

https://dspace.org/

On the application of partial differential equations and fractional partial differential equations to images and their methods of solution

Jacobs, Byron

2014-08-11
http://hdl.handle.net/10539/15144

Downloaded from a DSpace repository. https://dspace.org/

University of the Witwatersrand

Doctoral Thesis

On the Application of Partial Differential Equations and Fractional Partial Differential Equations to Images and Their Methods of Solution

Author:
Byron Jacobs

Supervisors:
Dr. Charis Harley
Prof Ebrahim Momoniat

A thesis submitted in fulfilment of the requirements
for the degree of Doctor of Philosophy

in the

Centre for Differential Equations, Continuum Mechanics and Applications
School of Computational and Applied Mathematics

May 2014

Declaration of Authorship

I, Byron Jacobs, declare that this thesis titled, 'On the Application of Partial Differential Equations and Fractional Partial Differential Equations to Images and Their Methods of Solution' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: 

Date: 22/05/2014

"The mathematician's best work is art, a high perfect art, as daring as the most secret dreams of imagination, clear and limpid. Mathematical genius and artistic genius touch one another."

Gosta Mittag-Leffler

UNIVERSITY OF THE WITWATERSRAND

Abstract

Faculty of Science
School of Computational and Applied Mathematics

Doctor of Philosophy

On the Application of Partial Differential Equations and Fractional Partial Differential Equations to Images and Their Methods of Solution

by Byron Jacobs

This body of work examines the plausibility of applying partial differential equations and time-fractional partial differential equations to images. The standard diffusion equation is coupled with a nonlinear cubic source term of the Fitzhugh-Nagumo type to obtain a model with diffusive properties and a binarizing effect due to the source term. We examine the effects of applying this model to a class of images known as document images; images that largely comprise text. The effects of this model result in a binarization process that is competitive with the state-of-the-art techniques. Further to this application, we provide a stability analysis of the method as well as high-performance implementation on general purpose graphical processing units. The model is extended to include time derivatives to a fractional order which affords us another degree of control over this process and the nature of the fractionality is discussed indicating the change in dynamics brought about by this generalization. We apply a semi-discrete method derived by hybridizing the Laplace transform and two discretization methods: finite-differences and Chebyshev collocation. These hybrid techniques are coupled with a quasi-linearization process to allow for the application of the Laplace transform, a linear operator, to a nonlinear equation of fractional order in the temporal domain. A thorough analysis of these methods is provided giving rise to conditions for solvability. The merits and demerits of the methods are discussed indicating the appropriateness of each method.

Acknowledgements

I would like to thank my supervisors Dr. Charis Harley and Prof Ebrahim Momoniat for their relentless guidance, endless patience and unwavering faith in me. I would like to acknowledge all the support, useful discussions and countless debates with my colleagues.

Contents

Declaration of Authorship i

Abstract iii

Acknowledgements iv

List of Figures viii

List of Tables x

1 Introduction 1

2 Methodologies 8
2.1 Introduction . . . 8
2.2 Fractional Calculus . . . 8
2.2.1 Riemann-Liouville Integration . . . 8
2.2.2 Caputo Derivative . . . 9
2.2.3 Modified Riemann-Liouville Fractional . . . 9
2.2.4 Laplace Transform . . . 9
2.3 Finite-Differences . . . 10
2.3.1 Explicit Finite-Difference Discretization . . . 10
2.3.2 Crank-Nicolson Discretization . . . 10
2.3.3 Alternating Direction Implicit Method . . . 11
2.3.4 Grünwald-Letnikov . . . 11
2.4 Chebyshev Collocation . . . 12
2.5 Boundary Conditions . . . 13
2.5.1 Dirichlet Boundary Conditions . . . 13
2.5.2 Neumann Boundary Conditions . . . 13
2.6 Quasi-Linearization . . . 14

3 A Novel Approach to Text Binarization via a Diffusion-Based Model 16
3.1 Introduction . . . 17
3.2 Model Derivation and Discussion . . . 19
3.2.1 One Dimensional Case . . . 21
3.2.1.1 Derivation . . . 21

3.2.1.2 Stability Analysis . . . 22
3.2.1.3 Implementation . . . 23
3.2.2 Two Dimensional Case . . . 25
3.2.2.1 Derivation . . . 26
3.2.2.2 Stability Analysis . . . 27
3.2.2.3 Implementation . . . 27
3.3 Experimental Results . . . 29
3.4 Conclusion . . . 31

4 A Locally Adaptive, Diffusion Based Text Binarization Technique 37
4.1 Introduction . . . 38
4.2 Model Derivation . . . 40
4.2.1 Diffusion and a Source Term . . . 40
4.2.2 Localizing the Model . . . 41
4.3 Implementation: A Massively Parallel Approach . . . 42
4.3.1 Algorithm Description . . . 44
4.4 Experimental Results . . . 45
4.4.1 Evaluation Measures . . . 45
4.4.2 Results . . . 46
4.5 Conclusion . . . 48

5 A Comparison of Two Hybrid Methods for Solving Linear Time-Fractional Partial Differential Equations on a Two-Dimensional Domain 49
5.1 Introduction . . . 50
5.2 Preliminaries . . . 51
5.3 Methods: Semi-Discrete Hybrid Transform Method . . . 52
5.3.1 Chebyshev Collocation . . . 53
5.3.1.1 Dirichlet Boundary Conditions . . . 55
5.3.1.2 Neumann Boundary Conditions . . . 56
5.3.2 Finite Difference Discretization . . . 56
5.3.2.1 Dirichlet Boundary Conditions . . . 57
5.3.2.2 Neumann Boundary Conditions . . . 57
5.4 Analysis . . . 58
5.4.1 Solvability . . . 58
5.4.1.1 Finite Difference Scheme . . . 58
5.4.1.2 Chebyshev Collocation . . . 59
5.4.2 Accuracy . . . 60
5.4.3 Finite Difference Scheme . . . 60
5.4.4 Chebyshev Collocation . . . 60
5.5 Results . . . 62
5.5.1 Example 1: One-Dimensional Time-Fractional Diffusion Equation with Homogenous Neumann Boundary Conditions . . . 62
5.5.2 Example 2: Diffusion Advection Equation with Dirichlet Boundary Conditions . . . 63
5.5.3 Example 3: Two-Dimensional Time-Fractional Diffusion Equation with Homogenous Dirichlet Boundary Conditions . . . 65
5.6 Discussion . . . 65

5.7 Conclusions 66

6 Application of Hybrid Laplace Method to Nonlinear Time-Fractional Partial Differential Equations via Quasi-Linearization 68
6.1 Introduction 69
6.2 Preliminaries 71
6.3 Methods 71
6.3.1 Chebyshev Collocation 72
6.3.1.1 Dirichlet Boundary Conditions 74
6.3.1.2 Neumann Boundary Conditions 75
6.4 Results 76
6.4.1 Example 1 76
6.4.2 Example 2 77
6.4.3 Example 3 77
6.4.4 Example 4 78
6.5 Image Processing Application 80
6.6 Discussion 81
6.7 Conclusions 82

7 Conclusion 84
7.1 Document Image Binarization 84
7.2 Fractional Order Partial Differential Equations 85
7.3 Conclusion 87

Bibliography 88

List of Figures

3.1 Fitzhugh-Nagumo Type Source Term with arrows indicating the directionality of the forcing effect. 20
3.2 Efficacy of proposed method on a one dimensional example of a Heaviside function with SNR of 10dB. 24
3.3 Binarized result of proposed method (solid purple line) and binarized noisy input (dashed blue line). 25
3.4 Binarization of an (a) initial noisy block, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result. 31
3.5 Binarization of an (a) initial noisy text sample, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result. 32
3.6 Binarization of an (a) initial noisy newspaper clip, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result. 32
3.7 Binarization of an (a) initial noisy restaurant bill, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result. 33
3.8 Binarization of an (a) initial noisy newspaper article, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result. 34
3.9 Binarization of an (a) initial noisy novel extract, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result. 34
3.10 Binarization of a (a) novel extract, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result. 35

4.1 (a) Sample image of a novel excerpt taken with a mobile device. (b) Ground truth image. (c) Sauvola's binarized result. (d) Our proposed method. 46
4.2 (a) Sample image of a novel excerpt taken with a mobile device. (b) Ground truth image. (c) Sauvola's binarized result. (d) Our proposed method. 46
4.3 (a) Sample image of a restaurant bill taken with a mobile device. (b) Ground truth image. (c) Sauvola's binarized result. (d) Our proposed method. 47

5.1 Integration Contour in the Real-Imaginary Plane Non-Intersecting with the lower bound for $\alpha = 0.75$. 59

5.2 Integration Contour in the Real-Imaginary Plane Non-Intersecting with the lower bound for $\alpha = 0.75$ and $N_x = 5$. 61
5.3 Integration Contour in the Real-Imaginary Plane Non-Intersecting with the lower bound for $\alpha = 0.75$ and $N_x = 20$. 62
5.4 Log Plot Illustrating the Diminishing Average Error in the Chebyshev Collocation Scheme with Time for Example 1. 64
5.5 Log Plot Illustrating the Diminishing Average Error in the Finite Difference Scheme with Time for Example 1. 64

6.1 Plot of $u(x = 0, t)$ for various values of α. 77
6.2 Plot of $u(x = 0, t)$ for various values of α. 78
6.3 Plot of $u(x = 0, y = 0, t)$ for various values of α. 79
6.4 Plot of $u(x = 0, y = 0, t)$ for various values of α. 80
6.5 Figures depicting results obtained at increasing time using the current method and $\alpha = 1$. (a) The original image, (b) Binarized original image at $t \approx 0$, (c) Processed binary image at $t = 0.003$, (d) Processed binary image at $t = 0.008$, (e) Over diffused image at $t = 0.01$. 81
6.6 Figures depicting results obtained at increasing time using the current method and $\alpha = 0.2$. (a) The original image, (b) Processed image at $t = 0.01$, (c) Processed binary image at $t = 0.001$, (d) Processed binary image at $t = 0.01$ 82

