

图像处理与分析

— 图像恢复 (2)

授课教师：孙剑

jiansun@mail.xjtu.edu.cn

<http://jiansun.gr.xjtu.edu.cn>

西安交通大学 数学与统计学院

目录

- 基于正则化能量建模的图像恢复算法
 - 基本思想和方法
 - 实例1：基于稀疏性正则化方法的图像去模糊算法
 - 实例2：基于稀疏性正则化方法的图像去噪算法

- 基于偏微分方程的图像恢复算法
 - 实例1：基于偏微分方程的图像填充算法

基于正则化能量建模的图像恢复

- 图像恢复的正则化能量模型：为解决图像恢复反问题，将图像恢复问题建模为能量函数优化问题，采用正则化技术约束优化问题的解空间。优化问题的解即为恢复得到的清晰图像。

低质量观测图像的生成模型（一般表示形势）：

$$O = AI + B$$

其中 O 为观测图像， I 为要恢复的高质量图像

$$I = \arg \min_I \left\{ E(I | O) = \|AI - O\|_2^2 + \lambda R(I) \right\}$$

正则化项

正则化项：基于图像的先验性性质，例如光滑性等。

基于正则化能量建模的图像恢复

图像去噪问题：



噪声模型： $O = I + n, n \sim N(0, \sigma)$

其中 O 为噪声图像， I 为要恢复的高质量图像

能量模型： $I = \arg \min_{I^*} \left\{ E(I | O) = \|I - O\|_2^2 + R(I) \right\}$

正则化项

基于正则化能量建模的图像恢复

图像去模糊：



模糊模型： $O = k * I + n, n \sim N(0, \sigma)$

其中 O 为观测到的模糊图像， I 为要恢复的高质量图像

能量模型： $I = \arg \min_{I,k} \left\{ E(I|O) = \|k * I - O\|_2^2 + R_1(I) + R_2(k) \right\}$

正则化项



基于正则化能量建模的图像恢复

图像超分辨率问题：



低分辨率模型： $O = (g_\sigma * I) \downarrow$

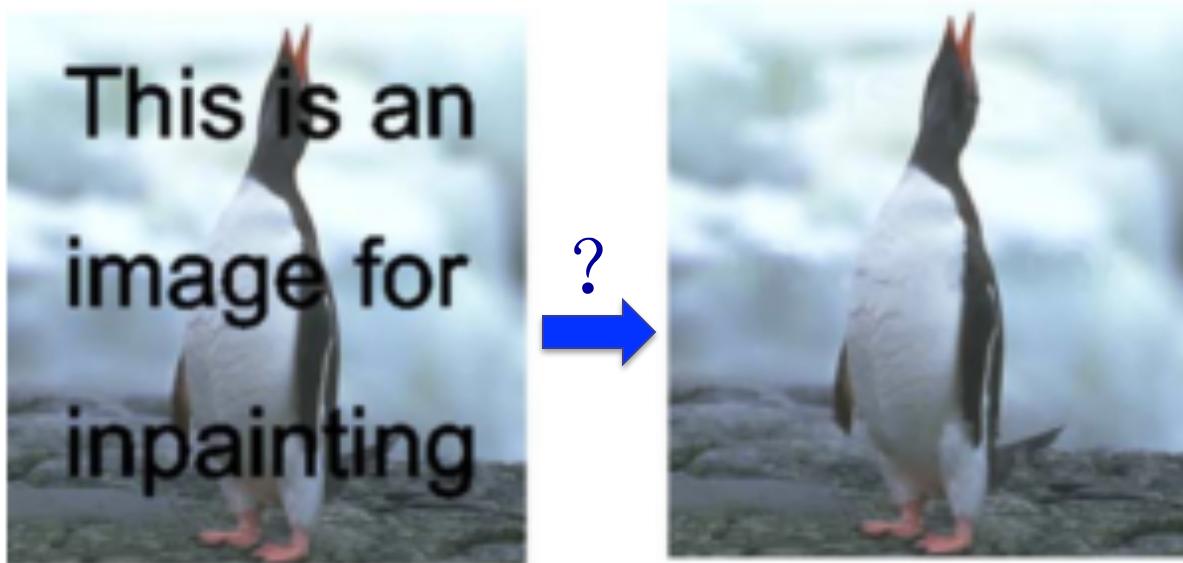
其中 O 为观测到的低分辨率图像， I 为要恢复的高分辨率图像

能量模型：
$$I^* = \arg \min_I \left\{ E(I | O) = \| g_\sigma * I - O \|_2^2 + R(I) \right\}$$

↑ 正则化项
 $R(I)$

基于正则化能量建模的图像恢复

图像填充问题：



$$\text{观测模型: } O(x) = M(x)I(x), \quad M(x) = \begin{cases} 0 & x \text{ 属于缺失区域} \\ 1 & x \text{ 属于已知区域} \end{cases}$$

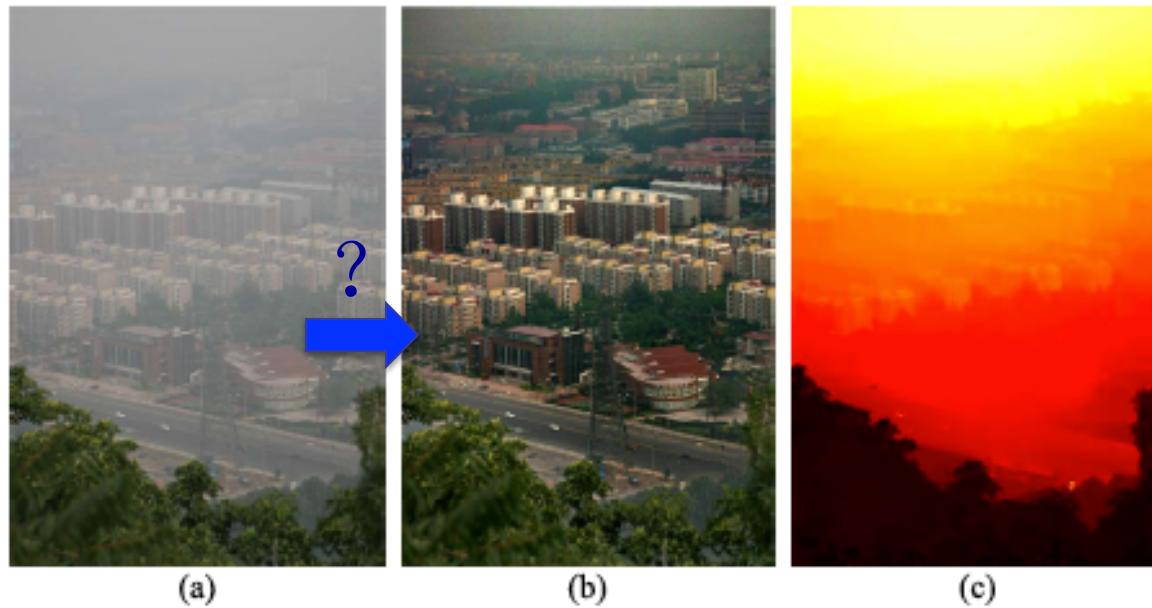
其中 O 为观测到的缺失信息图像, I 为要恢复的高质量图像

$$\text{能量模型: } I^* = \arg \min_I \left\{ E(I | O) = \| M \odot I - O \|_2^2 + \lambda R(I) \right\}$$

