CS 663
Fundamentals of Digital Image Processing
Project Report

Vector-Valued Image Regularization with PDEs

Ankita Pasad
12D070021

Himanshi Swarnkar
120100065

Praveen Agrawal
12D020030

November 23, 2015

1. Introduction

Nonlinear partial differential equations succeed in smoothing data while preserving large global features such as contours and corners (discontinuities of the signal) and their use within variational frameworks has opened new ways to handle classical image processing issues (restoration, segmentation, etc.). Thus, many PDE-based schemes have been presented so far in the literature, particularly for the regularization of 2D scalar images.Extensions of these nonlinear regularization PDEs to vector-valued images $I : \Omega \to \mathbb{R}^n$ have been recently proposed and this is what we have done in this project for colour images.

We start from an image $I_{(t=0)}$ which evolves until a finite number of iterations ($t = t_{end}$).

$$I^{(t=0)} = I_0$$

$$\text{(repeat until } t < t_{end}) \quad I_{x,y}^{t+dt} = I_{x,y}^{t} + dt \frac{\delta I}{\delta t}(x,y)$$

We can classify diffusion PDE's schemes proposed in the paper into one of these three following approaches, related to different interpretation levels of the regularization process:

- **Functional Minimization:** Regularizing an image I may be seen as the minimization of a functional $E(\mathbf{I})$ measuring a global image variation. The idea is that minimizing this functional will flatten the image variations, then gradually remove the noise.
- **Divergence Expressions:** A regularization process may be also more locally designed, as a diffusion of pixel values, viewed as chemical concentrations or temperatures, and directed by a $2x2$ *diffusion tensor* $\mathbf{D}$ (symmetric and positive-definite matrix).
- **Oriented Laplacians:** 2D image regularization may be finally seen as the simultaneous juxtaposition of two oriented 1D heat flows, leading to 1D Gaussian smoothing processes along orthonormal directions with different weights.

In this project we have implemented the third approach stated above, which is using the oriented laplacians.

2. Image Regularization as Oriented 1D Laplacians

Two simultaneous 1D heat flows, oriented in orthogonal directions $\xi_{(x,y)}$ and $\eta_{(x,y)}$ and weighted by two coefficients $c_{1(x,y)}$ and $c_{2(x,y)} > 0$:

$$\frac{\partial \mathbf{I}}{\partial t} = c_1 \frac{\partial^2 \mathbf{I}}{\partial \xi^2} + c_2 \frac{\partial^2 \mathbf{I}}{\partial \eta^2} = c_1 \mathbf{I}_{\xi\xi} + c_2 \mathbf{I}_{\eta\eta}$$

$$where, \quad \eta = \frac{\nabla \mathbf{I}}{\|\nabla \mathbf{I}\|} and \quad \xi = \eta^{\perp}$$

The smoothing weights c_1, c_2 and directions ξ, η are directly designed from the eigenvalues and eigenvectors of the structure tensor $\mathbf{G}$, where, $\mathbf{G} = \sum_{j=1}^{n} \nabla \mathbf{I}_j \nabla \mathbf{I}_j^T$, in order to perform edge-preserving smoothing, mainly along the direction orthogonal to the image discontinuities.

3. Diffusion Tensor

A second-order tensor is a symmetric and semi-positive definite p × p matrix. (p = 2 for images, p = 3 for volumetric images). It has p positive eigenvalues and p orthogonal eigenvectors. Tensors can describe a smoothing process, by telling how much the pixel values diffuse along given orthogonal orientations, i.e. the “geometry” of the smoothing.

$$\frac{\partial \mathbf{I}}{\partial t} = c_1 \mathbf{I}_{\xi\xi} + c_2 \mathbf{I}_{\eta\eta} = trace(\mathbf{T}\mathbf{H}_i)$$

where $\mathbf{H}_i$ is the Hessian matrix of the vector component I_i

$$\mathbf{H}_{i,j} = \frac{\partial^2 \mathbf{I}}{\partial x_i \partial x_j}$$

and $\mathbf{T}$ is the diffusion tensor with eigenvalues c_1, c_2 and eigenvectors ξ, η.

$$\mathbf{T} = c_1 \xi \xi^T + c_2 \eta \eta^T$$

where, c1 and c2 are defined as the following functions of the eigenvalues (λ_1 and λ_2),

$$c_1 = \frac{1}{(1+\lambda_1+\lambda_2)^p} \quad \text{and} \quad c_2 = \frac{1}{(1+\lambda_1+\lambda_2)^q}$$

p and q are chosen as needed depending on the application.