List of Tables

3.1 Table Listing Percentage of Character Correctly Identified. 35
3.2 Table Listing Percentage of Pixels Accurately Binarized. 35
3.3 Table Listing the l_1 Error Norm Between Obtained Image and Ground Truth. 36

4.1 Table of various performance measures for a variety of datasets. 47
4.2 Table indicating the maximal and minimal performance measures of all competition entrants for each year. 47

5.1 Maximum absolute error in the presented method's solution of the problem described by Example 1 at time $t = 0.5$. 63
5.2 Maximum absolute error in the presented method's solution of the problem described by Example 2 in the Laplace domain at $s = 50$. 65
5.3 Maximum absolute error in the presented method's solution of the problem described by Example 3 at time $t = 0.5$. 66

6.1 Maximum absolute error in the presented method's solution of the problem described by Example 1 at time $t = 0.5$. 76
6.2 Maximum absolute error in the presented method's solution of the problem described by Example 2 at time $t = 0.5$. 78
6.3 Maximum absolute error in the presented method's solution of the problem described by Example 3 at time $t = 0.5$. 79
6.4 Maximum absolute error in the presented method's solution of the problem described by Example 4 at time $t = 1$. 80

Dedicated to my family; my brother Jonathan and my mother Gail who supported me throughout my life and studies and for their endless pride. I would like to thank my wife Leigh without whom I would surely have never been writing this. I love you all.

Chapter 1

Introduction

"Anyone who cannot cope with mathematics is not fully human. At best he is a tolerable subhuman who has learned to wear shoes, bathe and not make messes in the house."

Robert Heinlein

This thesis approaches image processing via an investigation of partial differential equations. Many researchers have presented algorithmic approaches to image processing [1–4], defining techniques based on a set of rules with great success. Physically derived methods also hold a place in the field varying from imposing a diffusion equation that prefers to diffuse along a sharp gradient rather than across it [5] to reconstructing a degraded image by treating the image gradient field as a fluid and allowing the fluid to "fill-in" once lost information [6]. Ebihara et. al. [7] for instance implement a reaction-diffusion model as a method for image segmentation and edge detection.

We follow the approach of applying physically meaningful models to an input image by tailoring a model that exhibits desirable traits. The Fithugh-Nagumo equation [8, 9] originally used to describe the propagation of an impulse exhibits thresholding properties. Interestingly this source term exhibits a binarizing property when applied to an image. Diffusive processes are typically destructive, however coupling a diffusion process with the aforementioned source-term results in a process which thresholds an input image while smoothing noise and preserving details present in the given data.

Given our approach to the processing of an image we need to also consider the meaning of concepts such as differentiation and integration within this context. The concept of a derivative is perhaps one of the most fundamental in the field of mathematical sciences. This concept has physical meaning and interpretation is direct and apparent; a rate

of change, the relationship of distance moved to velocity to acceleration. Seemingly a completely different concept is integration, given the physical meaning of the area under a curve. Obviously there are many different geometric and physical interpretations for the concept of differentiation and integration, however that discussion is not relevant for our current work. Rather we consider the slightly more uncommon concept of fractional differentiation and fractional integration. The idea of fractional differentiation is attributed first to a series of letters written between Leibnitz and L'Hopital in 1695, in which L'Hopital asks the pertinent question when referring to $\frac{d^n y}{dx^n} \equiv D^n y$:"What if n = $\frac{1}{2}$?", to which Leibnitz wrote prophetically, "...Thus it follows that $d^{\frac{1}{2}}x$ will be equal to $x\sqrt[2]{\mathrm{dx} : x}$, an apparent paradox, from which one day useful consequences will be drawn." [10, 11] A simple enough question to ask is does the operator $D^{\frac{1}{2}}$ exist, such that when applied twice we get $(D^{\frac{1}{2}})^2 = D$? The answer to this is yes. Another question that is concisely answered is, given D^n what happens if n is negative? The operator described is nothing other than a fractional integral similar to the duality of integer differentiation and integration, where integration is the inverse operation to differentiation. This leads us immediately to the term 'differintegration' or 'differintegral', as dubbed by Oldham and Spanier in [11], notated by D^n where n is no longer constrained to be a positive integer.

There are three main approaches to fractional derivatives; the Riemann-Liouville derivative, the Caputo derivative and more recently the modified Riemann-Liouville derivative introduced by Jumarie in [12]. Jumarie substantiates the need for this derivative as a means for overcoming some shortfalls of the other two approaches and it's consistency with integer order derivatives. For example in order to compute the first order Caputo derivative one must already know the second order derivative [13]. The Riemann-Liouville derivative also exhibits inconsistencies such as the derivative of a constant is not 0. Jumarie has made extensive use of his modified Riemann-Louiville derivative and accompanying generalized Taylor Series for investigations of Brownian motion and Poisson processes to a fractional order, for solving stochastic differential equations governed by fractional Brownian motion, for solving fractional partial differential equations and developing a Fourier Transform of fractional order [12–17]. With this in mind the focus of our work is based primarily on the Caputo derivative due to its popularity amongst researchers particularly in the field of partial differential equations. Moreover the use of the Caputo derivative allows one to formulate problems with integer order boundary conditions where the Riemann-Liouville derivative requires fractional order boundary conditions which are difficult to interpret physically. The Caputo derivative also enjoys an elegant transformation by the Laplace transform making it an appealing choice to work with. These derivatives are discussed further in Chapter 2.

By generalizing the order of differentiation we may begin looking at fractional order partial differential equations and their applications. Hilfer collates a series of research papers on the application of fractional calculus in physics in [18]. For example diffusion in a disordered media can be interpreted as a fractional diffusion process. Many researches have interpreted anomalous diffusion via Brownian motion which is simplified to the ordinary diffusion equation on a macro scale. Havlin [19] says in uniform Euclidean space, the mean square displacement (MSD) of a random walker, $\langle R^2(t)\rangle$ is proportional to time for any number of spatial dimensions (Fick's law). However, Havlin goes on to mention that in disordered systems this law is not valid and the diffusion process becomes anomalous. As such we find that

$$\langle R^2(t)\rangle \sim t^{\alpha} \tag{1.1}$$

where for $\alpha < 1$ we have sub-diffusion, $\alpha = 1$ is the case of regular diffusion and $\alpha > 1$ implies super-diffusion. This is another meaningful interpretation of fractional diffusion: diffusion in a disordered media.

Chen and Holm [20] briefly allude to the physical significance of sub- and super-diffusion. Compounding the physical meaning of diffusion equations, Cattaneo's research [21] derives a more accurate model for diffusion. One explanation of sub-diffusion is a medium exhibiting diffusive memory, that is a medium that diffuses differently based on previous events. This may allow a more intelligent diffusion based denoising algorithm when compared to the usual local linear diffusion, which is why considering the fractional Fitzhugh-Nagumo equation is of interest.

The meaning and importance of fractional partial differential equations is wide ranging. Compte and Metzler [22] for example give derivations of other fractional partial differential equations, the generalized Cattaneo equations, which are used in the description of anomalous transport. Metzler and Klafter [23, 24] give excellent reviews on developments within the field of anomalous transport with focus on the importance of fractionality. Cattaneo [21] suggested that the classical heat equation is unphysical for short time diffusion processes because of the fact that, according to the heat equation, changes in temperature are instantly felt globally in the diffusive media, rather than locally. This implies an infinite velocity of propagation. To remedy this apparent flaw Cattaneo suggested introducing a wave term with small coefficient to model a finite velocity of propagation. Compte and Metzler [22] also mention that some fractional Cattaneo equations reproduce features that can be derived from other schemes such as the continuous time random walk (CTRW) picture or a non-local flux concept. This idea of non-locality is reiterated throughout the literature. In deriving the generalized

Cattaneo equations, Compte and Metzler [22] also mention that when pertaining to diffusion theory a time-fractional term would imply a non-conservation of heat or particle number. Several authors have investigated the use of fractional-order equations in mathematical modelling [25–30]. Bonilla et. al. [31] suggest the use of fractional-order models as a replacement for nonlinear models.

Chen and Holm [20] give a physical interpretation of the fractional diffusion-wave equation, attributing the fractionality of the equation to taking account of non-conservative systems and potential non-local and memory effects on energy dissipations of which medical ultrasonic wave propagation is an example. *Fractional Differential Equations* by Igor Podlubny [32] reiterates this interpretation. Podlubny [33] echoes the sentiment of non-local and non-conservative effects. Chen and Holm [20] go on to show that with regards to the application of the diffusion-wave equation, sub-diffusion does not agree with the power law except in physically rare circumstances. Furthermore they go on to say that super-diffusion is physically meaningful, for example the diffusion of cellular materials out of a cell aided by an active process.

In addition to his book [32], Podlubny published [33] which develops a new geometric and physical interpretation of fractional differintegration. Podlubny gives the left-sided Riemann-Liouville fractional integral, in the following form:

$$ {}_0I_t^{\alpha} = \int_0^t f(\tau) dg_t(\tau) \tag{1.2} $$

where $g_t(\tau)$ is given the physical meaning of mapping perceived time to absolute time. This follows from an in depth discussion in support of this interpretation. The example that Podlubny gives is based on the assumption that absolute time is not homogenous, and so if one were to integrate the velocity of an object with regards to absolute time, we can find the absolute distance covered, since the fractional integral is essentially integrating over the mapping of perceived (local) time to absolute (cosmic) time. Meerschaert and Tadjeran [34] give the following physical meaning to the space fractional derivative: Fractional space derivatives are used to model anomalous diffusion or dispersion, where a particle plume spreads at a rate inconsistent with the classical Brownian motion model. However, this work focuses on fractional ordered derivatives in the temporal domain and not the spatial domains.