正则化项

基于正则化能量建模的图像恢复

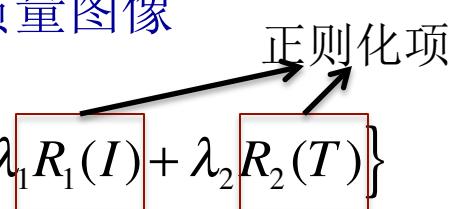
图像去雾问题：



观测模型： $O(x) = T(x)I(x) + (1 - T(x))A$

其中 O 为观测到的缺失信息图像， I 为要恢复的高质量图像

能量模型： $I^* = \arg \min_I \{E(I, t | O) = \|T \odot I + (1 - T)A - O\|_2^2 + \lambda_1 R_1(I) + \lambda_2 R_2(T)\}$



基于正则化能量建模的图像恢复

正则化模型的Bayes统计解释：极小化能量模型相当于极大化Bayes后验概率。

$$P(I|O) = P(O|I)P(I)$$

$$P(O|I) \propto \exp\left(-\frac{\|AI - O\|_2^2}{\sigma^2}\right)$$

极大化

极小化

$$\rightarrow I = \arg \min_I \{-\log(P(I|O)) = -\log(P(O|I)) - \log(P(I))\}$$

正则化项

$$I = \arg \min_I \left\{ E(I|O) = \|AI - O\|_2^2 - \lambda \log(P(I)) \right\}$$

基于正则化能量建模的图像恢复

如何设计正则化项？主要思想是基于图像的先验性

- 光滑性正则化：基于图像灰度具有光滑性 / 分片光滑性。

一阶连续性：

$$R(I) = \|\nabla_x I\|_p^p + \|\nabla_y I\|_p^p$$

$$R(I) = \sum_{(x,y) \in \Omega} \|I(x+1,y) - I(x,y)\|_p^p + \|I(x,y+1) - I(x,y)\|_p^p$$

基于正则化能量建模的图像恢复

- 基于稀疏性的正则化项

$$L_1 \text{ 正则化: } R(I) = |\nabla_x I| + |\nabla_y I|$$

$$R(I) = \sum_{(x,y) \in \Omega} |I(x+1,y) - I(x,y)| + |I(x,y+1) - I(x,y)|$$

L_0 正则化:

$$R(I) = \underbrace{\sum_{(x,y) \in \Omega} |I(x+1,y) - I(x,y)|_0 + |I(x,y+1) - I(x,y)|_0}_{|x|_0 = \delta\{x \neq 0\}}$$

非零元素的个数

- 基于自然图像统计正则化项: 例如Markov随机场模型

目录

- 基于正则化能量建模的图像恢复算法
 - 基本思想和方法
 - 实例1：基于稀疏性正则化方法的图像去模糊算法
 - 实例2：基于稀疏性正则化方法的图像去噪算法

- 基于偏微分方程的图像恢复算法
 - 实例1：基于偏微分方程的图像填充算法

基于稀疏性正则化方法的图像去模糊算法

一种基于稀疏性能量建模的图像去模糊算法

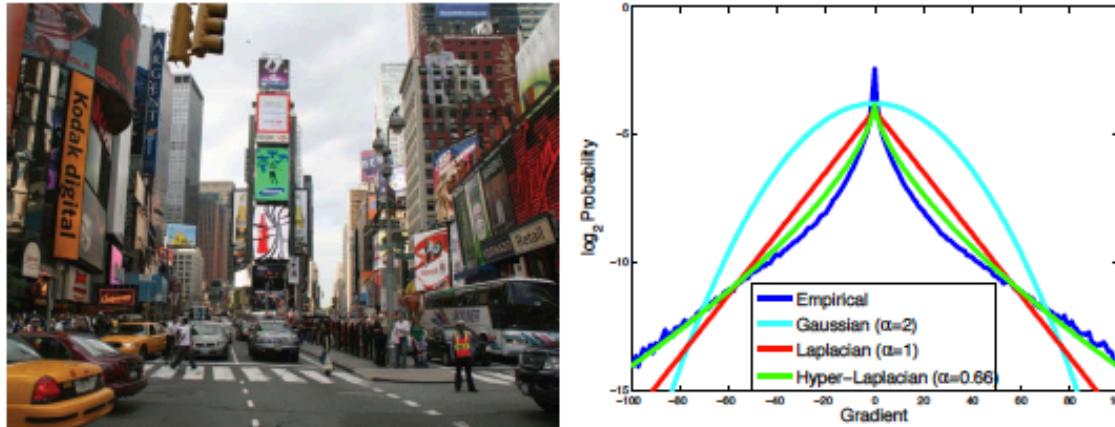


Figure 1: A hyper-Laplacian with exponent $\alpha = 2/3$ is a better model of image gradients than a Laplacian or a Gaussian. **Left:** A typical real-world scene. **Right:** The empirical distribution of gradients in the scene (blue), along with a Gaussian fit (cyan), a Laplacian fit (red) and a hyper-Laplacian with $\alpha = 2/3$ (green). Note that the hyper-Laplacian fits the empirical distribution closely, particularly in the tails.

参考文献： Dilip Krishnan, Rob Fergus. Fast Image Deconvolution using Hyper-Laplacian Priors, Nips, 2009

基于稀疏性正则化方法的图像去模糊算法

图像去模糊的Bayes模型：

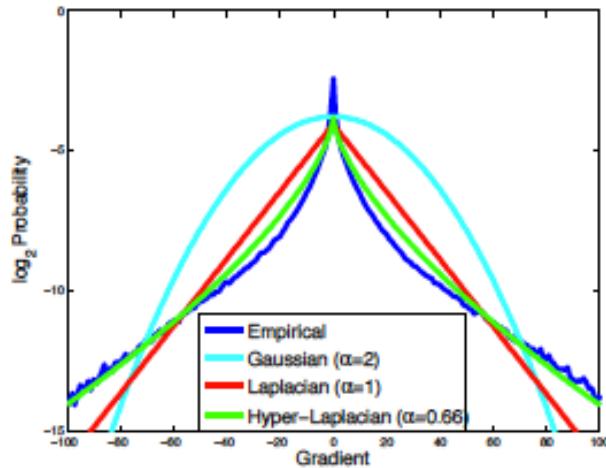
$$P(x | y) = P(y | x)P(x)$$

似然项：

$$P(y | x) \propto \exp\left(-\frac{\|k^*x - y\|_2^2}{\sigma^2}\right)$$

先验项：

$$P(x) \propto \exp(-\|(x \oplus f_j)_i\|^\alpha) \quad \alpha \leq 1$$



基于稀疏性正则化方法的图像去模糊算法

极大化后验概率，转化为极小化能量函数：

$$\min_{\mathbf{x}} \sum_{i=1}^N \left(\frac{\lambda}{2} (\mathbf{x} \oplus \mathbf{k} - \mathbf{y})_i^2 + \boxed{\sum_{j=1}^J |(\mathbf{x} \oplus f_j)_i|^\alpha} \right)$$

$\alpha \leq 1$

稀疏性正则化项

$$f_1 = [1, -1];$$
$$f_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

\mathbf{y} 是观测到的模糊图像； \mathbf{x} 是要估计的高质量图像； \mathbf{k} 为已知的模糊核

基于稀疏性正则化方法的图像去模糊算法

优化算法 (Half-quadratic splitting 算法)

引入辅助变量 w :

$$\min_{\mathbf{x}, \mathbf{w}} \sum_i \left(\frac{\lambda}{2} (\mathbf{x} \oplus \mathbf{k} - \mathbf{y})_i^2 + \frac{\beta}{2} (\|F_i^1 \mathbf{x} - w_i^1\|_2^2 + \|F_i^2 \mathbf{x} - w_i^2\|_2^2) + |w_i^1|^\alpha + |w_i^2|^\alpha \right)$$

