: $R^2 \rightarrow R$ may be viewed as a 2D surface in R^3,

$$\text{M}: S(x,y) = (x, y, I(x,y)) \subset R^3$$

⬇

$$ds^2 = dx^2 + dy^2 + dI^2 = dx^2 + dy^2 + (I_x dx + I_y dy)^2$$

$$= (1 + I_x^2)dx^2 + 2I_x I_y dxdy + (1 + I_y^2)dy^2$$

⬇

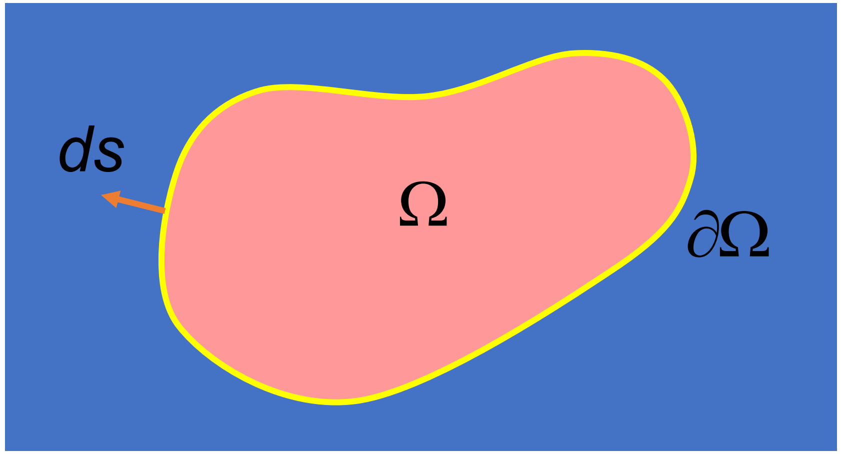
$$ds^2 = [dx \quad dy] G \begin{bmatrix} dx \\ dy \end{bmatrix}, G = \begin{bmatrix} 1 + I_x^2 & I_x I_y \\ I_x I_y & 1 + I_y^2 \end{bmatrix}$$

G: symmetric and positive definite matrix

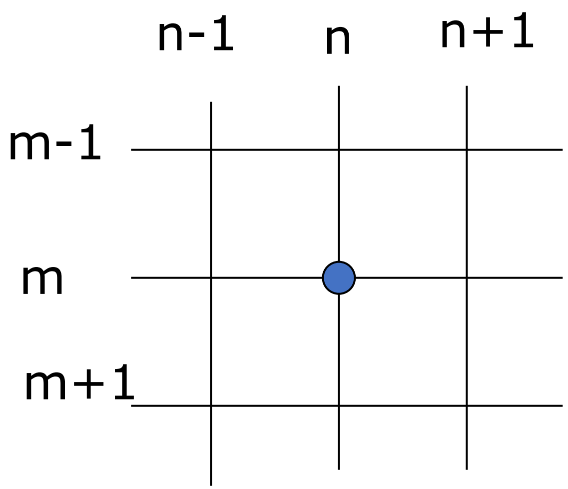
PDE-based Image Processing

- Image as a function (surface)

The solution of a PDE Equation is uniquely determined in Ω, if *Dirichlet* boundary conditions or *Neumann* boundary conditions are specified on $\partial\Omega$



Numerical Implementation of PDEs



$$\left.\frac{\partial I}{\partial x}\right|_{(m,n)} \approx \frac{I(m+1,n)-I(m-1,n)}{2h}$$

or

$$\left.\frac{\partial I}{\partial x}\right|_{(m,n)} \approx \lambda \frac{I(m+1,n)-I(m-1,n)}{2h} + \frac{1-\lambda}{2}\left(\frac{I(m+1,n+1)-I(m-1,n+1)}{2h} + \frac{I(m+1,n-1)-I(m-1,n-1)}{2h}\right)$$

$$\left.\Delta I\right|_{(m,n)} \approx \frac{I(m+1,n)+I(m-1,n)+I(m,n+1)+I(m,n-1)-4I(m,n)}{h^2}$$

or

$$\left.\Delta I\right|_{(m,n)} \approx \lambda \frac{I(m+1,n)+I(m-1,n)+I(m,n+1)+I(m,n-1)-4I(m,n)}{h^2} + (1-\lambda)\frac{I(m+1,n+1)+I(m-1,n+1)+I(m+1,n+1)+I(m-1,n-1)-4I(m,n)}{h^2}$$

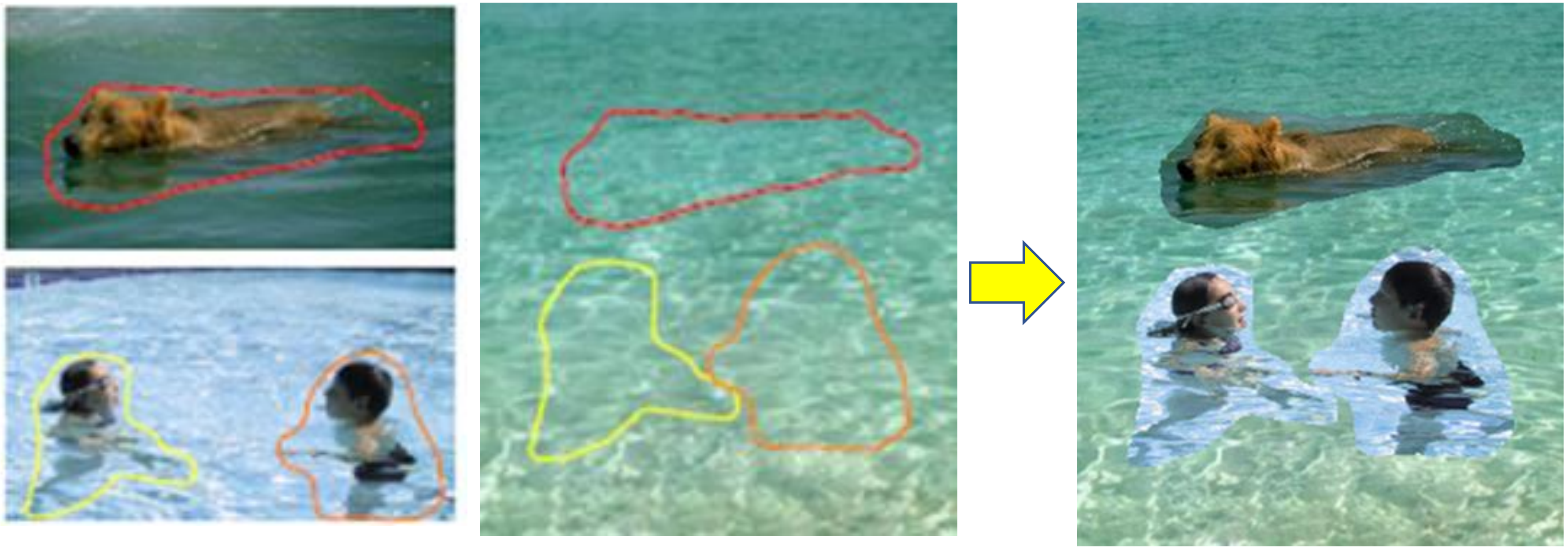
高阶插值例子：

Poisson Image Editing

Perez et al. Poisson image editing. Siggraph 2003.
https://www.cs.jhu.edu/~misha/Fall07/Papers/Perez03.pdf

图像移植

- 能否做到无缝移植?



Demo

如何做到的？

sources/destinations　cloning　seamless cloning

Poisson方程

Poisson Equation

$$\Delta\Phi = -4\pi G\rho(x, y)$$

$$\Delta \equiv \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$$

Siméon Denis Poisson （泊松）

- His teachers:
  - *Laplace*
  - *Lagrange*
- Poisson’s terms:
  - Poisson's equation
  - Poisson's integral
  - Poisson distribution
  - Poisson brackets
  - Poisson's ratio
  - Poisson's constant

1781-1840, France

Siméon Denis Poisson

"Life is good for only two things:

to study mathematics and to teach it."

1781-1840, France

Background

- Partial Differential Equations (PDE)

$$E(f, f_x, f_y, f_{xx}, f_{xy}, f_{yy}) = 0$$

- The PDE's which occur in physics are mostly second order and linear:

$$A \cdot f_{xx} + 2B \cdot f_{xy} + C \cdot f_{yy} + D \cdot f_x + E \cdot f_y + F \cdot f + G = 0$$

$$A \cdot f_{xx} + 2B \cdot f_{xy} + C \cdot f_{yy} + D \cdot f_x + E \cdot f_y + F \cdot f + G = 0$$

$A \cdot C < B^2$:
- Hyberbolic
  - wave equation: $$\Delta f = \frac{1}{v^2} \frac{\partial^2 f}{\partial t^2}$$

$A \cdot C = B^2$:
- Parobolic
  - heat equation: $$\frac{\partial f}{\partial t} = k \cdot \Delta f$$

$A \cdot C > B^2$:
- Elliptic
  - Laplace equation: $$\Delta f = 0$$
  - Poisson equation: $$\Delta f = -\rho$$

Poisson Equation

$$\Delta f = -\rho$$

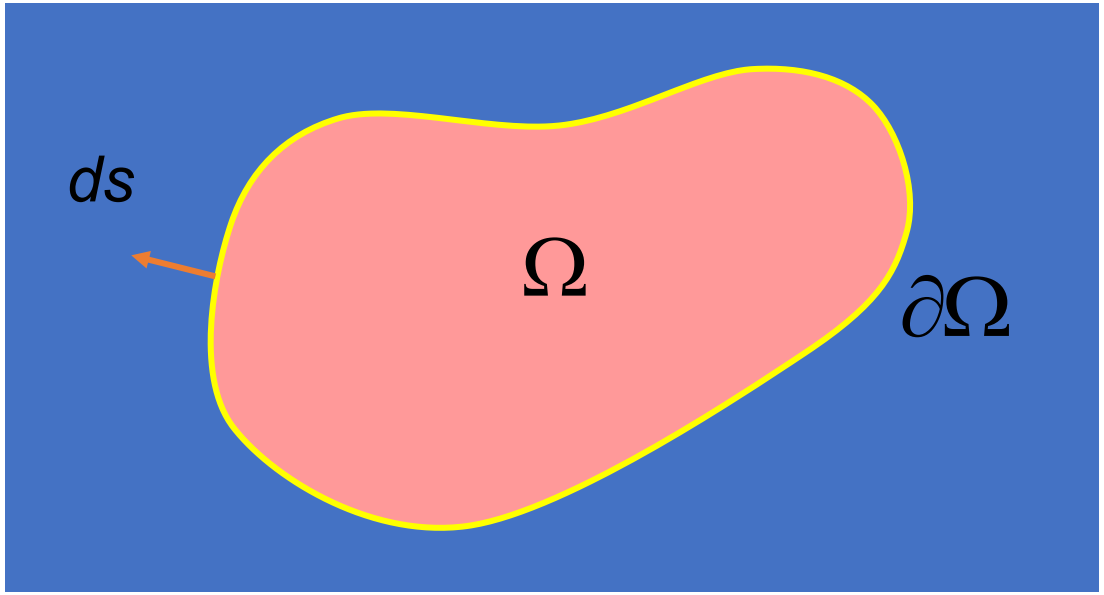
$$\Delta \equiv \frac{\partial^2}{\partial^2 x} + \frac{\partial^2}{\partial^2 y}$$