Within the context of this thesis we require computational methods for the solution of fractional partial differential equations. Acedo and Yuste [35] present an explicit-finite difference method for fractional diffusion equations as well as giving a new von Neumann-Type stability analysis. The method is a foward time centered space (FTCS)

discretization scheme, to which they perform the stability analysis. They conclude that their method is unconditionally consistent, also mentioning that second-order accuracy is attainable using the FTCS scheme for the standard diffusion equation but not for the fractional case because of the fractional operator. Following this publication Tadjeran and Meerschaert published "A second-order accurate numerical method for the two-dimensional fractional diffusion equation" [36] in which they showcase a method conglomerating the alternating directions implicit (ADI) [37] scheme with a Crank-Nicolson discretization and a Richardson extrapolation yielding a method that is unconditionally stable and second-order accurate. By using the Lax equivalence theorem the authors show this method to be convergent. Many other researchers have presented numerical approaches such as predictor-corrector methods, high-order schemes, ADI schemes and Runge-Kutta type methods [38–45]. In addition to numerical approaches, some authors are able to derive exact solutions to some fractional order equations. El-Kalhout et. al. [46] derive exact solutions to a class of time-fractional partial differential equations via the Mellin transform. Saxena et. al. [47] give closed form solutions to the fractional reaction and fractional diffusion equations in the form of Fox and Mittag-Leffler functions via asymptotic expansion. Many researchers have extended analytic methods such as the differential transform method, Adomian decomposition method and He's variational iteration method which provide closed form solutions to partial differential equations to a fractional order in space and time as well as being able to handle some nonlinearities [48–55]. While these methods are limited in some ways in terms of capability of dealing with specific boundary conditions the analytic solutions obtained by these methods provide an excellent framework with which one may compare numerical results.

Although diffusive partial differential equations have been thoroughly investigated in image processing as denoising techniques [56–62] there has been very little work done on the application of fractional partial differential equations to image processing. Bai and Feng [63] present a fractional-order anisotropic process for image denoising by extending the Perona-Malik equation [5] to include fractional-order derivatives via the energy functional. They then applied the algorithm using two dimensional discrete fourier transforms. This opens a field of research wherein one can employ physically substantiated diffusion-type processes in an attempt to obtain a desirable result in image processing.

Extending this concept, of applying partial differential equations to images, to include fractional order derivatives broadens the scope of equations that are of interest. If we consider a fractional order partial differential equation

$$\frac{\partial^\alpha u}{\partial t^\alpha} = \frac{\partial^\beta u}{\partial x^\beta}, \tag{1.3}$$

with $\alpha = 1$ and $\beta = 2$ this equation reduces to the classical diffusion equation. However, if $\beta = 2$ and $1 < \alpha < 2$ then the equation exhibits propagational effects as α approaches 2 and the equation, correspondingly, approaches the wave equation. Similarly if $\alpha = 1$ and $1 < \beta < 2$ the equation begins to resemble the simple advection equation, where again propagational effects are observed. This is meaningful because preserving the spatial topography of an image is imperative to maintaining the useful information in that image. It is for this reason that we focus on the diffusion equation in two-dimensions, since the introduction of an advective term into the model would propagate information. Similarly considering the time-fractional diffusion equation with $1 < \alpha < 2$ we obtain the diffusion-wave equation, which again has propagational properties. Hence we restrict our model to being time-fractional only and $0 < \alpha \leq 1$.

Although there has been extensive research done [64–71] on pseudo-spectral methods, more specifically Chebyshev collocation there has been very little attention paid to Chebyshev collocation applied to fractional order differential equations. Bueno-Orovio et. al. [72] however, introduce a Fourier spectral method for the integration of space fractional reaction diffusion equations.

This thesis is structured as follows: Chapter 2 presents the methods that are put to extensive use throughout this work.

In Chapter 3 we present a novel diffusion based method for the binarizing of noisy document images. Document images are images of text, either written or typed but are typically noisy. This method is the marriage of the linear diffusion process, which has denoising properties, and the Fitzhugh-Nagumo source term which has binarization properties. The result is a method that is isotropic, i.e. it acts homogenously in all directions, is insensitive to noise and produces excellent results when compared to the state-of-the-art methods in the field.

Chapter 4 extends the binarization model presented in Chapter 3 to an adaptive scheme. This extension allows the application of our method to images with non-uniform illumination typically associated with photographed documents, where the flash from a camera may light an image more intensely in the center of the image and less so at the boarders. This adaptive structure also allows the method to perform consistently even with shadows cast on to the document. Once more this process is isotropic, a simple extension of the global case and produces excellent results.

We consider the linear time-fractional model in Chapter 5 and present the difficulties associated with solving fractional-order partial differential equations. The Laplace

transform is used to reduces the fractional-order derivative to an algebraic constraint. Due to the linearity of the Laplace transform we are unable to apply the Laplace transform to the nonlinear source term and hence Chapter 5 examines the feasibility of this method for solving time-fractional partial differential equations as well as comparing two discretization methods for different types of boundary conditions.

Extending the approach presented in Chapter 5 we introduce a quasi-linearization technique in Chapter 6 which enables us to apply the linear technique to a nonlinear time-fractional partial differential equation. We employ Chebyshev collocation as a discretization based on results presented in Chapter 5 and conclude by addressing some shortcomings of this method. The application of this new methodology within the context of binarizing images is discussed.

Chapter 7 concludes by discussing the importance of this body of work as well as highlighting the cohesion between chapters.

Chapter 2

Methodologies

"Science is a wonderful thing if one does not have to earn one's living at it."

Albert Einstein

2.1 Introduction

The purpose of this Chapter is to introduce the methodologies used throughout this work and as a reference. Definitions of some less common operators and results are given as well as useful discretizations and descriptions of methods.

2.2 Fractional Calculus

The Riemann-Liouville fractional derivative and integral form the basis for the Caputo derivative. The definitions of the Riemann-Liouville derivative and Caputo derivates are given in [11] and [32].

2.2.1 Riemann-Liouville Integration

Definition 1. The Riemann-Louiville integral of order $\gamma > 0$ of a function $u(t)$ is [32]

$$J^{\gamma}u(t) = \frac{1}{\Gamma(\gamma)} \int_0^t (t-\tau)^{\gamma-1} u(\tau) d\tau, \quad x > 0. \tag{2.1}$$

Therefore the Riemann-Liouville fractional derivative may be written as follows.

Definition 2. The Riemann-Louiville derivative of order γ, where $m \leq \gamma < m+1$ of a function $u(t)$ is [32]

$$D^{m+1}J^{\gamma-m}u(t) = \frac{d^{m+1}}{dt^{m+1}}\frac{1}{\Gamma(\gamma-m)}\int_0^t (t-\tau)^{\gamma-m-1}u(\tau)d\tau, \quad x > 0. \tag{2.2}$$

2.2.2 Caputo Derivative

In this work we employ Caputo's definition of a fractional derivative [73] over the Riemann-Louiville derivative due to the fact that the Caputo derivative makes use of the physical boundary conditions, whereas the Riemann-Louiville derivative requires fractional order boundary conditions.

Definition 3. The fractional derivative of $u(t)$ according to the Caputo definition with $m-1 < \alpha \leq m$, $m \in \mathbb{N}$, is [32]

$$\frac{\partial^\alpha u(t)}{\partial t^\alpha} = J^{m-\alpha}D^m u(t) = D_*^\alpha u(t) := \frac{1}{\Gamma(m-\alpha)}\int_0^t (t-\tau)^{m-\alpha-1}u^{(m)}(\tau)d\tau. \tag{2.3}$$

If $\alpha = 1$ the Caputo fractional derivative reduces to the ordinary first order derivative.

2.2.3 Modified Riemann-Liouville Fractional

Definition 4. Jumarie [12] introduces the modified Riemann-Liouville fractional derivative as

$$u^{(\alpha)}(t) = \lim_{\triangle t \to 0}\frac{\triangle^\alpha u(t)}{\triangle t^\alpha}, \tag{2.4}$$

and

$$\triangle^\alpha u(t) = \sum_{k=0}^{\infty}(-1)^k\binom{\alpha}{k}u\left[t+(\alpha-k)\triangle t\right]. \tag{2.5}$$

2.2.4 Laplace Transform

Podlubny [32] illustrates the pleasing property of the Laplace Transform of a Caputo derivative, as can be seen in Eq. (2.6). In our case where $0 < \alpha < 1$ we have,

$$\mathcal{L}\left\{\frac{\partial^\alpha u(x,y,t)}{\partial t^\alpha}\right\} = s^\alpha U(x,y) - s^{\alpha-1}u(x,y,0). \tag{2.6}$$

This property allows one to treat fractional order derivatives algebraically.