4. Applications

4.1 Colour Image Restoration

Colour image with real noise (digital snapshot under low luminosity conditions). Our vector-valued regularization PDE can successfully remove the noise, while preserving the global features of the image.

Figures (1), (2) and (3) demonstrate some of the results that were obtained (in each of them, left image is the noisy input and the right image is the denoised output). In figure (4) we have regularized the input image to remove the wrinkles from the face image. The value of p was taken as 2 and that of q was taken as 0.5.

Figure 1: Image Denoising (After 120 iterations with step-size of 0.01)

Figure 2: Image Denoising (After 70 iterations with step-size of 0.01)

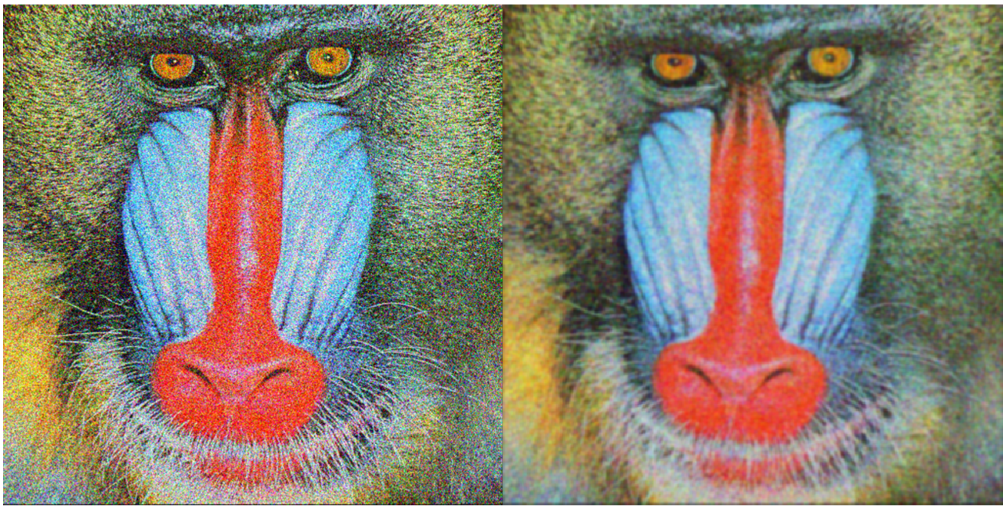


Figure 3: Image Denoising (After 200 iterations with step-size of 0.005)

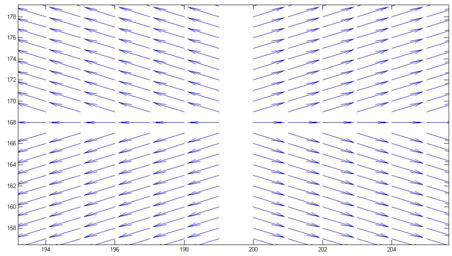
Figure 4: Image Denoising (After 50 iterations with step-size of 0.01)

4.2 Flow Visualization

We can visualize a 2D vector field $\mathscr{F} : \Omega \to \mathbb{R}^2$ by modifying our trace PDE to diffuse in the direction of the field $\mathscr{F}$ than according to the local geometry of the image. As, the evolution time increases, more global features of the flow appear.

