Introduction to Nonlinear Image Processing

IPAM Summer School on Computer Vision
July 22, 2013

Iasonas Kokkinos
Center for Visual Computing
Ecole Centrale Paris / INRIA Saclay
Mean and median

Observations in a 3x3 window: [1, 2, 100, 1, 3, 2, 1, 5, 3]

Mean:
\[
\frac{(1 + 2 + 100 + 1 + 3 + 2 + 1 + 5 + 3)}{9} = 13.1
\]

Median:
1) Sort: [1, 1, 1, 2, 2, 3, 3, 5, 100]
2) Pick mid-point

robust to outliers
non-linear
Mean
Introduction to nonlinear image processing

Median
Gaussian blur
Image Processing

\[ f \rightarrow \psi \rightarrow \psi(f) \]

Previous lecture: \[ \psi(\alpha f + \beta h) = \alpha \psi(f) + \beta \psi(h) \]

This lecture: remove this constraint

freedom! at the expense of control
Multi-scale Gaussian smoothing

\( g_\sigma(x, y) = \frac{1}{2\pi \sigma^2} \exp \left( -\frac{x^2 + y^2}{2\sigma^2} \right) \)

\( \sigma = 5 \)  \( \sigma = 10 \)  \( \sigma = 20 \)
Gaussian scale space

Rewrite Gaussian: \( g_t(x, y) = \frac{1}{4\pi t} \exp \left( -\frac{x^2 + y^2}{4t} \right) \) \( t = \frac{\sigma^2}{2} \)

Scale space: \( u(x, y, t) = g_t(x, y) * u_0(x, y), \quad t > 0 \)

\( u(x, y, 0) = u_0(x, y) \)

A. Witkin, Scale-space filtering, IJCAI, 1983.
J. Koenderink, The structure of images, Biological Cybernetics, 1984
T. Lindeberg, Scale-Space Theory in Computer Vision, Kluwer, 1994
J. Weickert, Linear scale space has first been proposed in Japan. JMiV, 1999.
Heat diffusion and image processing

Gaussian satisfies:
\[ \frac{\partial g}{\partial t} = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \]

Scale-space satisfies:
\[ u(x, y, t) = g_t(x, y) \ast u_0(x, y) \]

Associative property:
\[ f \ast [g \ast h] = [f \ast g] \ast h \]

Scale-space satisfies:
\[ \frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \]
\[ u(x, y, 0) = u_0(x, y) \]

Heat diffusion PDE (Partial Differential Equation)
Heat diffusion and image processing
Heat diffusion in 1D

\[ u_t = u_{xx} \]
Heat diffusion in 1D - inhomogeneous material

\[ u_t = \frac{d}{dx} \left( c \frac{d}{dx} u \right) \]
Heat diffusion in 2D

Homogeneous material

\[ u_x \rightarrow \nabla u = (u_x, u_y) \]

\[ \frac{d}{dx} u_x \rightarrow \text{div}(\nabla u) = \frac{\partial}{\partial x} u_x + \frac{\partial}{\partial y} u_y \]

\[ u_t = u_{xx} + u_{yy} \]

Inhomogeneous material

\[ \frac{d}{dx} (cu_x) \rightarrow \text{div}(c \nabla u) \]

\[ u_t = \text{div}(c \nabla u) \]
Perona-Malik Diffusion

Image-dependent conductivity

\[ u_t = \text{div} \left( g \left( |\nabla u| \right) \nabla u \right) \quad u(x, y, 0) = u_0(x, y) \]

\[ g(s) = \exp \left( -\frac{s^2}{a^2} \right) \]

Diffusion stops at strong image gradients (structure-preserving)

CLMC formulation: \( |\nabla u| \rightarrow |\nabla G \ast u| \)

P. Perona and J. Malik, Scale-space and edge detection using anisotropic diffusion, PAMI 1990
Nonlinear vs. linear diffusion

(a) Linear diffusion at $t=2$

(b) Nonlinear diffusion at $t=4.4$
Extension to vectorial images

- Extension of nonlinear diffusion to vectorial images:

\[
\mathbf{u} = (u_1, u_2, \ldots, u_N)
\]

\[
\frac{\partial \mathbf{u}}{\partial t} = \text{div} \left( g(\|\nabla \mathbf{u}\|) \nabla \mathbf{u} \right)
\]

**generalization**

\[
\frac{\partial u_i}{\partial t} = \text{div} \left( g(\|\nabla \mathbf{u}\|) \nabla u_i \right), \quad i = 1, \ldots, N
\]

where:

\[
\|\nabla \mathbf{u}\| = \sqrt{\sum_{i=1}^{N} \|\nabla u_i\|^2}
\]
Nonlinear diffusion for color image denoising

(a) Color Image with Noise  
(b) Perona-Malik diffusion
Variational interpretation of heat diffusion

- Cost functional:
  \[ E[u] = \iint_{\Omega} \left\| \nabla u \right\|^2 dxdy \]
  \[ = \iint_{\Omega} (u_x^2 + u_y^2) dxdy \]

- Euler-Lagrange:
  \[ \frac{\delta E}{\delta u} = \frac{\partial E}{\partial u} - \frac{\partial}{\partial x} \left( \frac{\partial E}{\partial u_x} \right) - \frac{\partial}{\partial y} \left( \frac{\partial E}{\partial u_y} \right) \]
  \[ = -2 \frac{\partial^2 u_x}{\partial x^2} - 2 \frac{\partial^2 u_y}{\partial y^2} \]
  \[ = -2(u_{xx} + u_{yy}) \]

- Heat diffusion: modifies temperature to decrease E quickly
Variational techniques

- Denoising as functional minimization
  - Functional: encodes *undesirable* properties

- *Total Variation:*

\[
TV[u] = \iint_{\Omega} \sqrt{u_x^2 + u_y^2} \, dx \, dy
\]

\[
\frac{\partial u}{\partial t} = \text{div} \left( \frac{\nabla u}{\| \nabla u \|} \right)
\]

Rudin, L. I.; Osher, S.; Fatemi, E. "Nonlinear total variation based noise removal algorithms". Physica D 60, 1992
Total Variation diffusion

(a) Noisy image
(b) Total Variation diffusion
Perona-Malik versus Total Variation

(a) Perona-Malik diffusion

\[ g(s) = \exp \left( -\frac{s^2}{K^2} \right) \]

(b) Total-Variation diffusion

\[ g(s) = \frac{1}{s} \]
What is the ‘right’ cost functional?