Definition 5. The Generalized Mittag-Leffler function of the argument z

$$E_{\alpha,\beta}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(k\alpha + \beta)}. \tag{2.7}$$

2.3 Finite-Differences

2.3.1 Explicit Finite-Difference Discretization

In two spatial dimensions we use a central finite difference scheme to approximate the first and second derivatives. The first derivative is typically used to impose derivative boundary conditions and the second derivative is used in the diffusion equation. The first derivative in the x and y directions are

$$\frac{\partial u}{\partial x} \approx \frac{u^n_{i+1,j} - u^n_{i-1,j}}{2\triangle x} + \mathcal{O}(\triangle x^2) \tag{2.8}$$

and

$$\frac{\partial u}{\partial y} \approx \frac{u^n_{i,j+1} - u^n_{i,j-1}}{2\triangle y} + \mathcal{O}(\triangle y^2) \tag{2.9}$$

respectively and for the second derivative we have in the x and y directions

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u^n_{i+1,j} - 2u^n_{i,j} + u^n_{i-1,j}}{\triangle x^2} + \mathcal{O}(\triangle x^2) \tag{2.10}$$

and

$$\frac{\partial^2 u}{\partial y^2} \approx \frac{u^n_{i,j+1} - 2u^n_{i,j} + u^n_{i,j-1}}{\triangle y^2} + \mathcal{O}(\triangle y^2) \tag{2.11}$$

respectively where $u(x_i, y_j, t_n) = u^n_{i,j}$.

A forward-difference discretization may be used for the first order derivative in time namely

$$\frac{\partial u}{\partial t} \approx \frac{u^{n+1}_{i,j} - u^n_{i,j}}{\triangle t} + \mathcal{O}(\triangle t) \tag{2.12}$$

2.3.2 Crank-Nicolson Discretization

The more stable Crank-Nicolson discretization may be used in two dimensions in the x and y directions we have

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{1}{2}\left(\frac{u^{n+1}_{i+1,j} - 2u^{n+1}_{i,j} + u^{n+1}_{i-1,j}}{\triangle x^2} + \frac{u^n_{i+1,j} - 2u^n_{i,j} + u^n_{i-1,j}}{\triangle x^2}\right) + \mathcal{O}(\triangle x^2) \tag{2.13}$$

and

$$\frac{\partial^2 u}{\partial y^2} \approx \frac{1}{2}\left(\frac{u_{i,j+1}^{n+1} - 2u_{i,j}^{n+1} + u_{i,j-1}^{n+1}}{\triangle x^2} + \frac{u_{i,j+1}^{n} - 2u_{i,j}^{n} + u_{i,j-1}^{n}}{\triangle y^2}\right) + \mathcal{O}(\triangle y^2) \tag{2.14}$$

respectively. While the Crank-Nicolson scheme doesn't improve the accuracy in the spatial discretization it does yield a scheme that is unconditionally numerically stable. When used in conjunction with an implicit discretization of the temporal derivate one derives a scheme that is accurate to $\mathcal{O}(\triangle t^2) + \mathcal{O}(\triangle x^2) + \mathcal{O}(\triangle y^2)$ instead of $\mathcal{O}(\triangle t) + \mathcal{O}(\triangle x^2) + \mathcal{O}(\triangle y^2)$.

2.3.3 Alternating Direction Implicit Method

If we consider the discretization of the two-dimensional diffusion equation

$$\frac{\mathbf{u}^{j+1} - \mathbf{u}^{j}}{\triangle t} = \mathbf{A}\mathbf{u}^{j+1} + \mathbf{u}^{j+1}\mathbf{B} \tag{2.15}$$

it is difficult to solve for $\mathbf{u}^{j+1}$. The Alternating Direction Implicit method [74] suggests we treat each dimension implicitly *in turn*. Hence we reduce the above equation to

$$\frac{\mathbf{u}^{j+1/2} - \mathbf{u}^{j}}{\triangle t} = \mathbf{A}\mathbf{u}^{j+1/2} + \mathbf{u}^{j}\mathbf{B} \tag{2.16}$$

and

$$\frac{\mathbf{u}^{j+1} - \mathbf{u}^{j+1/2}}{\triangle t} = \mathbf{A}\mathbf{u}^{j+1/2} + \mathbf{u}^{j+1}\mathbf{B}. \tag{2.17}$$

The result is two linear systems that are readily solved.

2.3.4 Grünwald-Letnikov

Definition 6. The Grünwald-Letnikov discretization is given as

$$D_t^{\alpha} u(t) = \lim_{\triangle t \to 0} \frac{1}{\triangle t^{\alpha}} \sum_{k=0}^{M} \omega_k^{\alpha} u(t - k\triangle t), \tag{2.18}$$

where

$$\omega_k^{\alpha} = (-1)^k \binom{\alpha}{k}. \tag{2.19}$$

Since the Grünwald-Letnikov derivative is a generalization of the standard first order finite difference derivative, the Grünwald-Letnikov derivative is also $\mathcal{O}(\triangle t)$ accurate. The parameter M, used in the above definition, is the maximum integer so that $t - M\triangle t$ is nonnegative. Computationally speaking, for long simulations M may become

unfeasibly large hence the short-memory principle may be employed. The short-memory principle as described in [32] specifies that M may be chosen to represent the degree of memory in the diffusive material. This means that we may truncate the sum to the past M time steps to represent a short memory.

2.4 Chebyshev Collocation

Chebyshev polynomials form a basis on $[-1,1]$ and hence we dictate the domain of our PDE to be $\Omega = [-1,1]^n$ where n indicates the number of spatial dimensions, here we choose $n = 2$. We note here however, that any domain in $\mathbb{R}^2$ can be trivially deformed to match Ω. We discretize our spatial domain using Chebyshev-Gauss-Labatto points,

$$x_i = \cos\left(\frac{i\pi}{N_x}\right), \quad i = 0,1,\ldots,N_x, \qquad y_j = \cos\left(\frac{j\pi}{N_y}\right), \quad j = 0,1,\ldots,N_y. \tag{2.20}$$

Given this choice of spatial discretization we have $x_0 = 1$, $x_{N_x} = -1$, $y_0 = 1$ and $y_{N_y} = -1$ indicating that the domain is in essence reversed and one must exercise caution when imposing the boundary conditions.

In mapping our domain to Ω we may assume that $N_x = N_y$, i.e. we have equal number of collocation points in each spatial direction. We now define a differentiation matrix $D^{(1)} = d_{kl}$,

$$d_{kl} = \begin{cases} \frac{c_k(-1)^{k+l}}{c_l(x_k - x_l)}, & k \neq l, \\ -\frac{x_k}{2(1-x^2{}_k)}, & k = l, \\ \frac{1}{6}\left(2N_x^2+1\right), & k = l = 0, \\ -\frac{1}{6}\left(2N_x^2+1\right), & k = l = N_x, \end{cases} \quad \text{where} \quad c_k = \begin{cases} 2, & k = 0, N_x, \\ 1, & k = 1,\ldots,N_x - 1. \end{cases} \tag{2.21}$$

Bayliss et al. [75] describe a method for minimizing the round off errors incurred in the calculations of higher order differentiation matrices. Since we write $D^{(2)} = D^{(1)}.D^{(1)}$ we implement the method, described in [75], in order to minimize propagation of round off errors for the second derivative in space.

The derivative matrices in the x direction are

$$\hat{D}_x^{(1)} = D^{(1)}, \qquad \hat{D}_x^{(2)} = D^{(2)}, \tag{2.22}$$

where $D^{(1)}$ is the Chebyshev differentiation matrix of size $(N_x+1)\times(N_y+1)$.
Because we have assumed $N_x = N_y$ we derive the pleasing property that $\hat{D}_y^{(1)} = \left(\hat{D}_x^{(1)}\right)^T$

and $\hat{D}_y^{(2)} = \left(\hat{D}_x^{(2)}\right)^T$.

2.5 Boundary Conditions

2.5.1 Dirichlet Boundary Conditions

Boundary conditions may be in the form of Dirichlet conditions,

$$u(-1, y, t) = a, \tag{2.23}$$
$$u(1, y, t) = b, \tag{2.24}$$
$$u(x, -1, t) = c, \tag{2.25}$$
$$u(x, 1, t) = d, \tag{2.26}$$

discretized as

$$u^n(x_{N_x}, y_j) = a, \tag{2.27}$$
$$u^n(x_0, y_j) = b, \tag{2.28}$$
$$u^n(x_i, y_{N_y}) = c, \tag{2.29}$$
$$u^n(x_i, y_0) = d, \tag{2.30}$$

since $x_{N_x} = -1$, $x_0 = 1$, $y_{N_y} = -1$ and $y_0 = 1$. The parameters a, b, c and d are potentially functions of the temporal variable and one of the spatial variables, i.e. $a = a(y, t)$. We assume that a, b, c and d are constant. Dirichlet boundary conditions can be imposed directly by substituting directly into the governing equations.

