Chapter 4

Efficient PDE-based nonlinear diffusion and Time dependent models for image denoising

4.1 Introduction

In this Chapter, we present a new nonlinear anisotropic diffusion model which incorporates adaptive information computed from the image at scale $t$. Following [3, 9], well-posedness of the proposed scheme is proved using the theory of viscosity solutions. We derive theoretical considerations for anisotropic diffusion model. We present proof of the viscosity solution of model (4.2.6). Besides, we propose a time dependent model for solving total variation (TV) minimization problem in image denoising. This is a constrained optimization type of numerical algorithm for removing noise from images. The constraints are imposed using Lagrange’s multipliers and the solution is obtained using the gradient projection method. 1D and 2D numerical experimental results by explicit numerical schemes are discussed.

The nonlinear diffusion method for image denoising and edge detection was first introduced by Perona and Malik [56]. This method is based on a diffusion process governed by a partial differential equation (PDE), where diffusion amount depends on the gradient of images.

Mathematically, $u_0 : \Omega \to \mathbb{R}$ represents a noisy version of a true image, and it is obtained by the following imaging process

$$ u_0(x, y) = u(x, y) + n(x, y), \quad (4.1.1) $$

59
where \( u(x, y) \) denotes the desired clean image, \( u_0(x, y) \) denotes the pixel values of a noisy image for \( x, y \in \Omega, \Omega \subset \mathbb{R}^2 \) is a bounded domain, usually a rectangle and \( n(x, y) \) is additive white noise assumed to be close to Gaussian. The values \( n(i, j) \) of \( n \) at the pixels \((i, j)\) are independent random variables, each with a Gaussian distribution of zero mean and variance \( \sigma^2 \).

In our tests, we will use the peak signal to noise ratio (PSNR) as a criteria for the quality of restoration:

\[
\text{PSNR} = 10 \log_{10} \left( \frac{R^2}{\frac{1}{mn} \sum_{i,j} (u(i,j) - u_{\text{new}}(i,j))^2} \right),
\]

where \( \{u(i,j) - u_{\text{new}}(i,j)\} \) is the difference of the pixel values between the restored and original images.

The choice of the diffusivity \( c \) is very important in controlling the smoothing and even enhancement of edges. The Charbonnier diffusivity \( c(s) = \frac{1}{\sqrt{1 + (|s|^2/K^2)}} \), that is related to the convex regularizer \( \psi(s^2) = \sqrt{K^4 + K^2 s^2 - K^2} \), see references [18, 79], is used in our experiments.

### 4.2 Physical background for anisotropic diffusion model for image denoising

In general, variational deblurring and denoising of an image can be achieved by minimizing the energy functional presented in [82],

\[
E(u) = \int_{\Omega} \psi(|\nabla u|^2) \, dx + \frac{\lambda}{2} \int_{\Omega} (u - u_0)^2 \, dx.
\]

The Euler-Lagrange equation associated with (4.2.1) with homogeneous Neumann boundary conditions is given by

\[
\begin{cases}
0 = -\text{div}(\psi'(\|\nabla u\|^2)\nabla u) + \lambda (u - u_0), & x \in \Omega, \\
\frac{\partial u}{\partial \bar{n}} = 0, & x \in \partial \Omega,
\end{cases}
\]

where \( \partial \Omega \) is the boundary of \( \Omega \) and \( \bar{n} \) is the outward normal to \( \partial \Omega \).
The resulting gradient descent equation is

\[ u_t = \text{div}(c(|\nabla u|)\nabla u) - \lambda (u - u_0), \tag{4.2.3} \]

with \(u(x, 0)\) given as initial data (the original noisy image \(u_0(x)\) used as initial guess), homogeneous Neumann boundary conditions, i.e., \(\frac{\partial u}{\partial n} = 0\) on the boundary of the domain. It is also known as diffusion-reaction equation where the diffusion term with diffusivity \(c(s) = \psi'(s^2)\) is related to the regulariser in the energy functional.

Applying a priori smoothness on the solution image, our nonlinear anisotropic diffusion model becomes,

\[ u_t = \text{div}(c(|\nabla G_\sigma * u|)\nabla G_\sigma * u) - \lambda (G_\sigma * u - u_0). \tag{4.2.4} \]

Witkin [83] noticed that the convolution of the signal with Gaussians at each scale was equivalent to solving the heat equation with the signal as initial datum. The term \((G_\sigma * \nabla u)(x, t) = (\nabla G_\sigma * u)(x, t)\), which appears inside the divergence term of (4.2.4), is simply the gradient of the solution at time \(\sigma\) of the heat equation with \(u(x, 0)\) as initial datum.

In order to preserve the notion of scale in the gradient estimate, it is convenient that this kernel \(G_\sigma\) depends on a scale parameter [45]. In fact, the function \(G_\sigma\) can be considered as “low-pass filter” or any smoothing kernel, i.e., a denoising technique is used before solving the nonlinear diffusion problem [3, 13].