$$\rho = \rho(x, y)$$

Boundary conditions

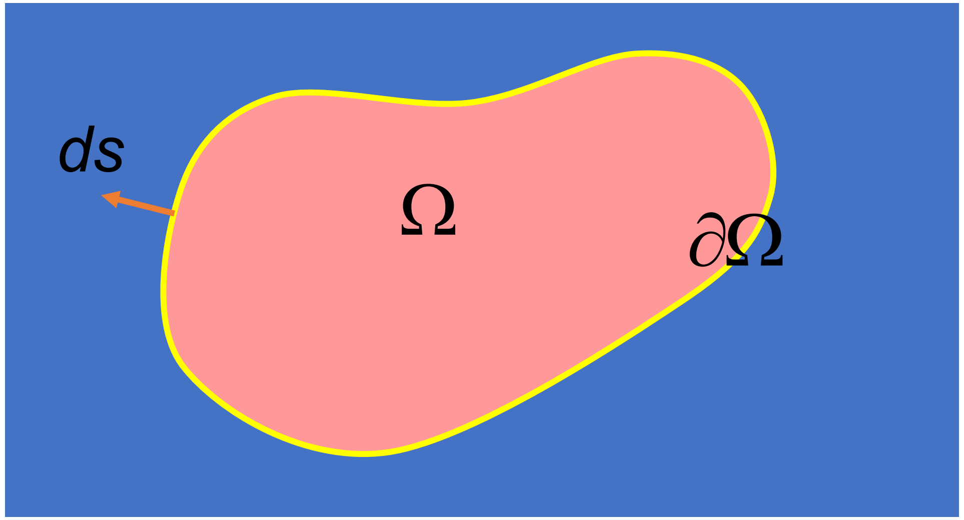
- *Dirichlet* boundary conditions: $f|_{\partial\Omega}$

- *Neumann* boundary conditions: $\frac{\partial f}{\partial \mathbf{s}}|_{\partial\Omega}$

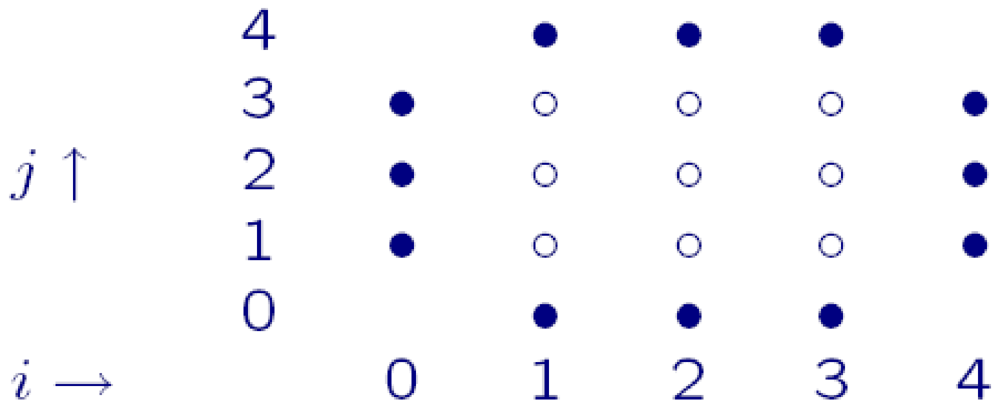


Existence of solution

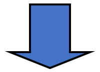
The solution of an Poisson Equation is uniquely determined in Ω, if *Dirichlet* boundary conditions or *Neumann* boundary conditions are specified on $\partial\Omega$



Discrete Poisson Equation



$$\Delta f = -\rho$$



$$\frac{f_{i+1,j} + f_{i-1,j} - 2f_{i,j}}{h^2} + \frac{f_{i,j+1} + f_{i,j-1} - 2f_{i,j}}{h^2} = \rho_{i,j}$$

Matrix Nature of Poisson Equation

- Sparse linear system

$$\begin{bmatrix} -4 & 1 & & & & & & & \\ 1 & -4 & 1 & & & & & & \\ & 1 & -4 & 1 & & & & & \\ & & 1 & -4 & 1 & & & & \\ & & & 1 & -4 & 1 & & & \\ & & & & 1 & -4 & 1 & & \\ & & & & & 1 & -4 & 1 & \\ & & & & & & 1 & -4 & 1 \\ & & & & & & & 1 & -4 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{21} \\ f_{31} \\ f_{12} \\ f_{22} \\ f_{32} \\ f_{13} \\ f_{23} \\ f_{33} \end{bmatrix} = \begin{bmatrix} b_{11} \\ b_{21} \\ b_{31} \\ b_{12} \\ b_{22} \\ b_{32} \\ b_{13} \\ b_{23} \\ b_{33} \end{bmatrix}$$

Poisson Equation Solver

- Direct method
- Iterative methods
  - Jacobi, Gauss-Seidel, SOR
- Multigrid method

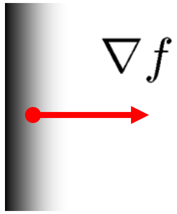
Gradient Image Processing

Image gradient

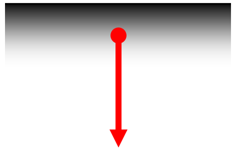
- The gradient of an image:

$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

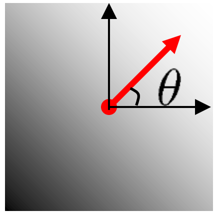
- The gradient points in the direction of most rapid change in intensity



$$\nabla f = \left[\frac{\partial f}{\partial x}, 0\right]$$



$$\nabla f = \left[0, \frac{\partial f}{\partial y}\right]$$



$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

The gradient direction is given by:

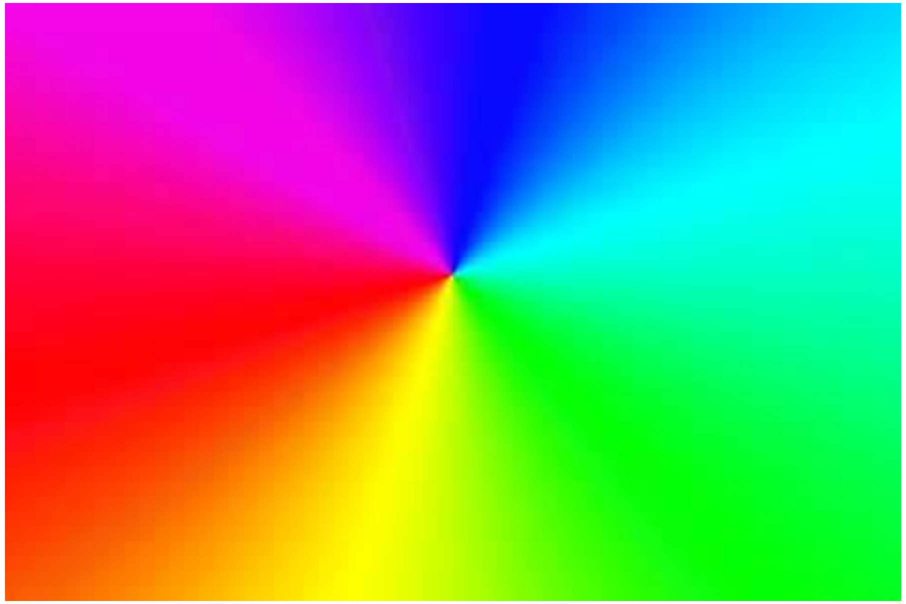
$$\theta = \tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right)$$

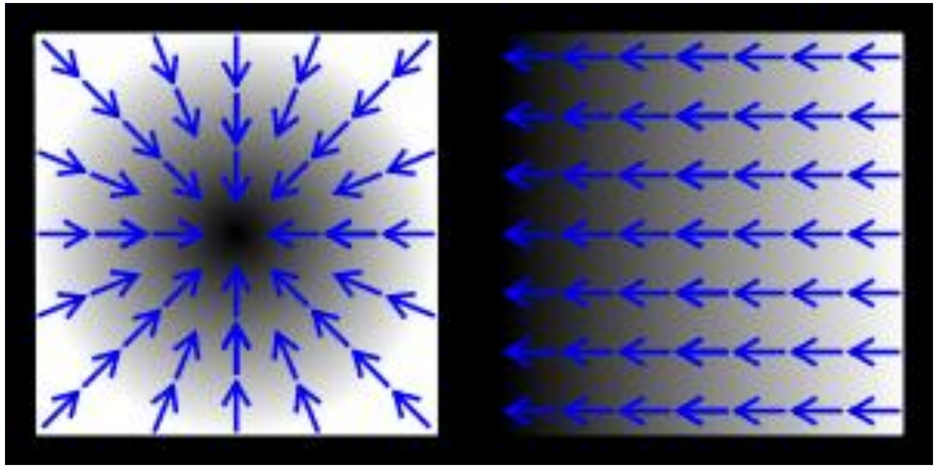
- how does this relate to the direction of the edge?

The *edge strength* is given by the gradient magnitude

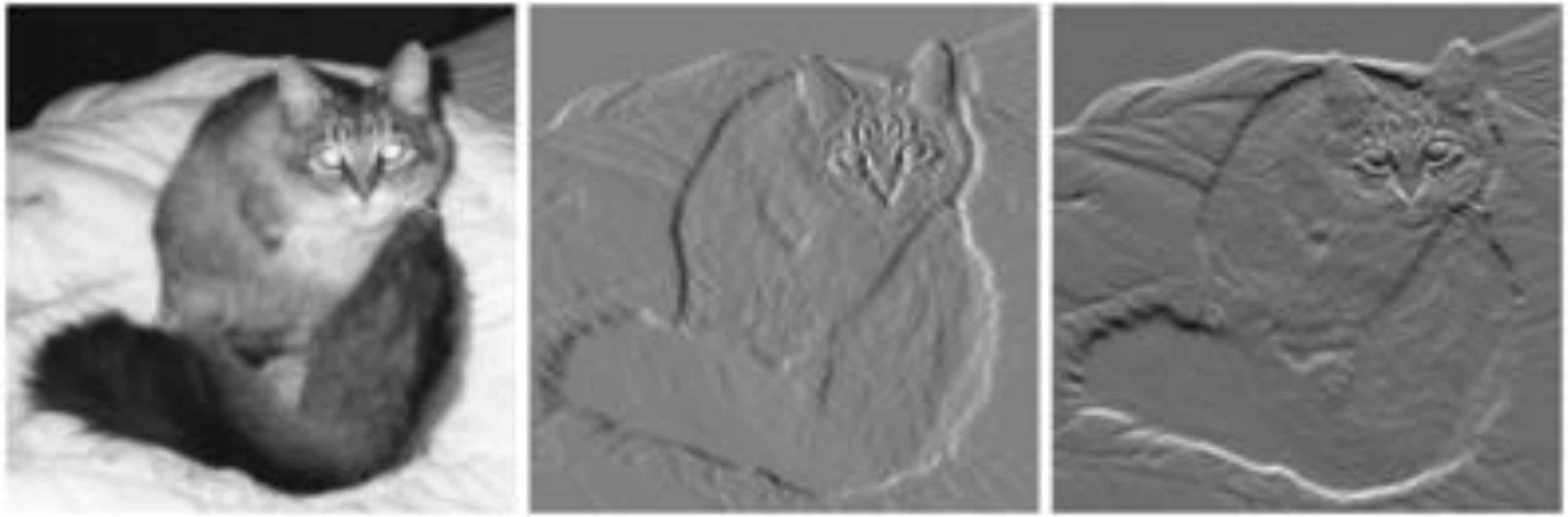
$$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

Gradient (梯度)





Gradients



X-gradient Y-gradient

Why gradients?