当 $\beta \rightarrow \infty$ 时, $F_i^1 x \rightarrow w_i^1; F_i^2 x \rightarrow w_i^2$, 则

$$\min_{\mathbf{x}, \mathbf{w}} \sum_i \left(\frac{\lambda}{2} (\mathbf{x} \oplus \mathbf{k} - \mathbf{y})_i^2 + \frac{\beta}{2} (\|F_i^1 \mathbf{x} - w_i^1\|_2^2 + \|F_i^2 \mathbf{x} - w_i^2\|_2^2) + |w_i^1|^\alpha + |w_i^2|^\alpha \right)$$

→

$$\min_{\mathbf{x}} \sum_{i=1}^N \left(\frac{\lambda}{2} (\mathbf{x} \oplus \mathbf{k} - \mathbf{y})_i^2 + \sum_{j=1}^J |(\mathbf{x} \oplus f_j)_i|^\alpha \right)$$

基于稀疏性正则化方法的图像去模糊算法

优化方法：迭代求解 \mathbf{x}, \mathbf{w} ，迭代中 $\beta \rightarrow \infty$

子问题1：求解 \mathbf{x}

$$\min_{\mathbf{x}, \mathbf{w}} \sum_i \left(\frac{\lambda}{2} (\mathbf{x} \oplus \mathbf{k} - \mathbf{y})_i^2 + \frac{\beta}{2} (\|F_i^1 \mathbf{x} - w_i^1\|_2^2 + \|F_i^2 \mathbf{x} - w_i^2\|_2^2) \right)$$

上述能量函数导数 = 0 

$$\left(F^{1^T} F^1 + F^{2^T} F^2 + \frac{\lambda}{\beta} K^T K \right) \mathbf{x} = F^{1^T} \mathbf{w}^1 + F^{2^T} \mathbf{w}^2 + \frac{\lambda}{\beta} K^T \mathbf{y}$$

左右两侧做FFT: $\left(F^{1^T} F^1 + F^{2^T} F^2 + \frac{\lambda}{\beta} K^T K \right) \mathbf{x} = F^{1^T} \mathbf{w}^1 + F^{2^T} \mathbf{w}^2 + \frac{\lambda}{\beta} K^T \mathbf{y}$

$$\mathbf{x} = \mathcal{F}^{-1} \left(\frac{\mathcal{F}(F^1)^* \circ \mathcal{F}(\mathbf{w}^1) + \mathcal{F}(F^2)^* \circ \mathcal{F}(\mathbf{w}^2) + (\lambda/\beta) \mathcal{F}(K)^* \circ \mathcal{F}(\mathbf{y})}{\mathcal{F}(F^1)^* \circ \mathcal{F}(F^1) + \mathcal{F}(F^2)^* \circ \mathcal{F}(F^2) + (\lambda/\beta) \mathcal{F}(K)^* \circ \mathcal{F}(K)} \right)$$

基于稀疏性正则化方法的图像去模糊算法

子问题2: 求解 \mathbf{w}

$$w^* = \arg \min_w |w|^\alpha + \frac{\beta}{2}(w - v)^2$$

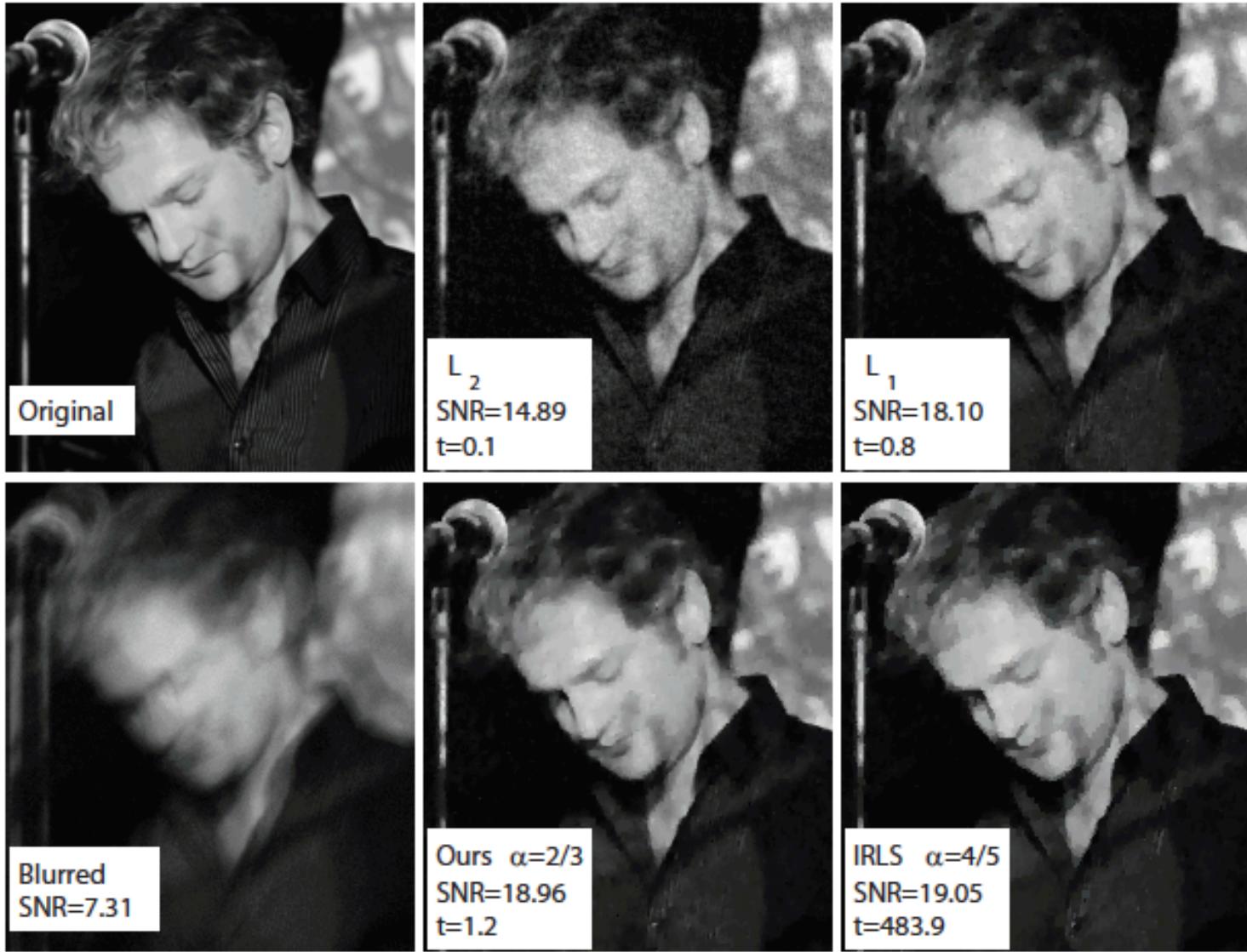
Algorithm 2: Solve Eqn. 5 for $\alpha = 1/2$

