Image Denoising Algorithms Based on the Dual Formulation of Total Variation

Master thesis in applied mathematics

Christoffer A. Elo

Department of Mathematics
University of Bergen

April 30, 2009

Forord

I min mastergrad har jeg studert bildebehandling med retning innen støyfjerning. Jeg vil gjerne takke min hovedveileder Alexander Malyshev, som introduserte meg til dette feltet og har gitt meg verdifull hjelp underveis. Jeg vil også takke min biveilder Talal Rahman som alltid har hatt svar på mine spørsmål.

En takk går også til Sveinung Fjær og Tor Harald Sandve for all hjelp og faglige diskusjoner.

Avslutningsvis takker jeg min samboer Eldbjørg, som alltid har gitt meg støtte underveis.

<div align="right">

Bergen, April 30, 2009
Christoffer A. Elo

</div>

# Contents

Chapter 1

Introduction

Image processing with the aid of computers has been a prominent research subject for the last decades. It started with basic statistic methods early in the seventies with applications for medical imaging, satellite imagery and photo enhancement. Real-time processing and more advanced methods based on partial differential equations (PDEs) started to evolve in the eighties, along with a better data performance and computational efficiency. With todays modern computers and dedicated hardware for image processing, the applications for image processing are still growing and creating more complicated problems that are yet to be solved.

Among the various image processing applications there is a strong need for mathematics. The image processing field covers a lot of different areas in mathematics, and solving image processing applications, using e.g nonlinear PDEs, requires the study of function spaces in order to get meaningful physical solutions. To be able to arrive at the nonlinear PDE equations, different mathematical frameworks may be needed: optimization techniques, functional analysis, convex analysis, variational methods, statistical analysis, parameter estimation, etc. As one can see, image processing based on PDE can be very sophisticated, and when a nonlinear PDE is established, there is a crucial need for fast computational algorithms. This includes important fields such as: numerical linear algebra, numerical analysis, finite difference/elements, multilevel methods, fourier analysis, approximation theory, etc.

The different image processing applications of interest, that are based on PDEs, include image segmentation [32, 30, 18], image registration [7, 22], image in-painting[3, 28, 38] and image denoising [33, 36, 40, 10, 9, 11]. The latter will be studied in this thesis. For further introduction to the different applications for image processing based on PDEs, see the books on the topic [2, 17] and references therein. Other methods are of cource also possible, and statistic methods and wavelet theory have shown to produce promising results, see the book [17] for furher introduction to other ways of solving applications from image processing.

The image denoising problems that are solved in this thesis uses techniques that aim to minimize a given energy functional, that describes the solutions to the problem. These problems are typically *ill-posed* inverse problems, i.e we observe the input image, and the main challange is to smooth out the noise while preserving the main features of the true image. The observed image is normally

given by an input image, which is composed by the right-handed terms:

input image   =   blurring   *   true image   +   noise
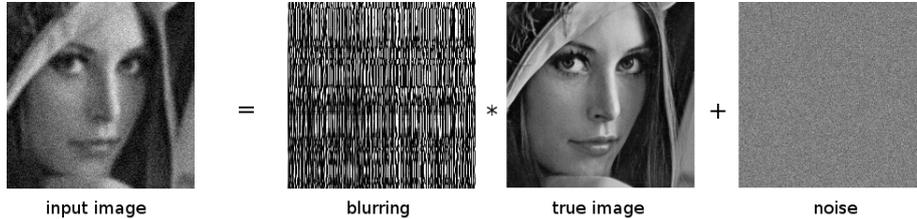
The above illustration shows that finding the true image requires knowing the blurring kernel and the noise data. This is not easily solved, as we only have one equation and three unknown variables. The kernel operator is often known, e.g the blurring kernel, and this can be approximated. But the noise is often only given by some statistical estimates, such that the noise is zero-mean white Gaussian noise of an estimated variance. By the nature of this problem we can only find an approximated solution.

To find any meaningful solutions to the problem, there is a need for additional assumptions. The variational approach is to regularize the solution, and in the early nineties, the total variational was proposed by the authors Rudin, Osher and Fatemi (ROF model). This was a huge breakthrough, since total variation preserves interesting features in the solution, such as sharp edges that are important to recover.

The thesis is arranged in the following way. First we introduce a brief overview of the mathematical framework, this includes the function spaces, differential operators and how to discretize the nonlinear equations that arise from the functional minimization.

The next chapter will give a brief introduction on how to restore a noisy image based on partial differential equations. These PDE methods evolve in time and depend on the the derivatives of different orders. A famous example is the heat equation, that smoothes the image by diffusion. It will be necessary to introduce nonlinearity, in order to keep the edges in the image. Based on the difficulties from the PDEs, the minimization of an edge preserving energy functional is given. The classical ROF model that uses the total variation regularization, with a L^2 fidelity term, is given with theoretical results such as existence and uniqueness. The corresponding Euler equation, that minimizes the energy functional, is then derived into a set of nonlinear partial differential equations. An overview of the drawbacks that total variation based restoration introduces into the solution, is also precented. Motivated by the numerical difficluties for the nonlinear term, a dual formulation is solved by the Chambolle iteration proposed in [9], that will improve the speed of convergence.

The second-to-last chapter is the main contribution in this thesis. This chapter is devoted to two models that reduce the staircase effect: the Fourth-Order model [10, 12, 26] and the TV-Stokes model [38, 34, 25, 21]. The latter is a two-step approach that decouples the Fourth-Order model into two second-order problems, since higher-order methods tend to have computational difficulties due to very large conditioning. The first step of the TV-Stokes model smoothes the tangential vector field with the condition that the vector field is divergence free. Once the smoothed tangential vector field is obtained, the second step reconstructs the image by fitting it to the normal vector field. Both the models

are derived in the dual formulation to improve the convergence speed, and solving the dual formulation is the main novelty in this thesis, see [21]. The discrete algorithm for the dual TV-Stokes exploits the fast Fourier transformation to solve the divergence free condition.

Finally, the last chapter is devoted to numerical experiments. This chapter discusses the numerical behaviour of the proposed dual TV-Stokes model and the choice for the smoothening parameter. Restored images are shown for the different denoising algorithms, aswell as difference images and contour plots to illustrate the denoising quality. The fast convergence for the dual TV-Stokes is compared to the very slow primal TV-Stokes [34, 25], that is solved by explicit schemes.

1.1 Mathematical preliminaries

This section will briefly cover some of the necessarily tools that are needed for modern image processing, based on partial differential equations. The following text is a standard topic in functional analysis, and if terms like function spaces are unknown to the reader, (s)he should refer to books on the topic, see for instance [37, 19]. Function spaces are very useful in image processing since they possess different features that represent an image. Lets start by recalling the Euclidean space followed by the inner product and some properties.

Definition 1 (Vector space). *Euclidean $n-space$ $\mathbb{R}^n$ is the set of all the $n-tuples$ of real numbers. If x denotes a vector in $\mathbb{R}^n$ then it is written as*

$$x = [x_1, \ldots, x_n]^T \tag{1.1}$$

with the length, called the Euclidean norm, defined as

$$\sqrt{(x,x)} = \|x\| = \sqrt{x_1^2 + \ldots + x_n^2} \tag{1.2}$$

Definition 2 (Inner product space). *The quantity $(x,y) = \sum_{i=1}^n x^i y^i$ is called the inner product. Let x, x_1, x_2 and y, y_1, y_2 be vectors in $\mathbb{R}^n$ then the following properties must hold*

1. $(x,y) = (y,x)$ *(symmetry)*

2. $(ax,y) = (x,ay) = a(x,y)$ *(bilinearity)*
   $(x_1 + x_2, y) = (x_1, y) + (x_2, y)$
   $(x, y_1 + y_2) = (x, y_1) + (x, y_2)$

3. $(x,x) \geq 0$, *and* $(x,x) = 0$ *iff* $x = 0$ *(positive definiteness)*

Proposition 1 (Properties of the norm). *If $x, y \in \mathbb{R}^n$ and $a \in \mathbb{R}$*

1. $\|x\| > 0$ *if* $x \neq 0$

2. $|(x,y)| \leq \|x\|\|y\|$

3. $\|x + y\| \leq \|x\| + \|y\|$

4. $\|ax\| = |a|\|x\|$

Definition 3 (Hilbert space). *A Hilbert space is a complete inner-product space.*

The L^p space is a Hilbert space with the following norm

$$\|f\|_p = \left(\int_\Omega |f|^p \, \mathrm{d}x \right)^{1/p}. \tag{1.3}$$

Example 1 (L^2-space). *L^2 is a Hilbert space with the following inner-product*

$$(f, g) = \int_\Omega fg \, \mathrm{d}x. \tag{1.4}$$

1.1.1 Linear mappings

Linear mappings are the tools for manipulating images. They may be described as transformations, e.g to rotate an image by 90 degrees. A more useful transformation in this thesis is the differentiation map, which will be the key for minimization later on.

Consider two vector spaces X and Y and a mapping $f : X \to Y$, then this mapping is linear if

$$f(\alpha x + \beta y) = \alpha f(x) + \beta f(y), \qquad \forall x, y \in X \text{ and } \alpha, \beta \in \mathbb{R} \tag{1.5}$$

If $A \subset X$ and $B \subset Y$ then the image and inverse image is defined by

$$f(A) = \{f(x) : u \in A\}, \qquad f^{-1}(B) = \{x : f(x) \in B\} \tag{1.6}$$

The mapping is called a functional if Y is the scalar field and an operator if Y is a vector space.

Definition 4 (Differentiation operator). *A operator $f : R^n \to R^m$ is differentiable at $a \in R^n$ if there is a linear transformation $\lambda : R^n \to R^m$ such that*

$$\lim_{h \to 0} \frac{\|f(x + h) - f(x) - \lambda(h)\|}{\|h\|} = 0 \tag{1.7}$$

The linear transformation $\lambda = (\nabla f(x), h)$ is denoted by $Df(x)$ and is called the derivative of f at x.

By the above definition for $f : R \to R^n$ and for any $h \neq 0$,

$$f'(x; h) = (\nabla f(x), h), \quad \forall h \tag{1.8}$$

In particular, for $j = 1, \ldots, n$

$$(\nabla f(x), e_j) = \lim_{h \to 0} \frac{f(x + he_j) - f(x)}{h} = \frac{\partial f}{\partial \xi_j}(x) \tag{1.9}$$

e_j is the vector of the jth row in a $n \times n$-identity matrix, and ξ_j is the jth component of $f(x)$ so that

$$\nabla f(x) = \left(\frac{\partial f}{\partial \xi_1}(x), \ldots, \frac{\partial f}{\partial \xi_n}(x) \right) \tag{1.10}$$

which is called the *gradient* of f. The second-order derivative of f is called the *Hessian matrix* and is denoted by D^2. The trace of this matrix gives the *Laplacian*

$$\Delta f(x) = \text{tr}(D^2 f(x)) = \sum_{i=1}^{n} \frac{\partial^2 f}{\partial \xi_i^2} \tag{1.11}$$

If $f : R^n \to R^n$ then the $n \times n$ matrix of $Df(x)$ is called the *Jacobian matrix*. The trace of the Jacobian matrix is called the divergence

$$\text{div} \, f(x) = \text{tr}(Df(x)) = \sum_{i=1}^{n} \frac{\partial f_i}{\partial \xi_i}. \tag{1.12}$$

Note that the Laplacian is also given by $\text{div} \, \nabla f(x) = \Delta f(x)$.

It is important to introduce a space that possesses derivatives of certain order in order to search for meaningful solutions when solving problems based on differential equations.

Definition 5 (The function space $H^1(\Omega)$ is called a Sobolev space)**.**

$$H^1(\Omega) = \left\{ f \in L^2 : Df \in L^2(\Omega) \right\} \tag{1.13}$$

In the above definition the Df is a distributional derivative such that

$$\int_{\Omega} Df\phi \, dx = - \int_{\Omega} f D\phi \, dx, \quad \phi \in C_0^1(\Omega) \tag{1.14}$$

Theorem 1 (The Sobolev space $H^1(\Omega)$ is complete)**.**

If $f \in L^1(\Omega)$ and $Df \in L^1(\Omega)$ in a distributional sense, then the finite total variation is given by

Definition 6 (Total variation of f)**.** *If $f \in L^1(\Omega)$ and $Df \in L^1(\Omega)$*

$$\int_{\Omega} |Df| \, dx = \sup \left\{ \int_{\Omega} f \text{div} \, g \, dx : g = (g_1, \ldots, g_n) \in C_0^1(\Omega)^n, |g| \le 1 \right\} \tag{1.15}$$

$C_0^1(\Omega)$ is the set of continuously differentiable function with compact support(functions that vanish on the boundary) in Ω.

Thus, total variation can be seen as a measure of the amount of oscillation in the function f.

Functions that have a finite value for the $\int_{\Omega} |Df| \, dx$ norm are said to have bounded variation. The space of all these functions is called the space of bounded variation and has the following definition

Definition 7 ($BV(\Omega)$ the space of function of bounded variation)**.**

$$BV(\Omega) = \left\{ f \in L^1(\Omega) : \int_{\Omega} |Df| \, dx < +\infty \right\} \tag{1.16}$$

If $f \in C^1(\Omega)$, then with integration by parts in equation (1.15) yields

$$\int_{\Omega} f \text{div} \, g \, dx = - \int_{\Omega} \sum_{i=1}^{n} \frac{\partial f}{\partial x_i} g \, dx. \tag{1.17}$$

Thus the adjoint of the gradient is the negative divergence

$$(\nabla f, g) = (f, -\text{div} \, g). \tag{1.18}$$

1.1.2 Optimization and Convexity

Convexity plays an important role in optimization. It gives the problem a sufficient condition for uniqueness. The references [20, 35] give a good introduction on optimization and convex problems.

In an Euclidean space, a set is convex if every point on a straight line lies within the set. This gives the mathematical definition of a convex set

Definition 8 (Convex set). *A subset V of a vector space is convex iff for every pair of elements $u, v \in V$ their closed segment $[u, v]$ is also in V.*

A strict convex space is e.g the $\| \cdot \|_2$ unit ball, while the unit balls of $\| \cdot \|_\infty$ and $\| \cdot \|_1$ are not strictly convex, since they contain a line segment on the boundary, thus the name borderline convex.

Definition 9 (Convex functional). *Let f be a functional from R^n to $[-\infty, \infty]$, then f is convex if and only if*

$$f(\lambda u + (1 - \lambda)v) \leq \lambda f(u) + (1 - \lambda)f(v) \quad \forall \lambda \in [0, 1] \tag{1.19}$$

if the inequality is strict, then f is strictly convex.

Note that a sum of convex functions is still convex, and a sum of a strictly convex function and a convex function is strictly convex.

Definition 10 (Epigraph of a function). *It is the set of points of V which lies above the graph of f.*

$$\{(x, \lambda : x \in V, \lambda \in \mathbb{R}, \lambda \geq f(x)\} \tag{1.20}$$

It can further be shown that f is convex iff its epigraph is convex.

1.1.3 Discretization

Discretization of our problems is really the practical part of the functional analysis, the following books give a good introduction on numerical linear algebra and analysis, [39, 24, 20]. Numerical algorithms and methods are used to find solutions to the mathematical problems that are typically described in the continuous setting. It is therefore natural to introduce a discrete framework to place images. The image is described in a two dimensional space X with a fixed size $N \times N$. We denote X by using the Euclidean space $\mathbb{R}^{N \times N}$ and Y will express the vector $X \times X$. The space X will be equipped with the Euclidean norm from (1.2)

$$\|x\|_X = \left(\sum_{1 \leq i,j \leq N} |x_{i,j}|^2 \right)^{1/2} \tag{1.21}$$

and Y has the following Euclidean norm

$$\|y\|_Y = \left(\sum_{1 \leq i,j \leq N} \left| y_{i,j}^1 + y_{i,j}^2 \right|^2 \right)^{1/2} \tag{1.22}$$

where $y = [y^1, y^2]^T$.