- Human visual system is very sensitive to gradient
- Gradient encode edges and local contrast quite well

- Method
  - Do your editing in the gradient domain
  - Reconstruct image from gradient

Cloning Gradient

sources/destinations

seamless cloning

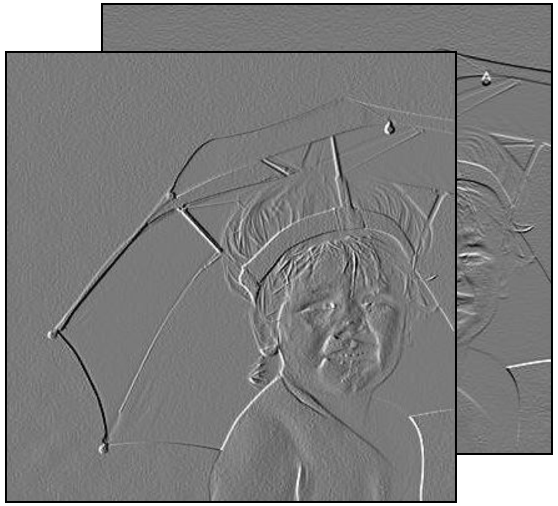
Variational interpretation

$$f^* = \text{argmin}_f \iint_\Omega \underbrace{\|\nabla f - \mathbf{v}\|^2}_{F} \quad \text{s.t } f^*|_{\partial\Omega} = f|_{\partial\Omega}$$

Euler Equation: $F_f - \frac{\partial}{\partial x} F_{f_x} - \frac{\partial}{\partial y} F_{f_y} = 0$

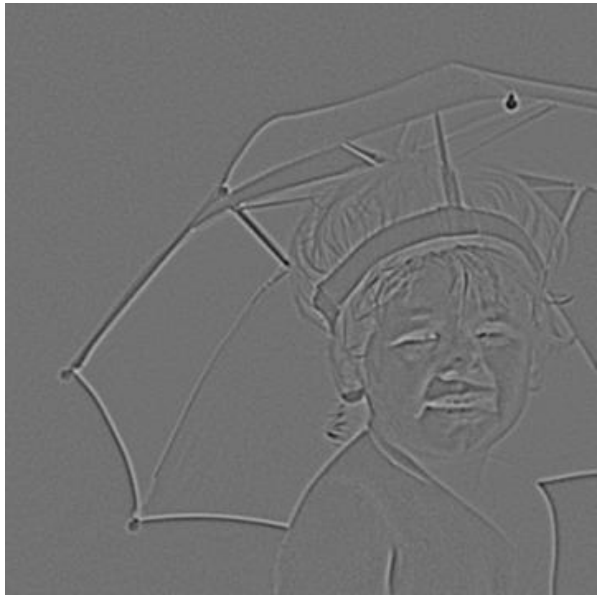
$$\Delta f = div(\mathbf{v}) \;\; \text{s.t } f^*|_{\partial\Omega} = f|_{\partial\Omega}$$

V is a guidance field, needs not to be a gradient field.

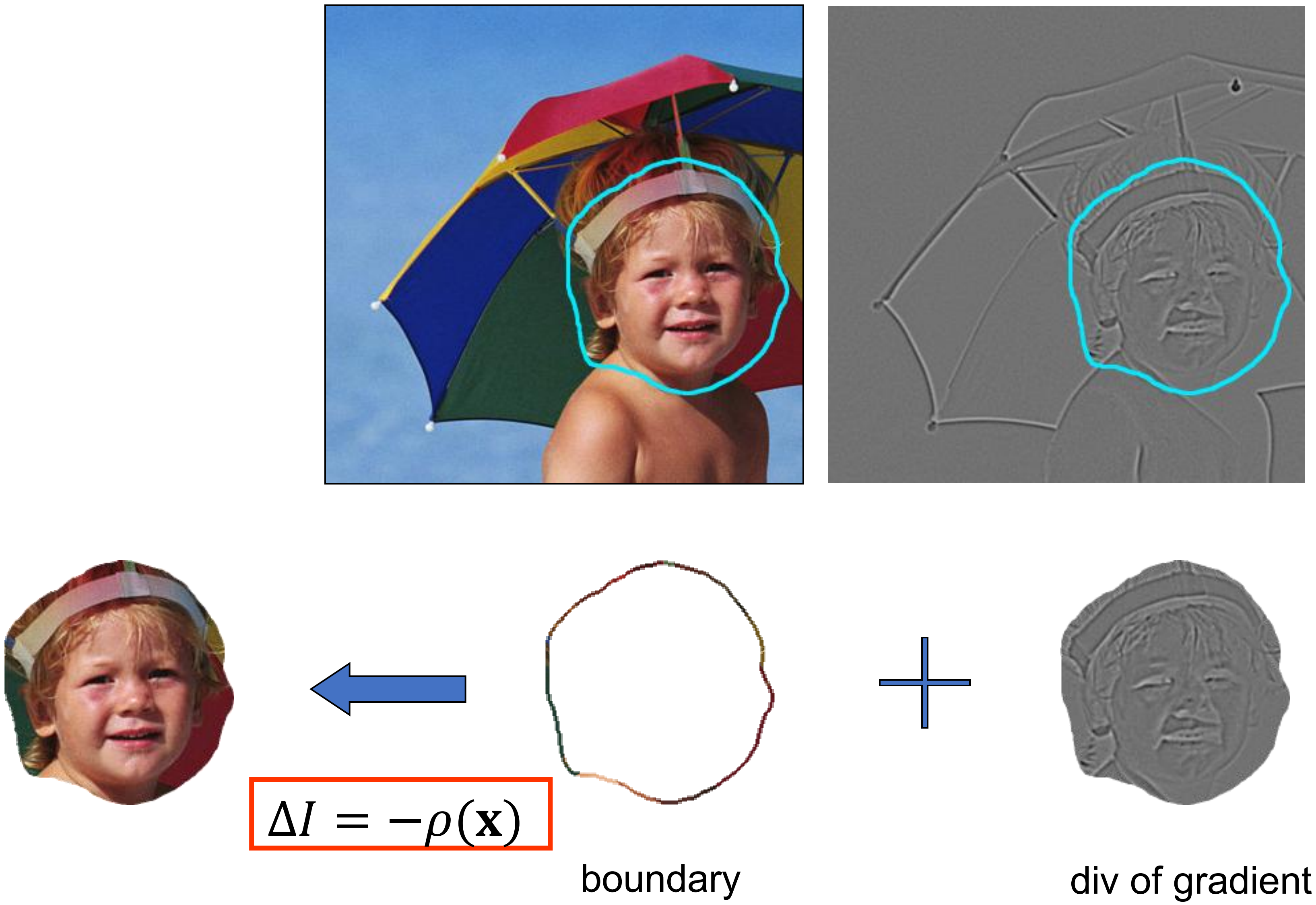


$$\mathbf{g} = -\nabla I$$

$$\rho(\mathbf{x}) = div(\mathbf{g}) \equiv \lim_{V \to 0} \frac{\oint_S \mathbf{g} \cdot d\mathbf{s}}{V}$$



$$\Delta I = -\rho(\mathbf{x})$$


$\Delta I = -\rho(\mathbf{x})$
boundary
div of gradient

Poisson Image Editing

[Perez et al., Siggraph 2003]

Seamless Cloning

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

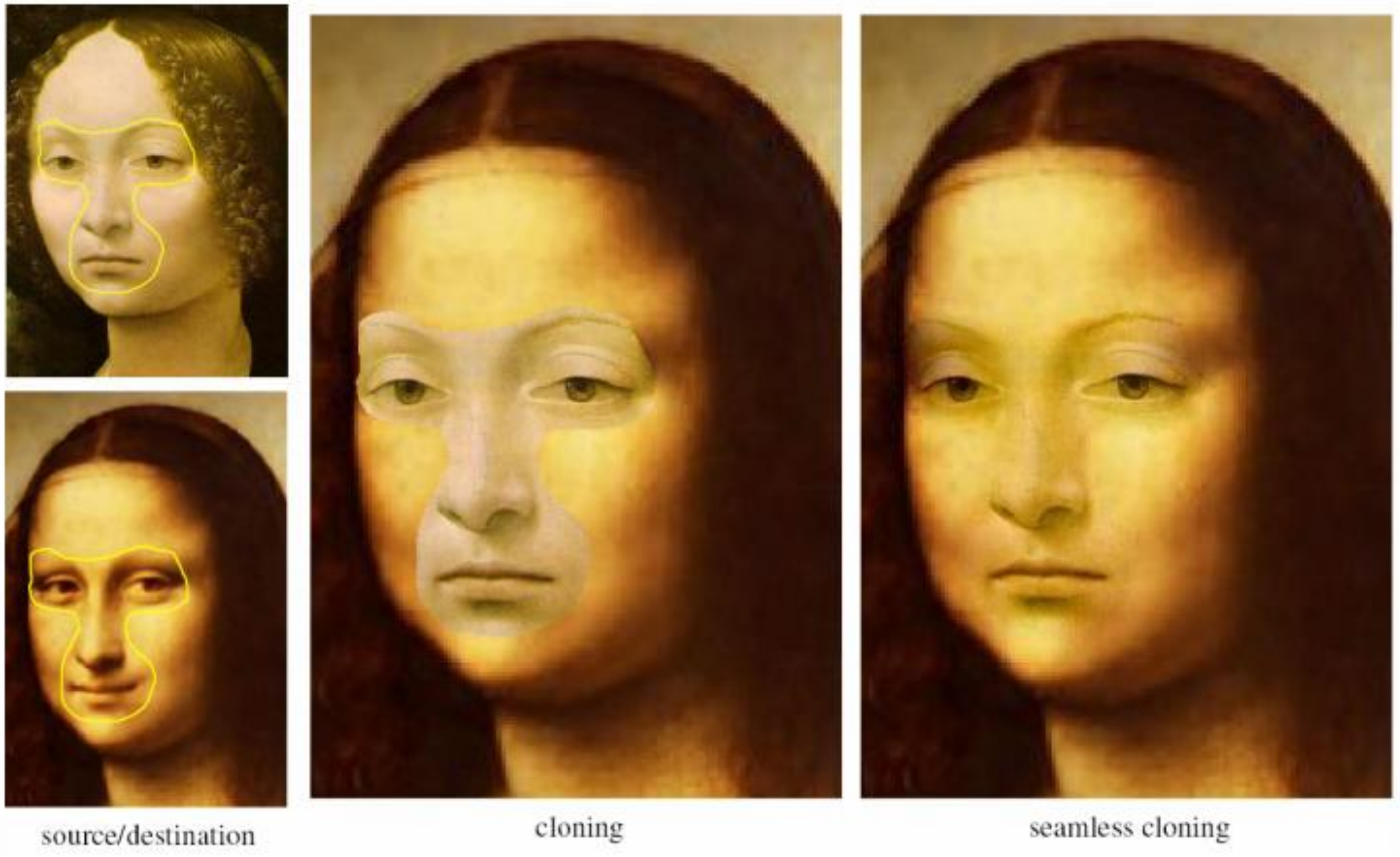
Cloning by solving Poisson Equation

$$\Delta I = div(\nabla I_A) \qquad \text{s.t.} \quad I|_{\partial\Omega} = I_B|_{\partial\Omega}$$