Require: Target value v , Weight β
1: $\epsilon = 10^{-6}$
2: {Compute intermediary terms m, t_1, t_2, t_3 }
3: $m = -\text{sign}(v)/4\beta^2$
4: $t_1 = 2v/3$
5: $t_2 = \sqrt[3]{-27m - 2v^3 + 3\sqrt{3}\sqrt{27m^2 + 4mv^3}}$
6: $t_3 = v^2/t_2$
7: {Compute 3 roots, r_1, r_2, r_3 :}
8: $r_1 = t_1 + 1/(3 \cdot 2^{1/3}) \cdot t_2 + 2^{1/3}/3 \cdot t_3$
9: $r_2 = t_1 - (1 - \sqrt{3}i)/(6 \cdot 2^{1/3}) \cdot t_2$
 $- (1 + \sqrt{3}i)/(3 \cdot 2^{2/3}) \cdot t_3$
10: $r_3 = t_1 - (1 + \sqrt{3}i)/(6 \cdot 2^{1/3}) \cdot t_2$
 $- (1 - \sqrt{3}i)/(3 \cdot 2^{2/3}) \cdot t_3$
11: {Pick global minimum from $(0, r_1, r_2, r_3)$ }
12: $r = [r_1, r_2, r_3]$
13: $c_1 = (\text{abs}(\text{imag}(r)) < \epsilon)$ {Root must be real}
14: $c_2 = \text{real}(r)\text{sign}(v) > (2/3 \cdot \text{abs}(v))$
 {Root must obey bound of Eqn. 13}
15: $c_3 = \text{real}(r)\text{sign}(v) < \text{abs}(v)$ {Root $< v$ }
16: $w^* = \max((c_1 \& c_2 \& c_3) \text{real}(r)\text{sign}(v))\text{sign}(v)$
return w^*

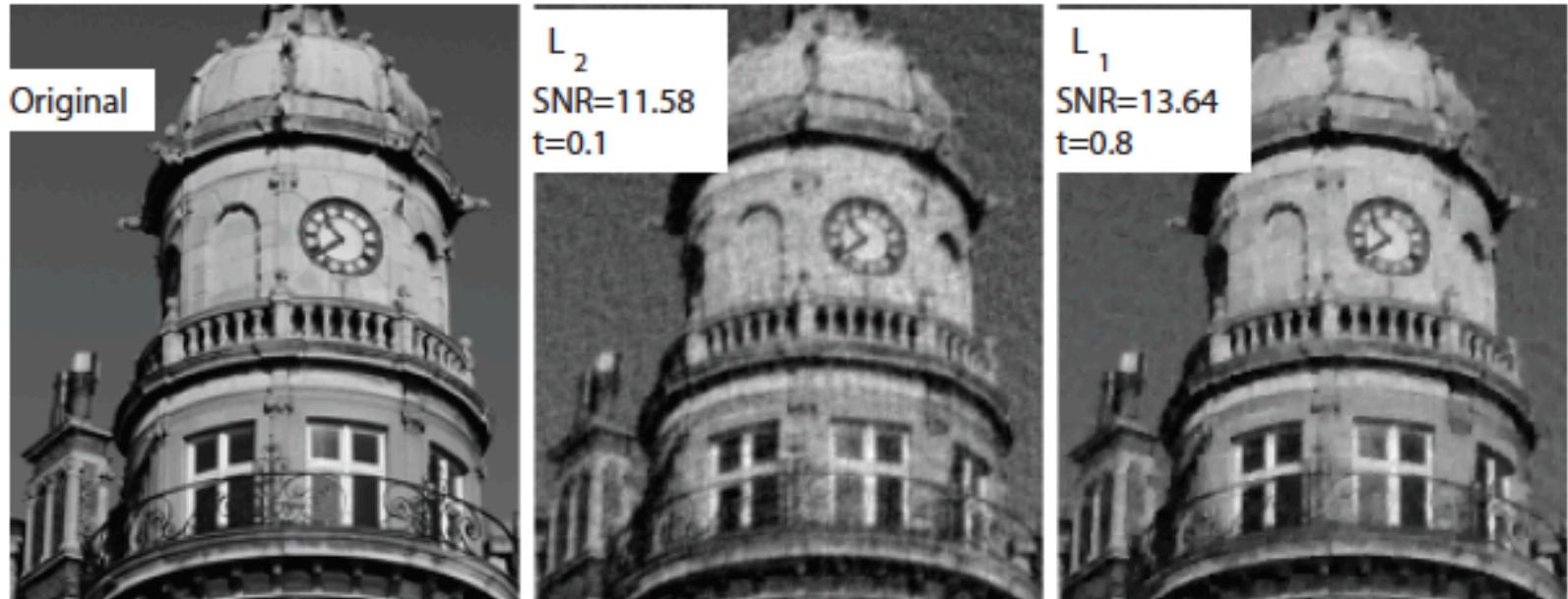
Algorithm 3: Solve Eqn. 5 for $\alpha = 2/3$

Require: Target value v , Weight β
1: $\epsilon = 10^{-6}$
2: {Compute intermediary terms m, t_1, \dots, t_7 :}
3: $m = 8/(27\beta^3)$
4: $t_1 = -9/8 \cdot v^2$
5: $t_2 = v^3/4$
6: $t_3 = -1/8 \cdot mv^2$
7: $t_4 = -t_3/2 + \sqrt{-m^3/27 + m^2v^4/256}$
8: $t_5 = \sqrt[3]{t_4}$
9: $t_6 = 2(-5/18 \cdot t_1 + t_5 + m/(3 \cdot t_5))$
10: $t_7 = \sqrt{t_1/3 + t_6}$
11: {Compute 4 roots, r_1, r_2, r_3, r_4 :}
12: $r_1 = 3v/4 + (t_7 + \sqrt{-(t_1 + t_6 + t_2/t_7)})/2$
13: $r_2 = 3v/4 + (t_7 - \sqrt{-(t_1 + t_6 + t_2/t_7)})/2$
14: $r_3 = 3v/4 + (-t_7 + \sqrt{-(t_1 + t_6 - t_2/t_7)})/2$
15: $r_4 = 3v/4 + (-t_7 - \sqrt{-(t_1 + t_6 - t_2/t_7)})/2$
16: {Pick global minimum from $(0, r_1, r_2, r_3, r_4)$ }
17: $r = [r_1, r_2, r_3, r_4]$
18: $c_1 = (\text{abs}(\text{imag}(r)) < \epsilon)$ {Root must be real}
19: $c_2 = \text{real}(r)\text{sign}(v) > (1/2 \cdot \text{abs}(v))$
 {Root must obey bound in Eqn. 13}
20: $c_3 = \text{real}(r)\text{sign}(v) < \text{abs}(v)$ {Root $< v$ }
21: $w^* = \max((c_1 \& c_2 \& c_3) \text{real}(r)\text{sign}(v))\text{sign}(v)$
return w^*