An easy way to implement differential operators, as discussed in the previous section, is to replace the operators with finite difference, i.e replace $h^+ \to 0$ to a fixed parameter $h > 0$, giving the *forward difference* in the one-dimentional case where $f \in \mathbb{R}^N$

$$D_+^h f(x) = \frac{(f+h) - f(x)}{h} \tag{1.23}$$

approaching the differential from $h^- \to 0$ gives the *backward difference*

$$D_-^h f(x) = \frac{f(x) - f(x-h)}{h}. \tag{1.24}$$

Another way to approximate the derivative would be to evaluate both sides, giving the *central difference*

$$D_0^h f(x) = \frac{D_+^h f(x) + D_-^h f(x)}{2} = \frac{f(x+h) - f(x-h)}{2h} \tag{1.25}$$

To introduce a discrete version of the gradient defined in (1.10), a natural choice is to use the forward difference in both directions, if $d \in X$ then the gradient ∇d is a vector in $Y = X \times X$ and is given by

$$(\nabla d)_{i,j} = \begin{bmatrix} (\nabla d)_{i,j}^1 \\ (\nabla d)_{i,j}^2 \end{bmatrix} \tag{1.26}$$

where

$$(\nabla d)_{i,j}^1 = D_+^{h_x} d(x,y), \quad \text{if } i < N \tag{1.27}$$

$$(\nabla d)_{i,j}^2 = D_+^{h_y} d(x,y), \quad \text{if } j < N \tag{1.28}$$

It is useful to introduce a differential matrix B that approximates the differential operator. This useful since the notation will not be so cluttered as with finite differences.

$$(\nabla d)_{i,j}^1 = d B_x^T \tag{1.29}$$

and

$$(\nabla d)_{i,j}^2 = B_y d \tag{1.30}$$

where $B_x(B_y)$ stands for differentiation in the x (resp. y) direction. The forward differential $(N) \times (N-1)$ matrix takes the following form

$$B = \frac{1}{h} \begin{bmatrix} -1 & 1 & & \\ & -1 & 1 & \\ & & \ddots & \ddots \\ & & & -1 & 1 \end{bmatrix} \tag{1.31}$$

and the adjoint of B is defined as the backward difference $(N-1) \times (N)$ matrix

$$B^* = -B^T = \frac{1}{h} \begin{bmatrix} 1 & & & \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix} \tag{1.32}$$

The discrete version of the total variation given in equation (1.15) is then easily followed by

$$J(d) = \sum_{1 \le i,j \le N} |(\nabla d)_{i,j}|. \tag{1.33}$$

The discrete version of the divergence operator can be defined as an analog in the continuous setting in equation (1.18), and thus for every $p \in Y$ and $d \in X$, $(-\operatorname{div} p, d)_X = (p, \nabla d)_Y$. We will use the backward difference to get symmetry for the discrete divergence operator, hence

$$(\operatorname{div} p)_{i,j} = -p_1 B_x - B_y^T p_2. \tag{1.34}$$

Staggered grid

As one notices from the Taylor expansion of the centred approximation given in (1.25), this yields second order accuracy

$$D_0^h f(x) = \frac{f(x+h) - f(x-h)}{2h} = Df(x) + \mathcal{O}(h^2). \tag{1.35}$$

However, this centred approximation evaluates the difference across two cells which can lead to undesirable approximations for short wavelengths (high frequency). A better way to approximate first-order derivatives is to consider *staggered grid*, or half-length approximation $x_{i+\frac{1}{2}} = x_i + \frac{h}{2}$

$$\frac{f(x+h) - f(x)}{h} = Df\left(x + \frac{h}{2}\right) + \mathcal{O}(h^2). \tag{1.36}$$

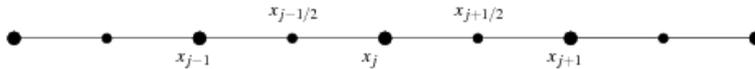This is just a matter of re-defining the grid as shown in figure 1.1.

Figure 1.1: Staggered grid in one dimension

1.1.4 Definition of an image

A digital image can be defined as a function, d, that is bounded and piecewise smooth on an open subset $\Omega \subset \mathbb{R}^2$ where Ω usually is square. $d(x,y)$ represents a pixel at the space coordinates (x, y). Is also useful to see the function $d \in X$ as a matrix d_{ij} where each component in the matrix is the finite gray-scale value, $d_{ij} = d(x = i, y = j)$, varying in the range from 0 (black) to 255 (white). In practice, the gray-values are usually normalized into $[0, 1]$.

To obtain a digital image from the continuous world, there is a need for discretization, also known as sampling and quantization. This is done by superimposing a regular grid on a continuous image, and each pixel will have the value of the average value from the continuous image. The regular grid will have a size (resolution) that is important for the quality of the obtained digital image. Low resolution will give blocky images, while the computation with these images will be fast, since the dimension of the matrix will be lower. Higher resolution of the discretized image will give a better approximation of

the continuous image. The problem with discretization of a continuous image is the natural superimposed noise due to defects in the sensors, transmission problems, interference, etc. This can be mathematically described as the observed image d_0 that consists of the original image d perturbed by an additive unknown random noise variable η,

$$d_0 = d + \eta. \tag{1.37}$$

The problem is given by the observed image d_0 and the assumption that η is Gaussian white noise. The values $\eta_{i,j}$ are independent random values, each with Gaussian distribution with zero-mean and an estimated variance $\|\eta\|_2^2 \approx \sigma^2$. The problem is to reconstruct d from (1.37) which is an inverse problem, and one can easily see that the problem is *ill-posed*, meaning that one could only get an approximated solution for the given problem.

Another problem that can occur is e.g a blur in the observed image created by e.g incorrect lens adjustment, this can be mathematically described as

$$d_0 = Rd + \eta \tag{1.38}$$

where R is a convolution. In the rest of this thesis, we will consider the convolution as an identical map $R = I$, i.e pure noise as described in (1.37).

There are many features in an image such as edges (discontinuities), flat areas (zero gradient), smooth areas and textures (highly oscillating patterns). A good denoising algorithm should try to keep these features in the image, while effectively removing the noise.

Chapter 2

The ROF model

This chapter will cover some of the pioneering methods that are based on variational approaches. The first section will give the primal formulation of the ROF model [36, 40, 10] and the last section will present the dual formulation [16, 8, 9, 13] which increases the computational efficiency.

We start by motivating for finding a solution to (1.37), a good start is to assume a smoothness of the solution. This is a well known processes called *Tikhonov regularization* of the inverse problem. The idea is to build a functional with a smoothening term and a data fitting-term and then try to minimize this functional

$$\inf_d f(d) = \inf_d \int_\Omega |\nabla d|^2 \, dx + \lambda \int_\Omega |d - d_0|^2 \, dx. \tag{2.1}$$

The minimizer $\inf_d f(d)$ will have a solution in the Sobolev space $H^1(\Omega)$, and it is unique since $f(d)$ is strictly convex.

The Euler equation, see for instance [20, Theorem 7.2-4], for this minimization is the following

$$\begin{cases} -\Delta d + \lambda(d - d_0) = 0 & \text{in } \Omega \\ \frac{\partial d}{\partial n} = 0 & \text{in } \partial\Omega. \end{cases} \tag{2.2}$$

An easy way to solve the above equation is to march forward in time

$$\frac{\partial d}{\partial t} = \Delta d - \lambda(d - d_0) \quad \text{in } \Omega, \tag{2.3}$$

this corresponds to the standard heat equation with a *force* term. The solution is obtained when the iteration is stationary $\frac{\partial d}{\partial t} \to 0$. However, the resulting image does not preserve interesting features in the image, since the method smoothes out the edges and overblurrs the image, as illustrated in figure 2.1. This is a huge drawback, as the solution d is $C^\infty(\Omega)$ for $\lambda = 0$, meaning that the heat equation has infinite speed of propagation.

Over the two last decades there has been a lot of research in image processing to restore the image d in (1.37), and one quite effective way is to introduce a nonlinear diffusion to preserve the edges. Perona and Malik introduced the diffusion coefficient $g(|\nabla d|)$ in their classical paper [33]

$$\frac{\partial d}{\partial t} = \text{div} \left(g(|\nabla d|)\nabla d \right) \quad \text{in } \Omega, \tag{2.4}$$

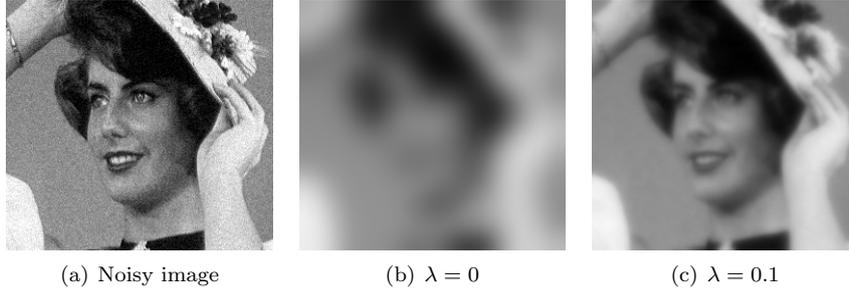| (a) Noisy image | (b) $\lambda = 0$ | (c) $\lambda = 0.1$ |

Figure 2.1: Heat equation with and without a force term

where g will be small when the gradient is large (detecting an edge) and large when g detects smooth areas.

2.1 Primal Formulation

Another and more popular approach is to use variational formulation, which has been successful and is still one of the most active areas in image processing. Variational formulation normally means minimizing an object functional subject to a fidelity term, and with image denoising this can be formulated as

$$\min_d J(d) \quad \text{subject to} \quad \|d - d_0\|_2^2 = \sigma^2, \tag{2.5}$$

where J will be the regularity term and is bounded below (lower semi-continuity). The last term ensures that the given image d_0 is close to d and this term should be proportional to the noise-level σ^2

$$\|d^* - d_0\|_2^2 \approx \sigma^2, \tag{2.6}$$

where d^* denotes an approximate solution to (2.5). σ is supposed to be a known estimate of the variance to the error data, i.e $\|\eta\|_2 = \sigma$.

Choosing $J(d)$ equal to $\int_\Omega |\nabla d|^2 \, \mathrm{d}x$ (H_2^1 semi-norm) will smooth the image very effectively including the discontinuous lines in the image which results in a blurry restored image. To preserve the edges, Rudin et al. suggested in [36] to lower the H_2^1 to the H_1^1 (weak derivatives in L^1), which defines the functions of bounded variation $BV(\Omega)$, recall the definition in (1.16). Thus a function in $BV(\Omega)$ is a function defined in $L^1(\Omega)$, whose distributional derivatives are a finite total variation over Ω. The total variation (1.15) will be restated here

$$\|\nabla d\|_1 = \sup \left\{ \int_\Omega d \operatorname{div} \xi(x) \, \mathrm{d}x : \xi \in C_0^1(\Omega, \mathbb{R}^2), |\xi(x)| \leq 1 \forall x \in \Omega \right\}. \tag{2.7}$$

The above functional does not penalize the discontinuities in d, and therefore recovers the edges from the original image. Integration by parts yields

$$\min_d \int_\Omega |\nabla d| \, \mathrm{d}x \quad \text{subject to} \quad \|d - d_0\|_2^2 = \sigma^2, \tag{2.8}$$

which is the famous formulation that Rudin, Osher and Fatemi published in 1992. This model is of great value to image processing, since images tend to have

discontinuities. Another important task in image processing, which is based on the similar model, is when an image is damaged by missing information. The process is then to fill the missing parts in the image with information from surrounding areas. This is called image in-painting, and pioneering works can be found in [3, 28].

Chambolle and Lions showed in [10] that (2.8) is naturally linked with the following unconstrained problem

$$\min_d f(u) = \min_d \int_\Omega |\nabla d|\, \mathrm{d}x + \frac{\lambda}{2}\|d - d_0\|_2^2, \tag{2.9}$$

where $f(d)$ is the objective function and λ is a non-negative Lagrange multiplier. This formulation is known as the TV-L^2 model, and is the convexification of (2.8) when the constraint is equal to $\|d - d_0\|_2^2 \le \sigma^2$.

The λ parameter balances between two terms $\lambda \to \infty$ and $\lambda \to 0$. If $\lambda \to \infty$, the solution is given by $d = d_0$, which is obviously not a good result. If $\lambda \to 0$, then the solution is zero, or any constant, since $\nabla d = 0$ and does not fit the image with respect to d_0. The optimal λ should take advantage of σ such that the constraint $\|d - d_0\| = \sigma$ is fulfilled.

The following results show that the ROF model has a unique solution, which is important when searching for solutions. Uniqueness and existence among other theoretical results concerning bounded variation, can be found in [10, 1].

Proposition 2 ($f(d)$ has a unique solution.)**.** *Suppose that f is coercive and that f is lower semi-continuous, then the problem has a solution. The solution is unique if f is strictly convex.*

Proof. Let d_n be a minimizing sequence of (2.9) such that

$$f(d_n) \to \inf_{d \in L^2(\Omega)} f(d) = \alpha, \quad n \ge 1. \tag{2.10}$$

Suppose $\|d_0\|_{L^2} = 0$ and that all the constrains are satisfied by d_n. $f(d)$ is coercive: $\lim f(d) \to +\infty$ for $\|d\| \to \infty$. Due to Poincaré inequalities, d_n is bounded in $BV(\Omega)$, see e.g [1, Theorem 2.5]. Thus, an extracted sequence from d_n converges weakly to some $\bar{d} \in L^2(\Omega)$. f is lower semi-continuous on L^2, hence

$$f(\bar{d}) \le \lim_{n_i \to \infty} f(d_{n_i}) = \alpha, \tag{2.11}$$

where $\bar{d}$ is a solution of (2.8) and $\alpha \ne -\infty$.

If f is strictly convex, two solutions is impossible due to

$$f\left(\frac{d_1 + d_2}{2}\right) < \frac{1}{2}\left(f(d_1) + f(d_2)\right) = \alpha. \tag{2.12}$$

$\square$

To derive a minimization of $f(d)$ one introduces an admissible variation ψ of d, i.e $\psi \in H^1(\Omega)$, which has no constraints

$$\phi : t \in I \to \phi(t) = f(d + t\psi). \tag{2.13}$$

ϕ will be minimal at $t = 0$ and $\phi'(0) = f'(d)\psi$

$$0 \ge \lim_{t \to 0^-} \frac{\phi(t) - \phi(0)}{t} = \phi'(0) = \lim_{t \to 0^+} \frac{\phi(t) - \phi(0)}{t} \ge 0, \tag{2.14}$$

14

which shows that $f'(d)\psi = 0$, and the ψ is arbitrary, therefore $f'(d) = 0$. This is known as the Euler equation, see [20] for further introduction on the topic.

The Euler equation of (2.9) is the following

$$\phi(t) = f(d + t\psi) = \int_\Omega |\nabla d + t\nabla\psi| + \frac{\lambda}{2}|d - d_0 + t\psi|^2 \,\mathrm{d}x$$

$$= \int_\Omega |\nabla d + t\nabla\psi| + \frac{\lambda}{2}|d - d_0|^2 + \lambda t\psi(d - d_0) + \frac{\lambda}{2}t^2\psi^2 \,\mathrm{d}x. \quad (2.15)$$

Using the fact that $\phi'(t)$ will give minimum at $\phi'(0) = 0$

$$\phi'(t) = \int_\Omega \left(\frac{\nabla d + t\nabla\psi}{|\nabla d + t\nabla\psi|}, \nabla\psi \right) + \lambda\psi(d - d_0) + \lambda t\psi^2 \,\mathrm{d}x \quad (2.16)$$

$$\psi'(0) = \int_\Omega \left(\frac{\nabla d}{|\nabla d|}, \nabla\psi \right) + \psi\lambda(d - d_0) \,\mathrm{d}x = 0, \quad (2.17)$$

integrating by parts (strong formulation) yields

$$\int_{\partial\Omega} \psi \frac{\partial d}{\partial n} \,\mathrm{d}x + \int_\Omega \psi \left(-\mathrm{div}\, \frac{\nabla d}{|\nabla d|} + \lambda(d - d_0) \right) \,\mathrm{d}x = 0. \quad (2.18)$$

Then for any ψ, we reach the nonlinear partial differential equation with homogeneous Neumann boundary condition

$$\begin{cases} \mathrm{div}\, \frac{\nabla d}{|\nabla d|} - \lambda(d - d_0) = 0 & \text{in } \Omega \\ \frac{\partial d}{\partial n} = 0 & \text{in } \partial\Omega. \end{cases} \quad (2.19)$$

Note that for $|\nabla d| = 0$, (2.19) is not well-defined, this is a problem since the solutions of the ROF model have large flat areas, i.e where $\nabla d = 0$ in the image. One way to solve this is to regularize $f(d)$

$$\inf_d f(d)^\beta = \int_\Omega \sqrt{|\nabla d|^2 + \beta} + \frac{\lambda}{2}|d - d_0|^2 \,\mathrm{d}x, \quad (2.20)$$

where $\beta > 0$ implies that $f(d)_\beta$ is differentiable and this yields the regularized Euler equation

$$\begin{cases} g(d) = \mathrm{div}\, \frac{\nabla d}{\sqrt{|\nabla d|^2 + \beta}} - \lambda(d - d_0) = 0 & \text{in } \Omega \\ \frac{\partial d}{\partial n} = 0 & \text{in } \partial\Omega, \end{cases} \quad (2.21)$$

with homogeneous Neumann boundary condition and initial solution, given by $d = d_0$. $g(d) = 0$ is a necessary and sufficient condition for d to be a solution of the convex minimization problem (2.20).