We use the following class of functions for the diffusion equation, given in [8, 71],

\[ c(x, |\nabla u|) = \alpha(x) c_g(|\nabla u|). \tag{4.2.5} \]

Here \(\alpha\) is the adaptive parameter estimated at each pixel \(x \in \Omega\). The function \(c_g\) depends on the gradient image \(|\nabla u|\) and can be chosen similar to \(c(s)\). If we choose \(\alpha(x) = 1, c_g = c(s)\) and \(G_\sigma * u\) as \(u\) then the model (4.2.4) can be written as:

\[ \frac{\partial u}{\partial t} = \text{div}(c(x, |\nabla u|)\nabla u) - \lambda (u - u_0). \tag{4.2.6} \]
4.3 Theoretical considerations

In this Section, motivated by Alvarez et al. [3], we want to present the viscosity solution for model (4.2.6).

\[
\frac{\partial u}{\partial t} = \text{div}(c(x, |\nabla u|)\nabla u) - \lambda(u - u_0), \quad x \in \mathbb{R}^n, \ t \in \mathbb{R}_+,
\]

\[
u(x, 0) = u_0(x), \quad x \in \mathbb{R}^n.
\]  

(4.3.1)

Let us first introduce two auxiliary functions depending on \( x \) and \( p \) from \( \mathbb{R}^n \), a symmetric matrix-valued one \( a \) and a vector one \( \chi \). We denote

\[
a_{ij}(x, p) = c(x, |p|)\delta_{ij} + c_y(x, |p|)\frac{p_ip_j}{|p|},
\]  

(4.3.2)

\[
\chi_i(x, p) = \frac{\partial c(x, |p|)}{\partial x_i}.
\]  

(4.3.3)

Here \( \delta_{ij} \) is Kronecker’s delta, and \( c_y \) is the partial derivative of \( c(x, y) \) with respect to the second variable.

Motivated by Alvarez et al. [2], we consider the case of spatially periodic boundary conditions. We will assume that there is an orthogonal basis \( b_i \) in \( \mathbb{R}^n \) so that

\[
u(., x + b_i) = \nu(., x), \quad x \in \mathbb{R}^n, \ i = 1, 2, ...., n.
\]  

(4.3.4)

Let \( u_0 \) is Lipschitz and satisfies (4.3.4). Of course, \( c \) (and thus \( a \) and \( \chi \)) should also satisfy the same spatial periodicity restriction (with respect to \( x \) but not to \( y \) or \( p \)). Functions \( a \) and \( \chi \) are continuous, bounded, periodic and continuously differentiable in \( x \) and their \( x \)-derivatives are uniformly (w.r.t. \( p \)) bounded,

\[
a_{ij}(x, p)\xi_i\xi_j \geq C \left[ \text{mod} \left( \frac{\partial a(x, p)}{\partial x_k} \right) \right]_{ij} \xi_i\xi_j, \quad k = 1, ......., n, \ \xi, x, p \in \mathbb{R}^n.
\]  

(4.3.5)

Here \( \lambda \geq 0 \) and below \( C \) stands for a generic positive constant, which can take different values in different lines.

We first recall the definition of viscosity sub-/supersolution of (4.3.1), if for any \( \phi \in C^2([0, T] \times \mathbb{R}^n) \) and any point \((x_0, t_0) \in (0, T] \times \mathbb{R}^n\), at which \( u - \phi \) attains local
maximum/minimum [29].

\[
\frac{\partial \phi(x_0, t_0)}{\partial t} - \text{div}(c(x_0, |\nabla \phi(x_0, t_0)|) \nabla \phi(x_0, t_0)) + \lambda(u(x_0, t_0) - u_0(x_0)) \leq 0; \geq 0. \tag{4.3.6}
\]

A viscosity solution is a function which is both a subsolution and a supersolution.

**Lemma 4.3.1.** Let \( A \) and \( B \) be quadratic matrices of order \( n \). Assume that \( B \) is symmetric and there is a constant \( M \geq 0 \) such that

\[
MA_{ij}\xi_i\xi_j \geq \text{mod}(B)_{ij}\xi_i\xi_j, \quad \forall \xi \in \mathbb{R}^n. \tag{4.3.7}
\]

Then for any matrix \( U \) (of the same order but not necessarily symmetric) one has

\[
\text{Tr}^2(BU^\top) \leq M\|B\|\text{Tr}(UAU^\top), \tag{4.3.8}
\]

where \( \|\cdot\| \) denotes the operator norm of a matrix and \( \text{mod}(B) \) be the matrix whose entries are the absolute values of the entries of \( B \).