基于稀疏性正则化方法的图像去模糊算法



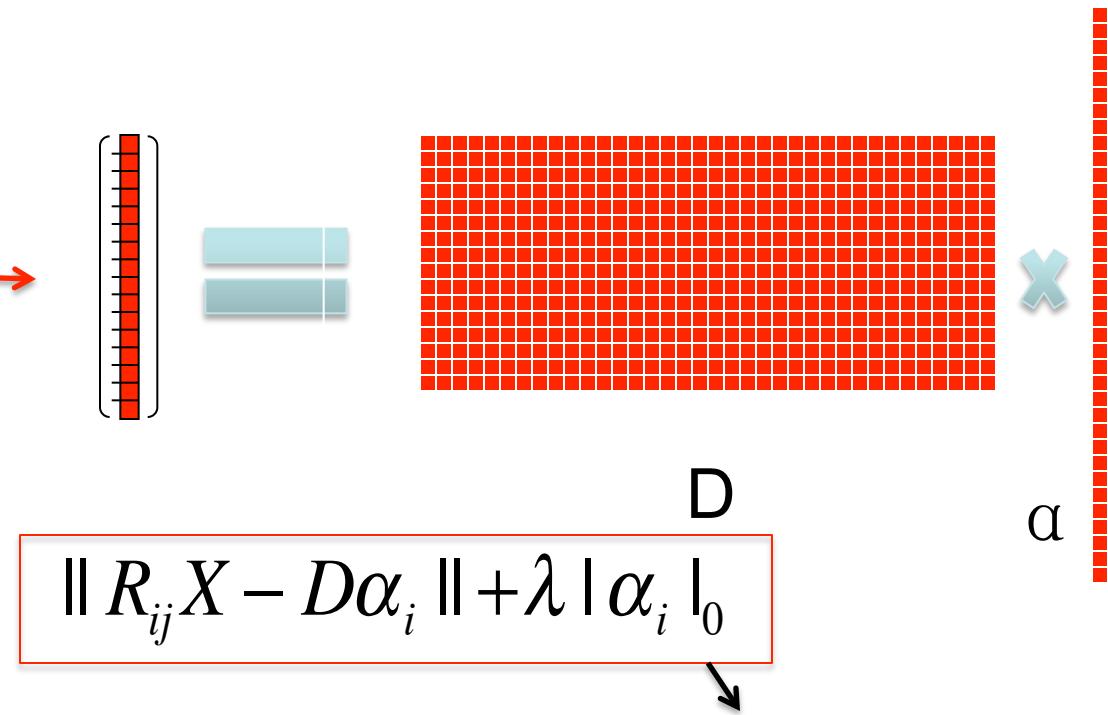
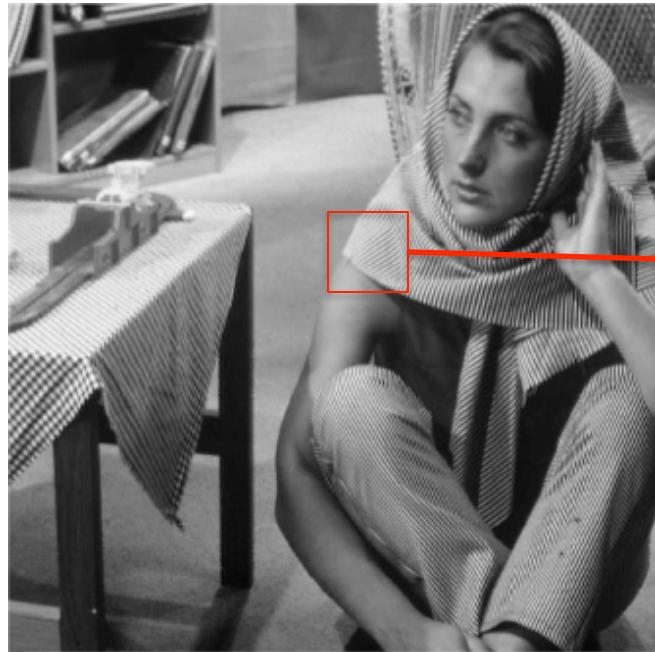
基于稀疏性正则化方法的图像去模糊算法



目录

- 基于正则化能量建模的图像恢复算法
 - 基本思想和方法
 - 实例1：基于稀疏性正则化方法的图像去模糊算法
 - 实例2：基于稀疏性正则化方法的图像去噪算法
- 基于偏微分方程的图像恢复算法
 - 实例1：基于偏微分方程的图像填充算法

基于稀疏性正则化方法的图像去噪算法



约束图像块是字典D的稀疏线性组合

$R_{ij}X$:图像 X 中像素 (i,j) 处的图像块, R_{ij} 是提取该图像块的矩阵。

Michael Elad and Michal Aharon, Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries, IEEE TIP 2006

基于稀疏性正则化方法的图像去噪算法

图像去噪的能量模型--1:

$$\begin{aligned}\{\hat{\alpha}_{ij}, \hat{\mathbf{X}}\} = \arg \min_{\alpha_{ij}, \mathbf{X}} & \lambda \|\mathbf{X} - \mathbf{Y}\|_2^2 \\ & + \sum_{ij} \mu_{ij} \|\alpha_{ij}\|_0 + \sum_{ij} \|\mathbf{D}\alpha_{ij} - \mathbf{R}_{ij}\mathbf{X}\|_2^2.\end{aligned}$$

模型的优化:

子问题1: $\hat{\alpha}_{ij} = \arg \min_{\alpha} \mu_{ij} \|\alpha\|_0 + \|\mathbf{D}\alpha - \mathbf{x}_{ij}\|_2^2$

可采用orthonormal matching pursuit算法求解；也可采用更高效的ADMM算法求解。

基于稀疏性正则化方法的图像去噪算法

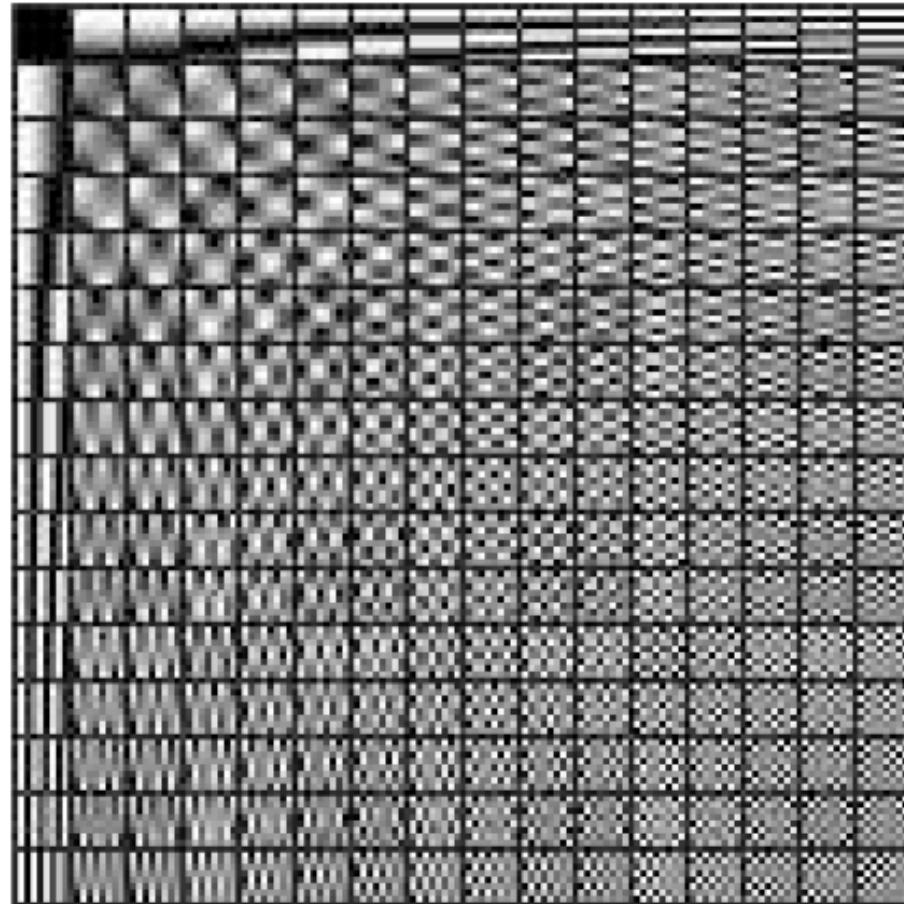
子问题2: $\hat{\mathbf{X}} = \arg \min_{\mathbf{x}} \lambda \|\mathbf{X} - \mathbf{Y}\|_2^2 + \sum_{ij} \|\mathbf{D}\hat{\alpha}_{ij} - \mathbf{R}_{ij}\mathbf{X}\|_2^2.$

有close-form解:

$$\hat{\mathbf{X}} = \left(\lambda \mathbf{I} + \sum_{ij} \mathbf{R}_{ij}^T \mathbf{R}_{ij} \right)^{-1} \left(\lambda \mathbf{Y} + \sum_{ij} \mathbf{R}_{ij}^T \mathbf{D}\hat{\alpha}_{ij} \right)$$