2.1.1 Drawbacks

Total variation restoration is very good for recovering images from noise. Unfortunately it may fail in some areas to give a satisfactorily result, due to artificial effects that are introduced while performing pure total variation with a L^2 fidelity term. These drawbacks can be found in the survey paper [11], and they will be restated here.

Regularization The first disadvantage one notices about the ROF model is the β-regularization. The drawback is that the smaller β responds to more iterations before convergence, and this makes the ROF model rather slow when solving for time marching schemes. Choosing higher β gives faster convergence, but smooths the image needlessly.

Staircase It is well known that the ROF model produces staircase effect for piecewise smooth areas. In figure 2.2 it is easy to see why this happens. The figure consists of a decreasing function from $[5, 25]$, and when noise is added, the function becomes piecewise constant for the decreasing parts. When running the ROF scheme, the nondecreasing parts will be smoothed out as illustrated in the figure below.

(a) Original function (red-line) and (b) Denoised function (blue-line)
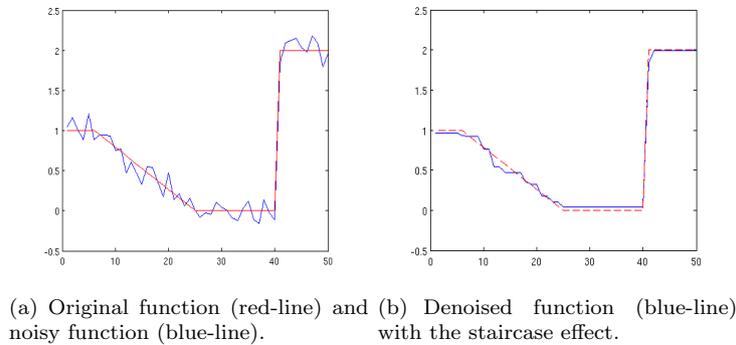noisy function (blue-line). with the staircase effect.

Figure 2.2: Illustration of staircase effect

Loss of texture Another disadvantage when using total variation based methods, is that texture is easily smoothed out during the denoising. In a recently proposed model by Osher et al. in [31], they prevent this by iterative regularization, which means running the ROF model multiple times, and each time adding the difference between the observed noisy image and the previous one, i.e $d_{i+1} = d_i + (d_0 - d_i)$. Thus, the noise (and the details!) are added back to the image.

Loss of constrast Loss of contrast is also a limitation for the ROF model. If the input image $d_0(x, y)$ is a disk, then the solution is given by $\alpha d(x, y)$, where $\alpha \in [0, 1)$, i.e the solution is never $\alpha = 1$. This disadvantage can also be seen in the one-dimensional example illustrated in figure 2.2, where the solution $f(x)$ is in the range $(0, 2)$.

One approach to this problem is given in [14], where they weaken the fidelity norm. They formulated the problem to measure the L^1-norm between the observed and denoised data. The TV-L^1 minimization problem is then given by

$$\inf_d \int_\Omega |\nabla d| + \frac{\lambda}{2}|d - d_0|\, \mathrm{d}x. \tag{2.22}$$

The above formulation is not strictly convex, which means that there is no unique global minimizer. The reason why this formulation has been studied, is

that it has some disirable advantages, e.g that the solution is contrast invariant. This means that if d is a solution for the problem (2.22), then αd is also a solution for αd_0, where d_0 is the initial input image and α is a scaling factor.

2.1.2 Numerical methods

There are various ways of solving (2.21), and in [36] the authors Rudin et al. suggested to forward the equation with artificial time to get a gradient descent scheme

$$d^{n+1} = d^n - kg(d^n), \quad n = 0, 1, 2, \ldots, \tag{2.23}$$

as $n \to \infty$. When steady-state is achieved $\frac{\partial d}{\partial t} \to 0$, the solution d^{n+1} will satisfy the unique solution of the minimizer given in (2.8). The timestep k must be chosen to satisfy the Courant-Friedrichs-Lewy (CFL) condition $k \leq c|\nabla d|$, for some constant $c > 0$. Due to this condition, the method converges rather slowly, since k must be chosen small. This is due to that $|\nabla d|$ is approximately zero in flat regions.

The main difficulty that equation (2.21) poses is the linearization of the highly nonlinear term $\operatorname{div}(\nabla d/|\nabla d|_\beta)$. Thus, one way to overcome the CFL condition is to linearise (2.21) by using the nonlinear term $1/|\nabla d|_\beta$ from the previous iteration. This was proposed in [40] and is known as lagged diffusivity fixed point iteration. The term d^{n+1} can then be solved by the following sparse system of linear equations

$$d^{n+1} = d^0 + \frac{1}{\lambda}\operatorname{div}\left(\frac{\nabla d^{n+1}}{|\nabla d^n|_\beta}\right). \tag{2.24}$$

This gives a faster iteration than (2.23), but reacts more to the β-regularization due to the highly nonlinear term. Vogel et al. also solve the above iteration with multigrid methods, but unfortunately, standard finite differences or finite elements discretization yield disappointing results when the input image d_0 is not sufficiently smooth. Thus, by achieving $\mathcal{O}(n)$ complexity, higher β-regularization is needed such that the nonlinear term is well behaved.

2.2 Dual Formulation

As seen in the previous section, a major drawback of the ROF model is that the formulation is non-differentiable in zero, due to the total variation term. To overcome further regularization, Chambolle in [9], Carter in [8] and Chan et al. in [16], studied the so called dual formulation for the ROF model. We will reduce the above presented primal formulation to the dual formulation. Lets consider the more general definition of the total variation, given in equation (2.7), and use the right-handed definition to obtain the $\inf - \sup$ problem

$$\inf_d \sup_{|p| \leq 1} h(d, p) = \inf_d \sup_{|p| \leq 1} \left\{ \int_\Omega (d, \operatorname{div} p) + \frac{1}{2\lambda}|d - d_0|^2 \, \mathrm{d}x \right\}. \tag{2.25}$$

Hence $h(\bar{d}, \bar{p})$ is a saddle point since $\bar{d}$ is the minimum of the function $h(\bar{d}, \cdot)$, and $\bar{p}$ is the maximum of the function $h(\cdot, \bar{p})$, such that

$$\sup_{|p| \leq 1} f(\bar{d}, p) = f(\bar{d}, \bar{p}) = \inf_d(d, \bar{p}). \tag{2.26}$$

Given any elements $\hat{d}$ and $\hat{p}$ gives the following inequality

$$\inf_d f(d, \hat{p}) \leq f(\hat{d}, \hat{p}) \leq \sup_{|p| \leq 1} f(\hat{d}, p), \qquad (2.27)$$

and as $\inf_d f(d, \hat{p})$ is a function of the single variable $\hat{p}$, and $\hat{d}$ is the single variable for $\sup_{|p| \leq 1} f(\hat{d}, p)$, this yields that the following inequality exists

$$\sup_{|p| \leq 1} \inf_d f(d, p) \leq f(d, p) \leq \inf_d \sup_{|p| \leq 1} f(d, p). \qquad (2.28)$$

The fact that $f(\bar{d}, \bar{p})$ is a saddle point from equation (2.26), gives that the necessary converse inequality must also hold. This is of great value for solving (2.25), since it does not matter which problem is solved first. Thus interchanging the sup and inf from equation (2.25) yields the inner minimization

$$\sup_{|p| \leq 1} \inf_d h(d, p) = \sup_{|p| \leq 1} \inf_d \int_\Omega (d, \operatorname{div} p) + \frac{1}{2\lambda} |d - d_0|^2 \, dx. \qquad (2.29)$$

Hence the equation is piecewise continuous and differentiation w.r.t d is easy

$$d = d_0 - \lambda \operatorname{div} p. \qquad (2.30)$$

The term $\lambda \operatorname{div} p = \pi_\lambda$ is the nonlinear projection that is to be solved and inserted into (2.30) instead of solving the primal problem. This projection can be solved by substituting (2.30) into (2.29) and deriving the following maximization problem

$$\sup_{|p| \leq 1} h(p) = \sup_{|p| \leq 1} \int_\Omega (d_0, \operatorname{div} p) - \frac{\lambda}{2} |\operatorname{div} p|^2 \, dx = \sup_{|p| \leq 1} \frac{1}{2\lambda} \int_\Omega |d_0|^2 - |\lambda \operatorname{div} p - d_0|^2 \, dx, \tag{2.31}$$

using $\sup \{h\} = -\inf \{-h\}$ yields the following minimization problem for the dual variable

$$\min_p \left\{ \|\operatorname{div} p - \lambda^{-1} d_0\|_2^2 : |p| \leq 1 \right\}. \qquad (2.32)$$

In one dimension p is pair of two vectors, and the above minimization is easily solved by e.g least-square methods. In our case, p is a pair of two matrices. Chambolle solved the minimization problem by considering the optimality condition, also known as the KKT conditions (due to Karush-Kuhn-Tucker, cf. [20, Theorem 9.2-3])

$$-(\nabla(\operatorname{div} p - \lambda^{-1} d_0)) + \alpha p = 0, \qquad (2.33)$$

where α is the Lagrangian multiplier associated with the constraint p that must satisfy

$$\begin{cases} \alpha > 0 \\ |p| = 1 \end{cases} \quad \text{or} \quad \begin{cases} \alpha = 0 \\ |p| < 0. \end{cases} \qquad (2.34)$$

He then did an important observation, that in either case

$$\alpha = |(\nabla(\operatorname{div} p - \lambda^{-1} d_0))|, \qquad (2.35)$$

such that the (2.35) can be substituted into equation (2.33) which yields the equation

$$(\nabla(\operatorname{div} p - \lambda^{-1} d_0)) - |(\nabla(\operatorname{div} p - \lambda^{-1} d_0))| p = 0. \qquad (2.36)$$

2.2.1 Discrete algorithm

By considering the minimization problem in a discrete framework we have the following analog to (2.32)

$$\min_{p} \left\{ \|\operatorname{div}^h p - \lambda^{-1} d_0\|_X^2 : |p_{i,j}| \leq 1, i,j = 1, \ldots N \right\}. \qquad (2.37)$$

Deriving with the same arguments as (2.33)-(2.36) and considering

$$\frac{p^{n+1} - p^n}{k} = \nabla^h \left(\operatorname{div}^h p^n - \lambda^{-1} d_0 \right) - \left| \nabla^h \left(\operatorname{div}^h p^n - \lambda^{-1} d_0 \right) \right| p^{n+1} \qquad (2.38)$$

yields a semi-implicit scheme known as Chambolle's iteration that is given in [9]

$$p^0 = 0, \quad p^{n+1} = \frac{p^n + k\nabla^h \left(\operatorname{div}^h p - \lambda^{-1} d_0 \right)}{1 + k \left| \nabla^h \left(\operatorname{div}^h p - \lambda^{-1} d_0 \right) \right|}. \qquad (2.39)$$

The iteration converges for $k \leq 1/8$ which is proven in [9]. In practice, convergence is observed by setting k equal to $1/4$. This gives a rapid convergence up to 50-100 iterations, and then the speed of convergence decays.

Chapter 3

Staircase reducing models

Section 2.1.1 pointed out some of the areas where the ROF model may suffer from an inefficient restoration. This chapter will first give a brief overview of some of the proposals from the literature that may improve these problems. Then the main focus will be on the staircasing effect. Two models that deal with this effect will be presented: the Fourth-Order model [10, 12, 26] and the TV-Stokes model [38, 34, 25, 21]. The latter reference is the novelty in research literature and it is the main contribution in this thesis. Both models will be given in dual formulation and the next section will give numerical experiments for these models. It should also be noted that these problems are still challenging problems in image processing and remain to be solved.

The ROF model treats flat regions and edges well, but favours piecewise constant functions in the solution, known as staircase effect. This is an adverse effect in images that contain regions with gradual image variations. To prevent this over-sharpening there have been some proposals to minimize different regularizations terms.

A more general version of the ROF model may be given as the following minimization problem

$$\inf_d \int_\Omega \phi(|\nabla d|) + \frac{\lambda}{2}(d - d_0)\,\mathrm{d}x, \tag{3.1}$$

where $\phi : R^+ \to R^+$ is a smooth function. The above minimization has the following regularized Euler equation

$$\mathrm{div}\left(\phi'(\sqrt{|\nabla d|^2 + \beta})\frac{\nabla d}{|\nabla d|}\right) - \lambda(d - d_0) = 0, \tag{3.2}$$

where $\beta > 0$ is a small parameter to avoid $|\nabla d| = 0$. Pure total variation corresponds to $\phi(g) = g$ and $\phi(g) = g^2$ corresponds to the H^1 norm. The question now is how to combine the two norms to benefit each of them.

In [4] they propose

$$\phi'(|\nabla d|) = |\nabla d|^{p-1} \tag{3.3}$$

and then choose p close to 1.

The smooth function ϕ in the minimization problem (3.1) will be more interesting when endowed with the following conditions

$$(C1): \quad \phi(0) = \phi'(0) = 0 \tag{3.4}$$

$$(C2): \quad \lim_{s \to +\infty} \frac{\phi(s)}{s} = 0. \tag{3.5}$$

The first condition (C1) implies that the first term in (3.1) is quadratic near the origin, thus smoothes the flat areas ($|\nabla d| \approx 0$) with the H^1 norm. The second condition (C2) implies that the function is sublinear in growth at infinity. This property will enhance the edges since the cost of edges are low. Using the above conditions for the smooth function ϕ results in an ill-posed problem since the function is obviously non-convex. A simple example would be choosing $\phi(s) = s^2/(1 + s^2)$. However, the non-convexity is irrelevant for the finite-dimensional case as stated in [2] on page 93, and the numerical experiments often give visually good results. This strategy is throughly presented in [23] where the authors also gave the dual formulation with a weighted total variation term of the non-convex problem.

In [5, 4] they proposed $\phi(|\nabla d|) = |\nabla d|^{p(|\nabla d|)}$ to adapt the behaviour of $|\nabla d|$, such that $p = 1$ near the edges and $p = 2$ in flat regions. In between these regions they use a fractional norm $H_p^1, p \in (1, 2)$. A simple example would be $p(|\nabla d|) = \frac{2}{1+2|\nabla d|}$. However, in [5, 4], examples are given in just the one dimensional case.

3.1 Fourth-Order denoising in dual formulation

One of the early approaches to remove staircasing in image denoising, was to introduce higher-order derivatives or norms into the energy. The first approach to combine the TV norm and H^1 norm (square norm of the gradient) is to consider the inf-convolution as the authors Chambolle and Lions proposed in [10]. The resulting minimization is equivalent to

$$\inf_d \int_{|\nabla d| \geq \epsilon} |\nabla d| \, \mathrm{d}x + \frac{\epsilon}{2} \int_{|\nabla| < \epsilon} |\nabla d|^2 \, \mathrm{d}x + \frac{\lambda}{2} \int_\Omega |d - d_0|^2 \, \mathrm{d}x, \tag{3.6}$$

where ϵ is a given threshold to separate the smooth and discontinuous parts of the image.

The same method can be used to minimize the total variation of the gradient, i.e a second-order functional. Some of the earliest variants of higher-order methods for image denoising can be found in the same paper [10], where they decompose the image into two parts, $d = d_1 + d_2$, where d_1 consists of discontinuous parts and d_2 consists of the smooth areas of the image, and then minimize the following problem

$$\inf_{d_1, d_2} \int_\Omega |\nabla d_1| + \alpha |\nabla(\nabla d_2)| + \frac{\lambda}{2} |d_1 + d_2 - d_0|^2 \, \mathrm{d}x, \tag{3.7}$$

such that each component is assigned to the appropriate norm, i.e $d_1 \in BV(\Omega)$ and $d_2 \in H_1^1(\Omega)$ with $\nabla d_2 \in BV(\Omega)$. The results given in [10] are very good for the one-dimensional case, but the model suffers from over-smoothening of the edges in two dimensions.

This is evident, since the problem is how to combine the two different norms, which are dependent on the gradient, and this operation is unstable.

One of the first approaches to minimize only the higher-order derivaives in the energy was proposed by the the authors Lundervold, Lysaker and Tai (LLT) in [26], where they gave the following minimization problem

$$\int_\Omega \left(|d_{xx}| + |d_{yy}| \right) \, \mathrm{d}x \, \mathrm{d}y \quad \text{subject to} \quad \|d - d_0\|_2 \leq \sigma \tag{3.8}$$

and another functional that is rotational invariant

$$\int_\Omega \sqrt{|d_{xx}|^2 + |d_{xy}|^2 + |d_{yy}|^2 + |d_{yx}|^2} \, \mathrm{d}x \, \mathrm{d}y \quad \text{subject to} \quad \|d - d_0\|_2 \leq \sigma. \tag{3.9}$$

The Euler equations for the two above functionals will result into a set of fourth-ordered nonlinear PDEs.