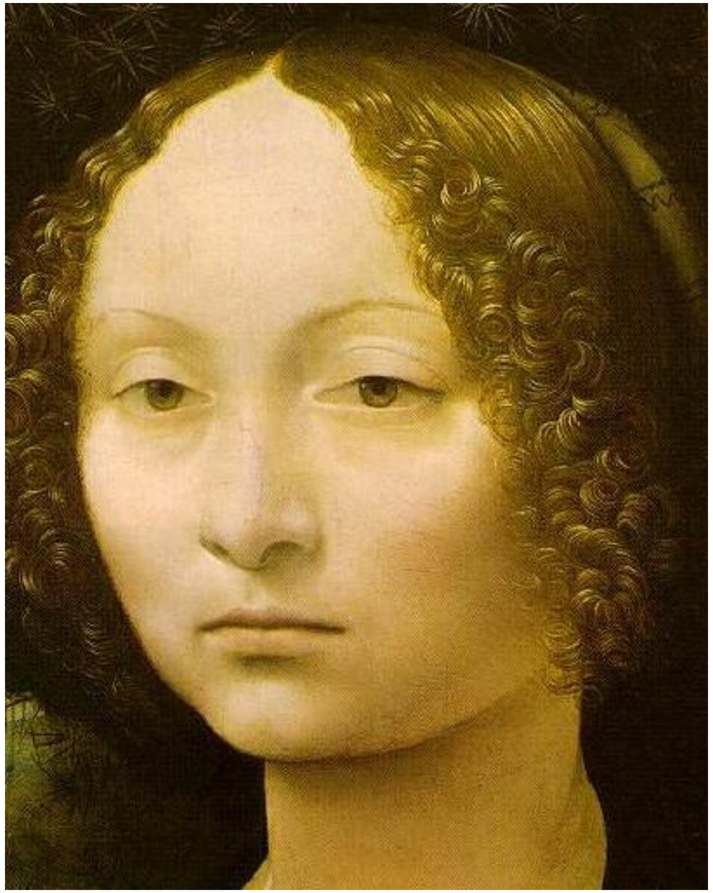
I_A $\qquad$ I_B

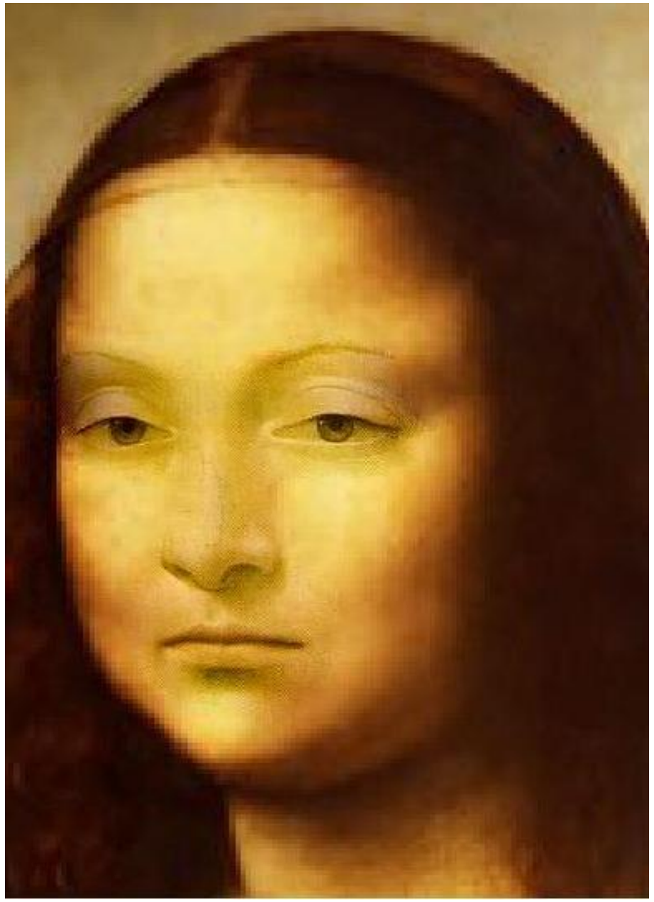
compose

Cloning Gradient



Cloning Gradient

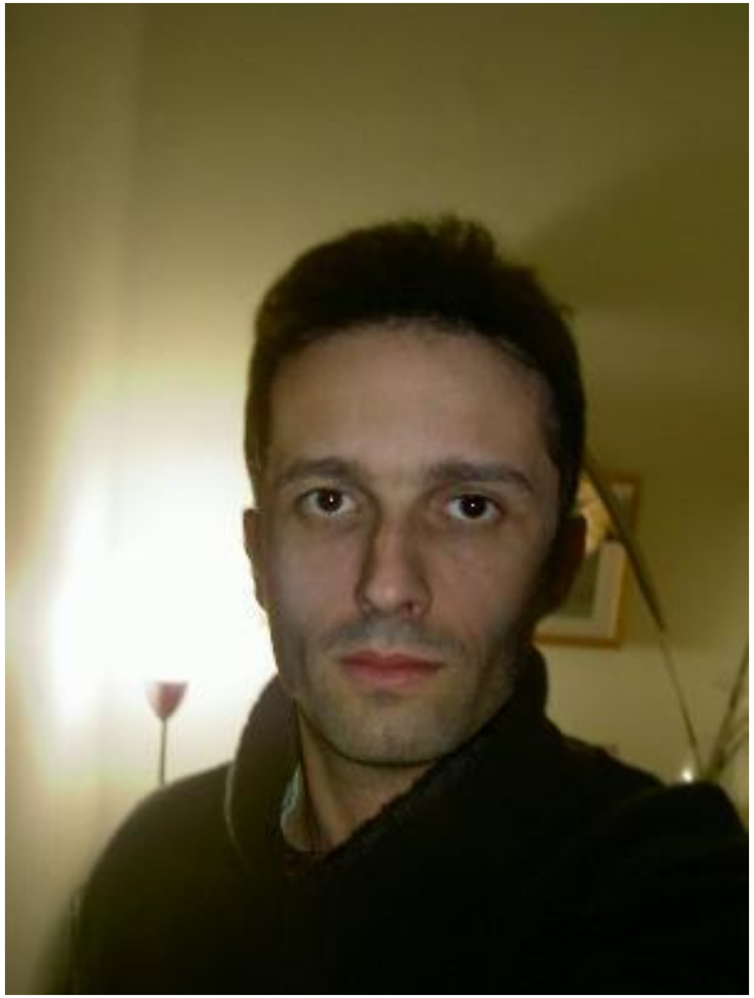




change texture

conceal

mix lights



change colors

change colors

Seamless Tiling

Multiple images tiled at random

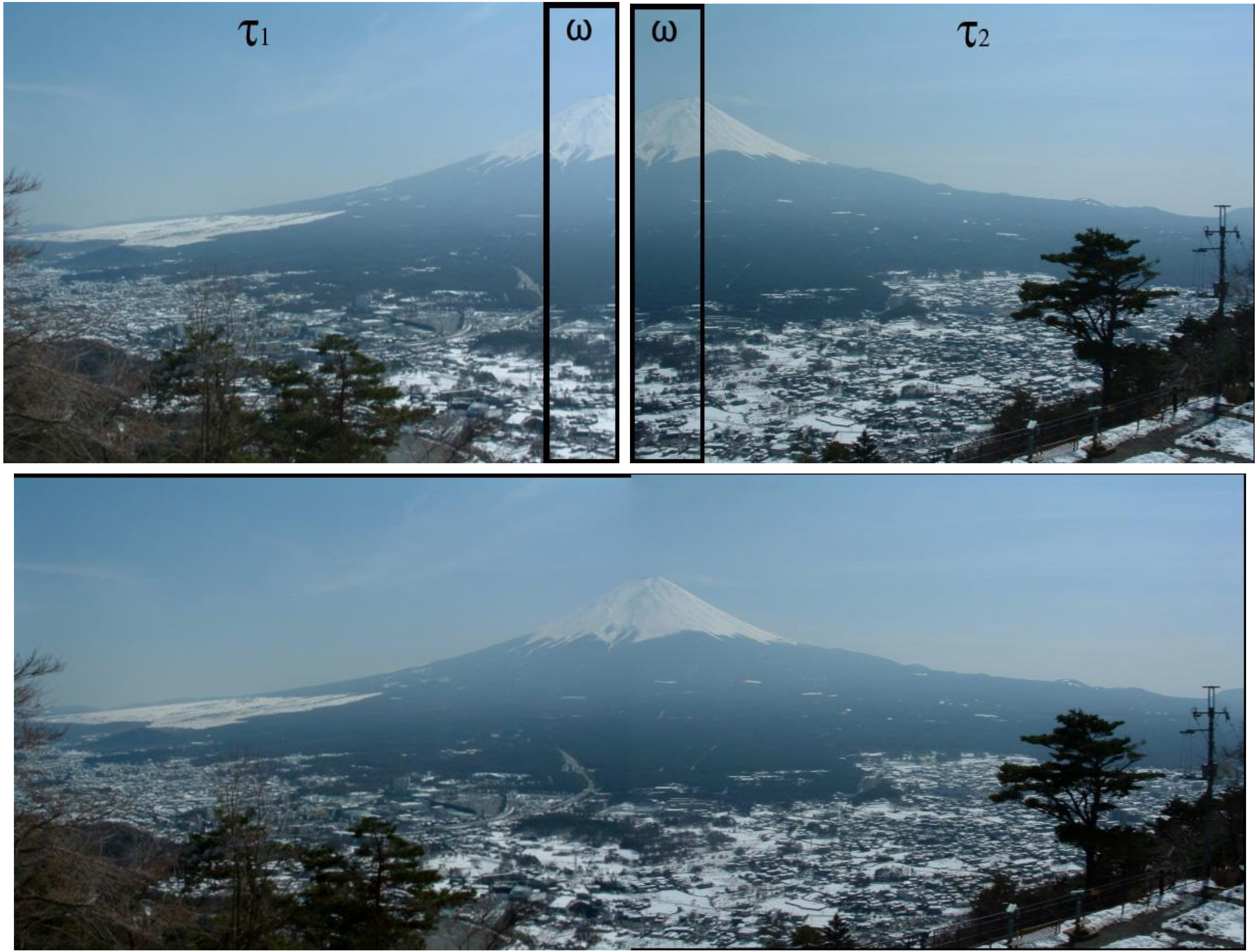
More on Gradient-based Method

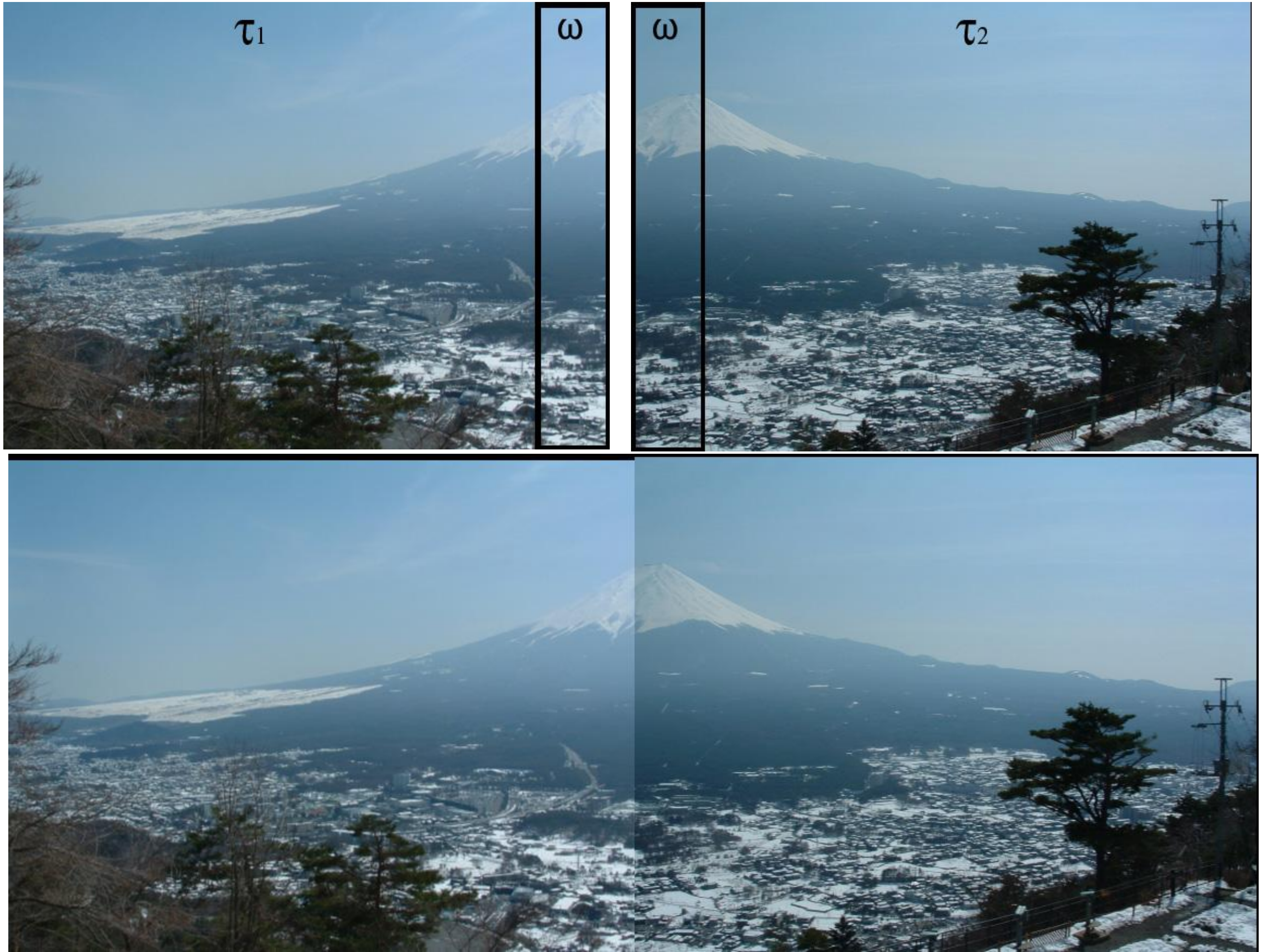
Gradients and grayscale images

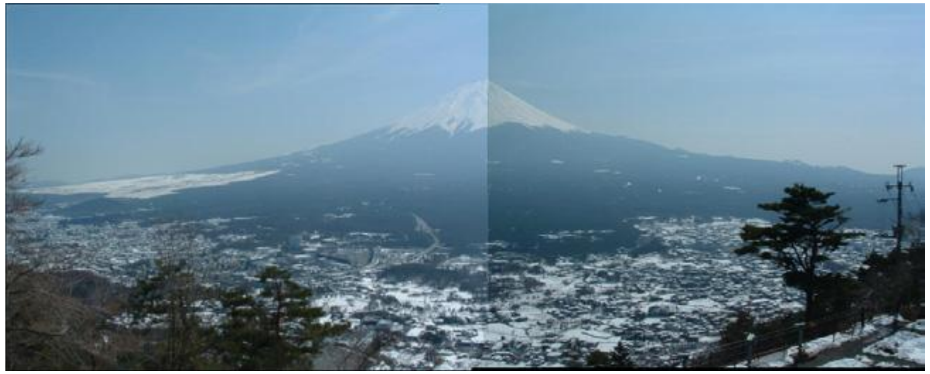
- **Grayscale image: n× n scalars**
- **Gradient: n× n *2D vectors***
- **Overcomplete!**
- **What's up with this?**
- **Not all vector fields are the gradient of an image!**
- **Only if they are curl-free (a.k.a. conservative)**
  - But it does not matter for us