**Proof.** Formula (4.3.7) and (4.3.8) are invariant with respect to orthogonal changes of bases. Thus, without loss of generality, we may assume that \( B \) is already diagonalized by an orthogonal transform. Then

\[
\text{Tr}^2(BU^\top) = (B_iU_i)^2 \leq \|B\|\|B_iU_i^2 \leq \|B\|\|\text{mod}(B)_{ij}U_iU_j \|
\]

\[
= \|B\|\|\text{mod}(B)_{ij}U_iU_j \leq M\|B\|A_{ij}U_iU_j = M\|B\|\text{Tr}(UAU^\top).
\]

**Theorem 4.3.1.** The problem (4.3.1) has a unique viscosity solution \( u \) in \( C([0, T] \times \mathbb{R}^n) \cap L^\infty(0, T, W^{1,\infty}(\mathbb{R}^n)) \) for any \( T \in [0, \infty) \), provided that \( u_0 \) is Lipschitz continuous on \( \mathbb{R}^n \), and if \( v \in C(\mathbb{R}^n \times [0, T]) \) is a viscosity solution of (4.3.1) with \( u_0 \) replaced by a Lipschitz continuous function \( v_0 \), then for all \( T \in [0, \infty) \), there exists a constant \( C > 0 \), depending only on \( u_0 \), \( v_0 \) and \( T \), such that

\[
\sup_{0 \leq t \leq T} \|u(x, t) - v(x, t)\|_{L^\infty(\mathbb{R}^n)} \leq C\|u_0 - v_0\|_{L^\infty(\mathbb{R}^n)}. \tag{4.3.9}
\]

Moreover, \( \inf_{\mathbb{R}^n} u_0 \leq u(x, t) \leq \sup_{\mathbb{R}^n} u_0 \).
Proof. If \( u \) is a viscosity solution of equation (4.3.1) on \( \mathbb{R}^n \times \mathbb{R}_+ \), then

\[
\inf_{\mathbb{R}^n} u_0 \leq u(x,t) \leq \sup_{\mathbb{R}^n} u_0, \quad \text{on} \quad \mathbb{R}^n \times \mathbb{R}_+.
\] (4.3.10)

Let \( \phi(x,t) = \delta t \), then, at the point \((x_0,t_0)\), \( t_0 > 0 \), of the global maximum of \( u(x,t) - \delta t \), (4.3.6) gives \( \delta + \lambda(u(t_0,x_0) - u_0(x_0)) \leq 0 \), when \( u(x_0,t_0) < u_0(x_0) \), so we get a contradiction since \( u(x_0,t_0) - \delta t_0 \geq u_0(x_0) \) due to the fact that \((x_0,t_0)\) is the global maximum point of \( u(x,t) - \delta t \); thus the function \( u(x,t) - \delta t \) attains its global maximum at \( t = 0 \), and it remains to let \( \delta \to 0^+ \), we get (4.3.10).

Now, we establish a formal a priori estimate for \( \sup_{\mathbb{R}^n} |\nabla u| \). Observe that (4.3.1) is equivalent to

\[
\frac{\partial u}{\partial t} = [a_{ij}(x,\nabla u)u_{x_i}u_{x_j} + \chi_i(x,\nabla u)u_{x_i}] - \lambda(u - u_0).
\] (4.3.11)

Differentiating (4.3.11) with respect to each \( x_k \), \( k = 1, \ldots, n \), multiplying by \( 2u_{x_k} \) and taking a summation w.r.t. \( k \), we get

\[
\gamma(|\nabla u|^2) := \frac{\partial |\nabla u|^2}{\partial t} - a_{ij}(x,\nabla u) \frac{\partial^2 |\nabla u|^2}{\partial x_i \partial x_j} - \frac{\partial a_{ij}(x,\nabla u)}{\partial p_l} u_{x_i}u_{x_j} \frac{\partial |\nabla u|^2}{\partial x_l} \\
- \chi_i(x,\nabla u) \frac{\partial |\nabla u|^2}{\partial x_i} u_{x_k} \frac{\partial |\nabla u|^2}{\partial x_l} u_{x_j} + 2\lambda(u_{x_k} - (u_0)_{x_k})u_{x_k} \\
= -2a_{ij}(x,\nabla u)u_{x_i}u_{x_j}u_{x_k} + 2\frac{\partial a_{ij}(x,\nabla u)}{\partial x_k} u_{x_i}u_{x_j}u_{x_k} + 2\frac{\partial \chi_{ij}(x,\nabla u)}{\partial x_k} u_{x_i}u_{x_j}u_{x_k}.
\] (4.3.12)

The Lemma 4.3.1 gives opportunity to discharge the undesired influence of the second term in the right-hand side of (4.3.12). For the second term, due to the Lemma 4.3.1 and Cauchy’s inequality, we have

\[
\left| 2\frac{\partial a_{ij}(x,\nabla u)}{\partial x_k} u_{x_i}u_{x_j}u_{x_k} \right| \leq C|u_{x_k}| \sqrt{a_{ij}(x,\nabla u)u_{x_i}u_{x_j}u_{x_k}u_{x_k}} \\
\leq a_{ij}(x,\nabla)u_{x_k}u_{x_k}u_{x_k} + C|\nabla u|^2.
\] (4.3.13)