基于稀疏性正则化方法的图像去噪算法

字典D的设置：
采用DCT变换作为基元，构成字典



基于稀疏性正则化方法的图像去噪算法

图像去噪的能量模型--2:

$$\begin{aligned}\{\hat{\mathbf{D}}, \hat{\boldsymbol{\alpha}}_{ij}, \hat{\mathbf{X}}\} = & \arg \min_{\hat{\mathbf{D}}, \hat{\boldsymbol{\alpha}}_{ij}, \hat{\mathbf{X}}} \lambda \|\mathbf{X} - \mathbf{Y}\|_2^2 \\ & + \sum_{ij} \mu_{ij} \|\boldsymbol{\alpha}_{ij}\|_0 + \sum_{ij} \|\mathbf{D}\boldsymbol{\alpha}_{ij} - \mathbf{R}_{ij}\mathbf{X}\|_2^2.\end{aligned}$$

子问题1: $\hat{\boldsymbol{\alpha}}_{ij} = \arg \min_{\boldsymbol{\alpha}} \mu_{ij} \|\boldsymbol{\alpha}\|_0 + \|\mathbf{D}\boldsymbol{\alpha} - \mathbf{x}_{ij}\|_2^2$

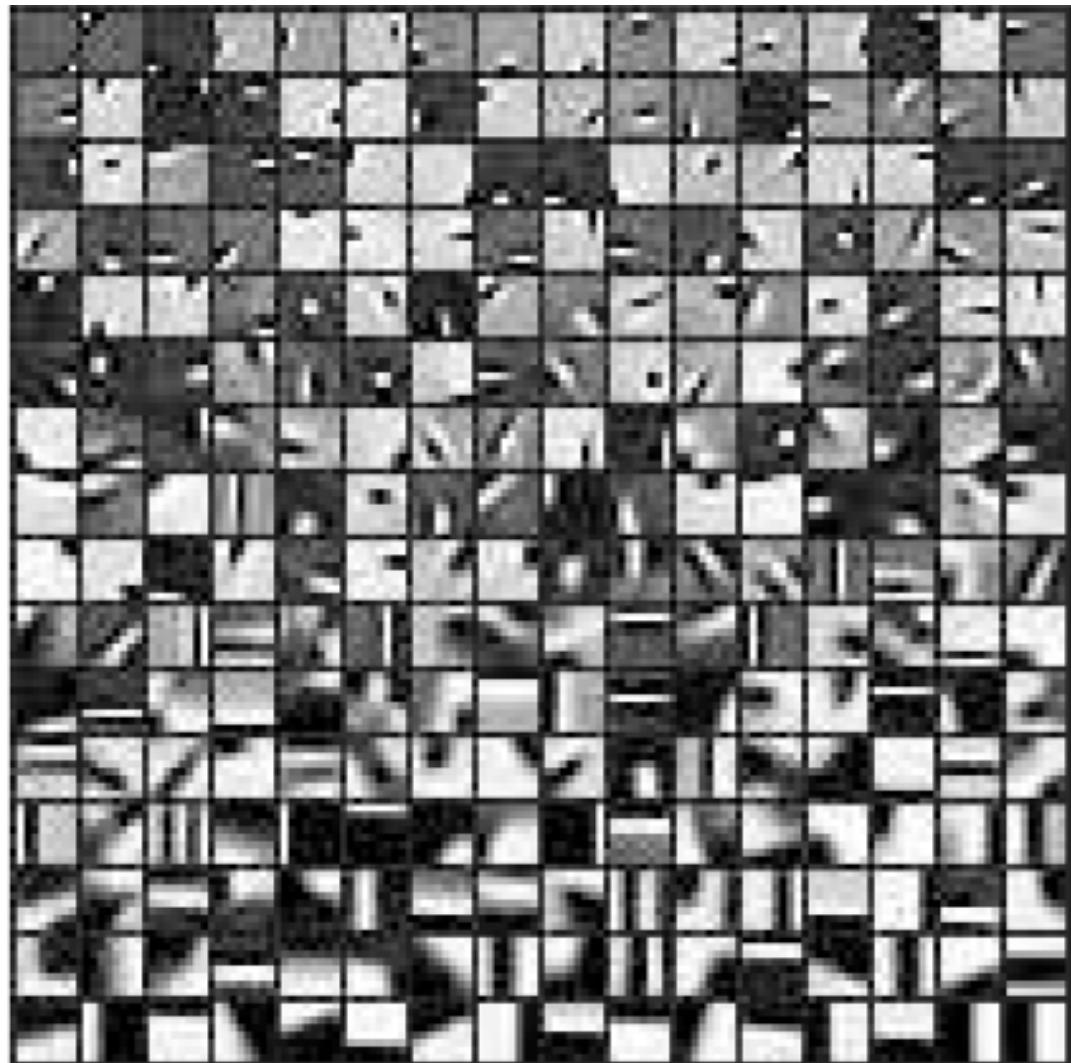
子问题2: $\hat{\mathbf{X}} = \arg \min_{\mathbf{x}} \lambda \|\mathbf{X} - \mathbf{Y}\|_2^2 + \sum_{ij} \|\mathbf{D}\hat{\boldsymbol{\alpha}}_{ij} - \mathbf{R}_{ij}\mathbf{X}\|_2^2.$

子问题3: $D = \arg \min_D \|\mathbf{D}\boldsymbol{\alpha}_{ij} - \mathbf{R}_{ij}\mathbf{X}\|^2$

可采用KSVD算法进行求解。

基于稀疏性正则化方法的图像去噪算法

算法自动学习到的
字典D:







基于稀疏性正则化方法的图像去噪算法

- 字典学习算法思想总结：

将图像块表达为一个字典的稀疏线性组合形式；字典是由图像块的表达基元组成的集合。字典学习就是自动的从图像块集合中学习到字典，并用字典进行图像块的表达。

字典学习算法被广泛应用于图像恢复、图像识别等问题，是图像处理和模式识别的一种重要的学习算法。

目录

- 基于正则化能量建模的图像恢复算法
 - 基本思想和方法
 - 实例1：基于稀疏性正则化方法的图像去模糊算法
 - 实例2：基于稀疏性正则化方法的图像去噪算法

- 基于偏微分方程的图像恢复算法
 - 实例1：基于偏微分方程的图像填充算法

基于偏微分方程的图像恢复算法

- 偏微分方程图像处理方法: PDE是图像处理的一种重要方法, 主要思想是将图像处理过程建模一个偏微分方程演化过程。偏微分方程的解即为图像处理的结果。

[参考文献]: Gilles Aubert, Pierre Kornprobst. Mathematical Problems in Image Processing, Springer

基于偏微分方程的图像恢复算法

热传导方程：

$$\partial_t u(x, y, t) = \nabla^2 u(x, y, t)$$

$$u(x, y, 0) = I(x, y)$$

方程的解

高斯滤波

$$u(x, y, t) = \underbrace{G(x, y, t)}_{\text{高斯滤波器}} * I(x, y)$$

高斯滤波器

$$G_\sigma(x, y) = C\sigma^{-1/2} \exp\left(-\frac{(x^2 + y^2)}{4\sigma}\right)$$

缺点：各向同性扩散方程。在各个方向上同等扩散，滤波的同时破坏图像内容，即图像边缘。

基于偏微分方程的图像恢复算法

非线性偏微分方程（各向异性偏微分方程）：

PM模型 (Perona和Malik)

热传导方程 (线性)