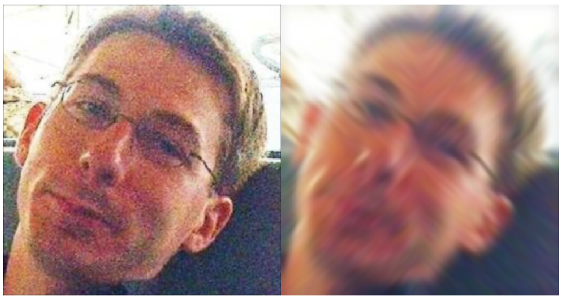
$$\frac{\delta I_i}{\delta t} = trace\Big(\Big[\frac{1}{||\mathscr{F}||}\mathscr{F}\mathscr{F}^T\Big]\mathbf{H}_i\Big)$$

(a)Visualization flow using trace PDEs

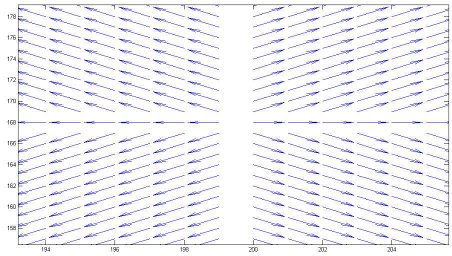


(b)Visualization of flow in MATLAB

Figure 5: Flow Visualization on clean image (After 1000 iterations with step-size of 0.01)

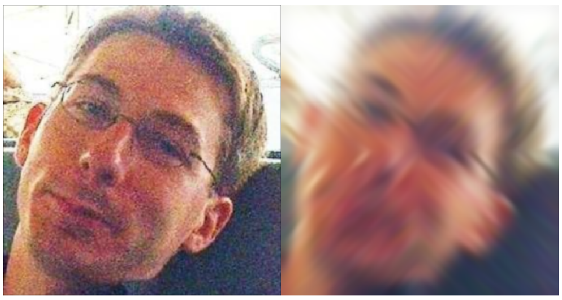


(a)Visualization flow using trace PDEs

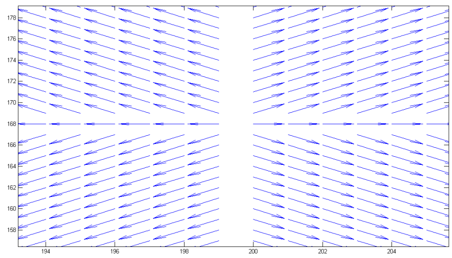


(b)Visualization of flow in MATLAB

Figure 6: Flow Visualization on noisy image (After 500 iterations with step-size of 0.01)



(a)Visualization flow using trace PDEs



(b)Visualization of flow in MATLAB

Figure 7: Flow Visualization on noisy image (After 1000 iterations with step-size of 0.01)

4.3 Color Image Inpainting

Image inpainting consists of filling undesired holes (defined by the user) in an image by interpolating the data located at the neighbourhood of the holes. It is possible to do that by applying our PDE only in the holes to fill: boundaries pixels will be diffused until they completely fill the missing regions, in a structure-preserving way.

Figures (8) and (9) demonstrate this where we have removed the wire mesh of the cage and the spectacles respectively (again input on the left and output on the right). The value of p was taken as 0 and that of q was taken as 1.2.

Figure 8: Image Inpainting (After 8000 iterations with step-size of 0.01)

Figure 9: Image Inpainting (After 5000 iterations with step-size of 0.01)

4.4 Image Resizing

Starting from a linear interpolation of a small image, and applying our PDE on the image, we can retrieve nonlinear magnified images without jagging or bloc effects, inherent to classical linear interpolation techniques.

Figures (10) and (11) demonstrate some of the results that were obtained (leftmost image is the input image,middle image is the output obtained after bilinear interpolation and rightmost image is the output after running the PDE on the interpolated image). The value of p was taken as 2 and that of q was taken as 0.5.

Figure 10: Image Resizing (After 200 iterations with step-size of 0.01)

Figure 11: Image Resizing (After 200 iterations with step-size of 0.01)

5. References

- David Tschumperlé, Rachid Deriche
  Vector-Valued Image Regularization with PDEs: A Common Framework for different Applications
  IEEE transactions on pattern analysis and machine intelligence, vol. 27, no.4, April 2005
- Chapter 3 of the book on Digital Color Imaging
- Slides on Tensor-Directed Smoothing of Multi-Valued Images with Curvature-Preserving Diffusion PDE's by David Tschumperlé
- Slides on Numerical Differentiation for the method to find gradients