The sum of the absolute values of the subsequent terms of the right-hand side of (4.3.12) does not exceed \( C(1 + |\nabla u|^2) \). Thus,

\[
\gamma(|\nabla u|^2) \leq C(1 + |\nabla u|^2),
\] (4.3.14)
so

$$\gamma(e^{-Ct}(1 + |\nabla u|^2)) \leq 0. \quad (4.3.15)$$

From the weak maximum principle for the weakly parabolic operator $\gamma$ one easily concludes that

$$|\nabla u|^2 \leq C. \quad (4.3.16)$$

Using (4.3.10) and (4.3.16), by means of the approach from [2] we can get the uniform Holder estimate

$$|u(x, t) - y(x, s)|^2 \leq C|t - s|. \quad (4.3.17)$$

From (4.3.10), (4.3.16) and (4.3.17), the solutions of these problems are uniformly bounded and equicontinuous on $\mathbb{R}^n \times [0, T]$. Then we can select a uniformly converging sequence of approximate solutions, and pass to the limit in the viscosity sense using the general consistency/stability results from [24]. The uniqueness of solutions follows from the stability estimate (4.3.9). This bound may be shown by revisiting the proof of a similar bound in [66]. We only point out that the matrix $\tau$ [66] is replaced by

$$\tau = \begin{pmatrix} D_1 & \sqrt{D_1D_2} \\ \sqrt{D_1D_2} & D_2 \end{pmatrix}, \quad (4.3.18)$$

where

$$D_1 = a \left( x_0, \frac{|x_0 - y_0|^2(x_0 - y_0)}{\delta} \right), \quad D_2 = a \left( y_0, \frac{|x_0 - y_0|^2(x_0 - y_0)}{\delta} \right),$$

and the notation within is taken from [66]. Note that the $2n \times 2n$ matrix $\tau$ is symmetric and positive-semidefinite.

### 4.4 Numerical experiments for anisotropic diffusion model

We have used two gray scale images as shown in Figure 2.5.1. The pixel values of all images lie in interval $[0, 255]$. The Gaussian white noise is added by the normal imnoise function imnoise (I,'Gaussian', M, $\sigma^2$), i.e., the mean $M$ and variance $\sigma^2$ in Matlab. We first scale the intensities of the images into the range between zero and one before we begin our experiments. We have taken $\Delta t/\Delta x^2 = 0.4$, see [41], Charbonnier diffusivity $K = 5$ and Lagrange multiplier = 0.85 as in [15] and [17] in our all experiments.
Figure 4.4.1: (top row) Noisy Lena images with different levels of Gaussian noise (a)-(c), $\sigma^2 = 0.002, 0.004, 0.006$, respectively; (second row) (d)-(f) corresponding denoised images by model (4.2.3); (third row) (g)-(i) by model (4.2.4).
Figure 4.4.2: (top row) Noisy Boat images with different levels of Gaussian noise (a)-(c), $\sigma^2 = 0.002, 0.004, 0.006$, respectively; (second row) (d)-(f) corresponding denoised images by model (4.2.3); (third row) (g)-(i) by model (4.2.4).
Table 4.4.1: Results obtained by using models (4.2.3) and (4.2.4) applied to the images in Figures 4.4.1(a) and 4.4.2(a) with Gaussian white noise ($\sigma^2 = 0.002$).

<table>
<thead>
<tr>
<th>Images</th>
<th>PSNR</th>
<th>Images</th>
<th>PSNR</th>
<th>Images</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noisy Image</td>
<td>(4.2.3)</td>
<td>(4.2.4)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fig. 4.4.1(a)</td>
<td>27.02</td>
<td>Fig. 4.4.1(d)</td>
<td>30.33</td>
<td>Fig. 4.4.1(g)</td>
<td>30.48</td>
</tr>
<tr>
<td>Fig. 4.4.2(a)</td>
<td>27.05</td>
<td>Fig. 4.4.2(d)</td>
<td>29.75</td>
<td>Fig. 4.4.2(g)</td>
<td>30.01</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>No. of 400</td>
<td>-</td>
<td>No. of 200</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.4.2: Results obtained by using models (4.2.3) and (4.2.4) applied to the images in Figures 4.4.1(b) and 4.4.2(b) with Gaussian white noise ($\sigma^2 = 0.004$).

<table>
<thead>
<tr>
<th>Images</th>
<th>PSNR</th>
<th>Images</th>
<th>PSNR</th>
<th>Images</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noisy Image</td>
<td>(4.2.3)</td>
<td>(4.2.4)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fig. 4.4.1(b)</td>
<td>24.08</td>
<td>Fig. 4.4.1(e)</td>
<td>27.14</td>
<td>Fig. 4.4.1(h)</td>
<td>28.78</td>
</tr>
<tr>
<td>Fig. 4.4.2(b)</td>
<td>24.09</td>
<td>Fig. 4.4.2(e)</td>
<td>26.80</td>
<td>Fig. 4.4.2(h)</td>
<td>28.45</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>No. of 400</td>
<td>-</td>
<td>No. of 200</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.4.3: Results obtained by using models (4.2.3) and (4.2.4) applied to the images in Figures 4.4.1(c) and 4.4.2(c) with Gaussian white noise ($\sigma^2 = 0.006$).