PM 扩散方程 (非线性)

$$\begin{cases} \partial_t u = \nabla \cdot (c(|\nabla u|) \nabla u), & u(x, y, 0) = I(x, y) \\ c(|\nabla u|) = \frac{1}{1 + |\nabla u|^2 / k^2} \end{cases}$$

扩散系数 $c(|\nabla u|)$

- 作用： $c(|\nabla u|)$ 可以控制扩散速率，使得图像的边缘处扩散较慢。
- 缺点：对孤立噪声敏感。

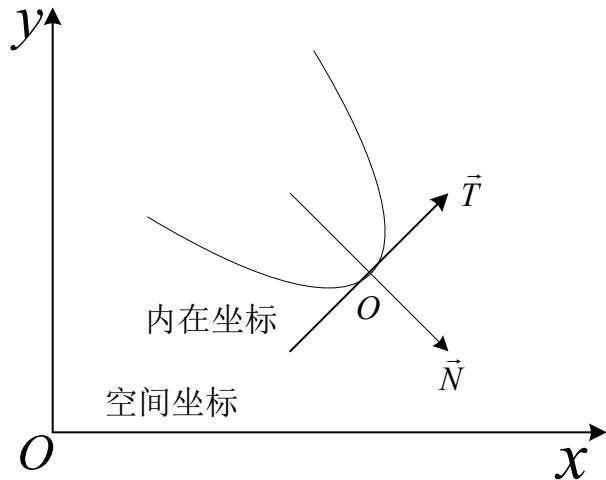
基于偏微分方程的图像恢复算法

平均曲率流各向异性扩散方程：

Alvarez “平均曲率流”各向异性扩散模型：

$$\partial_t u = |\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right), \quad u(x, y, 0) = I(x, y)$$

优点：在垂直于边缘的方向上限制了方程的扩散，滤波同时保护图像的边缘。



空间坐标与基于图像特征的内在坐标示意图

$$N = \frac{\nabla u}{|\nabla u|} = \frac{(u_x, u_y)}{\sqrt{u_x^2 + u_y^2}} \quad T = \frac{\nabla^\perp u}{|\nabla u|} = \frac{(-u_y, u_x)}{\sqrt{u_x^2 + u_y^2}}$$

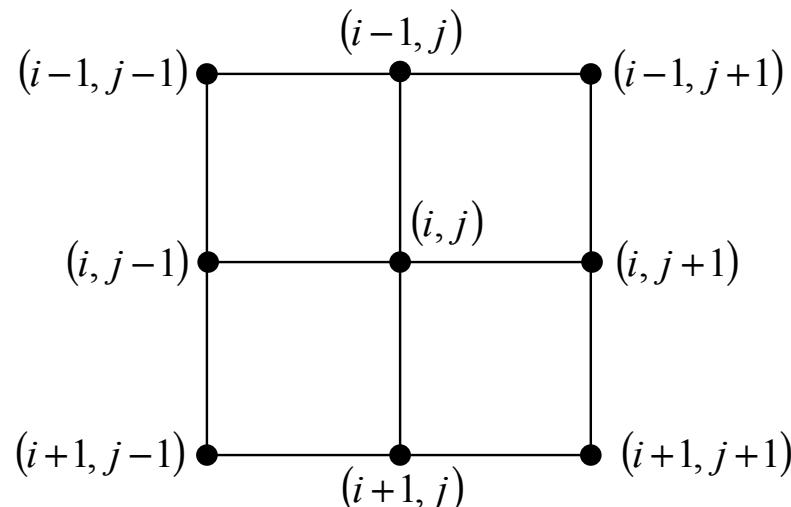
基于偏微分方程的图像恢复算法

偏微分方程的数值解法

——对连续的微分方程进行离散差分求解

假设要处理的图像是一幅 $M \times N$ 的灰度图像 I ，

$I_{i,j}$ 代表 $1 \leq i \leq M$, $1 \leq j \leq N$ 的任意一点。



目标像素 (i, j) 与它的邻域

基于偏微分方程的图像恢复算法

采用的空间差分为：

$$\Delta_{xx} u_{i,j}^n = u_{i+1,j}^n + u_{i-1,j}^n - 2u_{i,j}^n$$

$$\Delta_{yy} u_{i,j}^n = u_{i,j+1}^n + u_{i,j-1}^n - 2u_{i,j}^n$$

边界条件

$$u_{i,0}^n = u_{i,1}^n, \quad u_{i,N+1}^n = u_{i,N}^n, \quad i = 1, 2, \dots, M$$

$$u_{0,j}^n = u_{1,j}^n, \quad u_{M+1,j}^n = u_{M,j}^n, \quad j = 1, 2, \dots, N$$

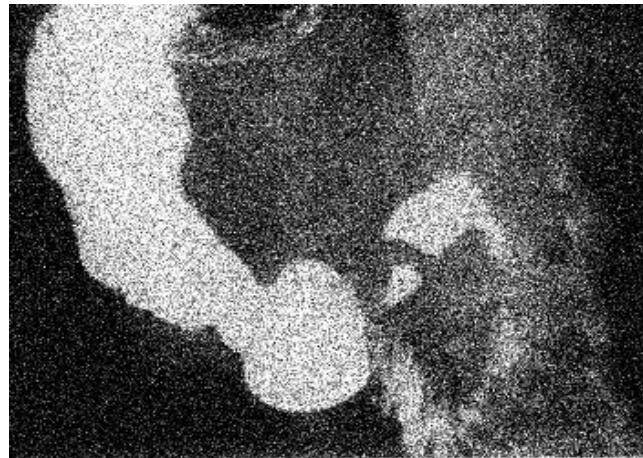
初值条件

$$u_{i,j}^0 = I_{i,j}$$

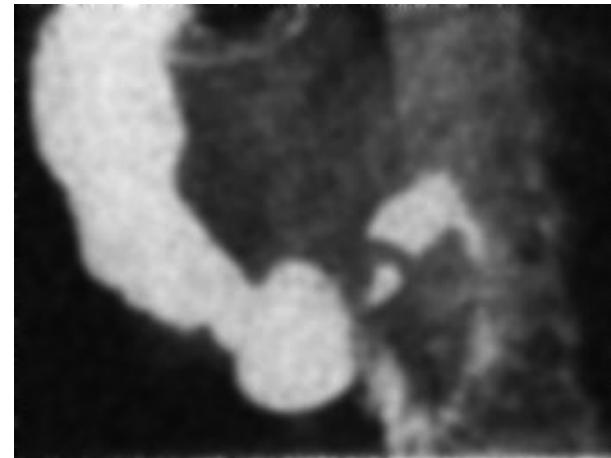
由此，可得热传导扩散方程 $\partial_t u = \nabla^2 u$ 的差分格式为：

$$u_{i,j}^{n+1} = u_{i,j}^n + \Delta t \left(u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n - 4u_{i,j}^n \right)$$