Example: Image Stitching

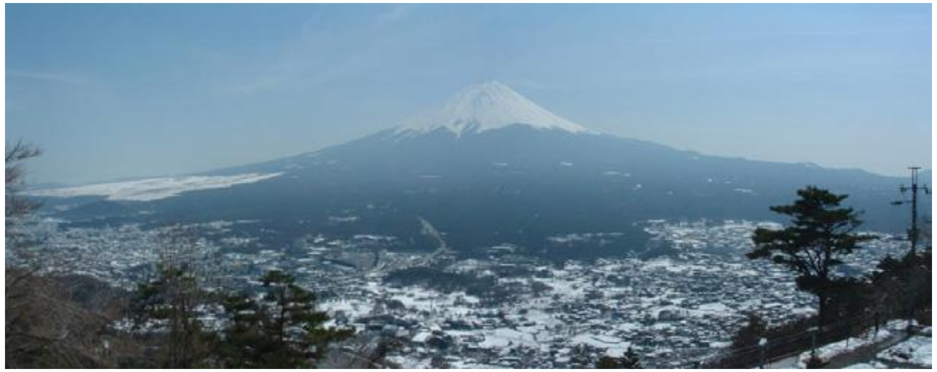
http://www.cs.huji.ac.il/~alevin/papers/eccv04-blending.pdf
http://eprints.pascal-network.org/archive/00001062/01/tips05-blending.pdf


τ1
ω
ω
τ2

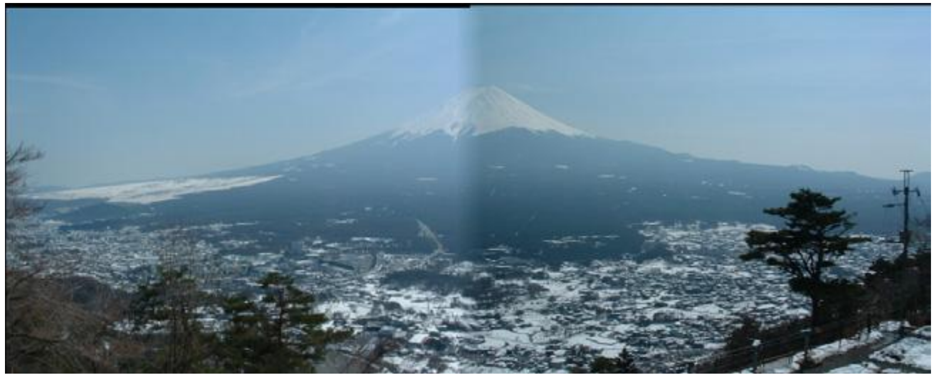

τ_1
ω
ω
τ_2



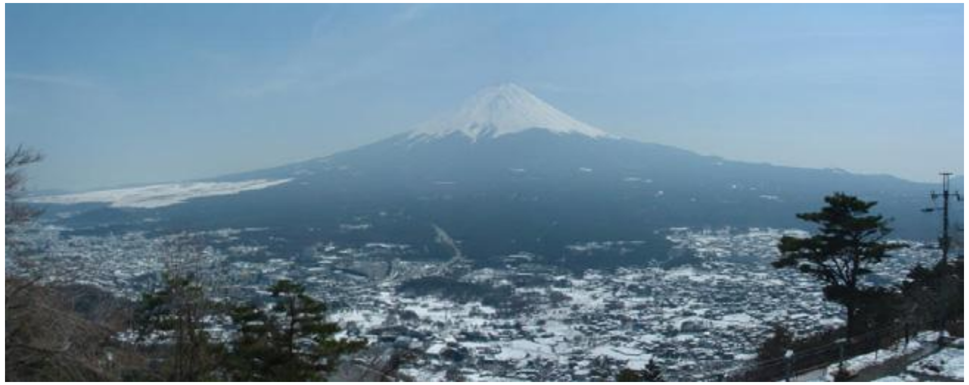
Optimal seam



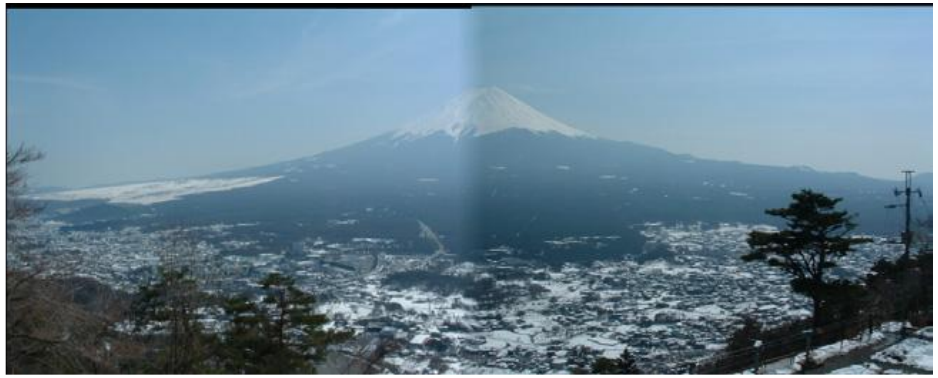
Optimal seam on the gradients



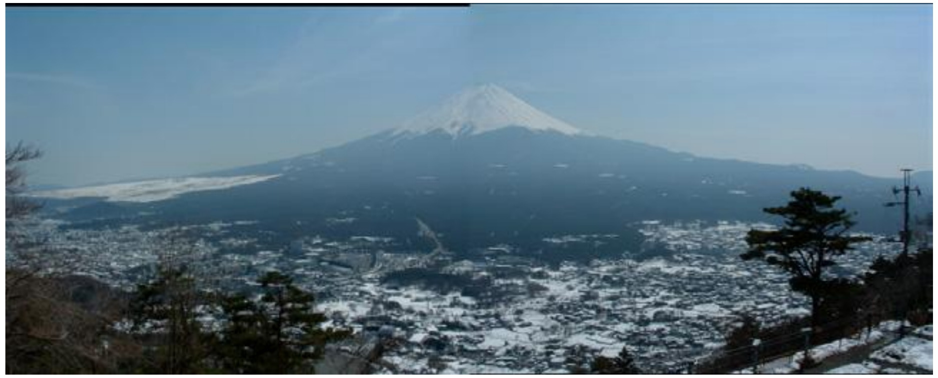
Feathering



Pyramid blending on the gradients



Pyramid blending



GIST1

PDE based Image Processing

1. Mean Curvature Evolution

Mean Curvature

Unit normal of this surface is

$$N(x,y) = \frac{S_x \times S_y}{||S_x \times S_y||} = \frac{(-I_x, -I_y, 1)}{\sqrt{1 + I_x^2 + I_y^2}}$$