<table>
<thead>
<tr>
<th>Images</th>
<th>PSNR</th>
<th>Images</th>
<th>PSNR</th>
<th>Images</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noisy Image</td>
<td>(4.2.3)</td>
<td>(4.2.4)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fig. 4.4.1(c)</td>
<td>22.36</td>
<td>Fig. 4.4.1(f)</td>
<td>25.17</td>
<td>Fig. 4.4.1(i)</td>
<td>27.46</td>
</tr>
<tr>
<td>Fig. 4.4.2(c)</td>
<td>22.33</td>
<td>Fig. 4.4.2(f)</td>
<td>24.96</td>
<td>Fig. 4.4.2(i)</td>
<td>27.28</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>No. of 400</td>
<td>-</td>
<td>No. of 200</td>
<td></td>
</tr>
</tbody>
</table>
4.5 Time dependent model for 2D

Most conventional variational methods involve a least squares $L^2$ fit because this leads to linear equations. The first attempt along these lines was made by Phillips [57] and later refined by Twomey et al. [73, 74] in one-dimensional case.

The total variation based image denoising model, which is based on the constrained minimization problem appeared in [61], is as follows:

\[
\text{minimize} \int \Omega |\nabla u| \, dx \, dy = \int \Omega \sqrt{u_x^2 + u_y^2} \, dx \, dy, \quad (4.5.1)
\]

subject to constraints

\[
\int \Omega u \, dx \, dy = \int \Omega u_0 \, dx \, dy, \quad (4.5.2)
\]

and

\[
\int \Omega \frac{1}{2} (u - u_0)^2 \, dx \, dy = \sigma^2. \quad (4.5.3)
\]

The first constraint corresponds to the assumption that the noise has zero mean, and the second constraint uses a priori information that the standard deviation of the noise $n(x,y)$ is $\sigma$.

The Euler-Lagrange equation is given by,

\[
0 = \nabla \cdot \left( \frac{\nabla u}{|\nabla u|} \right) - \lambda_1 - \lambda_2 (u - u_0), \quad (4.5.4)
\]

in $\Omega$, with $\frac{\partial u}{\partial n} = 0$ on the boundary of the domain.

Since (4.5.4) is not well defined at points where $\nabla u = 0$, due to the presence of the term $\frac{1}{|\nabla u|}$, it is common to slightly perturb the TV algorithm to become

\[
\int \Omega |\nabla u|_\beta \, dx \, dy = \int \Omega \sqrt{u_x^2 + u_y^2 + \beta} \, dx \, dy, \quad (4.5.5)
\]

where $\beta$ is a small positive parameter [17].

The solution procedure uses a parabolic equation with time as an evolution parameter, or equivalently, the gradient descent method. This means that we solve
\[ u_t = \nabla \cdot \left( \frac{\nabla u}{|\nabla u|} \right) - \lambda (u - u_0), \quad (4.5.6) \]

for \( t > 0, \; x, \; y \in \Omega \) with \( u(x, y, 0) \) given as initial data and \( \frac{\partial u}{\partial n} = 0 \) on the boundary of the domain.

Applying a priori smoothness on the solution image, our new time dependent model becomes,

\[ u_t = \nabla \cdot \left( \frac{\nabla G_{\sigma} * u}{|\nabla G_{\sigma} * u|} \right) - \lambda (G_{\sigma} * u - u_0), \quad (4.5.7) \]

for \( t > 0, \; x, \; y \in \Omega \) with \( u(x, y, 0) \) given as initial data and \( \frac{\partial u}{\partial n} = 0 \) on the boundary of the domain. It should be noticed that (4.5.7) only replaces \( u \) in (4.5.6) by its estimate \( G_{\sigma} * u \).

Witkin [83] noticed that the convolution of the signal with Gaussians at each scale was equivalent to solving the heat equation with the signal as initial datum. The term \( (G_{\sigma} * \nabla u)(x, y, t) = (\nabla G_{\sigma} * u)(x, y, t) \), which appears inside the divergence term of (4.5.7), is simply the gradient of the solution at time \( \sigma \) of the heat equation with \( u(x, y, 0) \) as initial datum. In order to preserve the notion of scale in the gradient estimate, it is convenient that this kernel \( G_{\sigma} \) depends on a scale parameter [45]. In fact, the function \( G_{\sigma} \) can be considered as “low-pass filter” or any smoothing kernel, i.e., a denoising technique is used before solving the nonlinear diffusion problem [3, 13].

The first constraint (4.5.4) is dropped because it is automatically enforced by the evolution procedure, i.e., the mean of \( u(x, y, 0) \) is the same as that of \( u_0(x, y) \). As \( t \) increases, a denoised version of image is realised.

To compute \( \lambda(t) \), we multiply (4.5.6) by \( (u - u_0) \) and integrate by parts over \( \Omega \). If steady state has been reached, the left side of (4.5.6) vanishes. We then have,

\[ \lambda = -\frac{1}{2\sigma^2} \int_{\Omega} \left[ \nabla u - \left( \frac{(u_0)_x u_x}{|\nabla u|} + \frac{(u_0)_y u_y}{|\nabla u|} \right) \right] \, dx \, dy. \quad (4.5.8) \]

This gives us a dynamic value \( \lambda(t) \), which appears to converge as \( t \to \infty \). The theoretical justification for this approach comes from the fact that it is merely the gradient projection method of Rosen [59].