2.5.2 Neumann Boundary Conditions

Alternatively Neumann boundary conditions give,

$$u_x(-1, y, t) = a, \tag{2.31}$$
$$u_x(1, y, t) = b, \tag{2.32}$$
$$u_y(x, -1, t) = c, \tag{2.33}$$
$$u_y(x, 1, t) = d. \tag{2.34}$$

Neumann boundary conditions are discretized as

$$\frac{\partial u}{\partial x}\left(x_{N_x}=-1,y_j,t_n\right)\approx\sum_{k=0}^{N_x}d_{N_xk}u^n(x_k,y_j)=\mathcal{L}\left\{a\right\},\tag{2.35}$$

$$\frac{\partial u}{\partial x}\left(x_0=1,y_j,t_n\right)\approx\sum_{k=0}^{N_x}d_{0k}u^n(x_k,y_j)=\mathcal{L}\left\{b\right\}.\tag{2.36}$$

Similarly for the boundary conditions on y,

$$\frac{\partial u}{\partial y}\left(x_i,y_{N_y}=-1\right)\approx\sum_{k=0}^{N_y}d_{kN_y}u^n(x_i,y_k)=\mathcal{L}\left\{c\right\},\tag{2.37}$$

$$\frac{\partial u}{\partial y}\left(x_i,y_0=1\right)\approx\sum_{k=0}^{N_y}d_{k0}u^n(x_i,y_k)=\mathcal{L}\left\{d\right\}.\tag{2.38}$$

By extracting the first and last terms in the sum, the discretizations can be written as

$$\begin{pmatrix} d_{N_x0} & d_{N_xN_x} \\ d_{00} & d_{0N_x} \end{pmatrix}\begin{pmatrix} u^n(x_0,y_j) \\ u^n(x_{N_x},y_j) \end{pmatrix}=\begin{pmatrix} \mathcal{L}\left\{b\right\}-\sum_{k=1}^{N_x-1}d_{0k}u^n(x_k,y_j) \\ \mathcal{L}\left\{a\right\}-\sum_{k=1}^{N_x-1}d_{N_xk}u^n(x_k,y_j) \end{pmatrix}\tag{2.39}$$

and

$$\begin{pmatrix} d_{0N_y} & d_{N_yN_y} \\ d_{00} & d_{N_y0} \end{pmatrix}\begin{pmatrix} u^n(x_i,y_0) \\ u^n(x_i,y_{N_y}) \end{pmatrix}=\begin{pmatrix} \mathcal{L}\left\{d\right\}-\sum_{k=1}^{N_y-1}d_{k0}u^n(x_i,y_k) \\ \mathcal{L}\left\{c\right\}-\sum_{k=1}^{N_y-1}d_{kN_y}u^n(x_i,y_k) \end{pmatrix}\tag{2.40}$$

The solutions to these linear systems are then substituted into the governing equations.

2.6 Quasi-Linearization

The quasi-linearization technique is employed in Chapter 6 to obtain a quasi-linear form of our time-fractional nonlinear diffusion equation to which a Laplace transform may be applied.

The quasi-linearization technique can be viewed as a generalized Newton-Raphson method in functional space. An iterative scheme is constructed creating a sequence of linear equations that approximate a nonlinear equation and boundary conditions. Furthermore, this sequence of solutions converges quadratically and monotonically [76–78].

If we consider a general parabolic two-dimensional nonlinear PDE of the form

$$\Phi(u,u_x,u_y,u_{xx},u_{yy},u_t)=0\tag{2.41}$$

then the quasi-linear form is given by

$$X_1^n u^{n+1} + X_2^n u_x^{n+1} + X_3^n u_y^{n+1} + X_4^n u_{xx}^{n+1} + X_5^n u_{yy}^{n+1} + X_6^n u_t^{n+1} = X_7^n \tag{2.42}$$

where

$$X_1^n = \left.\frac{\partial \Phi}{\partial u}\right|^n, \tag{2.43}$$

$$X_2^n = \left.\frac{\partial \Phi}{\partial u_x}\right|^n, \tag{2.44}$$

$$X_3^n = \left.\frac{\partial \Phi}{\partial u_y}\right|^n, \tag{2.45}$$

$$X_4^n = \left.\frac{\partial \Phi}{\partial u_{xx}}\right|^n, \tag{2.46}$$

$$X_5^n = \left.\frac{\partial \Phi}{\partial u_{yy}}\right|^n, \tag{2.47}$$

$$X_6^n = \left.\frac{\partial \Phi}{\partial u_t}\right|^n, \tag{2.48}$$

$$\tag{2.49}$$

and if $\underline{u} = (u, u_x, u_y, u_{xx}, u_{yy}, u_t)$ indexed by j then

$$X_7^n = \sum_j \underline{u}_j^n \frac{\partial \Phi}{\partial \underline{u}_j}. \tag{2.50}$$

In Chapter 6 we derive the quasi-linear form of the time-fractional diffusion equation in two dimensions with nonlinear source term. The purpose of obtaining a linear form is the ability to apply the Laplace transform from which we derive the benefits described above in Section 2.2.4.

Chapter 3

A Novel Approach to Text Binarization via a Diffusion-Based Model

"We especially need imagination in science. It is not all mathematics, nor all logic, but it is somewhat beauty and poetry."

Maria Montessori

This work has appeared in:

Jacobs, B. A., and E. Momoniat. "A novel approach to text binarization via a diffusion-based model." *Applied Mathematics and Computation* 225 (2013): 446-460.

This paper presents a new approach to document image binarization. The method is based on the dynamic process of diffusion, coupled with a nonlinear Fitzhugh-Nagumo type source term that exhibits binarizing properties. These desirable properties lead to a method that is robust to noise and is able to successfully binarize an input document image. We measure the efficacy of our proposed method against industry standards by two methods, a pixel by pixel comparison with the ground truth image and a standard optical character recognition test. Through these measures we illustrate a progressive method that performs at the highest level in the field.

3.1 Introduction

This paper aims to present a new text binarization algorithm based on a linear diffusion model with a nonlinear source term. The model is given as

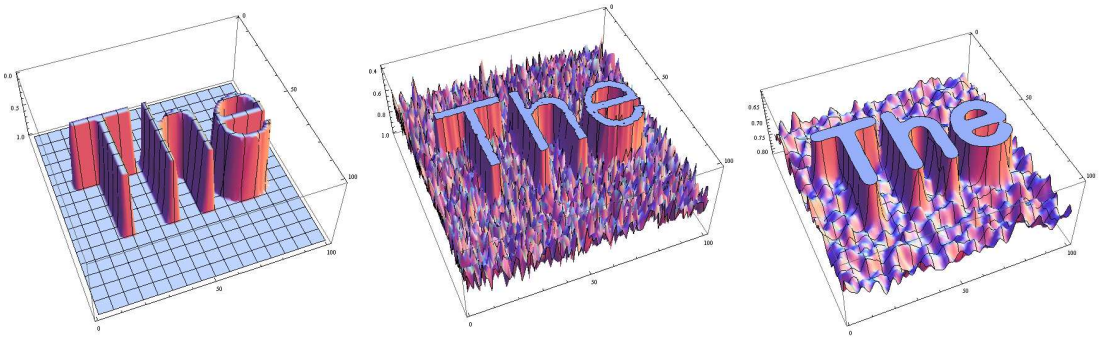
$$u_t = c_d \nabla^2 u + c_s u(1-u)(u-a) \tag{3.1}$$

where u is our dependant variable initially set to our input image. The model then dictates the evolution of this image with time, in a manner similar to the physical diffusion or heat transfer process. $u_t = \partial u / \partial t$ and ∇^2 is the typical Laplacian operator, taken to be

$$\nabla^2 u = \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$

since our application has two spatial dimensions. We choose the coefficient of diffusion to be c_d, this coefficient determines how aggressive the diffusive property of the model is and allows one to select the degree to which the image is denoised. Our source term is a cubic polynomial and is discussed further in Section 3.2. Finally c_s is the source term scaling coefficient, this is analogous to the diffusion coefficient and determines the degree to which the source term affects the process.

Our process is isotropic since there is no explicit dependence on the image directionality. The source term is chosen based on the Fitzhugh-Nagumo (FN) [8, 9] source term because of its thresholding properties. Diffusive processes have been thoroughly investigated in image processing as denoising techniques [56–61], and the revolutionary Perona-Malik [5] anisotropic diffusion model smoothes noise in an image while preserving the sharpness of edges. The application of diffusive processes to image denoising is an intuitive one. If we consider our initial image as a height map, where pixel luminosity corresponds to an (x, y) coordinate height, then by applying a diffusion process to this, the peaks that exist in this height map will be softened and reduced. Figures 3.1(a), 3.1(b) and 3.1(c) show a 3D representation of the ground truth image, the image with additive white gaussian noise and the corresponding diffused image respectively. From this representation we may visualize the binarizing process as slicing this height map with a plane to optimally separate the foreground and the background. Our method benefits from the denoising property, inherent to diffusion processes, in that noise is reduced by the diffusion term and image data is binarized by the source term simultaneously.



(a) (b) (c)

There is a vast literature of binarization techniques and an exhaustive survey of these methods has been done in [1–4]. Typically these methods are categorized as global or adaptive thresholding techniques. Global techniques generally use a histogram analysis to optimally choose a threshold so that we correctly classify as many pixels as possible, as black or white. Adaptive methods locally threshold sub-images, which allows different threshold values to be selected based on local properties. This gives a robustness to noise and nonuniform illumination. He et al. [1] propose adaptive variants of Niblack's method and of Sauvola's method. Recently Drira et al. [79, 80] proposed a modified method based on nonlinear diffusion tensors, where this anisotropy aids in edge and corner preservation of the document image data. Moghaddam and Cheriet [81] introduce a method based on reversing the physical process of document degradation. The authors consider the physical process of ink diffusing into the paper and construct a model to reverse the process via a backward diffusion. This method is tested on synthesized and real test images with excellent results. The application of PDE based methods to low quality input images is also investigated by Mahani et al. [82] where low quality camera phone images are used as inputs. The aforementioned PDE methods all make use of some anisotropy in the document image reconstruction, in contrast to this our method uses an algebraic nonlinearity coupled with a linear diffusion to binarize a document image effectively. Compounded by the simplicity of an algebraic nonlinearity the proposed method still remains isotropic. These methods may involve a number of steps and can become difficult to implement while computing diffusion tensors. The method proposed in this paper is simple to implement, robust to noise and effective at image binarization.

This chapter is divided into the following sections: Section 3.2 substantiates our choice of source term and diffusion process. We go on to present a one dimensional case in Section 3.2.1, this is used as an illustration of the degree of noise corruption introduced into our images as well as showing efficacy of our method. The main two dimensional model is showcased in Section 3.2.2, this has direct application to our given input images. We

use two methods as a source of comparison of the presented methods: the first is using Google's open source optical character recognition package Tesseract [83] to determine how frequently it is able to identify characters based on the preprocessing method. The second source of comparison is a direct calculation of pixels correctly binarized using a hand crafted binary image as a ground truth. An analysis of these results is given in Section 3.3 and concluding remarks are made in Section 3.4.