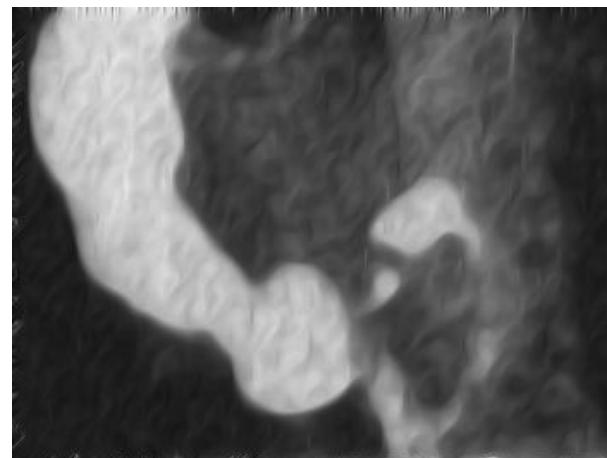
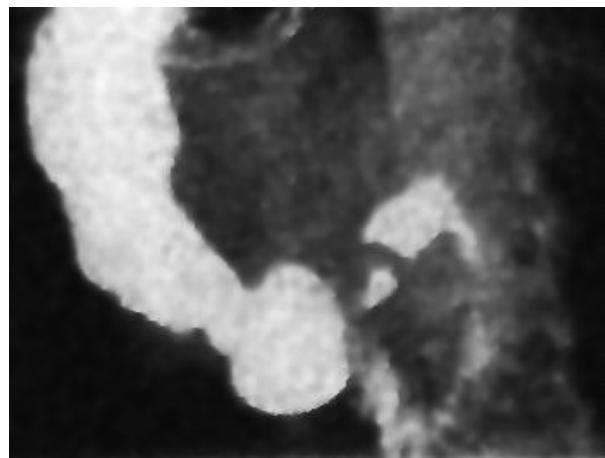
基于偏微分方程的图像恢复算法



原始图像



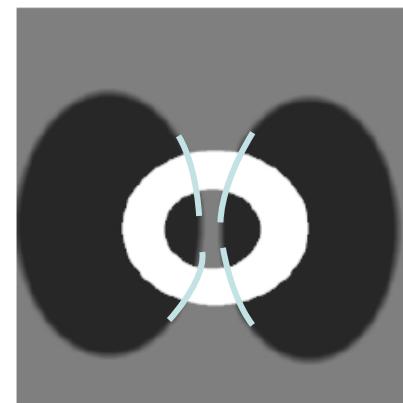
各向同性(线性)扩散滤波(热传导方程)



基于偏微分方程的图像填充算法

● 问题研究背景

- 老旧艺术品修复、电影修复
- 艺术家采用的修复方法：
 - **步骤1：**整体观察图像，由已知信息明确需要填充的内容、如何去填充；
 - **步骤2：**将待填充区域外的结构(即边缘)光滑延拓到待填充区域内部；
 - **步骤3：**将待填充区域内由结构轮廓线所围成的区域依据所连通的外部已知区域分别填充颜色；
 - **步骤4：**填充细节纹理。



基于偏微分方程的图像填充算法

● 基于扩散的方法

- 基本想法：以像素点为单位，将待填充区域边缘的图像信息扩散到填充区域内部；
- 优缺点：可以较好地填充较小的缺失区域（例如划痕移除），但难以填充大范围的缺失区域（例如物体移除）。



扩散填充过程（随着迭代次数增大）

基于偏微分方程的图像填充算法

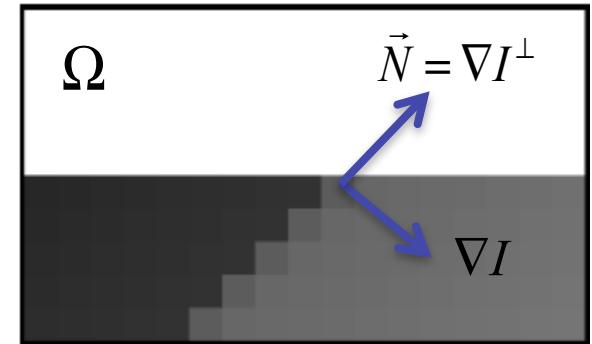
- [M. Bertalmio, image inpainting, Siggraph 2000]

- 基本思想：采用偏微分方程(PDE)的差分迭代格式进行图像信息扩散。
- 方法：沿着等灰度方向扩散图像信息，目的是平滑地将结构（即边缘）由已知区域延拓至未知区域。

迭代扩散格式：

$$I^{n+1}(i, j) = I^n(i, j) + \Delta t I_t^n(i, j), \forall (i, j) \in \Omega$$

Isophote(等灰度)方向： $\vec{N} = \nabla I^\perp$



$$I_t^n(i, j) = \nabla L^n(i, j) \cdot \vec{N}^n(i, j)$$

L 沿着等灰度方向的方向导数

$$L^n(i, j) = I_{xx}^n(i, j) + I_{yy}^n(i, j)$$

迭代收敛



$I_t^n(i, j) = 0$ 即 L 沿着等灰度方向平滑。

基于偏微分方程的图像填充算法

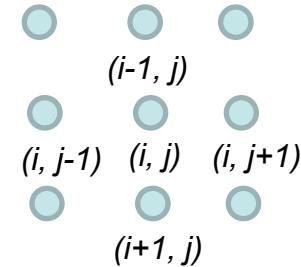
● 算法与实现

- 偏微分方程的差分实现

$$\frac{\vec{N}(i, j, n)}{|\vec{N}(i, j, n)|} := \frac{(-I_y^n(i, j), I_x^n(i, j))}{\sqrt{(I_x^n(i, j))^2 + (I_y^n(i, j))^2}},$$

$$\overrightarrow{\delta L}(i, j) := (L^n(i+1, j) - L^n(i-1, j), L^n(i, j+1) - L^n(i, j-1))$$

```
im_dx = (im([2:end,end],:) - im([1,1:end-1,:]))/2;  
im_dy = (im(:,[2:end,end]) - im(:,[1,1:end-1]))/2;
```

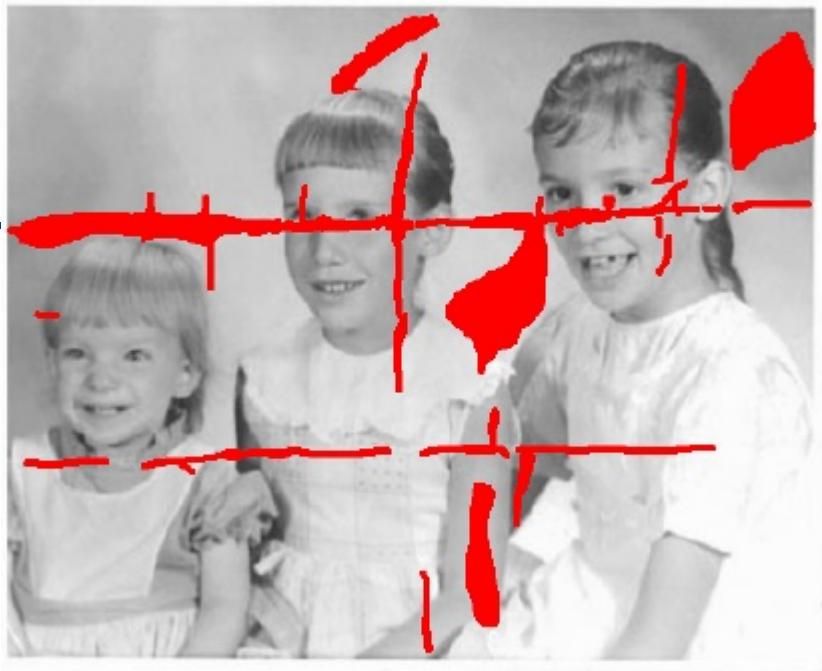


- 扩散迭代

$$I^{n+1}(i, j) = I^n(i, j) + \Delta t I_t^n(i, j), \forall (i, j) \in \Omega$$

- 以初始待填充图像为初始图像





作业

- 选择课堂讲解的三方面内容中的至少一个方面，阅读相应的论文。三方面内容的文献如下：

- [1] Dilip Krishnan, Rob Fergus. Fast Image Deconvolution using Hyper-Laplacian Priors, Nips, 2009
- [2] Michael Elad and Michal Aharon, Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries, IEEE TIP 2006
- [3] M. Bertalmio, image inpainting, Siggraph 2000