We still write \( G_{\sigma} * u \) as \( u \). Let \( u^n_{ij} \) be the approximation to the value \( u(x_i, y_j, t_n) \), where

\[ x_i = i\Delta x, \; y_j = j\Delta x, \; i, j = 1, 2, \ldots, N, \]
Efficient PDE-based nonlinear diffusion and Time dependent models for image denoising

\[ N \Delta x = 1, \ t_n = n \Delta t, \ n = 0, 1, \ldots, \]

\[ u^n_{i,j} = u(x_i, y_j, t_n), \]

\[ u^0_{i,j} = u_0(i \Delta x, j \Delta x) + \sigma \phi(i \Delta x, j \Delta x). \]  \hspace{1cm} (4.5.9)

The modified initial data are chosen so that the constraints are satisfied initially, i.e., \( \phi \) has mean zero and \( L^2 \) norm one.

The explicit partial derivatives of model (4.5.6) and model (4.5.7) can be expressed as:

\[ u_t = \frac{u_{xx}(u_y^2 + \beta) - 2u_{xy}u_xu_y + u_{yy}(u_x^2 + \beta)}{(u_x^2 + u_y^2 + \beta)^{\frac{3}{2}}} - \lambda (u - u_0). \]  \hspace{1cm} (4.5.10)

We define the derivative terms as,

\[ u^x_{ij} = \frac{u^n_{i+1,j} - u^n_{i-1,j}}{2\Delta x}; \quad u^y_{ij} = \frac{u^n_{i,j+1} - u^n_{i,j-1}}{2\Delta x}; \]

\[ u^{xx}_{ij} = \frac{u^n_{i+1,j} - 2u^n_{i,j} + u^n_{i-1,j}}{\Delta x^2}; \quad u^{yy}_{ij} = \frac{u^n_{i,j+1} - 2u^n_{i,j} + u^n_{i,j-1}}{\Delta x^2}; \]

\[ u^{xy}_{ij} = \frac{u^n_{i+1,j+1} - u^n_{i-1,j+1} - u^n_{i+1,j-1} + u^n_{i-1,j-1}}{4\Delta x \Delta x}; \quad u^t_{ij} = \frac{u^n_{i,j} + 1 - u^n_{i,j-1}}{\Delta t}. \]

We let,

\[ r^n_{ij} = u^{xx}_{ij}((u^y_{ij})^2 + \beta) - 2u^{xy}_{ij}u^x_{ij}u^y_{ij} + u^{yy}_{ij}((u^x_{ij})^2 + \beta), \]  \hspace{1cm} (4.5.11)

and

\[ p^n_{ij} = ((u^x_{ij})^2 + (u^y_{ij})^2 + \beta)^{\frac{3}{2}}. \]  \hspace{1cm} (4.5.12)

Then (4.5.10) reads as follows:

\[ u^t_{ij} = \frac{r^n_{ij}}{p^n_{ij}} - \lambda(u^n_{ij} - u_0(i \Delta x, j \Delta x)), \]  \hspace{1cm} (4.5.13)

with boundary conditions

\[ u^n_{i,N} = u^n_{i,N-1}, \quad u^n_{N,j} = u^n_{N-1,j}, \quad u^n_{1,j} = u^n_{2,j}, \quad u^n_{i,1} = u^n_{i,2}. \]  \hspace{1cm} (4.5.14)

The explicit method is stable and convergent for \( \frac{\Delta t}{\Delta x^2} \leq 0.5 \), see [41].
4.6 Time dependent model for 1D

The 2D model described before is more regular than the corresponding 1D model because the 1D original optimization problem is barely convex. For the sake of understanding the numerical behavior of our schemes, we also discuss the 1D model. The Euler-Lagrange equation in the 1D case reads as follows:

\[ 0 = \left( \frac{u_x}{|u_x|} \right)_x - \lambda_1 - \lambda_2(u - u_0). \]  

(4.6.1)

This equation can be written either as

\[ 0 = \left( \frac{u_x}{|u_x|} \frac{\beta}{\beta} \right)_x - \lambda_1 - \lambda_2(u - u_0), \]  

(4.6.2)

using the small regularizing parameter \( \beta > 0 \) introduced in [46], or

\[ 0 = \delta(u_x)u_{xx} - \lambda_1 - \lambda_2(u - u_0), \]  

(4.6.3)

using the \( \delta \)-function.

Our model in 1D will be

\[ u_t = \frac{\beta}{(\beta + u_x^2)^2}u_{xx} - \lambda(u - u_0), \]  

(4.6.4)

where \( \beta > 0 \) is small regularizing parameter. The parameter \( \beta > 0 \) in this model is estimated from the local amount of noise. We have found for our model, through our numerical experiments in 1D, that \( \beta \) can be estimated as the standard deviation of the noise.