Mean curvature is

$$H(x,y) = \frac{I_{xx}(1 + I_y^2) - 2I_x I_y I_{xy} + I_{yy}(1 + I_x^2)}{2(1 + I_x^2 + I_y^2)^{3/2}}$$

<u>Minimal Surface Theorem</u>

Surfaces of zero mean curvature have minimal areas

$$\frac{I_{xx}(1 + I_y^2) - 2I_x I_y I_{xy} + I_{yy}(1 + I_x^2)}{2(1 + I_x^2 + I_y^2)^{3/2}} = 0$$

Discrete Mean Curvature

$$S(M) = \iint_{\Omega} \sqrt{\det(G)}dxdy = \iint_{\Omega} \sqrt{1 + I_x^2 + I_y^2}dxdy$$

minimize S(M) leads to

Euler-Lagrange Equation:

$$\frac{\partial}{\partial x}\left(\frac{I_x}{\sqrt{1 + I_x^2 + I_y^2}}\right) + \frac{\partial}{\partial y}\left(\frac{I_y}{\sqrt{1 + I_x^2 + I_y^2}}\right) = 0 \quad \text{(A)}$$

$$\frac{I_{xx}(1 + I_y^2) - 2I_x I_y I_{xy} + I_{yy}(1 + I_x^2)}{2(1 + I_x^2 + I_y^2)^{3/2}} = 0 \quad \text{(B)}$$

Mean Curvature Diffusion

Diffusion equation

$$I_t = \frac{I_{xx}(1 + I_y^2) - 2I_xI_yI_{xy} + I_{yy}(1 + I_x^2)}{2(1 + I_x^2 + I_y^2)^{3/2}}$$

Before post-processing

After post-processing

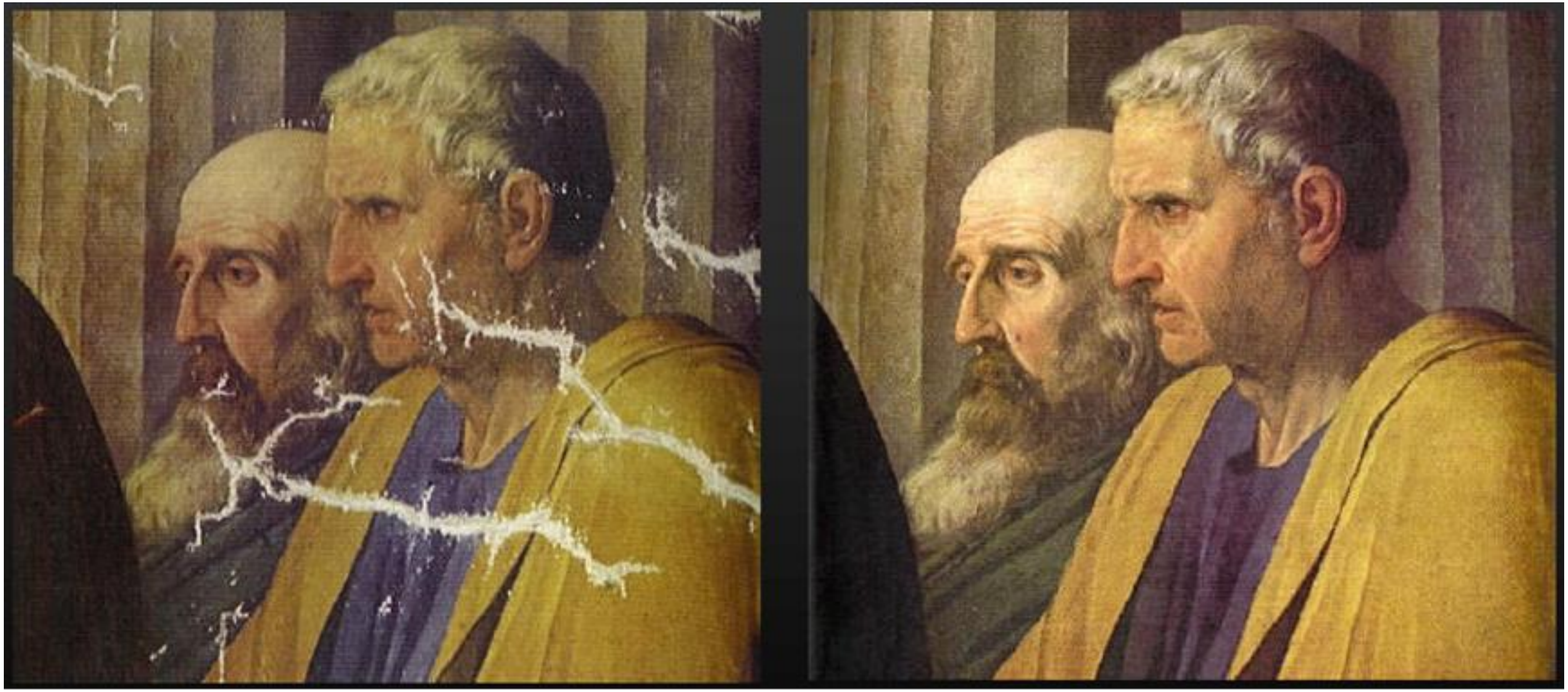
Further Diffusion

After 3 iterations

After 10 iterations

2. Image Inpainting

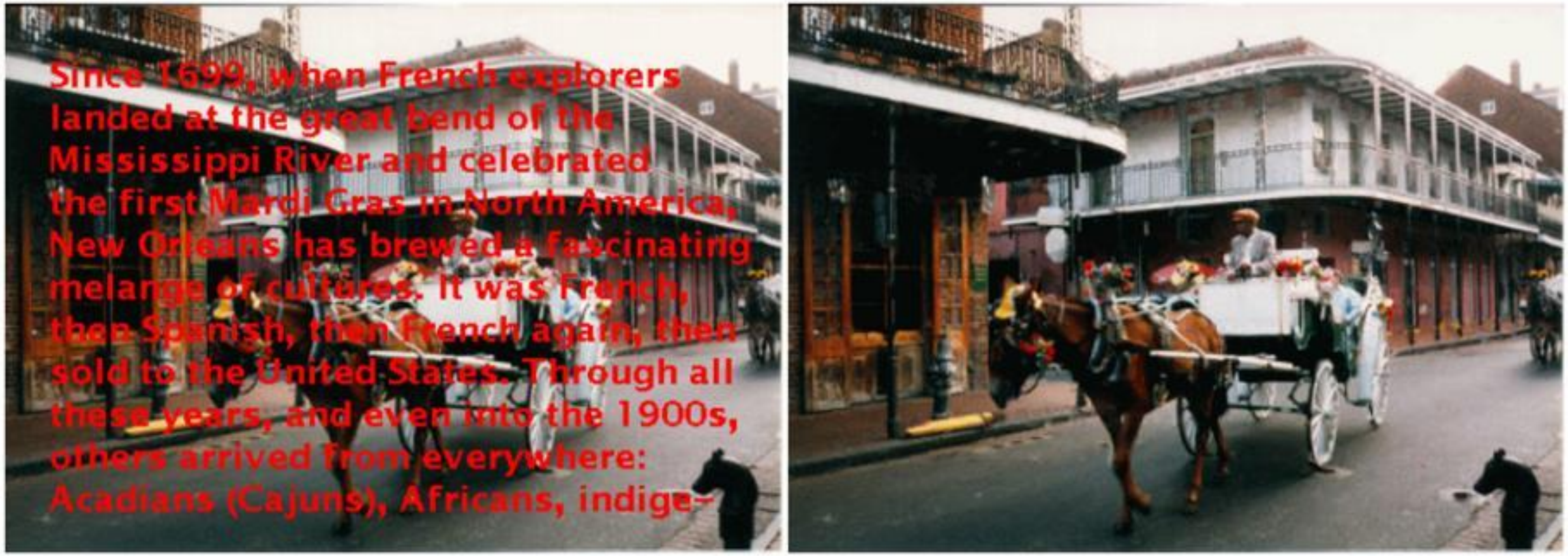
Image Inpainting



"Image Inpainting : An Overview",
Guillermo Sapiro

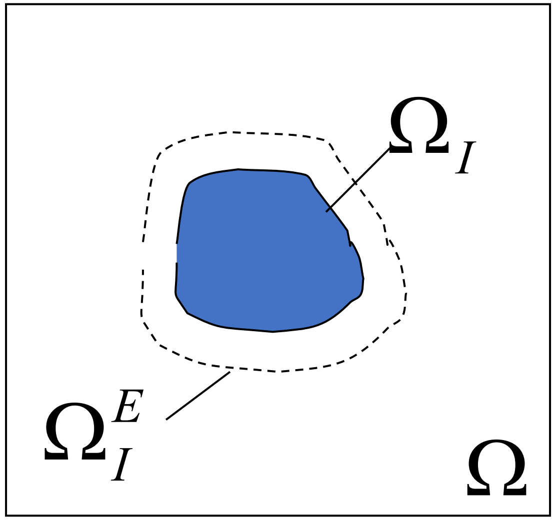
Image Inpainting

"Digital Image Inpainting is an iterative method for repairing damaged pictures or removing unnecessary elements from pictures"



"Fast Digital Image Inpainting",
Manuel M. Oliveira, Brian Bowen, Richard McKenna and Yu-Sung Chang