3.2 Model Derivation and Discussion

The application of diffusive processes to image denoising is a well researched area. By exploiting the denoising properties of the diffusion equation

$$\frac{\partial u}{\partial t} = c_d \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \tag{3.2}$$

and combining the FN type source term, given as

$$f(u) = u(1-u)(u-a) \tag{3.3}$$

where u is any given pixel's luminosity, we can derive a process that will denoise and binarize an image simultaneously. By combining (3.2) and (3.3) we obtain our model (3.1). The FN type source term exhibits a binarizing property by 'forcing' any given pixel toward a stable luminosity state, namely black or white. The direction of this 'force' is dependent on the pixel's intensity relative to the threshold parameter a and the cubic nature of the source term helps to stabilize pixels very quickly. We choose the value for a by using Otsu's method [84] for finding the threshold of our input image, but a can be chosen in many ways being dependent on the mean intensity of the image or even based on local information. Figure 3.1 illustrates the directionality and intensity of the forcing effect of $f(u)$.

For example, if a pixel, u, has intensity such that $a < u < 1$, then $f(u)$ will contribute energy to that pixel until it has an energy of 1. Note that $f(1) = 0$, so this represents a stable state. If some pixel obtains a value greater than 1, $u > 1$, $f(u)$ will then withdraw energy from u to force it back into the stable state of 1. Similarly if $0 < u < a$ then $f(u)$ performs diametrically, withdrawing energy until u obtains the stable state of 0. Furthermore, if $u = a$ then $f(u) = 0$ and the process reverts back to a simple diffusion model. This is desirable in that, if a pixel lies exactly on the threshold of our binarization we are unsure of how to characterize it. We then look to the diffusive

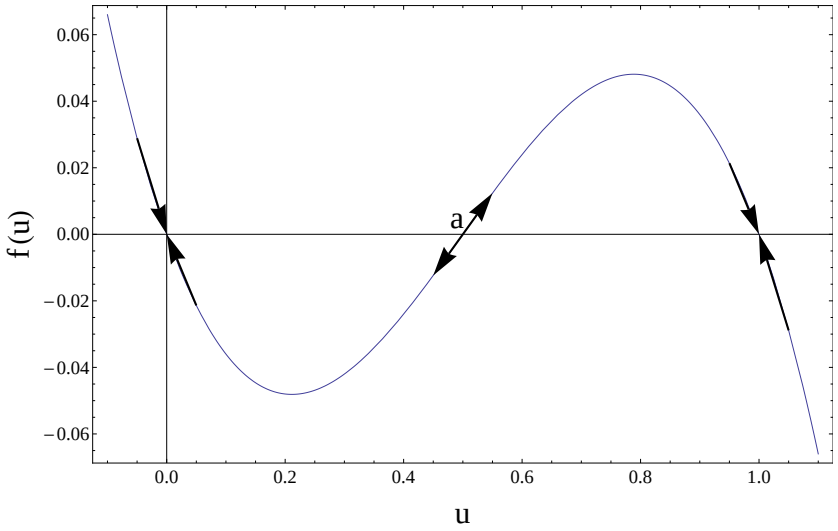


FIGURE 3.1: Fitzhugh-Nagumo Type Source Term with arrows indicating the directionality of the forcing effect.

process to inform us. If a pixel lies exactly on the threshold and all of its neighbours are brighter than it, then the diffusive process will draw said pixel's intensity toward its neighbours, making it brighter. The source term can then do its job by forcing it in the direction suggested by the diffusion process. This also aids in dealing with noise, since generally noise will behave as an outlier, the diffusive process will blend the outlying noise into its background colour and homogenize noisy areas.

Other binarizing processes [84–86] can be sensitive to noise and may classify a pixel incorrectly due to this noise. The proposed method classifies points not only based on their intensity but also based on their neighbourhood, this is simply a consequence of the design of the diffusion process itself. If, for example, a noisy pixel is close to white on a black background, then binarizing based solely on some threshold will classify the noisy pixel as white. However, by diffusing by some small amount, the noisy pixel will be drawn toward its neighbour's intensity and hence will be more likely to be classified correctly as black. Combining a linear diffusion model with the discussed source term we derive equation (3.1).

For both the one-dimensional and two-dimensional case we perform a Von Neumann stability analysis which is typically only applicable to linear problems. However, we prove boundedness on u as well as on the source term $f(u)$, so that the stability of the linearized system is analyzed under the pathological cases of the source term and appropriate conditions are derived.

3.2.1 One Dimensional Case

In this section we introduce the one dimensional application of our method. The aim here is to illustrate how robust the algorithm is to noise. In one spatial dimension we have

$$\frac{\partial u}{\partial t} = c_d \frac{\partial^2 u}{\partial x^2} + c_s u(1-u)(u-a) \tag{3.4}$$

with the initial state defined as

$$u(x,0) = \begin{cases} 1 & : \frac{1}{4} < x < \frac{3}{4} \\ 0 & : \text{otherwise,} \end{cases} \tag{3.5}$$

We then corrupt the initial data with additive gaussian noise with a mean of zero and a standard deviation of 0.15. Finally we impose zero sheer boundary conditions at $x = 0$ and $x = 1$

$$u_x(0,t) = u_x(1,t) = 0. \tag{3.6}$$

3.2.1.1 Derivation

The first step in implementing our scheme is discretizing our model so that we may operate on our discretized image. Our image can be thought of as a two dimensional matrix with values corresponding to luminosity. In one dimension this matrix simply becomes a vector of data points. We discretize our spatial dimension as $x_i = i\triangle x$ where $i = 0, 1, \ldots, p$. In discretized form $u(x_i, t^n)$ is represented as u_i^n, meaning the ith position in x and the nth step in time, t.

We approximate the time derivative in (3.4) using the forward difference scheme

$$\frac{\partial u}{\partial t} \approx \frac{u_i^{n+1} - u_i^n}{\triangle t} \tag{3.7}$$

and the spatial difference using the Crank-Nicolson Scheme

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{1}{2}\left(\frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{\triangle x^2} + \frac{u_{i+1}^{n} - 2u_i^{n} + u_{i-1}^{n}}{\triangle x^2} \right). \tag{3.8}$$

Substituting (3.7) and (3.8) into (3.4) and taking $\lambda = c_d \triangle t / 2\triangle x^2$ we obtain an iterative scheme

$$\begin{aligned} &-\lambda u_{i+1}^{n+1} + u_i^{n+1}(1+2\lambda) - \lambda u_{i+1}^{n+1} = \\ &\lambda u_{i+1}^n + u_i^n(1-2\lambda) + \lambda u_{i+1}^n + \triangle t c_s u_i^n (1-u_i^n)(u_i^n - a). \end{aligned} \tag{3.9}$$

or in matrix form

$$A\underline{u}^{n+1} = B\underline{u}^n + c_s\triangle t\underline{u}^n \circ (1 - \underline{u}^n) \circ (\underline{u}^n - a), \tag{3.10}$$

where $\circ$ denotes the Hadamard product or element-wise multiplication.

3.2.1.2 Stability Analysis

The presence of a nonlinear source term introduces a complication in analyzing the stability of the proposed process. We may, however, provide a bound for the function $f(u)$. Given that initially $u \in [0, 1]$ we require that the iteration formula

$$\underline{u}^{n+1} = A^{-1}\left(B\underline{u}^n + c_s\triangle t\underline{u}^n \circ (1 - \underline{u}^n) \circ (\underline{u}^n - a)\right) \tag{3.11}$$

does not violate this bound. Since A is positive definite with eigenvalues in $[1, 1 + 4\lambda]$, by Gerschgorin's Theorem, then A^{-1} is also positive definite with eigenvalues in $\left[\frac{1}{1+4\lambda}, 1\right]$. Hence A^{-1} will never scale an argument beyond $[0, 1]$. Since $a \in [0, 1]$ and at explicit time $u \in [0, 1]$, then by inspection it is easy to see that $f(u) = u(1 - u)(u - a)$ is maximized in a when $a = 0$ and minimized when $a = 1$. The turning points of $f(u)$ occur when $f'(u) = 0$. Given this, $f(u)$ is at a maximum when $u = 2/3$ and $a = 0$, similarly $f(u)$ is minimized when $u = 1/3$ and $a = 1$. Therefore

$$|f(u)| \leq \frac{4}{27}. \tag{3.12}$$

Considering the bounds of the right-hand side of (3.9) using (3.12) we have

$$|Bu_i^n + c_s\triangle t u_i^n(1 - u_i^n)(u_i^n - a)| \leq |Bu_i^n + c_s\triangle t\frac{4}{27}|. \tag{3.13}$$

This can be written as

$$Bu_i^n + c_s\triangle t\frac{4}{27} \leq 1 \tag{3.14}$$

or

$$Bu_i^n - c_s\triangle t\frac{4}{27} \geq 0 \tag{3.15}$$

which, in conjunction with (3.9), result in the same condition

$$c_d\frac{\triangle t}{\triangle x^2} + \triangle t c_s\frac{4}{9} \leq 1, \tag{3.16}$$

which ensures $u \in [0, 1]$. This reduces to the standard C-F-L condition for the heat equation with $c_s = 0$ which implies convergence.