We will further focus on the Fourth-Order model, and derive a dual formulation of the following minimization problem

$$\inf_d \int_\Omega (|\nabla^2 d|) + \frac{1}{2\lambda}(d - d_0)^2 \, \mathrm{d}x, \tag{3.10}$$

where the regularization term is defined by $|\nabla^2 d| = \sqrt{|d_{xx}|^2 + |d_{yy}|^2}$. This term is usually replaced by $|\nabla^2 d|_\beta = \sqrt{|\nabla^2 d|^2 + \beta^2}$, which is a necessary regularization to avoid the non-differentiability. The model shares the same numerical difficulties as the ROF model, and a dual formulation will now be given to overcome the β-regularization and the slow iteration. Similar works can be found in [15].

We can define the regularization term as an analog to the total variation defined in (2.7), hence

$$\|\nabla^2 d\|_1 = \sup \left\{ \int_\Omega d \, \Delta \xi(x) \, \mathrm{d}x : \xi \in C_0^2(\Omega, \mathbb{R}^2), |\xi(x)| \leq 1 \forall x \in \Omega \right\}. \tag{3.11}$$

Thus, we use the right-handed definition of the above equation and deduce

$$\inf_d \sup_{|p| \leq 1} \int_\Omega (d, \Delta p) + \frac{1}{2\lambda}(d - d_0)^2 \, \mathrm{d}x, \tag{3.12}$$

where p is the dual variable. By the convexity of d and the concave dual function p, we can use the same argument as in (2.25)-(2.29) to switch the inf and sup in (3.12)

$$\sup_{|p| \leq 1} \inf_d \int_\Omega (d, \Delta p) + \frac{1}{2\lambda}(d - d_0)^2 \, \mathrm{d}x. \tag{3.13}$$

Solving for the inner minimization with respect to d yields

$$d = d_0 - \lambda \Delta p. \tag{3.14}$$

Thus, by the simple equation above we find our restored image by looking for the projection $\lambda \Delta p$. We deduce a minimization problem for the dual variable by substituting equation (3.14) into (3.13) and setting $\sup\{\cdot\} = -\inf\{-\cdot\}$, and end up with the following dual problem

$$\inf_p \left\{ \|\Delta p - \lambda^{-1} d_0\|_2^2 : |p| \leq 1 \right\}. \tag{3.15}$$

The dual minimization problem is solved by considering the optimality condition for equation (3.15)

$$\nabla^2 \left(\Delta p - \lambda^{-1} d_0 \right) + \alpha p = 0, \tag{3.16}$$

where either $\alpha = 0$ and $|p| \le 1$ or $\alpha > 0$ and $|p| = 1$, both cases yield

$$\alpha = |\nabla^2 \left(\Delta p - \lambda^{-1} d_0 \right)|. \tag{3.17}$$

Thus, inserting equation (3.17) into (3.16) gives

$$\nabla^2 \left(\Delta p - \lambda^{-1} d_0 \right) + |\Delta p - \lambda^{-1} d_0|p = 0, \tag{3.18}$$

which may be solved in the same manner as the second-order Chambolle iteration

$$p^{n+1} = \frac{p^n + \tau \left(\nabla^2 \left(\Delta p^n - \lambda^{-1} d_0 \right) \right)}{1 + \tau \left| \nabla^2 \left(\Delta p^n - \lambda^{-1} d_0 \right) \right|}. \tag{3.19}$$

3.1.1 Discrete algorithm

The continuous iteration from equation (3.19) is discretized by using the matrix operators from section 1.1.3. Thus, the second-order operator ∇^2 may have the following discrete definition

$$\nabla_h^2 d = \begin{bmatrix} -dB_x^T B_x \\ -B_y^T B_y d \end{bmatrix}, \tag{3.20}$$

and the discrete laplacian operator in the same manner

$$\Delta^h d = -dB_x^T B_x - B_y^T B_y d. \tag{3.21}$$

The full discrete version of (3.19) is then given by using the above discrete operators

$$p^{n+1} = \frac{p^n + \tau \left(\nabla_h^2 \left(\Delta^h p^n - \lambda^{-1} d_0 \right) \right)}{1 + \tau \left| \nabla_h^2 \left(\Delta^h p^n - \lambda^{-1} d_0 \right) \right|}. \tag{3.22}$$

The following result gives a condition on τ for fast convergence.

Theorem 2. *Let $\tau \le 1/64$. Then $\lambda \Delta^h p^n$ converges to solution $\lambda \Delta^h p$ of the minimization problem given in (3.15) as $n \to \infty$.*

A proof of the theorem can be found in [15] where they propose a dual algorithm for the minimization problem given in (3.7).

As an alternative argument for convergence and a better condition for τ, we can consider the following

$$p^{n+1} = p^n + \tau \left(\nabla_h^2 \left(\Delta^h p^n - \lambda^{-1} d_0 \right) \right) = (1 + \tau \nabla_h^2 \Delta^h) p^n - \tau \nabla_h^2 \Delta^h \lambda^{-1}. \tag{3.23}$$

By denoting the above equation as $p^{n+1} = \phi(p^n)$ and requiring $\phi'(p^n) \le 1$ (1-Lipschitz) we deduce

$$\|\tau \nabla_h^2 \Delta^h\| - \|1\| \le \|1 + \tau \nabla_h^2 \Delta^h\| \le 1. \tag{3.24}$$

Hence, we get the following condition for τ

$$\tau \|\nabla_h^2 \Delta^h\| \le 2. \tag{3.25}$$

To estimate the norm of the operator $\nabla_h^2 \Delta^h$, we note that $\|B\|_\infty = 2$ and the SVD of the operator $\nabla_h^2 \Delta^h$ have the following form in x (resp. y) direction

$$\nabla_h^2 \Delta^h = -C^T \left[\begin{bmatrix} 0 & \\ & \Sigma_x^4 \end{bmatrix} + \begin{bmatrix} 0 & \\ & \Sigma_{xy}^4 \end{bmatrix} \right] C. \tag{3.26}$$

Thus, the following bound in x (resp. y) direction can be establibshed

$$\lambda_{\max}\left(\nabla_h^2 \Delta^h\right) = \lambda_{\max}\left(\Sigma_x^4\right) + \lambda_{\max}\left(\Sigma_{xy}^4\right) \le \|B\|_2^4 + \|B\|_2^4 \le \|B\|_\infty^4 + \|B\|_\infty^4 = 32, \tag{3.27}$$

giving the optimal condition for τ to be equal to 1/16. Note that the operator $\nabla_h^2 \Delta^h$ needs to be calculated in each iteration, and this operation is *ill-conditioned*, i.e the condition number $\kappa(\nabla_h^2 \Delta^h)$ is close to 10^{18}.

3.2 TV-Stokes denoising in dual formulation

3.2.1 Motivation

To tackle the numerical problems in higher-order methods, one can split the problem into two steps, and solve a second-order problem in each step. There has been some research activity to formulate a two step method, and Lysaker, Osher and Tai (LOT model) proposed in [27] to smooth the normal vector field in the first step with the following minimization

$$\inf_{|\mathbf{n}|=1} \int_\Omega |\nabla \mathbf{n}| + \frac{\delta}{2} |\mathbf{n} - \mathbf{n}_0|^2 \, \mathrm{d}x. \tag{3.28}$$

This will find the smoothed flow field, i.e directions of the gradient: $\mathbf{n} = \nabla d/|\nabla d|$ and $\mathbf{n}_0 = \nabla d_0/|\nabla d_0|$. Once the normal vector field is obtained, the image can be reconstructed by a second step, which matches the normal vector field by the following minimization problem

$$\inf_d \int_\Omega |\nabla d| - (\mathbf{n}^0, \nabla d) + \frac{\mu}{2}|d - d_0|^2 \, \mathrm{d}x, \tag{3.29}$$

where $\mathbf{n}^0 = [n_1, n_2]$. The LOT method gives a better result than the ROF, since it is more edge preserving, cf. [27]. However, it still suffers from the staircase effect, since it can not recover images containing affine regions of noise.

Another way to smooth the normal vector fields in step one, which is the crucial part of the two-step algorithm, is to notice that the isophote lines in the image must be constant, see Figure 3.1. Hence, the tangential vector field is divergence free. The first step should then minimize with respect to the tangential vector field given by

$$\tau = \nabla^\perp d = [-d_y, d_x]^T = [v, u]^T, \tag{3.30}$$

subject to the divergence of the tangential vector field is equal to zero, known as incompressibility in fluid mechanics

$$\mathrm{div}\,\tau = 0. \tag{3.31}$$

This is formulated in [34] by the authors Rahman, Tai and Osher. The model will be called the primal TV-Stokes model to indicate that the energy
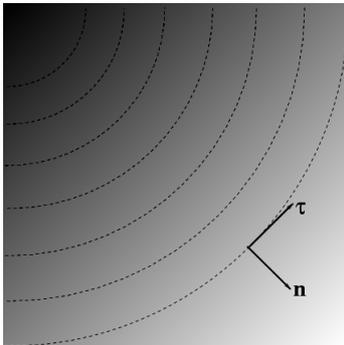
Figure 3.1: Normal and tangent vectors to the isophote lines

functionals are solved with respect to the primal variables τ and d. The first step of the primal TV-Stokes model has the following minimization problem

$$\inf_{\tau} \int_{\Omega} |\nabla \tau| + \frac{\delta}{2} |\tau - \tau_0|^2 \, \mathrm{d}x \quad \text{subject to} \quad \operatorname{div} \tau = 0. \tag{3.32}$$

where τ_0 is given by $\tau_0 = [v_0, u_0]^T$. Once the smoothed tangential vector field is obtained, the corresponding normal vector field is then calculated by $\mathbf{n} = [u, -v]$. The normal vector field can be used to reconstruct the noisy image by fitting it with the following second step minimization

$$\inf_{d} \int_{\Omega} |\nabla d| - \left(\nabla d, \frac{\mathbf{n}}{|\mathbf{n}|} \right) \, \mathrm{d}x \quad \text{subject to} \quad \int_{\Omega} (d - d_0)^2 \, \mathrm{d}x = \sigma^2. \tag{3.33}$$

The authors then solve the problem by finding the Euler-Lagrange equations for (3.32-3.33) and then iterating the nonlinear PDEs with an explicit gradient descent method until steady-state is reached. This, however, is very slow due to the nature of explicit forward schemes. A modified TV-Stokes model is proposed by Litvinov, Rahman and Tai in [25], where they establish existence and uniqueness to the minimization problems for both steps of the model.

The next sections will reformulate the primal equations given by (3.32) and (3.33) into a dual formulation, which improves the speed drastically.

3.2.2 First step: tangent field smoothing

To derive a dual formulation of the primal problem (3.32) we introduce the definition of the total variation of the tangential vector field

$$\int_{\Omega} |\nabla \tau| \, \mathrm{d}x = \sup_{p \in K} \left\{ \int_{\Omega} (\tau, \operatorname{div} p_i) \, \mathrm{d}x : i = 1, 2 \right\} \tag{3.34}$$

and K is closure of the convex set

$$\left\{ \operatorname{div} \xi_i : \xi_i \in C_0^1(\Omega, \mathbb{R}^2), |\xi_i(x)| \leq 1, \forall x \in \Omega, i = 1, 2 \right\}. \tag{3.35}$$

The dual variable p is now a vector composed by the two dual variables for both directions in the tangential vector field. The divergence of this field then

given by

$$\operatorname{div} p = (\operatorname{div} p_1, \operatorname{div} p_2)^T, \quad \operatorname{div} p_1 = \frac{\partial p_1^1}{\partial x} + \frac{\partial p_1^2}{\partial y}, \quad \operatorname{div} p_2 = \frac{\partial p_2^1}{\partial x} + \frac{\partial p_2^2}{\partial y}. \quad (3.36)$$

This definition of the divergence is similar to the vectorial dual norm from [6] for vectorial images, e.g. color images.

Using the dual formulation of the total variation norm, the primal problem (3.32) can be written as minimization of the following dual problem

$$\inf_{\operatorname{div} \tau = 0} \sup_{|p| \le 1} \int_{\Omega} (\tau, \operatorname{div} p) + \frac{1}{2\delta} |\tau - \tau_o|^2 \, \mathrm{d}x, \quad (3.37)$$

interchange sup and inf with the same argument as given in the section 2.2

$$\sup_{|p| \le 1} \inf_{\operatorname{div} \tau = 0} \int_{\Omega} (\tau, \operatorname{div} p) + \frac{1}{2\delta} |\tau - \tau_o|^2 \, \mathrm{d}x. \quad (3.38)$$

Let us introduce the orthogonal projection Π_P onto the constrained subspace $P = \{\tau : \operatorname{div} \tau = 0\}$. This projection may be given as

$$\Pi_P \begin{bmatrix} \pi_1 \\ \pi_2 \end{bmatrix} = \begin{bmatrix} \pi_1 \\ \pi_2 \end{bmatrix} - \nabla \Delta^+ \operatorname{div} \begin{bmatrix} \pi_1 \\ \pi_2 \end{bmatrix}. \quad (3.39)$$

Thus, finding the minimum for the inner problem, without constraint $\operatorname{div} \tau = 0$, is obtained by using the property from orthogonal projection, $(\tau, \operatorname{div} p) = (\Pi_P \tau, \operatorname{div} p) = (\tau, \Pi_P \operatorname{div} p)$, and the Euler equation. The equation (3.38) with the orthogonal projection

$$\sup_{|p| \le 1} \inf_{\operatorname{div} \tau = 0} \int_{\Omega} (\tau, \Pi_P \operatorname{div} p) + \frac{1}{2\delta} |\tau - \tau_o|^2 \, \mathrm{d}x \quad (3.40)$$

is solved by obtaining the following derivation for the Euler equation

$$\phi(t) = E(\tau + t\psi, \lambda) = \int_{\Omega} (\tau + t\psi, \Pi_P \operatorname{div} p) + \frac{1}{2\delta} |\tau - \tau_0 + t\psi|^2 \, \mathrm{d}x$$

$$= \int_{\Omega} (\tau, \Pi_P \operatorname{div} p) + (t\psi, \Pi_P \operatorname{div} p) + \frac{1}{2\delta} |\tau - \tau_0|^2 + \delta^{-1}(t\psi, \tau - \tau_0) + \frac{1}{2\delta} |t\psi|^2 \, \mathrm{d}x.$$
$$(3.41)$$

$\phi'(t)$ will give minimum at $\phi'(0) = 0$

$$\phi'(t) = \int_{\Omega} (\psi, \Pi_P \operatorname{div} p) + \delta^{-1}(\psi, \tau - \tau_0) + \delta^{-1} |t\psi| \, \mathrm{d}x, \quad (3.42)$$

$$\phi'(0) = \int_{\Omega} (\psi, \Pi_P \operatorname{div} p) + \delta^{-1}(\psi, \tau - \tau_0) \, \mathrm{d}x = 0, \quad (3.43)$$

then for any ψ, minimum in (3.40) is obtained by

$$\tau = \tau_o - \delta \Pi_P \operatorname{div} p. \quad (3.44)$$

Note that by the means of a Lagrangian multiplier λ, equation (3.40) may be written as

$$\sup_{|p|\leq 1} \inf_{\tau,\lambda} \int_\Omega (\tau, \operatorname{div} p) + \frac{1}{2\delta}|\tau - \tau_o|^2 + (\lambda, \operatorname{div} \tau)\, \mathrm{d}x, \tag{3.45}$$

that gives the Euler equations

$$\tau = \tau_0 - \delta \operatorname{div} p - \delta \nabla \lambda \tag{3.46}$$

$$\operatorname{div} \tau = 0. \tag{3.47}$$

Left multiplying the equation (3.46) with the div operator, lets us solve the equation $-\Delta \lambda = \operatorname{div} \operatorname{div} p$ with the help of the pseudoinverse Δ^+. Inserting λ into equation (3.46) gives the primal variable τ with the orthogonal projection (3.39) that satisfies $\operatorname{div} \tau = 0$.