We can also state our model in terms of the \( \delta \) function as

\[ u_t = \delta(u_x)u_{xx} - \lambda(u - u_0). \]  

(4.6.5)

In this chapter, we approximate \( \delta \), see the reference [46], by

\[ \delta(k) \approx \beta.(k^2 + \beta)^{-\frac{3}{2}}. \]  

(4.6.6)

These evolution models are initialized with the noisy signal \( u_0 \), homogeneous Neu-
mann boundary conditions, and with a prescribed Lagrange multiplier for slightly noisy signals.

We have estimated $\lambda > 0$ near the maximum value such that the explicit scheme is stable under appropriate CFL ($\frac{\Delta t}{\Delta x^2} < 0.25$) restrictions [46], provided $\beta$ is chosen to be the standard deviation of the noise.

The following is the explicit numerical scheme of model (4.6.4). Let $u^n_i$ be the approximation to the value $u(x_i, t_n)$, where $x_i = i\Delta x$ and $t_n = n\Delta t$, $n \geq 1$. We define the derivative terms as,

\[ u_x = \frac{u^n_{i+1} - u^n_{i-1}}{2\Delta x}; \quad u_{xx} = \frac{u^n_{i+1} - 2u^n_i + u^n_{i-1}}{\Delta x^2}; \]

\[ u_t = \frac{u^n_{i+1} - u^n_i}{\Delta t}. \]

We let,

\[ b_i = \frac{u^n_{i+1} - u^n_{i-1}}{2\Delta x}. \quad (4.6.7) \]

Then (4.6.4) reads as follows:

\[ u^{n+1}_i = u^n_i + \Delta t \left[ \frac{\beta}{(\beta + b_i^2)^{\frac{3}{2}}} \frac{u^n_{i+1} - 2u^n_i + u^n_{i-1}}{\Delta x^2} \right] - \Delta t\lambda(u^n_i - u_0(x_i)). \quad (4.6.8) \]

### 4.7 Numerical experiments for time dependent model for 1D

We, as an example, have taken 1D signals $u(x) = \exp(0.1x)$, $|\sin(x)|$, $x \in [0, 0.25]$ and $u(x) = |\sin(x) + \cos(x)|$, $x \in [0, 0.25]$ given in figures 4.7.1(a) and 4.7.1(b) respectively. When Gaussian white noise is added to them, we get noisy signals.
In our test, we will use the signal to noise ratio (SNR) of the signal $u$ to measure the level of noise, defined as

$$\text{SNR} = \frac{||u - \bar{u}||_{L^2}}{\sigma},$$  

(4.7.1)

where $\bar{u}$ is the mean of the signal $u$, i.e., the ratio of the standard deviation of the signal over the standard deviation of the noise.

The standard deviation of noisy signals (given in Figures 4.7.1(c) and 4.7.1(d)) are approximately $\sigma \approx 0.5$ and $\sigma \approx 0.04$ respectively whereas their SNR are 0.99 and 0.95 respectively.

We use $\beta = \sigma$ ($\sigma$ is the standard deviation of the noise) and the Langrange multiplier $\lambda = 0.005$ [46]. Figures 4.7.1(e) and 4.7.1(f) represent the denoised signals after 80 iterations with SNR $\approx 1.1$ and 1.12 respectively.

We have performed many other experiments on 1D signals obtaining similar results.

---

Figure 4.7.1: (a)-(b) Original signals; (c)-(d) corresponding noisy signals; (e)-(f) corresponding denoised signals.
4.8 Numerical experiments for time dependent model for 2D

In this Section, we have used three gray scale images, Goldhill (256 × 256), Rice (256 × 256) and Boat (512 × 512) shown in Figure 4.8.1 for our denoising experiments.

When Gaussian white noise with mean zero and variance $\sigma^2$ is added to the original images, we get noisy images. In our experiment, we have considered the images corrupted with different levels of Gaussian noise. Figure 4.8.2(a)-(c), Figure 4.8.3(a)-(c) and Figure 4.8.4(a)-(c) contain noisy images with different levels of Gaussian noise. The results obtained by using models (4.5.6) and (4.5.7) are shown in Figure 4.8.2-4.8.4 and Tables 1, 2 and 3. We have taken Lagrange multiplier $\lambda = 0.85$ as was used in references [15] and [17]. We can choose $\beta = 10^{-32}$ [17], the smallest positive machine number.

The values of PSNR obtained using model (4.5.7) given in Tables 1, 2 and 3 are larger than that of using model (4.5.6) at the same iteration number. Thus based on PSNR values and also on human perception, we conclude that the model (4.5.7) gives better denoised images than that of model (4.5.6).