We perform a Von Neumann stability analysis by substituting

$$u_i^n = U^n e^{I\omega i \triangle x} \tag{3.17}$$

where $I^2 = -1$, into (3.9). Then using the triangle inequality and the boundedness of $\cos(\theta)$, we may deduce the following

$$\begin{aligned} |U^{n+1}| &\leq \left| U^n \frac{(1 - 2\lambda + 2\lambda\cos(\omega\triangle x))}{(1 + 2\lambda - 2\lambda\cos(\omega\triangle x))} \right| + \\ &\quad \left| \frac{4\triangle t c_s}{27(1 + 2\lambda - 2\lambda\cos(\omega\triangle x))} \right| \\ &\leq \left| U^n \frac{(1 - 4\lambda)}{(1 + 4\lambda)} \right| + \left| \frac{4\triangle t c_s}{27(1 + 4\lambda)} \right|. \end{aligned} \tag{3.18}$$

We now define

$$|g| = \frac{|U^{n+1}|}{|U^n|} \tag{3.19}$$

and then from the stability condition, $|g| < 1$, we derive a bound on the source term scaling factor, c_s.

$$-54\lambda < c_s \triangle t < 54\lambda \tag{3.20}$$

Therefore we may conclude that this numerical scheme is stable for the above values of c_s, but c_s is chosen to be positive so that the direction of the 'force' of the source term is preserved.

3.2.1.3 Implementation

Our initial data are given by (3.5), i.e. u_i^0 is a known vector. Discretizing equation (3.6), using a standard forward difference scheme, and evaluating it at $x = 0$ and $x = 1$, or $i = 0$ and $i = p$ respectively, we get $u_{-1}^n = u_1^n$ and $u_{p+1}^n = u_{p-1}^n$ which are needed in the evaluation (3.9) at $i = 0$ and $i = p$. These criteria satisfy the boundary conditions of the physical model.

We implement our numerical approximation of (3.4), (3.5) and (3.6) using (3.7) and (3.8) where $x_i = x_0 + i\triangle x$ and $t_n = n\triangle t$ and $u_i^n = u(x_i, t_n)$. By using a standard forward difference scheme for u_t and a Crank-Nicolson scheme for u_{xx} we obtain the following:

$$A\underline{u}^{n+1} = B\underline{u}^n + c_s \triangle t \underline{u}^n \circ (1 - \underline{u}^n) \circ (\underline{u}^n - a) \tag{3.21}$$

where $A =$

$$\begin{bmatrix} 1+2\lambda & -2\lambda & 0 & \dots & 0 \\ -\lambda & 1+2\lambda & -\lambda & \dots & 0 \\ 0 & -\lambda & 1+2\lambda & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 1+2\lambda & -\lambda & 0 \\ 0 & \dots & -\lambda & 1+2\lambda & -\lambda \\ 0 & \dots & 0 & -2\lambda & 1+2\lambda \end{bmatrix} \tag{3.22}$$

and $B =$

$$\begin{bmatrix} 1-2\lambda & 2\lambda & 0 & \dots & 0 \\ \lambda & 1-2\lambda & \lambda & \dots & 0 \\ 0 & \lambda & 1-2\lambda & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 1-2\lambda & \lambda & 0 \\ 0 & \dots & \lambda & 1-2\lambda & \lambda \\ 0 & \dots & 0 & 2\lambda & 1-2\lambda \end{bmatrix}. \tag{3.23}$$

Implementing this scheme in `MATHEMATICA` and using $c_d = 1$, $c_s = 1/\triangle x^2$ and $a = 0.5$ we obtain Figure 3.2. Figure 3.3 is obtained by using a simple binarizing technique based on a mean global threshold. These figures illustrate the efficacy of the proposed method in classifying points as 1 or 0. The dashed blue line is the initial noisy data and the solid purple line is the result obtained after 150 iterations of our method.

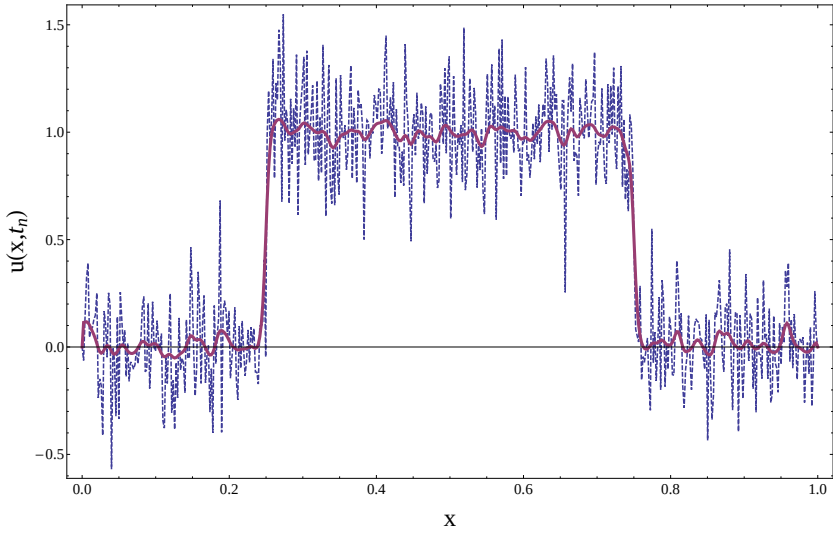


FIGURE 3.2: Efficacy of proposed method on a one dimensional example of a Heaviside function with SNR of 10dB.

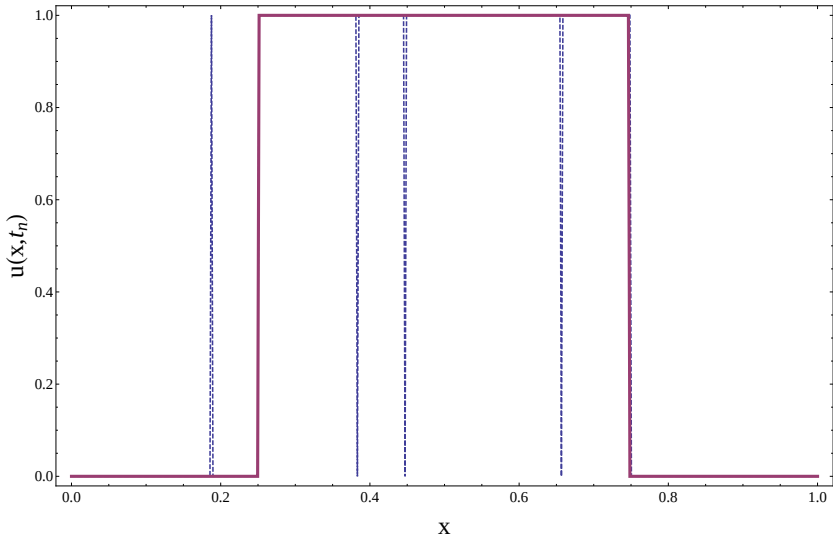


FIGURE 3.3: Binarized result of proposed method (solid purple line) and binarized noisy input (dashed blue line).

Figure 3.3 clearly illustrates several points have been incorrectly classified as 0 when they should be classified with their neighbours as 1 or vice versa. This is an important result since it illustrates that sensitivity to noise potentially leads to false classification, this further substantiates our use of a diffusion based algorithm.

3.2.2 Two Dimensional Case

We now extend our model to two spatial dimensions and get

$$\frac{\partial u}{\partial t} = c_d(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}) + c_s u(1-u)(u-a) \tag{3.24}$$

where the initial state is dictated by the input image

$$u(x, y, 0) = \text{Img}(x, y). \tag{3.25}$$

$\text{Img}(x, y)$ is a matrix with pixel intensity at position (x, y). Given the image resolution, p pixels by q pixels, we define our spatial discretization as $x_i = i\triangle x$ where $i = 0, 1, \ldots, p$ and $y_j = j\triangle y$ where $j = 0, 1, \ldots, q$. Our space steps are determined by the input image's dimensions, $\triangle x = 1/p$ and $\triangle y = 1/q$. We choose our time step to be $\triangle t = 0.01\triangle x\triangle y$ to ensure stability. We again impose zero sheer boundary conditions on

$$u_x(0, y, t) = u_x(p, y, t) = u_y(x, 0, t) = u_y(x, q, t) = 0. \tag{3.26}$$

3.2.2.1 Derivation

In two dimensions the discretized form of $u(x, y, t)$ is $u(x_i, y_j, t^n)$ and we represent it for simplicity as $u^n_{i,j}$. We again approximate the time derivative in (3.24) using the forward difference scheme

$$\frac{\partial u}{\partial t} \approx \frac{u^{n+1}_{i,j} - u^n_{i,j}}{\triangle t} \tag{3.27}$$

and the spatial differences using the Crank-Nicolson Scheme

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{1}{2}\left(\frac{u^{n+1}_{i+1,j} - 2u^{n+1}_{i,j} + u^{n+1}_{i-1,j}}{\triangle x^2} + \frac{u^n_{i+1,j} - 2u^n_{i,j} + u^n_{i-1,j}}{\triangle x^2}\right) \tag{3.28}$$

and

$$\frac{\partial^2 u}{\partial y^2} \approx \frac{1}{2}\left(\frac{u^{n+1}_{i,j+1} - 2u^{n+1}_{i,j} + u^{n+1}_{i,j-1}}{\triangle x^2} + \frac{u^n_{i,j+1} - 2u^n_{i,j} + u^n_{i,j-1}}{\triangle y^2}\right). \tag{3.29}$$