Image Inpainting: Modeling



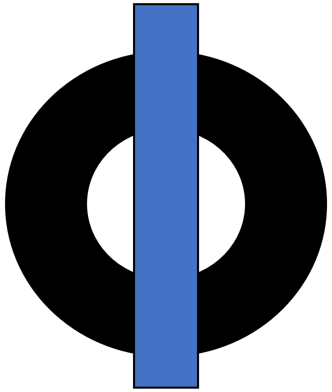


Image example

Ω_I^E Extended inpainting domain

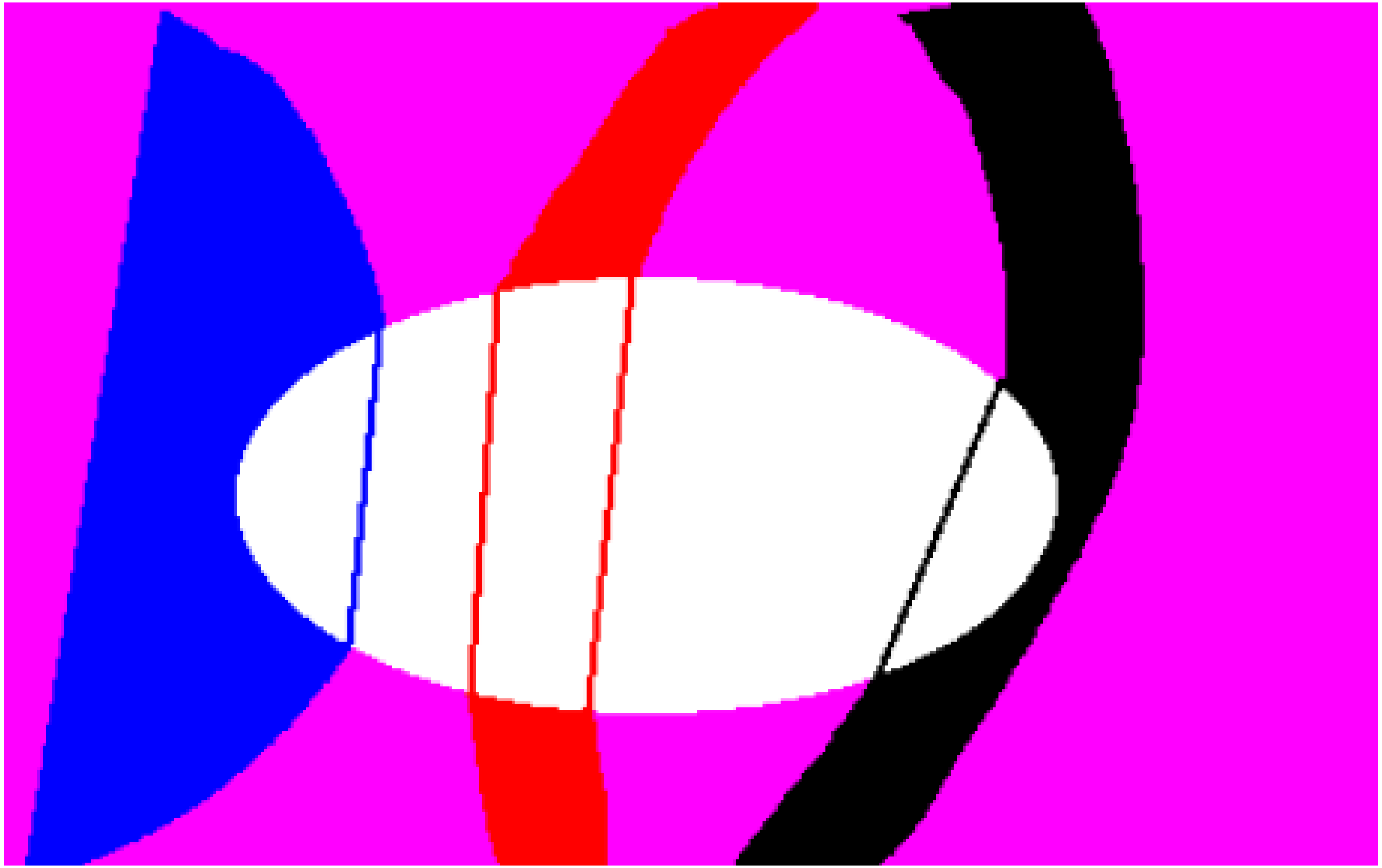
Assumption: inpainting domain is local and does not contain texture
(complimentary to texture-synthesis based inpainting techniques)

2.1 Level Line Based Method

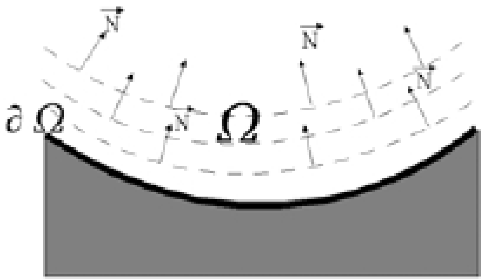
- Joining with geodesic curves the points of the isophotes (lines of equal gray values) arriving at the boundary of the region to be inpainted

- Drawbacks
  - Inpainted region should have simple topology
  - Angle of level lines is not preserved

Example



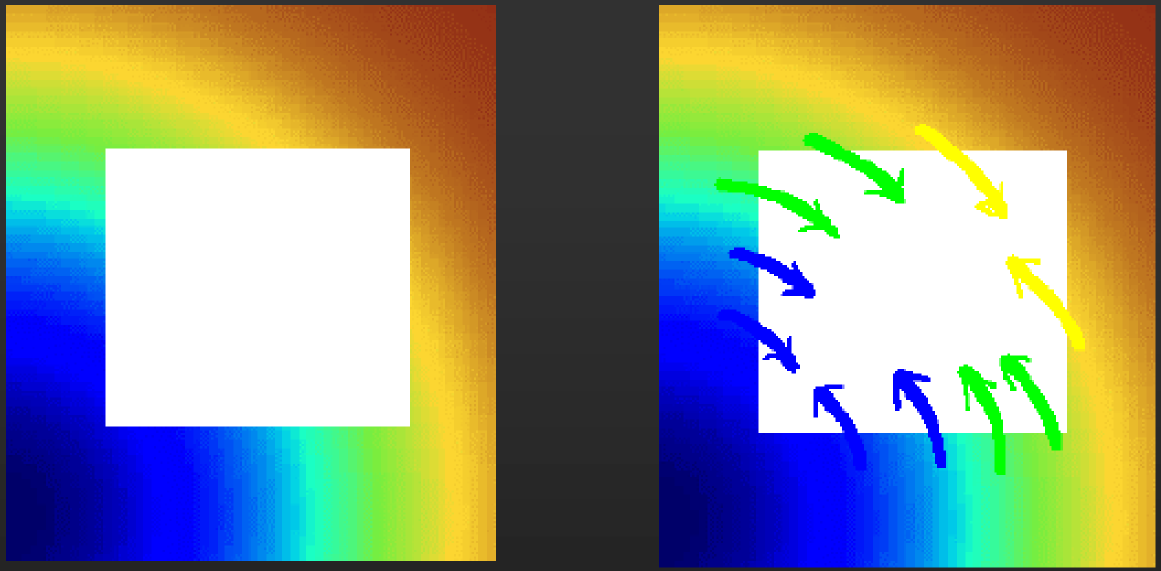
Extension



[Siggraph 2000]

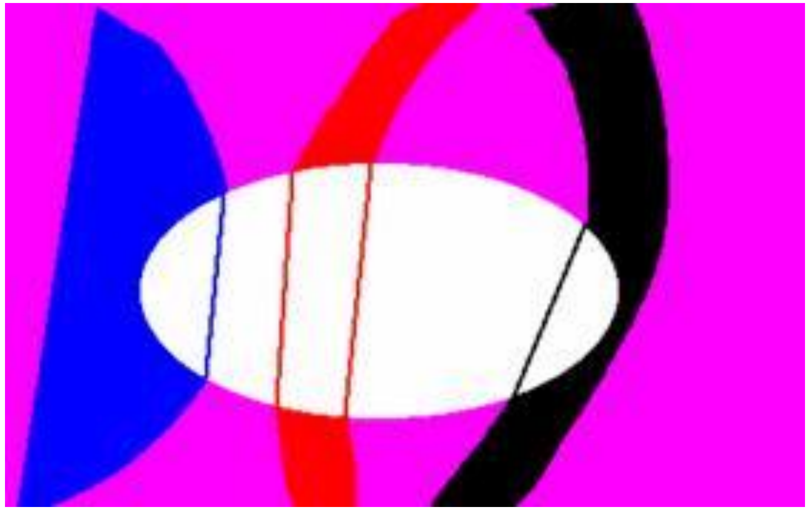
- Apply diffusion to the original image to avoid noise
- Updates to the values of pixels inside the region are made, information propagated in the direction of the isophotes
- After every few iterations, diffusion process is applied
- Propagation of gray values and the isophotes direction is critical
- Color images are considered as a group of 3 images and this technique is applied independently to them

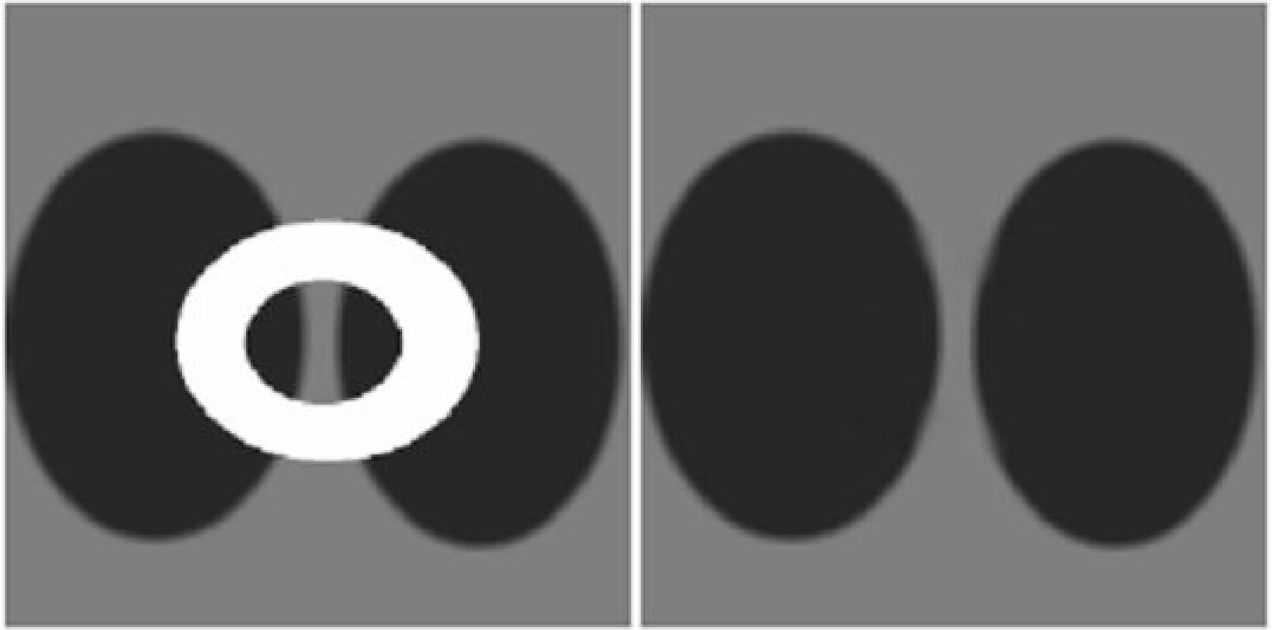
Idea



- **Propagate information** $\nabla L \bullet \vec{N} = 0$
- **Evolutionary form** $\frac{\partial I}{\partial t} = \nabla L \bullet \vec{N}$