We arrive at the minimization of the distance $\|\Pi_P \operatorname{div} p - \delta^{-1}\tau_0\|$ by substituting (3.44) back into (3.40)

$$\sup_{|p|\leq 1} \int_\Omega (\tau_0, \Pi_P \operatorname{div} p) - \delta|\Pi_P \operatorname{div} p|^2 + \frac{\delta}{2}|\Pi_P \operatorname{div} p|^2 + =$$

$$\sup_{|p|\leq 1} \int_\Omega (\tau_0, \operatorname{div} p) - \frac{\delta}{2}|\Pi_P \operatorname{div} p|^2 = \inf_{|p|\leq 1} \int_\Omega \frac{\delta}{2}\Pi_P|\operatorname{div} p|^2 - (\tau_0, \operatorname{div} p)\, \mathrm{d}x =$$

$$= \inf_{|p|\leq 1} \frac{1}{2\delta} \int_\Omega \delta^2 \Pi_P |\operatorname{div} p|^2 - 2\delta\,(\tau_0, \operatorname{div} p) + (\tau_0, \tau_0) - (\tau_0, \tau_0)\, \mathrm{d}x. \tag{3.48}$$

The problem now consists of solving the quadratic minimization problem

$$\min_p \left\{ \left\|\Pi_P \operatorname{div} p - \delta^{-1}\tau_0\right\|_2^2 : |p_i| \leq 1,\ i = 1, 2 \right\}. \tag{3.49}$$

Note that Π_P is an orthogonal projection of a convex set, hence P is another convex set, therefore the objective function in (3.49) is still convex. This satisfies the KKT conditions, such that the existence of the Lagrange multipliers $\alpha_i \geq 0$ associated to each constraint in (3.49), are given by

$$- \left(\nabla\left(\Pi_P \operatorname{div} p - \delta^{-1}\tau_0\right)\right)_i + \alpha_i p_i = 0. \tag{3.50}$$

Looking at the optimal conditions for the constraint functions, Chambolle noticed that α_i can be eliminated since $\alpha_i > 0$ and $|p_i| = 1$ or $\alpha_i = 0$ and $|p_i| < 1$, in both cases

$$\alpha_i = \left|\nabla\left(\Pi_P \operatorname{div} p - \delta^{-1}\tau_0\right)_i\right|. \tag{3.51}$$

Thus eliminating α_i by substituting (3.51) into (3.50)

$$- \left(\nabla\left(\Pi_P \operatorname{div} p - \delta^{-1}\tau_0\right)\right)_i + \left|\nabla\left(\Pi_P \operatorname{div} p - \delta^{-1}\tau_0\right)_i\right| p_i = 0. \tag{3.52}$$

Now we introduce an artificial time variable t and solve the time-dependent problem until steady-state, i.e $t \to \infty$ s.t $\frac{\partial p_i}{\partial t} \to 0$, this gives the iteration

$$\frac{\partial p_i}{\partial t} = \left(\nabla\left(\delta\Pi_P \operatorname{div} p - \tau_0\right)\right)_i - \left|\left(\nabla\left(\delta\Pi_P \operatorname{div} p - \tau_0\right)\right)_i\right| p_i. \tag{3.53}$$

The solution for the smoothed tangential vector field is given by equation (3.44).

3.2.3 Second step: image reconstruction

In the second step the noisy image d_0 is reconstructed with the help of the smoothed normal vector field from step one. The minimization problem is as follows

$$\inf_d \int_\Omega |\nabla d| - \left(\nabla d, \frac{\mathbf{n}}{|\mathbf{n}|}\right) \mathrm{d}x \quad \text{subject to} \quad \int_\Omega (d - d_0)^2 \, \mathrm{d}x = \sigma^2. \qquad (3.54)$$

Similar to the derivation for the dual ROF model in section 2.2, the total variation can be expressed as the dual total variation, and together with the adjoint operator of the gradient, the primal functional (3.54) can be rewritten into the $\inf - \sup$ problem

$$\inf_d \sup_{|r|\leq 1} \int_\Omega \left(d, \mathrm{div}\left(r + \frac{\mathbf{n}}{|\mathbf{n}|}\right)\right) \mathrm{d}x \quad \text{subject to} \quad \int_\Omega (d - d_0)^2 \, \mathrm{d}x = \sigma^2. \quad (3.55)$$

Equation (3.55) solves the same problem as the following unconstrained problem where $\mu > 0$ is a Lagrangian multiplier

$$\sup_{|r|\leq 1} \inf_d \int_\Omega \left(d, \mathrm{div}\left(r + \frac{\mathbf{n}}{|\mathbf{n}|}\right)\right) + \frac{1}{2\mu}(d - d_0)^2 \, \mathrm{d}x. \qquad (3.56)$$

The μ constant will act as a global smoothening parameter, i.e it controls the global smoothing effect and $\mathrm{div}(\mathbf{n}/|\mathbf{n}|)$ will act as a local smoothening term. Using the similar derivation of the Euler equation as in the first step, we obtain the primal variable d by the simple expression

$$d = d_0 - \mu \mathrm{div}\left(r + \frac{\mathbf{n}}{|\mathbf{n}|}\right), \qquad (3.57)$$

where r is the dual variable. Substituting (3.57) into (3.56) gives a maximization problem for the dual variable

$$\sup_{|r|\leq 1} \int_\Omega \left(d_0, \mathrm{div}\left(r + \frac{\mathbf{n}}{|\mathbf{n}|}\right)\right) - \mu \left|\mathrm{div}\left(r + \frac{\mathbf{n}}{|\mathbf{n}|}\right)\right|^2 + \frac{\mu}{2}\left|\mathrm{div}\left(r + \frac{\mathbf{n}}{|\mathbf{n}|}\right)\right|^2 \mathrm{d}x =$$

$$= \sup_{|r|\leq 1} \int_\Omega |d_0|^2 - \left|\mu \mathrm{div}\left(r + \frac{\mathbf{n}}{|\mathbf{n}|}\right) - d_0\right|^2 \mathrm{d}x \quad (3.58)$$

Finally, one can solve the above dual problem by using $\sup\{\cdot\} = -\inf\{-\cdot\}$ which yields the final minimization problem

$$\min_r \left\{ \left\|\mathrm{div}\left(r + \frac{\mathbf{n}}{|\mathbf{n}|}\right) - \mu^{-1}d_0\right\|_2^2 : |r| \leq 1 \right\}. \qquad (3.59)$$

As before, the KKT conditions give existence to the Lagrangian multipliers $\alpha \geq 0$ associated to the constraint $|r| \leq 1$ in (3.59)

$$-\nabla\left(\mathrm{div}\left(r + \frac{\mathbf{n}}{|\mathbf{n}|}\right) - \mu^{-1}d_0\right) + \alpha r = 0 \qquad (3.60)$$

with either $\alpha > 0$ and $|r| = 1$ or $\alpha = 0$ and $|r| < 1$, in any case

$$\alpha = \left| \nabla \left(\operatorname{div} \left(r + \frac{\mathbf{n}}{|\mathbf{n}|} \right) - \mu^{-1} d_0 \right) \right|. \tag{3.61}$$

Substituting (3.61) into (3.60) and introducing the artificial time t gives the time-dependent problem that needs to be solved until steady-state

$$\frac{\partial r}{\partial t} = -\nabla \left(\mu \operatorname{div} \left(r + \frac{\mathbf{n}}{|\mathbf{n}|} \right) - d_0 \right) + \left| \nabla \left(\mu \operatorname{div} \left(r + \frac{\mathbf{n}}{|\mathbf{n}|} \right) - d_0 \right) \right| r = 0, \tag{3.62}$$

and the denoised image is then given by equation (3.57).

3.2.4 Discrete orthogonal projection

We start by giving the discrete algorithm for the dual TV-Stokes model by solving the orthogonal projection (3.39). This projection will project elements from the finite set K_h which is defined in similar manner as the convex set K

$$K_h = \{ \operatorname{div}^h p_i : p_i \in Y, \ |p_{i,j,k}| \leq 1, \ i = 1, 2, \ \forall \, j, k = 1, \dots N \}, \tag{3.63}$$

onto the finite constrained subset

$$P_h = \left\{ \tau \in Y : \operatorname{div}^h \tau = 0 \right\}. \tag{3.64}$$

The discrete orthogonal projection $\Pi_P : K_h \to P_h$ has the form

$$\Pi_P^h = I - \nabla^h (\Delta^h)^+ \operatorname{div}^h, \tag{3.65}$$

where the gradient and the divergence operators are defined by the matrix operators from equations (1.26) (1.34). The discrete definition of the pseudoinverse operator has the following definition

Definition 11. *The pseudoinverse of an operator* $(\Delta^h) = U\Sigma V^T$, *denoted* $(\Delta^h)^+$, *has the following form*

$$(\Delta^h)^+ = V\Sigma^+ U^T, \tag{3.66}$$

where Σ^+ *is obtained by transposing* Σ *and inverting all nonzero entries.*

Thus, to complete the definition of (3.65), we need a description of the Laplacian. Let us consider two possibilities, the zero Dirichlet boundary condition

$$\Delta_D^h = -dB_x B_x^T - B_y B_y^T d = \begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix} \tag{3.67}$$

and the zero Neumann boundary condition

$$\Delta_N^h = -dB_x^T B_x - B_y^T B_y d = \begin{bmatrix} -1 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -1 \end{bmatrix}. \tag{3.68}$$

Note that the latter Laplacian matrix is singular.

We start solving the Poisson equation $f = \Delta_D^h d$ by considering the $N \times N$ Discrete Cosine Transformation matrix, C, which is defined by `dct(eye(N))` in MATLAB. The Discrete Sine Transformation matrix, $\widetilde{S}$, is defined in the same manner as `dst(eye(N-1))`, which satisfies the equation $\widetilde{S}^T \widetilde{S} = (N/2)I$, where I is the identity matrix. Thus, an orthogonal symmetric matrix $S = -\widetilde{S}/\sqrt{N/2}$ of order $N-1$ will be used. The singular value decomposition of B has the following form

$$B = S[0, \Sigma]C, \quad \Sigma = \text{diag}\,(\sigma_1, \ldots, \sigma_{N-1}), \tag{3.69}$$

where the diagonal matrix Σ has the entries

$$\sigma_k = \frac{2}{h} \sin \frac{\pi k}{2N}, \quad k = 1, 2, \ldots, N-1. \tag{3.70}$$

Applying the singular value decomposition to the Laplacian with zero Dirichlet boundary condition (3.67) yields

$$\Delta_D^h d = -dS^T[0; \Sigma_x]C^T C[0, \Sigma_x]S - S^T[0; \Sigma_y]C^T C[0, \Sigma_y]Sd, \tag{3.71}$$

by the orthogonality of C, the above equation reduces to

$$f = \Delta_D^h d = -dS^T \Sigma_x^2 S - S^T \Sigma_y^2 Sd. \tag{3.72}$$

Denoting the transformations $\widehat{d} = SdS^T$ and $\widehat{f} = SfS^T$ gives the equation

$$\widehat{f} = -\widehat{d}\Sigma_x^2 - \Sigma_y^2 \widehat{d} \tag{3.73}$$

which is easily solved by $\widehat{d}$ with the aid of (3.66), such that $\widehat{d} = G(\widehat{f})$ is given by

$$\widehat{d} = -\widehat{f}/(\sigma_x^2 + \sigma_y^2). \tag{3.74}$$

Thus the pseudoinverse operator $(\Delta_D^h)^+$ can be efficiently computed with the help of the Discrete Sine Transform

$$(\Delta_D^h)^+ f = S^T G(SfS^T)S, \tag{3.75}$$

where G is defined in (3.74).

The pseudoinverse for equation (3.68) is found in the similar manner, the equation can be rewritten as

$$f = \Delta_N^h d = -dC^T \begin{bmatrix} 0 & \\ & \Sigma_x^2 \end{bmatrix} C - C^T \begin{bmatrix} 0 & \\ & \Sigma_y^2 \end{bmatrix} Cd. \tag{3.76}$$

Denoting $\widetilde{f} = CfC^T$ and $\widetilde{d} = CdC^T$ we arrive at the equation

$$\widetilde{f} = -\widetilde{d} \begin{bmatrix} 0 & \\ & \Sigma_x^2 \end{bmatrix} - \begin{bmatrix} 0 & \\ & \Sigma_y^2 \end{bmatrix} \widetilde{d}. \tag{3.77}$$

This equation is easily solved with respect to $\widetilde{d}$. Suppose that the matrices $\widetilde{f}$ and $\widetilde{d}$ have the entries $\widetilde{f}_{ij}$ and $\widetilde{d}_{ij}$ for $i, j = 0, 1, \ldots$. Note that in our case

$\widetilde{f}_{00} = 0$. Then the solution $\widetilde{d} = H(\widetilde{f})$ is as follows:

$$
\begin{aligned}
\widetilde{d}_{00} &= 0, \\
\widetilde{d}_{i,0} &= -\widetilde{f}_{i,0}/\sigma_{i,y}^2, & i = 1, 2, \ldots, \\
\widetilde{d}_{0,j} &= -\widetilde{f}_{0,j}/\sigma_{j,x}^2, & j = 1, 2, \ldots, \\
\widetilde{d}_{ij} &= -\widetilde{f}_{ij}/(\sigma_{i,y}^2 + \sigma_{j,x}^2), & i, j = 1, 2, \ldots.
\end{aligned}
\tag{3.78}
$$

Thus the pseudoinverse operator $(\Delta_N^h)^+$ can be efficiently computed with the help of the Discrete Cosine Transform:

$$
(\Delta_N^h)^+ f = C^T H(C f C^T) C,
\tag{3.79}
$$

where the function H is defined in (3.78).

3.2.5 Discrete algorithm

The images will be two dimensional matrices with rectangular size $N \times N$. We will approximate the derivatives by finite difference introduced in section 1.1.3, and take advantage of the staggered grid, or cell centred approximation, given in section 1.1.3.

First step The full discrete version of equation (3.53) is then the following iteration

$$
p_i^{n+1} = p_i^n + k \left(\nabla^h \left(\delta \Pi_P^h \operatorname{div}^h p_i^n - \tau_0 \right) - \left| \nabla^h \left(\delta \Pi_P^h \operatorname{div}^h p^n - \tau_0 \right) \right| p_i^{n+1} \right)
\tag{3.80}
$$

or

$$
p_i^0 = 0, \quad p_i^{n+1} = \frac{p_i^n + k \nabla^h \left(\delta \Pi_P^h \operatorname{div}^h p_i^n - \tau_0 \right)}{1 + k \left| \nabla^h \left(\delta \Pi_P^h \operatorname{div}^h p_i^n - \tau_0 \right) \right|}; \quad i = 1, 2.
\tag{3.81}
$$

Section 3.2.5 discusses stopping critera for the semi-implicit iteration (3.81), and the following condition on k enables the fast convergence. We can prove the below theorem by following similar steps as in the proof of Theorem 1 in [9].

Theorem 3. *Let $k \leq 1/8$. Then $\delta \Pi_P^h \operatorname{div}^h p^n$ converges to $\delta \Pi_P^h \operatorname{div}^h p$ as $n \to \infty$.*

Proof. By induction, $\forall n \leq 0, |p_{i,j,k}^n| \leq 1, \forall i, j, k$. Thus by fixing $n \geq 0$, $\eta = (p^{n+1} - p^n)/k$ and considering

$$
\begin{aligned}
\|\Pi_P^h \operatorname{div}^h p^{n+1} - \delta^{-1} \tau_0\|^2 &= \|\Pi_P^h \operatorname{div}^h \eta + \Pi_P^h \operatorname{div}^h p^n - \delta^{-1} \tau_0\|^2 \\
&= \|\Pi_P^h \operatorname{div}^h p^n - \delta^{-1} \tau_0\|^2 + 2k < \Pi_P^h \operatorname{div}^h \eta, \Pi_P^h \operatorname{div}^h p^n - \delta^{-1} \tau_0 > \\
&\quad + k^2 \|\Pi_P^h \operatorname{div}^h \eta\|^2.
\end{aligned}
\tag{3.82}
$$

Using integration by parts in the second term, and the fact that the Π_P^h is an orthogonal projection $\Pi_P^h = \left(\Pi_P^h \right)^2$ and $\left(\Pi_P^h \right)^* = \Pi_P^h$, gives

$$
\begin{aligned}
\|\Pi_P^h \operatorname{div}^h p^{n+1} - \delta^{-1} \tau_0\|^2 &= \|\Pi_P^h \operatorname{div}^h p^n - \delta^{-1} \tau_0\|^2 \\
&\quad - k \left(2 < \eta, \nabla(\Pi_P^h \operatorname{div}^h p^n - \delta^{-1} \tau_0) > -k \|\Pi_P^h \operatorname{div}^h \eta\|^2 \right).
\end{aligned}
\tag{3.83}
$$

31

We can bound the last term by $\|\Pi_P^h \mathrm{div}^h \eta\| \leq \kappa \|\eta\|$, where $\kappa = \sup_{|p|\leq 1} \|\Pi_P^h \mathrm{div}^h p\|$. It is then clear that $\|\delta \Pi_P^h \mathrm{div}^h p^{n+1} - \delta^{-1}\tau_0\|$ decreses if k is positive, hence we need to estimate κ and show that k is positive when $p^{n+1} \neq p^n$ and $k \leq 1/\kappa^2$.