Can we learn the cost function?


- **Filters**: Use Gabor filters, Difference-of-Gaussians, Gaussian filters,
- **Random Fields**: Construct distribution that reproduces their histograms
- **And Maximum Entropy**: while being as random as possible
From FRAME to GRADE

GRADE: Gibbs Reaction & Diffusion Equation
GRADE: maximize image probability using Euler-Lagrange PDEs

S. Roth and M. Black, ‘Fields of Experts’, IJCV 2009
Second Moment Matrix

Distribution of gradients:

$$J = \begin{bmatrix}
\sum_{x',y'} I_{x}^2 \\
\sum_{x',y'} I_{x} I_{y} \\
\sum_{x',y'} I_{x} I_{y} \\
\sum_{x',y'} I_{y}^2
\end{bmatrix}$$
Second Moment Matrix

Distribution of gradients:

\[ J = \sum_{x', y'} (I_x, I_y)^T (I_x, I_y) \]
Second Moment Matrix

Distribution of gradients:

\[ J = \sum_{x', y'} (\nabla G_\sigma \ast u)^T (\nabla G_\sigma \ast u) \]
Second Moment Matrix

Distribution of gradients:

\[ J = G_{\rho} \ast \left[ (\nabla G_{\sigma} \ast u)^T (\nabla G_{\sigma} \ast u) \right] \]
Second Moment Matrix

\[ J = G_\rho \ast \left[ (\nabla G_\sigma \ast u)^T (\nabla G_\sigma \ast u) \right] \]

- Eigenvectors \( w_+ \), \( w_- \): directions of maximal and minimal variation of \( u \)
- Eigenvalues: amounts of minimal and maximal variation \( u \)
Anisotropic diffusion

Nonlinear diffusion

\[ u_t = \text{div} \left( g \left( |\nabla u| \right) \nabla u \right) \]

Nonlinear Anisotropic diffusion

\[ u_t = \text{div} \left( T \left( J_\rho \left( \nabla u_\sigma \right) \nabla u \right) \right) \]

A. Roussos and P. Maragos, Reversible Interpolation of vectorial Images, IJCV 2009
Anisotropic Diffusion example

Slide credits: A. Roussos
Anisotropic Diffusion example
Nonlinear anisotropic diffusion

\[
\frac{\partial u(x, t)}{\partial t} = \text{div} \left( T(J_{\rho}(\nabla u_{\sigma})) \nabla u \right)
\]

Extension to vectorial images

\[
u = (u_1, u_2, \ldots, u_N)
\]

\[
\frac{\partial u_i(x, t)}{\partial t} = \text{div} \left( T(J_{\rho}(\nabla u_{\sigma})) \nabla u_i \right), \quad i = 1, \ldots, M
\]

Structure tensor for vectorial image:

\[
J_{\rho}(\nabla u_{\sigma}) = K_{\rho} \ast \sum_{i=1}^{N} \nabla u_{i,\sigma} (\nabla u_{i,\sigma})^T, \quad \mu \in u_{\sigma} = K_{\sigma} \ast u
\]

Slide credits: A. Roussos
Nonlinear anisotropic diffusion

Noisy input

Nonlinear Anisotropic Diffusion

Slide credits: A. Roussos
Nonlinear vs. Nonlinear and anisotropic diffusion

Nonlinear Diffusion

Nonlinear Anisotropic Diffusion

Slide credits: A. Roussos
Nonlinear vs. Nonlinear and anisotropic diffusion

Nonlinear Diffusion

Nonlinear Anisotropic Diffusion

Slide credits: A. Roussos
Inpainting problem

(a) Image with missing region
(b) Initial prediction (constant)
(c1) 300 iterations
(c2) 600 iterations
(c3) 1500 iterations (converged)

Application: text removal

Digital Image Inpainting can put out of sight this annoying and boring text!

(a) Input  (b) Total Variation Inpainting

Slide credits: A. Roussos
Application: fake bravado

Slide credits: A. Roussos
Interpolation problem

(a) Low resolution input image
(b) Zero-Padding initialization
(c) 4x4 PDE-based magnification

A. Roussos and P. Maragos, Reversible Interpolation of vectorial Images, IJCV 2009
PDE-based interpolation

(a) Input Signal

(b) Bicubic interpolation (4 x 4)

(d) PDE-based interpolation (4 x 4)

A. Roussos and P. Maragos, Reversible Interpolation of vectorial Images, IJCV 2009
PDE-based interpolation

(a) Low-resolution Input
(b) Initialization: Zero-Padding
(c) 4x4 PDE-based interpolation
PDE-based interpolation

(a) Input

(b) Bilinear Interpolation

(c) TV based Interpolation

(d) Structure-tensor based interpolation

A. Roussos and P. Maragos, Reversible Interpolation of vectorial Images, IJCV 2009
Further study

Fast numerical solutions

A. Chambolle, 'An Algorithm for Total Variation Minimization and Applications', JMIV 2004

Not covered in this talk

Bilateral Filter, Non-Local Means


Online software

http://www.ipol.im/