![Figure 4.8.1: Original Test Images used for different experiments (a) Goldhill: 256×256, (b) Rice: 256 × 256 and (c) Boat: 512 × 512.](image-url)
Figure 4.8.2: (top row) Noisy Goldhill images with different levels of Gaussian noise (a)-(c), $\sigma^2 = 0.06, 0.08, 0.10$, respectively; (second row) (d)-(f) corresponding denoised images by model (4.5.6); (third row) (g)-(i) by model (4.5.7).
Figure 4.8.3: (top row) Noisy Rice images with different levels of Gaussian noise (a)-(c), $\sigma^2 = 0.06$, 0.08, 0.10, respectively; (second row) (d)-(f) corresponding denoised images by model (4.5.6); (third row) (g)-(i) by model (4.5.7).
Figure 4.8.4: (top row) Noisy Boat images with different levels of Gaussian noise (a)-(c), $\sigma^2 = 0.06, 0.08, 0.10$, respectively; (second row) (d)-(f) corresponding denoised images by model (4.5.6); (third row) (g)-(i) by model (4.5.7).
Table 4.8.1: Results obtained by using models (4.5.6) and (4.5.7) applied to the images in Figure 4.8.2 with three different levels of Gaussian noise ($\sigma^2 = 0.06, 0.08$ and $0.10$).

<table>
<thead>
<tr>
<th>Images (Noisy Images)</th>
<th>PSNR</th>
<th>Images (4.5.6)</th>
<th>PSNR</th>
<th>Images (4.5.7)</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fig. 4.8.2(a)</td>
<td>13.18</td>
<td>Fig. 4.8.2(d)</td>
<td>18.79</td>
<td>Fig. 4.8.2(g)</td>
<td>19.30</td>
</tr>
<tr>
<td>Fig. 4.8.2(b)</td>
<td>12.23</td>
<td>Fig. 4.8.2(e)</td>
<td>17.43</td>
<td>Fig. 4.8.2(h)</td>
<td>18.06</td>
</tr>
<tr>
<td>Fig. 4.8.2(c)</td>
<td>11.52</td>
<td>Fig. 4.8.2(f)</td>
<td>16.35</td>
<td>Fig. 4.8.2(i)</td>
<td>17.10</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>No. of 5</td>
<td>-</td>
<td>No. of 5</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 4.8.2: Results obtained by using models (4.5.6) and (4.5.7) applied to the images in Figure 4.8.3 with three different levels of Gaussian noise ($\sigma^2 = 0.06, 0.08$ and $0.10$).

<table>
<thead>
<tr>
<th>Images (Noisy Images)</th>
<th>PSNR</th>
<th>Images (4.5.6)</th>
<th>PSNR</th>
<th>Images (4.5.7)</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fig. 4.8.3(a)</td>
<td>13.36</td>
<td>Fig. 4.8.3(d)</td>
<td>19.10</td>
<td>Fig. 4.8.3(g)</td>
<td>19.39</td>
</tr>
<tr>
<td>Fig. 4.8.3(b)</td>
<td>12.38</td>
<td>Fig. 4.8.3(e)</td>
<td>17.85</td>
<td>Fig. 4.8.3(h)</td>
<td>18.26</td>
</tr>
<tr>
<td>Fig. 4.8.3(c)</td>
<td>11.64</td>
<td>Fig. 4.8.3(f)</td>
<td>16.86</td>
<td>Fig. 4.8.3(i)</td>
<td>17.38</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>No. of 5</td>
<td>-</td>
<td>No. of 5</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 4.8.3: Results obtained by using models (4.5.6) and (4.5.7) applied to the images in Figure 4.8.4 with three different levels of Gaussian noise ($\sigma^2 = 0.06, 0.08$ and $0.10$).

<table>
<thead>
<tr>
<th>Images (Noisy Images)</th>
<th>PSNR</th>
<th>Images (4.5.6)</th>
<th>PSNR</th>
<th>Images (4.5.7)</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fig. 4.8.4(a)</td>
<td>12.96</td>
<td>Fig. 4.8.4(d)</td>
<td>17.07</td>
<td>Fig. 4.8.4(g)</td>
<td>17.50</td>
</tr>
<tr>
<td>Fig. 4.8.4(b)</td>
<td>11.97</td>
<td>Fig. 4.8.4(e)</td>
<td>15.75</td>
<td>Fig. 4.8.4(h)</td>
<td>16.28</td>
</tr>
<tr>
<td>Fig. 4.8.4(c)</td>
<td>11.27</td>
<td>Fig. 4.8.4(f)</td>
<td>14.73</td>
<td>Fig. 4.8.4(i)</td>
<td>15.33</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>No. of 5</td>
<td>-</td>
<td>No. of 5</td>
<td>-</td>
</tr>
</tbody>
</table>
4.9 Conclusion

In this Chapter, we have presented a second order PDE based new nonlinear diffusion model (4.2.4) and a new time dependent model (4.5.7) to solve the nonlinear total variation problem for image denoising in 2D. The main idea is to apply a priori smoothness on the solution images. The model (4.2.4) gives larger PSNR values than that of model (4.2.3) even at relatively small iteration numbers and model (4.5.7) gives larger PSNR values than that of model (4.5.6), at the same iteration numbers. Besides, a new time dependent model (4.6.4) to solve the signal denoising in 1D has also been given.