To achieve the conditions on k we deduce the following

$$2\eta_i \nabla^h \left(\Pi_P^h \mathrm{div}^h p_i - \delta^{-1}\tau_0 \right) - \kappa^2 k |\eta_i|^2$$
$$= \left(1 - \kappa^2 k \right) |\eta_i|^2 - |\eta_i|^2$$
$$+ 2\eta_i \nabla^h \left(\Pi_P^h \mathrm{div}^h p_i - \delta^{-1}\tau_0 \right), \text{ for } i = 1, 2, \quad (3.84)$$

if we insert $\eta_i = \nabla^h \left(\Pi_P^h \mathrm{div}^h p_i - \delta^{-1}\tau_0 \right) - \left| \nabla^h \left(\Pi_P^h \mathrm{div}^h p_i - \delta^{-1}\tau_0 \right) \right| p_i^{n+1}$ in the above equation we deduce

$$2\eta_i \nabla^h \left(\Pi_P^h \mathrm{div}^h p_i - \delta^{-1}\tau_0 \right) - \kappa^2 k |\eta_i|^2$$
$$= \left(1 - \kappa^2 k \right) |\eta_i|^2$$
$$- \left| \nabla^h \left(\Pi_P^h \mathrm{div}^h p_i - \delta^{-1}\tau_0 \right) \right|^2 - \left| \nabla^h \left(\Pi_P^h \mathrm{div}^h p_i - \delta^{-1}\tau_0 \right) p_i^{n+1} \right|^2, \text{ for } i = 1, 2.$$
$$(3.85)$$

Since $|p_i^{n+1}| \leq 1$ we have the following

$$2\eta_i \nabla^h \left(\Pi_P^h \mathrm{div}^h p_i - \delta^{-1}\tau_0 \right) - \kappa^2 k |\eta_i|^2 \leq \left(1 - \kappa^2 k \right) |\eta_i|^2. \quad (3.86)$$

Thus, $\left\| \delta \Pi_P^h \mathrm{div}^h p_i^{n+1} - \tau_0 \right\|^2$ decreses if and only if k satisfies $1 - \kappa^2 k \geq 0$. If $p_i^{n+1} = p_i^n$ then $\eta_i = 0$, which is also true for $\kappa^2 k = 1$.

There exists a $m \geq 0$ such that $m = \lim_{n\to\infty} \left\| \delta \Pi_P^h \mathrm{div}^h p_i^{n+1} - \tau_0 \right\|^2$ since the norm is uniformly bounded. Let $\bar{p}_i$ be the limit of a converging subsequence $(p_i^{n_k})$ of (p_i^n) and $\bar{p}_i{}'$ be the limit of $p_i^{n_k+1}$. Inserting the subsequences into (3.81) and applying the limit gives

$$\bar{p}_i' = \frac{\bar{p}_i + k \nabla^h \left(\Pi_P^h \mathrm{div}^h \bar{p}_i - \delta^{-1}\tau_0 \right)}{1 + k \left| \nabla^h \left(\Pi_P^h \mathrm{div}^h \bar{p}_i - \delta^{-1}\tau_0 \right) \right|} \text{ for } i = 1, 2. \quad (3.87)$$

We see from the above equation that $m = \left\| \delta \Pi_P^h \mathrm{div}^h \bar{p}_i - \tau_0 \right\|^2 = \left\| \delta \Pi_P^h \mathrm{div}^h \bar{p}_i' - \tau_0 \right\|^2$, i.e $\eta_i = \left(\bar{p}_i' - \bar{p}_i \right)/k = 0$ and $\bar{p}_i' = \bar{p}_i$, hence

$$- \nabla^h \left(\Pi_P^h \mathrm{div}^h \bar{p}_i - \delta^{-1}\tau_0 \right) + \left| \nabla^h \left(\Pi_P^h \mathrm{div}^h \bar{p}_i - \delta^{-1}\tau_0 \right) \right| \bar{p}_i = 0 \text{ for } i = 1, 2$$
$$(3.88)$$

which is the Euler equation for a solution of (3.49). One can deduce that the $\delta \Pi_P^h \mathrm{div}^h p^n$ is the projection $\delta \Pi_P^h \mathrm{div}\, p$ and that this projection is unique. The theorem is proved if we can show $\kappa^2 \leq 8$.

Again, we use the orthogonal property for Π_P^h such that we have the reduced SVD, $\Pi_P^h = \hat{Q}I\hat{Q}^*$, such that $\left\|\Pi_P^h\right\|_2 = 1$. Then the estimate of κ is the following

$$
\begin{aligned}
\|\Pi_P^h \text{div}^h p\|^2 &= \sum_{1 \leq j,k \leq n} (p_{i,j,k}^1 - p_{i,j-1,k}^1 + p_{i,j,k}^2 - p_{i,j,k-1}^2)^2 \\
&\leq 4 \sum_{1 \leq j,k \leq n} \left(p_{i,j,k}^1\right)^2 + \left(p_{i,j-1,k}^1\right)^2 + \left(p_{i,j,k}^2\right)^2 + \left(p_{i,j,k-1}^2\right)^2 \leq 8\|p\|_Y^2 \leq 8, i = 1, 2,
\end{aligned}
\tag{3.89}
$$

which gives the inequality $\kappa^2 \leq 8$. $\qquad\square$

In practice the optimal condition for stability and convergence is achieved by choosing k equal $1/4$ and not $1/8$. The iteration converges rapidly for the first steps, and after n iterations the solution is given by (3.44). By standard theory from stability analysis and contraction theory we can get a better bound on k. Let us consider the following linearised iteration

$$
p^{n+1} = p^n + k\nabla^h \left(\Pi_P^h \text{div}^h p^n - \delta^{-1}\tau_0\right).
\tag{3.90}
$$

This can be denoted as

$$
p^{n+1} = \phi\left(p^n\right),
\tag{3.91}
$$

such that $\phi'\left(p^n\right) \leq 1$ is needed for stability (1-Lipstichz), $\phi'\left(p^n\right)$ can be bounded below by the following

$$
\|k\nabla^h \Pi_P^h \text{div}^h\| - \|1\| \leq \|1 + k\nabla^h \Pi_P^h \text{div}^h\| \leq 1,
\tag{3.92}
$$

which gives the condition on k

$$
k\|\nabla^h \Pi_P^h \text{div}^h\| \leq 2.
\tag{3.93}
$$

Thus, we need an estimate on the norm of the operator $\nabla^h \Pi_P^h \text{div}^h$. Note that $\|B\|_\infty = 2$ and $\|\nabla^h \Pi_P^h \text{div}^h\|_2 = \|\nabla^h \text{div}^h\|_2$, hence the SVD of $\nabla^h \text{div}^h$ is the following

$$
B_x^T \text{div}^h = -C^T \left[\begin{bmatrix} 0 \\ & \Sigma_x^2 \end{bmatrix} + \begin{bmatrix} 0 \\ & \Sigma_{xy}^2 \end{bmatrix}\right] C
\tag{3.94}
$$

$$
B_y \text{div}^h = -C^T \left[\begin{bmatrix} 0 \\ & \Sigma_y^2 \end{bmatrix} + \begin{bmatrix} 0 \\ & \Sigma_{yx}^2 \end{bmatrix}\right] C.
\tag{3.95}
$$

Now we can give a bound on the operator, e.g in the x direction, by the following

$$
\lambda_{\max}\left(\nabla^h \text{div}^h\right) = \lambda_{\max}\left(\Sigma_x^2\right) + \lambda_{\max}\left(\Sigma_{xy}^2\right) \leq \|B\|_2^2 + \|B\|_2^2 \leq \|B\|_\infty^2 + \|B\|_\infty^2 = 8,
\tag{3.96}
$$

which gives the optimal condition on k, i.e $k \leq 1/4$.

Second step The discrete version of the minimization in the second step is given by

$$r^0 = 0, \quad r^{n+1} = \frac{r^n + k\nabla^h \left(\text{div}^h \left(r^n + n\right) - \mu^{-1}d_0\right)}{1 + k\left|\nabla^h \left(\text{div}^h \left(r^n + n\right) - \mu^{-1}d_0\right)\right|}, \quad (3.97)$$

where $n = (n^1, n^2)$ is denoted as the normal vector field, calculated by the tangential vector field found in the first step

$$n^1 = \frac{v}{\sqrt{v^2 + u^2 + \epsilon}}, \quad n^2 = \frac{-u}{\sqrt{v^2 + u^2 + \epsilon}}. \quad (3.98)$$

The $\epsilon > 0$ is introduced as a small constant to avoid dividing by zero. In all the numerical examples for the second step, ϵ is chosen equal to 10^{-11}.

Theorem 4. *Let $k \leq 1/8$. Then $\mu\text{div}^h \left(r^n + n\right)$ converges to $\mu\text{div}^h \left(r + n\right)$ as $n \to \infty$.*

To get a better bound on k we consider the following

$$r^{n+1} = r^n + k\left(\nabla^h \left(\text{div}^h \left(r^n + n\right) - \mu^{-1}d_0\right)\right). \quad (3.99)$$

The above equation can be viewed as

$$r^{n+1} = \phi(r^n), \quad (3.100)$$

where $\phi'(r^n) \leq 1$ is needed for stability. Thus we need a lower bound which gives

$$\|k\nabla^h\text{div}^h\| - \|1\| \leq \|1 + k\nabla^h\text{div}^h\| \leq 1, \quad (3.101)$$

which yields the following condition on k

$$k\|\nabla^h\text{div}^h\| \leq 2, \quad (3.102)$$

hence we need to estimate $\|\nabla^h\text{div}^h\|_2 = \lambda_{\max}\left(\nabla^h\text{div}^h\right)$. Note that $\|B\|_\infty = 2$, and the SVD of $\nabla^h\text{div}^h$ is the following

$$B_x^T\text{div}^h = -C^T\left[\begin{bmatrix} 0 & \\ & \Sigma_x^2 \end{bmatrix} + \begin{bmatrix} 0 & \\ & \Sigma_{xy}^2 \end{bmatrix}\right]C \quad (3.103)$$

$$B_y\text{div}^h = -C^T\left[\begin{bmatrix} 0 & \\ & \Sigma_y^2 \end{bmatrix} + \begin{bmatrix} 0 & \\ & \Sigma_{yx}^2 \end{bmatrix}\right]C \quad (3.104)$$

$$\lambda_{\max}\left(\nabla^h\text{div}^h\right) = \lambda_{\max}\left(\Sigma_x^2\right) + \lambda_{\max}\left(\Sigma_{xy}^2\right) \leq \|B\|_2^2 + \|B\|_2^2 \leq \|B\|_\infty^2 + \|B\|_\infty^2 = 8. \quad (3.105)$$

This gives a better estimate on k which is less or equal to $1/4$.

As we can see, the second step has the same condition number as the Chambolle method for the ROF model, $\kappa\left(\nabla^h\text{div}^h\right) \approx 10^4$.

Finally, we can put all pieces together and give the complete algorithm

Algorithm: Dual TV-Stokes

Given d_0, k, δ and μ ;

Step one;

Let $p^0 = 0$ and $q^0 = 0$;

Calculate $\tau^0 = (v^0, u^0) : v^0 = -Bd$ and $u^0 = dB$;

Initialize counter: $n = 0$;

while *not converged* **do**

Calculate projections:

$$(\pi_p, \pi_q) = \Pi_K^h(\text{div}^h\, p^n, \text{div}^h\, q^n) \tag{3.106}$$

$$p^{n+1} = \frac{p^n + k\left(\nabla^h\left(\pi_p - \delta^{-1}v_0\right)\right)}{1 + k\left|\left(\nabla^h\left(\pi_p - \delta^{-1}v_0\right)\right)\right|}. \tag{3.107}$$

$$q^{n+1} = \frac{q^n + k\left(\nabla^h\left(\pi_q - \delta^{-1}u_0\right)\right)}{1 + k\left|\left(\nabla^h\left(\pi_q - \delta^{-1}u_0\right)\right)\right|}. \tag{3.108}$$

Update counter: $n = n + 1$;

end

Calculate τ:

$$\tau = \tau_0 - \Pi_K^h(\delta\text{div}^h\, p^{n+1}, \delta\text{div}^h\, q^{n+1}) \tag{3.109}$$

Step two;

Let $r^0 = 0$ and calculate the normal vector field:
$\mathbf{n} = (n_1, n_2), n_1 = u(v^2 + u^2)^{-1/2}$ and $n_2 = -v(v^2 + u^2)^{-1/2}$;
Initialize counter: $n = 0$;

while *not converged* **do**

Calculate projections:

$$r^{n+1} = \frac{r^n + k\left(\nabla^h\left(\text{div}^h\left(r^n + \mathbf{n}\right) - \mu^{-1}d_0\right)\right)}{1 + k\left|\left(\nabla^h\left(\text{div}^h\left(r^n + \mathbf{n}\right) - \mu^{-1}d_0\right)\right)\right|}. \tag{3.110}$$

Update counter: $n = n + 1$;

end

Recover image d:

$$d = d_0 - \mu\text{div}^h\left(r^{n+1} + \mathbf{n}\right) \tag{3.111}$$

Algorithm 1: Dual TV-div algorithm for image denoising

The operator with the highest computational cost in the above algorithm is the orthogonal projection, but with the aid of the fast Fourier transformation, the operators S and $S^T = S^{-1}$ or C and $C^T = C^{-1}$ only require $\mathcal{O}(N^2 \log_2 N)$ arithmetical operators. All other computations have the cost of $\mathcal{O}(N^2)$ arithmetical operators.

Stopping criteria

This section discusses the termination for Algorithm 1. The decision is based on computing the difference of the norm of the minimization functional or the residual to the iteration. If the difference is below a given tolerance, then the algorithm should terminate, such that the resulted image has a sufficient restored quality.

Measuring the dual energies for the TV-Stokes model in the equations (3.49) and (3.59) or for the Fourth-Order model in equation (3.15), and then to compute its relative error is an easy way to see if the iteration is stable. The dual energy function is given by $E : Y \rightarrow \mathbb{R}^+$ and is measured by $\| \cdot \|_Y$ from equation (1.21). The difference in the dual energy is then given by

$$\frac{\left| E(f^{n+1}) - E(f^n) \right|}{E(f^{n+1})} \leq tol \tag{3.112}$$

for a given tolerance.

The residuals from equations (3.81) and (3.97) will give a better measurement for convergence. The residuals will compute the difference at each pixel from the current iteration to the previous iteration. Thus, by looking at the difference

$$r = \frac{p^{n+1} - p^n}{k}, \tag{3.113}$$

and then measuring the residual by $\|r\|_X$, will give the residual number in each iteration. A given tolerance can then be used to stop the iteration, i.e to compute $\|r\|_X$ and check if it is below a given tolerance.

It should be pointed out that the Chambolle iteration is a fast method for achiving a denoised image, which is done in a few iterations. After 50-100 iterations the convergence speed gets slower i.e reducing the residual to e.g 10^{-1} or lower is a slower process. However, this is rarely needed, since the human eye stops notifying changes in the image when the residual is below 10^{-1}. For practical situations it is more natural to measure the energy, since the tolerance can be given as a magnitude.

The incompressibility condition in the first step is always fulfilled when solved by the orthogonal projection Π_P, such that measuring the objective function in the first step is a satisfactory measure for convergence. The $\|d^n - d^0\| = \sigma$ constraint in the second step, however, must be measured in each iteration in addition to the objective function. When both these estimates are below a given tolerance, we can say that the method has converged. If the noise-level σ is a known measure or an estimate of the real noise-level, we can get a fast convergence for the constraint by updating the smoothing parameter in each iteration, see the choice of smoothing parameter in 4.1.

Chapter 4

Numerical experiments

This chapter reports on the computational experiments for various test images with different degrees of difficulty. The numerical experiments will present a subset of all the the experiments that have been thoroughly studied during this thesis. The experiments are done mainly with the proposed algorithm described in the last section, as well as some brief comparison with other models based on total variation.

We need some tools to measure the efficiency for the different algorithms, so that we can judge the quality of the denoised image. The difference image will be used as a measure on how the algorithms preserve the edges, i.e if the edges are smoothed out by an algorithm, then the difference image will clearly contain edges. The difference image is given as $d^n - d^0$, where d^n is the n-th solution. For an optimal denoised image the difference image contains only noise.