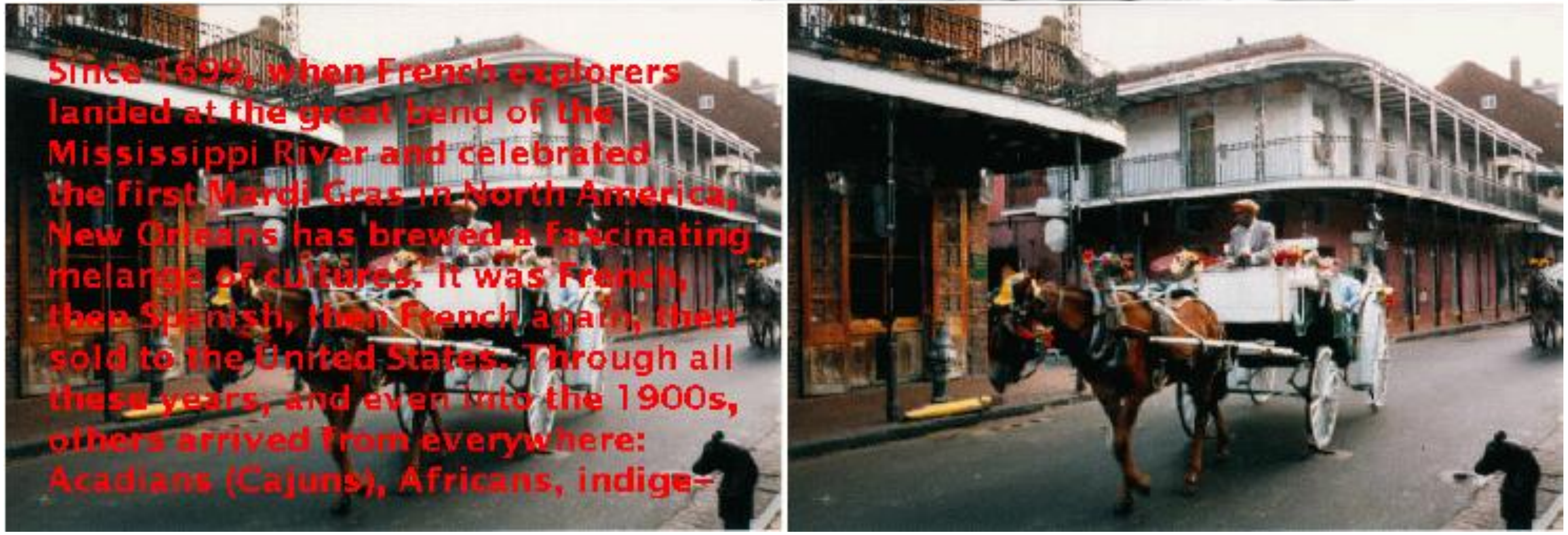
Comparison





Examples

Examples


Since 1699, when French explorers
landed at the great bend of the
Mississippi River and celebrated
the first Mardi Gras in North America,
New Orleans has brewed a fascinating
melange of cultures. It was French,
then Spanish, then French again, then
sold to the United States. Through all
these years, and even into the 1900s,
others arrived from everywhere:
Acadians (Cajuns), Africans, indige-

2.2 Variational Method

$$\min J(\hat{u}) = \int_{\Omega_I^E} |\nabla \hat{u}| dxdy + \frac{\lambda}{2} \int_{\Omega_I^E \backslash \Omega_I} |\hat{u} - u|^2 dxdy$$

↓ Total variation (TV)

$\hat{u}$ Restored image u degraded image

Rational:

- The 1st term describes the smoothness constraint within the extend inpainting domain
- The second term describes the observation constraint

How to obtain the corresponding PDE?

Euler-Lagrangian Equation

$$-\nabla \cdot \left(\frac{\nabla \hat{u}}{|\nabla \hat{u}|} \right) + \lambda_E (\hat{u} - u) = 0$$

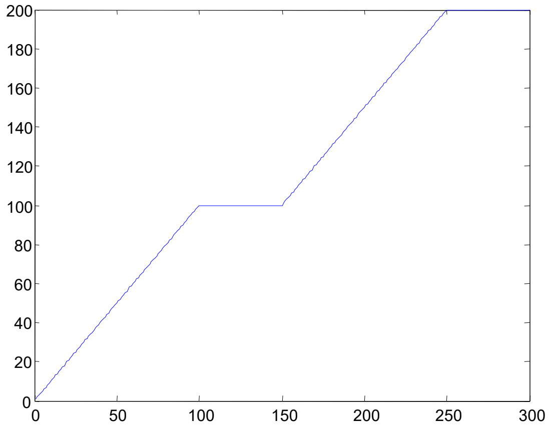
Where $\lambda_E = \begin{cases} \lambda & (x, y) \in \Omega_I^E \backslash \Omega_I \\ 0 & (x, y) \in \Omega_I \end{cases}$

⬇

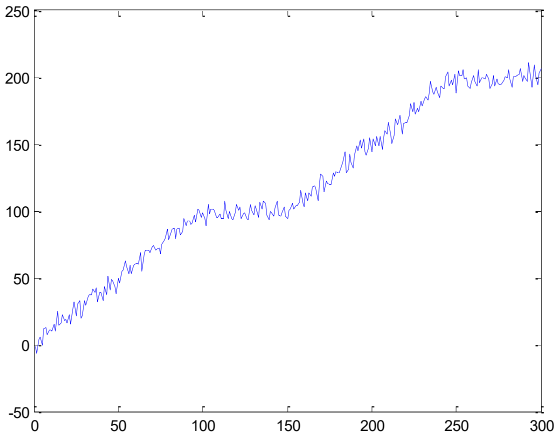
TV-inpainting: $\dfrac{\partial \hat{u}}{\partial t} = \nabla \cdot \left(\frac{\nabla \hat{u}}{|\nabla \hat{u}|} \right) + \lambda_E (u - \hat{u})$

Total Variation (TV)

$$V(f,\Omega) = \int_{\Omega} |\nabla f|$$



Clean (TV small)



noisy (TV large)

Key idea: it is L_1 instead of L_2 norm
(minimizing L_2 will not preserve edges)

Inpainting Example

Image to be inpainted

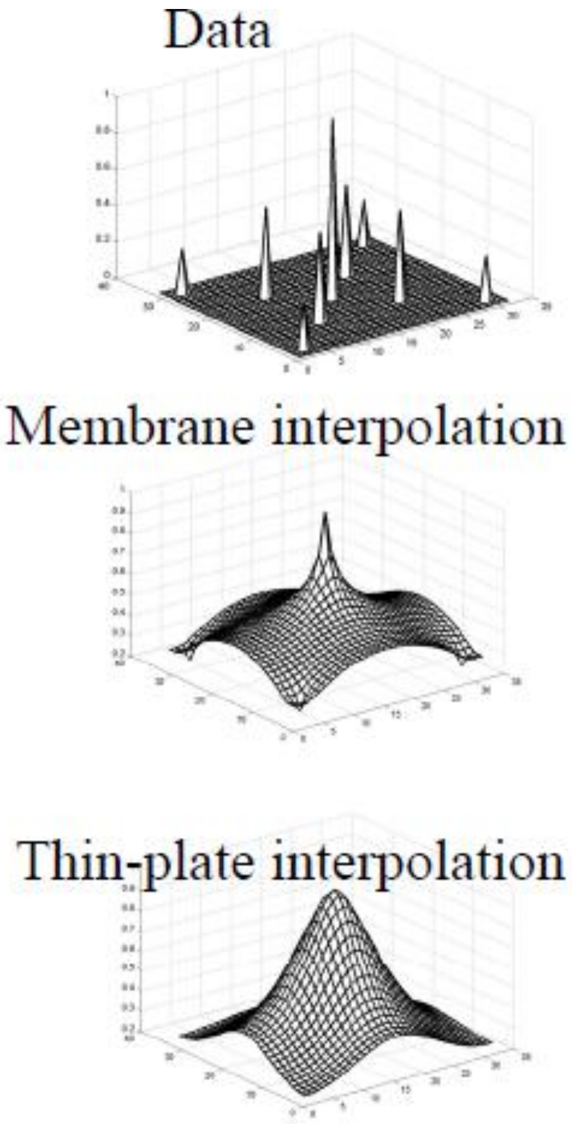
Inpainting domain (or mask)

(Courtesy: Jackie Shen, UMN MATH)

PDE: Thin Plate Energy

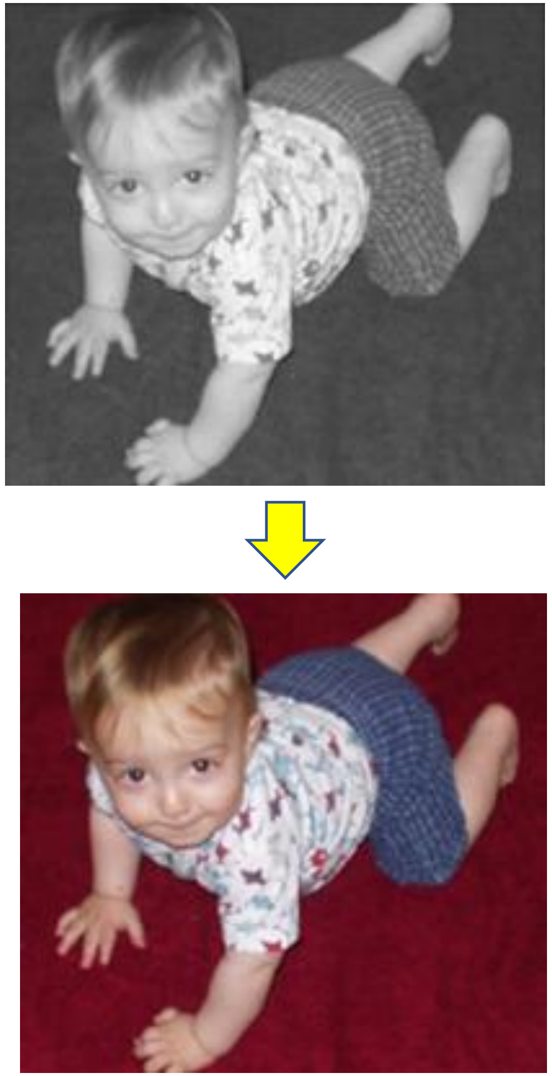
- **Thin plate:**
  minimize *second* derivative

$$\min_f \int\int f_{xx}^2 + 2f_{xy}^2 + f_{yy}^2 \,\mathrm{dxdy}$$



偏微分方程的广泛应用


Since 1699, when French explorers
landed at the great bend of the
Mississippi River and celebrated
the first Mardi Gras in North America,
New Orleans has brewed a fascinating
melange of cultures. It was French,
then Spanish, then French again, then
sold to the United States. Through all
these years, and even into the 1900s,
others arrived from everywhere:
Acadians (Cajuns), Africans, indige-



中国科学技术大学
University of Science and Technology of China

谢 谢 ！