The contour plots, `imcontour` in MATLAB, also help to evaluate the results as they reveal the level curves of the denoised image. As a quantitative measure, one can measure the restoration performance by the SNR that is given by

$$SNR = 20 \log_{10} \left(\frac{\int_\Omega (d - \overline{d})^2 \, \mathrm{d}x}{\int_\Omega (\eta - \overline{\eta})^2 \, \mathrm{d}x} \right), \tag{4.1}$$

where

$$\overline{d} = \frac{1}{|\Omega|} \int_\Omega d \, \mathrm{d}x, \quad \text{and} \quad \overline{\eta} = \frac{1}{|\Omega|} \int_\Omega \eta \, \mathrm{d}x. \tag{4.2}$$

The SNR will be reported for the noisy image. Finally there will be given a presentation on how the proposed algorithm compares in speed of convergence to the primal TV-Stokes algorithm. Using these tools we will be able to judge the efficiency of the different algorithms.

The basis of comparison will be equal for all the methods that are reported. In all the experiments the images are normalized into $[0, 1]$, where 0 is white and 1 is black. The only information that is given concerning the problem is the noisy image, i.e the exact same noisy image, and a new noisy will not be generated for each test. The noise-level σ will also be given, measured from the generated noise. When the noise-level is known, the smoothing parameter will be estimated such that the constraint $\|d^n - d^0\| \approx \sigma$ is fulfilled, this has the effect of choosing the most regular solution to the ill-posed problem. Thus, making

the comparison of the TV-Stokes model and the ROF model more correct, since they share the same quality of fidelity.

In practical situations the noise-level may be given as an estimate from statistical approaches, see for instance [29]. Many authors from the literature choose to find the smoothing parameter by trial-and-error, as this parameter is constant and may give a clearer indication on how a proposed algorithm performs. Finding the smoothing parameter is not an easy task and the parameter should be sufficiently small, i.e not over-smooth and lose edges.

We will use equation (3.112) to measure the dual energy difference in order to terminate the iteration. It should also be noted that all the algorithms are coded in MATLAB and it is likely that the performance would improve with a low-level programming language such as C or C++, especially exploiting the GPU (Graphical Processor Unit) with e.g CUDA which is a C-dialect for parallelling the GPU.

(a) Original image.   (b) Noisy image.
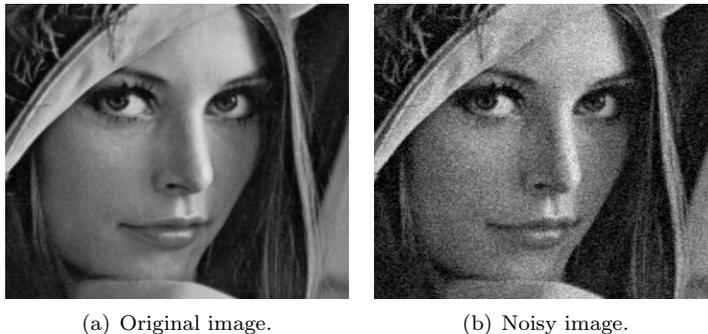
Figure 4.1: The original Lena image d and a noisy version $d^0 = d + \eta$, where $\|\eta\|_2 = \sigma = 10.9$ and $SNR \approx 16.0$.

The first numerical example is the well known Lena image, seen in Figure 4.1, that has been used in image processing since the seventies. Gaussian noise is added to the image, generated by the MATLAB function `imnoise` with zero mean and the variance paramter equal to 0.003. In Figure 4.2, the noisy image has been applied to the denoising algorithms, and the figures show how the different denoising algorithms perform. The next Figure 4.4 is a close up of Figure 4.2, to show the reader a subjective view on how the methods handle staircasing. The final experiment for the Lena image is showing how the dual TV-Stokes performs on convergence. All the models that have been given are solved by the dual formulation.

For the dual TV-Stokes method we have used δ equal to 0.06 and the timestep k equal to 1/4. The Figure 4.2.(a)-(b) shows that the TV-Stokes has performed quite well, and managed to completely remove the staircase while effectively smoothing out the noise and preserving the edges quite well. The Figure 4.2.(c)-(d) shows the dual TV-Stokes model with the identity operator for the orthogonal projection $\Pi_P = I$, i.e it solves the minimization on a non-constrained space. One see it is very close to the Figure 4.2.(a), and the respective Figures in 4.4 show a close up and where one can see that (c) has some small details smoothed out. The contour plots reveal this effect even more, since it is shown that the level curves are smoother when the div $\tau = 0$ condition is fulfilled.

The Fourth-Order method is shown in Figure 4.2.(e)-(f) and respectively in Figure 4.4. This method is also known to remove staircase effect as seen in the experiments, but it has the side effect that "black spots" are introduced in the restored image. This is also clearly seen in the contour plots, where the level curves are oscillating. Since the Fourth-Order method is a smoother solution, the restored image keeps the edges, but not as sharp as e.g the ROF model, thus resulting in a denoised image that may be more blurred.

The last Figures in 4.2 show how the classical ROF model performs. It is well known that this model introduces staircasing effect in images that are not piecewise constant, and this is clearly illustrated in the restored images.

The TV-Stokes model solved by the primal and the dual formulation has nearly the same quality. For the Lena image in Figure 4.3 the noise-level is equal to 14.0. The smoothing parameter δ is equal to 0.0835 and μ is equal to 0.17 for the dual TV-Stokes restored image shown in Figure 4.3.(b). For the primal TV-Stokes algorithm, δ was equal to 0.045. The Cameraman in Figure 4.3 is taken directly from the paper [34], where the SNR is the same as the one we report, $20 \log_{10}(8.21) \approx 18.28$. Figure 4.3.(e) is solved by the dual TV-Stokes with δ equal to 0.055 and μ equal to 0.08. Figure 4.3.(f) shows the primal TV-Stokes reconstruction for the same noisy image, where the δ parameter is equal to 0.06.

The main difference between the two formulations is that the dual formulation solves the problem more effectively. Clearly, from Table 4.1, we see that the dual formulation improves the speed drastically. The runtimes in the table was given for only one runtime, since computing an avarage of many runtimes is very time consuming for the TV-Stokes method.

Algorithm	Dual TV-Stokes algorithm		TV-Stokes algorithm, [34]	
Image	First step	Second step	First step	Second step
Lena	9.8	1.12	9083.2	1992.5
Cameraman	17.4	2.2	11189.0	2259.4
Barbara	128.2	20.7	80602.5	14926.3

Table 4.1: Runtimes of the dual TV-Stokes algorithm compared to the TV-Stokes algorithm [34]. The test system is a 2 Opteron 270 dualcore 64-bit processor and 8GB RAM. Both steps in the dual TV-Stokes algorithm are computed with 150 iterations, while the first step in the primal TV-Stokes algorithm is calculated with 75000 iterations and the second step with 25000 iterations.

We will again turn to the dual TV-Stokes model, and this time to see how fast the dual formulation performs. The energy plots in Figure 4.5.(a)-(b) and Figure 4.6.(a) show that the method is rapidly converging for the first 50-100 iterations, and then the performance is decreasing. The slow speed of convergence is well-known and there have been some research to improve the convergence rate. In [13] they proposed a multigrid method to accelerate the convergence, and by convergence rate analysis they showed that there are some difficulties for the nonlinear multigrid method. To improve the convergence they proposed a modified model that introduces regularization back into the iteration, as opposed to the Chambolle iteration, which manages to avoid regularization. This illustrates the difficulty of finding fast solvers for nonlinear PDES.

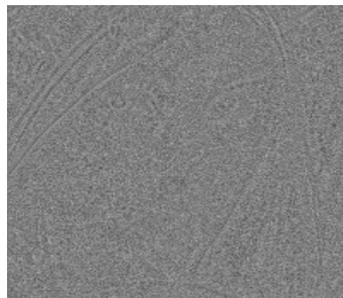The authors of [41] proposed various gradient projection algorithms to solve

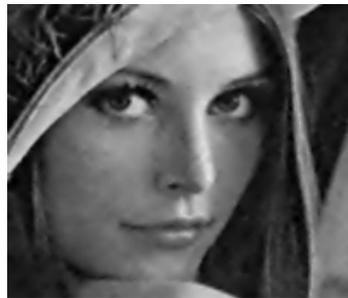(a) Denoised with the TV-Stokes model. Timestep k equal to 1/4.
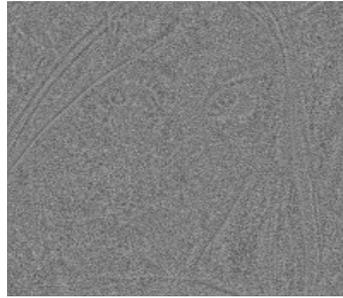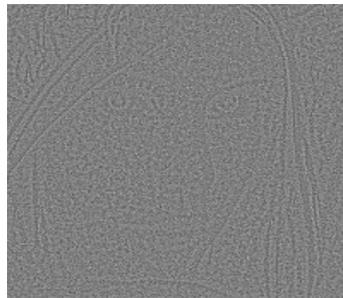
(b) Difference image of (a).

(c) Same parameters as (a) and $\Pi_P = I$.

(d) Difference image of (c).

(e) Denoised with the Fourth-Order model. Timestep τ equal to 1/16.

(f) Difference image of (e).

(g) Denoised with the ROF model. Timestep τ equal to 1/4.

(h) Difference image of (g).

Figure 4.2: Comparison of the Lena image with the following algorithms: the dual TV-Stokes model, the Fourth-Order model and the ROF model. The figures on the right show the corresponding difference image $d^n - d^0$.

(a) Noisy lena, $SNR \approx 14.0$. (b) Denoised with the dual TV-Stokes model. (c) Denoised with the primal TV-Stokes model.

(d) Noisy Cameraman. (e) Denoised with the dual TV-Stokes model. (f) Denoised with the primal TV-Stokes model.
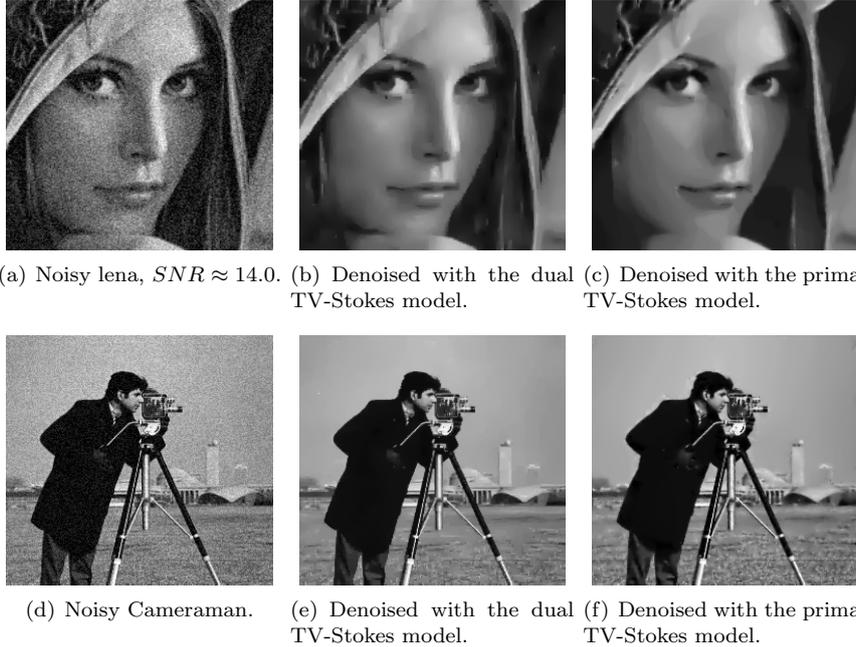
Figure 4.3: Comparison of the primal and dual TV-Stokes. The denoised quality for the primal and the dual model is nearly equal.

the quadratic minimization, e.g (2.32), with different strategies to select line-search and step-length. They report that some of the proposed approaches perform significantly faster, particularly when the accuracy for the residual is modest.

The next image, is the classical Cameraman image. The image is harder to restore than the Lena image due to the low intensity details and the flat background. Total variation based methods often struggle with these kind of features while denoising. Because of the smoothing, low intensity details get smeared out and the flat background is transformed into an area where the staircase effect dominates. This is easily seen in Figure 4.8 where the Figure 4.8.(c) is restored by the ROF method. Restoring the method with a higher-order method will improve the staircase effect in the background restoration, but fails to preserve the low intensity details, as this method provides a smoother solution. Figure 4.8.(a) shows the restored image by the Fourth-Order dual method, and the Cameraman reveals that the black spots, that higher-order methods are known to introduce, are presented in the restored result.

The dual TV-Stokes model also has some numerical difficulities with this image. The large amount of constant skyline in the background causes the algorithm to have more iterations, as the residual needs to be lower than e.g the Lena image. Besides the ineffective convergence, the restored quality is visually more pleasant for the TV-Stokes model than the ROF model or the Fourth-Order model, since the staircase effect does not appear in the final restored result, seen in the Figure 4.7.(e) with residual tolerance given as $tol = 1.0 \times 10^{-1}$.
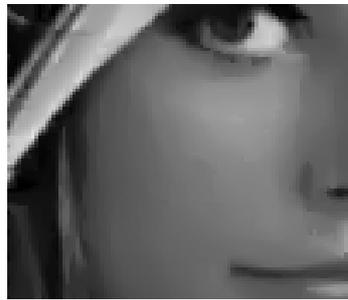
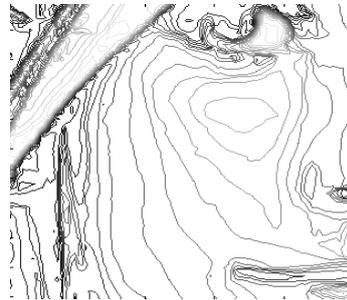The last example is the Barbara image, which is detailed image with high
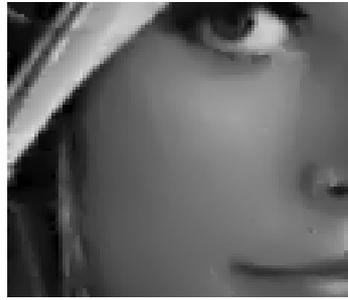
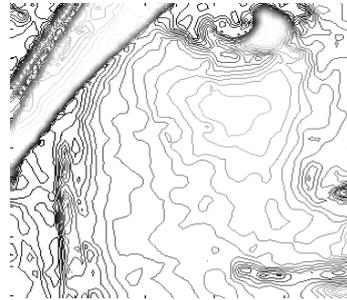(a) Denoised with the TV-Stokes model.
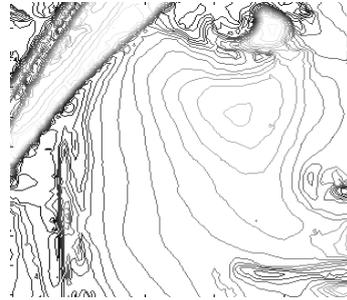
(b) Contour plot of (a).

(c) Denoised with $\Pi_P = I$.
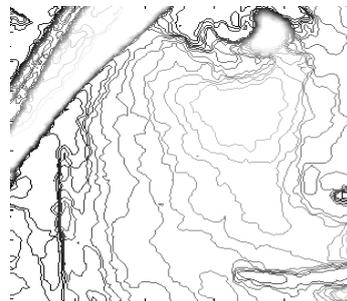
(d) Contour plot of (c).

(e) Denoised with the Fourth-Order model.
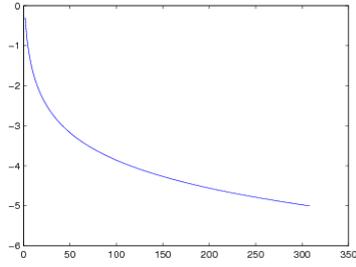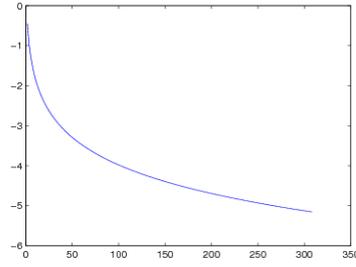
(f) Contour plot of (e).

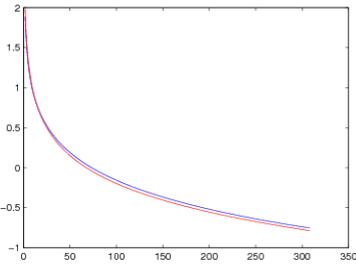(g) Denoised with the ROF model.

(h) Contour plot of (g).

Figure 4.4: Zoomed result for the restored images generated by the TV-Stokes (with and without Π_P), Fourth-Order model and the ROF model.
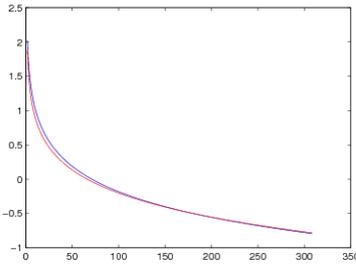
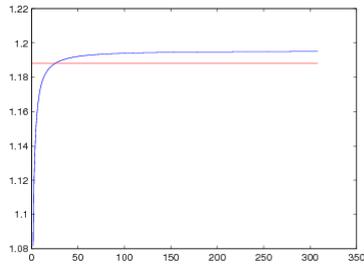(a) Difference in the energy for the $v-$ component $|E(p_1^{n+1}) - E(p_1^n)|/E(p_1^{n+1})$.

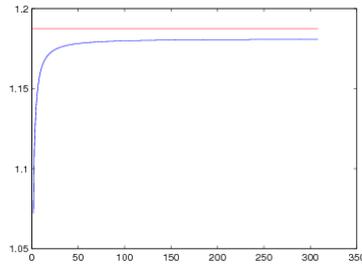(b) Difference in the energy for the $u-$ component $|E(p_2^{n+1}) - E(p_2^n)|/E(p_2^{n+1})$.

(c) Residual for the v component $v-$. p_1^1 (blue line) and p_1^2 (red line).

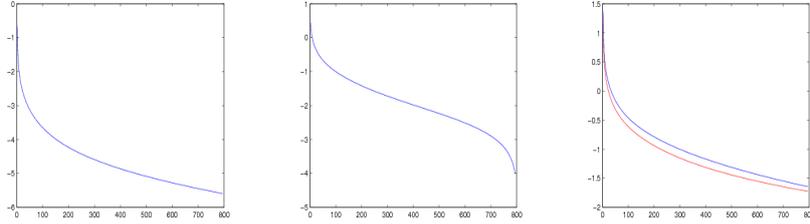(d) Residual for the $u-$ component. p_2^1 (blue line) and p_2^2 (red line).

(e) Constraint for the $v-$ component (blue line) and $\|\eta_v\|_2 = \sigma_1$ (red line).

(f) Constraint for the $u-$ component (blue line) and $\|\eta_u\|_2 = \sigma_2$ (red line).

Figure 4.5: Convergence behaviour for the tangential vector field $\tau = [v, u]^T$ of the first step. Smoothening parameter δ equal to 0.06 and timestep k equal to 1/4.

(a) Difference in the energy $|E(r^{n+1}) - E(r^n)|/E(r^{n+1})$.

(b) Relative error of the constraint, $|\sigma - e^n|/\sigma$, $e = \|d^n - d^0\|_2$.

(c) Residuals for the projections. r^1 blue line and r^2 red line.

Figure 4.6: Convergence behaviour of the second step. Energy tolerance is equal to 10^{-5} and timestep k is equal to $1/4$.

and low intensity textures. The high intensity textures and the smooth areas are preserved quite well, but the low intensity textures disappear in the same way as with the Cameraman. This image is 512×512 in size, which makes the algorithm slower because of the rather large matrix operations per iteration. However, achieveing a result for the optimal paramters is still obtainable since the method is stable after a few steps. Therefore one can run the method multiple times to find the optimal parameters. For this image we found the optimal δ to be equal 0.046.

4.1 Choice of the smoothing parameters

It is important to keep mandatory parameters to a minimum, such that automatic processing can be implemented and used in industrial cases.

The smoothing parameter δ in the first step controls the fidelity terms $\|v - v_0\| \approx \sigma_1$ and $\|u - u_0\| \approx \sigma_2$ where $[v, u]^T = \tau$ and $[v_0, u_0]^T = \tau_0$. In the experiments the smoothing parameter δ for the first step has been tested in the interval $0.01 \leq \delta \leq 0.1$. By starting with a small δ, the tangential vector field should contain some noise. Then, carefully increasing the δ such that the tangential vector field is sufficiently restored, the noise is smoothed out and the edges are restored. Using this tuning strategy, one should find the optimal δ for the first step.

The smoothing parameter μ for the second step controls the fidelity of $\|d^n - d^0\|_2 = \sigma$. If the parameter is chosen high, the resulted image is over-smoothed, and respectively under-smoothed if chosen low, as seen in Figure 4.10. The iteration starts with an under-smoothed image, i.e $\|d^i - d^0\|_2 \leq \sigma$ for $i = 0, 1 \ldots n$, and the algorithm gradually iterates to a restored image such that $\|d^n - d^0\| \approx \sigma$.

If the noise-level σ is a known measure or an estimated value, the $\mu > 0$ parameter can be updated in each iteration by considering the equality of the constrained and unconstrained problems (3.54) (3.56) such that μ^n is updated by the following iteration

$$\mu^n = \frac{\sigma}{\|\mathrm{div}^h (r^n + \mathbf{n})\|_2}. \tag{4.3}$$

(a) *tol* = 1.0, 74 iterations for the 1-step and 104 iterations for the 2-step.

(b) Contour plot of (a).

(c) *tol* = 5.0×10^{-1}, 131 iterations for the 1-step and 201 iterations for the 2-step.

(d) Contour plot of (c).

(e) *tol* = 1.0×10^{-1}, 552 iterations for the 1-step and 1381 iterations for the 2-step.
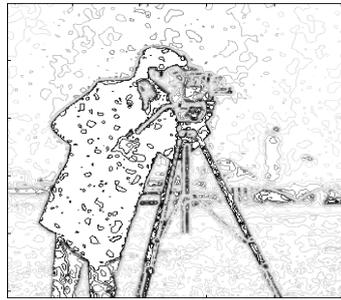

(f) Contour plot of (e).

Figure 4.7: Restored Cameraman with different residual tolerances. The figure clearly illustrates the need for low residual to give a sufficiently restored image. The figures on the right show the corresponding contour plots.

The iteration gives a fast convergence for the constraint.

If the noise-level is not a known measure or a good estimate can not be obtained, the μ parameter must be found by trial-and-error. This is normally done by choosing μ low, and then carefully increasing the parameter until the noise in the restored image is satisfactorily smoothed out. Anyhow, the difference should converge to zero, and is a useful estimate to see if the constraint has reached a stable state.
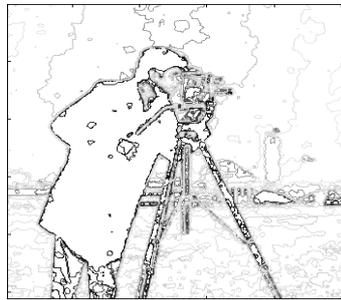
(a) Denoised with the Fourth-Order model. Timestep τ equal to 1/16.

(b) Contour plot of (a).

(c) Denoised with the ROF model. Timestep τ equal to 1/4.

(d) Contour plot of (c).

Figure 4.8: The Cameraman restored with the ROF model and the Fourth-Order model. Residual tolerance is equal to 1.0 for both models.
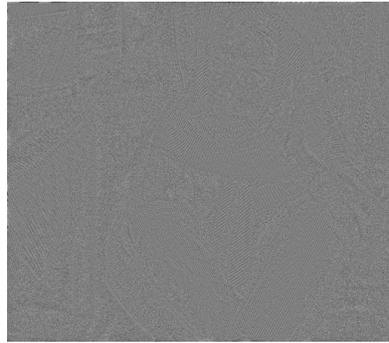
(a) Original image.

(b) Noisy image, $SNR \approx 20.0$.

(c) Denoised image.

(d) Difference image of (a) and (c).

(e) Zoomed result for the denoised image in (c).

(f) Contour plot of (e).

Figure 4.9: Barbara denoised with the dual TV-Stokes model. Residual tolerance is equal to 0.5×10^{-1} and smoothening paramter δ equal to 0.046.

(a) Under smoothed. (b) Optimal smoothed. (c) Over smoothed.

(d) $\|d - d_0\|_2 < \sigma$. (e) $\|d - d_0\|_2 \approx \sigma$. (f) $\|d - d_0\|_2 > \sigma$.
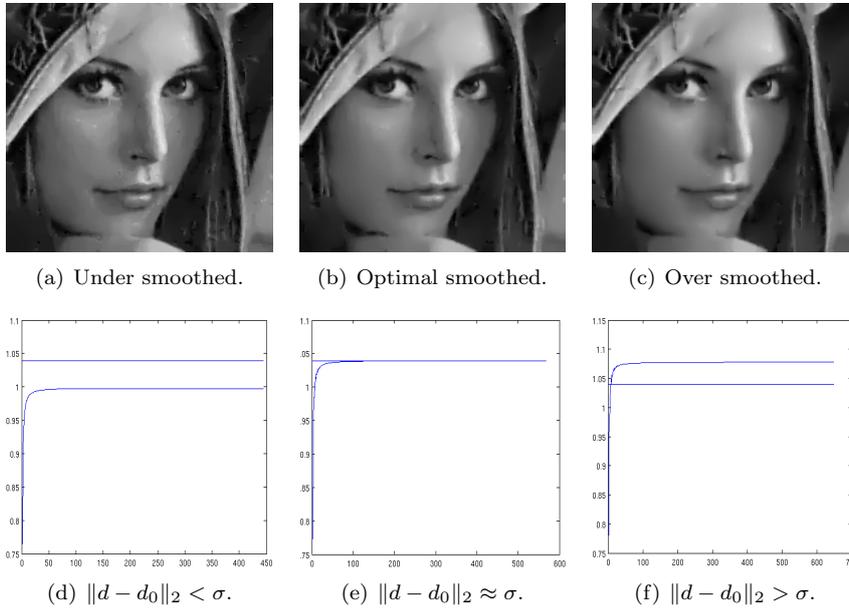
Figure 4.10: The figures illustrates different choices for the smoothening parameter μ for the second step. The straight line in the plots are the noise-level σ.

Bibliography

[1] Robert Acar and Curtis R. Vogel. Analysis of bounded variation penalty methods for ill-posed problems. In *Inverse Problems*, volume 10, pages 1217–1229, 1994.

[2] Gilles Aubert and Pierre Kornprobst. *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*, volume 147 of *Applied Mathematical Sciences*. Springer Verlag, New York, second edition, 2006.

[3] Marcelo Bertalmio, A.L. Bertozzi, and Guillermo Sapiro. Navier-stokes, fluid dynamics, and image and video inpainting. In *Computer Vision and Pattern Recognition*, volume 1 of *IEEE Computer Society Conference*, pages 355– I–362. IEEE Computer Society, 2001.

[4] Peter Blomgren, Tony F. Chan, and Pep Mulet. Extensions to total variation denoising. *SIAM Journal on Scientific Computing*, 20:1964–1977, 1999.

[5] Peter Blomgren, Tony F. Chan, Pep Mulet, and Chak-Kuen Wong. Total variation image restoration: Numerical methods and extensions. In *IEEE International Conference on Image Processing*, pages 384–387, 1997.

[6] Xavier Bresson and Tony F. Chan. Fast minimization of the vectorial total variation norm and applications to color image processing. *Computational and Applied Mathematics Report 07-25*, 2007.

[7] Lisa Gottesfeld Brown. A survey of image registration techniques. *ACM Computing Surveys*, 24:325–376, 1992.

[8] Jamylle L. Carter. *Dual methods for total variation-based image restoration*. PhD thesis, UCLA, 2001.

[9] Antonin Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1-2):89–97, 2004.

[10] Antonin Chambolle and Pierre-Louis Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76:167–188, 1997.

[11] T. Chan, S. Esedoglu, F. Park, and A. Yip. Recent developments in total variation image restoration. In *Mathematical Models of Computer Vision*, pages 17–32. Springer Verlag, 2005.

[12] Tony Chan, Antonio Marquina, and Pep Mulet. High-order total variation-based image restoration. *SIAM Journal on Scientific Computing*, 22(2):503–516, 2000.

[13] Tony F. Chan, Ke Chen, and Jamylle L. Carter. Iterative methods for solving the dual formulation arising from image restoration. *Electronic Transactions on Numerical Analysis*, 26:299–311, 2007.

[14] Tony F Chan and Selim Esedoglu. Aspects of total variation regularized l^1 function approximation. *SIAM Journal on Applied Mathematics*, 65(5):1817–1837, 2005.

[15] Tony F. Chan, Selim Esedoglu, and Frederick E. Park. A fourth order dual method for staircase reduction in texture extraction and image restoration problems. *Computational and Applied Mathematics Report 05-28*, 2005.

[16] Tony F. Chan, Gene H. Golub, and Pep Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 20(6):1964–1977, 1999.

[17] Tony F. Chan and Jian Hong (Jackie) Shen. *Image Processing and Analysis: variational, PDE, wavelet, and stochastic methods*. SIAM, Philadelphia, first edition, 2005.

[18] Tony F. Chan and Luminita A. Vese. A level set algorithm for minimizing the mumford-shah functional in image processing. In *IEEE Workshop on Variational and Level Set Methods*, page 161, Los Alamitos, CA, USA, 2001. IEEE Computer Society.

[19] Ward Cheney. *Analysis for Applied Mathematics*. Springer Verlag, New York, first edition, 2001.

[20] Philippe G. Ciarlet, Thomas Jean-Marie, and Miara Bernadette. *Introduction to numerical linear algebra and optimisation*. Cambridge texts in applied mathematics. Cambridge University Press, New York, 1989.

[21] Christoffer A. Elo, Alexander Malyshev, and Talal Rahman. A dual formulation of the TV-Stokes algorithm for image denoising. In *Scale Space and Variational Methods in Computer Vision*, volume 5567 of *Lecture Notes in Computer Science*, pages 307–318. Springer Verlag, 2009.

[22] Bernd Fischer and Jan Modersitzki. A unified approach to fast image registration and a new curvature based registration technique. *Linear Algebra and its Applications*, 380:107–124, 2004.

[23] Massimo Fornasier and Carola-Bibiane Schönlieb. Subspace correction methods for total variation and ℓ_1-minimization. *SIAM Journal on Numerical Analysis*, Submitted 2007.

[24] David Kincaid and Ward Cheney. *Numerical Analysis: Mathematics of Scientific Computing*. Brooks/Cole Publishing Company, Pacific Grove, third edition, 2002.

[25] William Litvinov, Talal Rahman, and Xue-Cheng Tai. A modified TV-Stokes model for image processing. *SIAM Journal on Scientific Computing*, Submitted 2008.

[26] Marius Lysaker, Arvid Lundervold, and Xue-Cheng Tai. Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time. *IEEE Transaction on Image Processing*, 12:1579–1590, 2003.

[27] Marius Lysaker, Stanley Osher, and Xue-Cheng Tai. Noise removal using smoothed normals and surface fitting. *IEEE Transaction on Image Processing*, 13(10):1345–1357, 2004.

[28] Simon Masnou and Jean michel Morel. Level lines based disocclusion. In *IEEE International Conference on Image Processing*, volume 3, pages 259–263. IEEE Computer Society, 1998.

[29] Peter Meer, Jean-Michel Jolion, and Azriel Rosenfeld. A fast parallel algorithm for blind estimation of noise variance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(2):216–223, 1990.

[30] David Mumford and Jayant Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989.

[31] Stanley Osher, Martin Burger, Donald Goldfarb, Jinjun Xu, and Wotao Yin. An iterative regularization method for total variation-based image restoration. *SIAM Journal on Multiscale Modeling and Simulation*, 4:460–489, 2005.

[32] Stanley Osher and James A. Sethian. Fronts propagating with curvature dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.

[33] Pietro Perona and Jitendra Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990.

[34] Talal Rahman, Xue-Cheng Tai, and Stanley Osher. A TV-Stokes denoising algorithm. In *Scale Space and Variational Methods in Computer Vision*, volume 4485 of *Lecture Notes in Computer Science*, pages 473–483. Springer Verlag, 2007.

[35] R. Tyrrell Rockafellar. *Convex analysis*. Princeton University press, New Jersey, 1972.

[36] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1-4):259–268, 1992.

[37] Walter Rudin. *Functional analysis*. International Series in Pure and Applied Mathematics. McGraw-Hill Inc., New York, second edition, 1991.

[38] Xue-Cheng Tai, Stanley Osher, and Randi Holm. Image inpainting using TV-Stokes equation. In *Image Processing Based on Partial Differential Equations*, Mathematics and Visualization, pages 3–22. Springer Verlag, Heidelberg, 2006.

[39] Lloyd N. Trefethen and David Bau. *Numerical Linear Algebra*. SIAM, Philadelphia, third edition, 1997.

[40] Curtis R. Vogel and Mary E. Oman. Iterative methods for total variation denoising. *SIAM Journal on Scientific Computing*, 17(1):227–238, 1996.

[41] Mingqiand Zhu, Stephen J. Wright, and Tony F. Chan. Duality-based algorithms for total-variation regularized image restoration. *Computational and Applied Mathematics Report 08-33*, 2008.