We may now substitute (3.27), (3.28) and (3.29) into (3.24) to obtain the following

$$\begin{aligned} u^{n+1}_{i,j}\left(1 + 2\lambda_1 + 2\lambda_2\right) - \lambda_1 u^{n+1}_{i+1,j} - \lambda_1 u^{n+1}_{i-1,j} - \\ \lambda_2 u^{n+1}_{i,j+1} - \lambda_2 u^{n+1}_{i,j-1} = u^n_{i,j}\left(1 - 2\lambda_1 - 2\lambda_2\right) + \\ \lambda_1 u^n_{i+1,j} + \lambda_1 u^n_{i-1,j} + \lambda_2 u^n_{i,j+1} + \lambda_2 u^n_{i,j-1} + \\ \triangle t c_s u^n_{i,j}(1 - u^n_{i,j})(u^n_{i,j} - a) \end{aligned} \tag{3.30}$$

where $\lambda_1 = c_d \triangle t / 2\triangle x^2$ and $\lambda_2 = c_d \triangle t / 2\triangle y^2$. The above scheme is implemented using an alternating direction implicit method, written as

$$A_1 \underline{u}^{n+1/2} = \underline{u}^n B_1^T + c_s \triangle t \underline{u}^n \circ (1 - \underline{u}^n) \circ (\underline{u}^n - a) \tag{3.31}$$

and

$$\underline{u}^{n+1} B_2^T = A_2 \underline{u}^{n+1/2} + c_s \triangle t \underline{u}^n \circ (1 - \underline{u}^n) \circ (\underline{u}^n - a), \tag{3.32}$$

where A_1, A_2, B_1 and B_2 are defined in the following section, which treats each direction implicitly in turn, rather than at once. Since we effectively treat each dimension independently, using the one dimensional argument that if $u^n \in [0, 1]$ then $u^{n+1/2} \in [0, 1]$ provided

$$c_d \frac{\triangle t}{\triangle x^2} + \triangle t c_s \frac{4}{9} \leq 1 \tag{3.33}$$

and if $u^{n+1/2} \in [0,1]$ then $u^{n+1} \in [0,1]$ provided

$$c_d \frac{\triangle t}{\triangle y^2} + \triangle t c_s \frac{4}{9} \leq 1. \tag{3.34}$$

3.2.2.2 Stability Analysis

We may again exploit the boundedness of $f(u)$, and perform a Von Neumann stability analysis by substituting

$$u_{i,j}^n = U^n e^{Iij\omega \triangle x \triangle y} = U^n e^{Iij\omega h^2} \tag{3.35}$$

where, for simplicity, $h^2 = \triangle x \triangle y$. Then by the triangle inequality and the boundedness of $\cos(\theta)$, we get

$$\begin{aligned}
&|U^{n+1}| \leq \\
&\left| U^n \frac{1 - 2(\lambda_1 + \lambda_2) + 2\lambda_1 \cos(j\omega h^2) + 2\lambda_2 \cos(i\omega h^2)}{1 + 2(\lambda_1 + \lambda_2) - 2\lambda_1 \cos(j\omega h^2) - 2\lambda_2 \cos(i\omega h^2)} \right| \\
&+ \left| \frac{4c_s \triangle t}{27(1 + 2(\lambda_1 + \lambda_2) - 2\lambda_1 \cos(j\omega h^2) - 2\lambda_2 \cos(i\omega h^2))} \right| \\
&\leq \left| U^n \frac{1 - 4(\lambda_1 + \lambda_2)}{1 + 4(\lambda_1 + \lambda_2)} \right| + \left| \frac{4c_s \triangle t}{27(1 + 4(\lambda_1 + \lambda_2)} \right|
\end{aligned} \tag{3.36}$$

again, from the stability condition, $|g| < 1$, we derive the following bound on the source term scaling coefficient, c_s,

$$-54(\lambda_1 + \lambda_2) < c_s \triangle t < 54(\lambda_1 + \lambda_2). \tag{3.37}$$

As in the one dimensional case this scheme is stable for the above values of c_s. Typically Crank-Nicolson schemes are unconditionally stable, however the inclusion of a source term affects the bounds of stability.

3.2.2.3 Implementation

Our implementation is achieved using an Alternating-Direction Implicit (ADI) method [74]. This is a two-step process in which we first approximate $\underline{u}^{n+1/2}$ using an implicit scheme in the x direction and an explicit scheme in the y direction. We then approximate $\underline{u}^{n+1}$ using an explicit scheme in the x direction and an implicit scheme in the y direction. Our variables are defined as before, $\underline{u}^0$ is our initial image, we then solve two linear systems to obtain $\underline{u}^1$ and continue as iterating procedure. In matrix form we have

$$A_1 \underline{u}^{n+1/2} = \underline{u}^n B_1^T + c_s \triangle t \underline{u}^n \circ (1 - \underline{u}^n) \circ (\underline{u}^n - a) \tag{3.38}$$

and

$$B_2\underline{v}^{n+1} = \underline{v}^{n+1/2}A_2^T + c_s\triangle t\underline{v}^{n+1/2} \circ (1 - \underline{v}^{n+1/2}) \circ (\underline{v}^{n+1/2} - a) \tag{3.39}$$

where

$$\underline{v}^n = (\underline{u}^n)^T \tag{3.40}$$

$A_1 =$

$$\begin{bmatrix} 1+2\lambda_1 & -2\lambda_1 & 0 & 0 & \dots & 0 \\ -\lambda_1 & 1+2\lambda_1 & -\lambda_1 & 0 & \dots & 0 \\ 0 & -\lambda_1 & 1+2\lambda_1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1+2\lambda_1 & -\lambda_1 & 0 \\ 0 & 0 & \dots & -\lambda_1 & 1+2\lambda_1 & -\lambda_1 \\ 0 & 0 & \dots & 0 & -2\lambda_1 & 1+2\lambda_1 \end{bmatrix} \tag{3.41}$$

$B_1 =$

$$\begin{bmatrix} 1-2\lambda_2 & 2\lambda_2 & 0 & 0 & \dots & 0 \\ \lambda_2 & 1-2\lambda_2 & \lambda_2 & 0 & \dots & 0 \\ 0 & \lambda_2 & 1-2\lambda_2 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1-2\lambda_2 & \lambda_2 & 0 \\ 0 & 0 & \dots & \lambda_2 & 1-2\lambda_2 & \lambda_2 \\ 0 & 0 & \dots & 0 & 2\lambda_2 & 1-2\lambda_2 \end{bmatrix} \tag{3.42}$$

$A_2 =$

$$\begin{bmatrix} 1-2\lambda_1 & 2\lambda_1 & 0 & 0 & \dots & 0 \\ \lambda_1 & 1-2\lambda_1 & \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_1 & 1-2\lambda_1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1-2\lambda_1 & \lambda_1 & 0 \\ 0 & 0 & \dots & \lambda_1 & 1-2\lambda_1 & \lambda_1 \\ 0 & 0 & \dots & 0 & 2\lambda_1 & 1-2\lambda_1 \end{bmatrix} \tag{3.43}$$

and $B_2 =$

$$\begin{bmatrix} 1+2\lambda_2 & -2\lambda_2 & 0 & 0 & \dots & 0 \\ -\lambda_2 & 1+2\lambda_2 & -\lambda_2 & 0 & \dots & 0 \\ 0 & -\lambda_2 & 1+2\lambda_2 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1+2\lambda_2 & -\lambda_2 & 0 \\ 0 & 0 & \dots & -\lambda_2 & 1+2\lambda_2 & -\lambda_2 \\ 0 & 0 & \dots & 0 & -2\lambda_2 & 1+2\lambda_2 \end{bmatrix}. \tag{3.44}$$

Ultimately the process becomes a simple solution to a linear system, the righthand side of equations (3.38) and (3.39) is always known and can be reduced to a simple vector. The solutions to the linear systems (3.38) and (3.39) are found using `MATHEMATICA` for a variety of sample images the results of which are shown Section 3.3.

3.3 Experimental Results

This section illustrates the results we obtained after applying a number of methods, Otsu [84], Sauvola [86], and Perona-Malik [5], to the heavily-corrupted input images. These results are given in Figures 3.4, 3.5, 3.6, 3.7, 3.8, 3.9 and 3.10. The input images were corrupted with additive white Gaussian noise to give a signal to noise ratio (SNR) of 10dB with the exception of 3.10(a) which was left uncorrupted to illustrate the comparative decrease in efficacy when noise is introduced, this can be seen in Tables 3.1 and 3.2. This level of noise illustrates the excellent robustness to noise of our method, where the sensitivity of other standard binarization algorithms tends to yield less desirable results. We present the l_1 error norms with respect to the ground truth images for each example in Table 3.3. These results indicate a superior performance of the current method as well as reiterating the robustness to noise corruption.

Isotropic diffusion equations tend to blur and lose important edge information. Figure 3.4 shows the binarization of a two dimensional Heaviside function. Typically the blurring induced by diffusion will produce high local errors with regards to a l_1 error norm or a pixelwise comparison with the ground truth. However, the parameters c_d and c_s are chosen so that the effect of the source term dominates the diffusion process, which helps to prevent local errors introduced by diffusion and preserve edges in the image.

Figure 3.5 is an illustrative example for text denoising. Given a relatively small amount of noise Otsu's method and Sauvola's method will binarize this example perfectly. However, due to the high level of noise in the image both of these methods suffer from incorrect classification. This "proof of concept" allows us to move onto more practical examples. Figure 3.6(a) gives an example of poor printing and ink smudging, which are commonly difficult to overcome in binarizing printed text. By binarizing this image we hope to better distinguish between noise and the actual text. Once more, the sensitivity to this noise is evident in the comparative algorithms, and our proposed method achieves a good result.

One potential application of our proposed technique is a mobile application for binarizing text taken with a low quality camera. Typically low resolution cameras found on mobile devices are extremely susceptible to noise from electronic components, particulary in low

light scenarios. Figure 3.7 is a photographed restaurant bill taken on a mobile device in low light. Our method is able to substantially reduce noise and binarize the text which will aid in optical character recognition. When compared with Otsu's method, Figure 3.7(b), and Sauvola's method, Figure 3.7(c), our method has reduced noise and correctly classified the majority of pixels in the image.

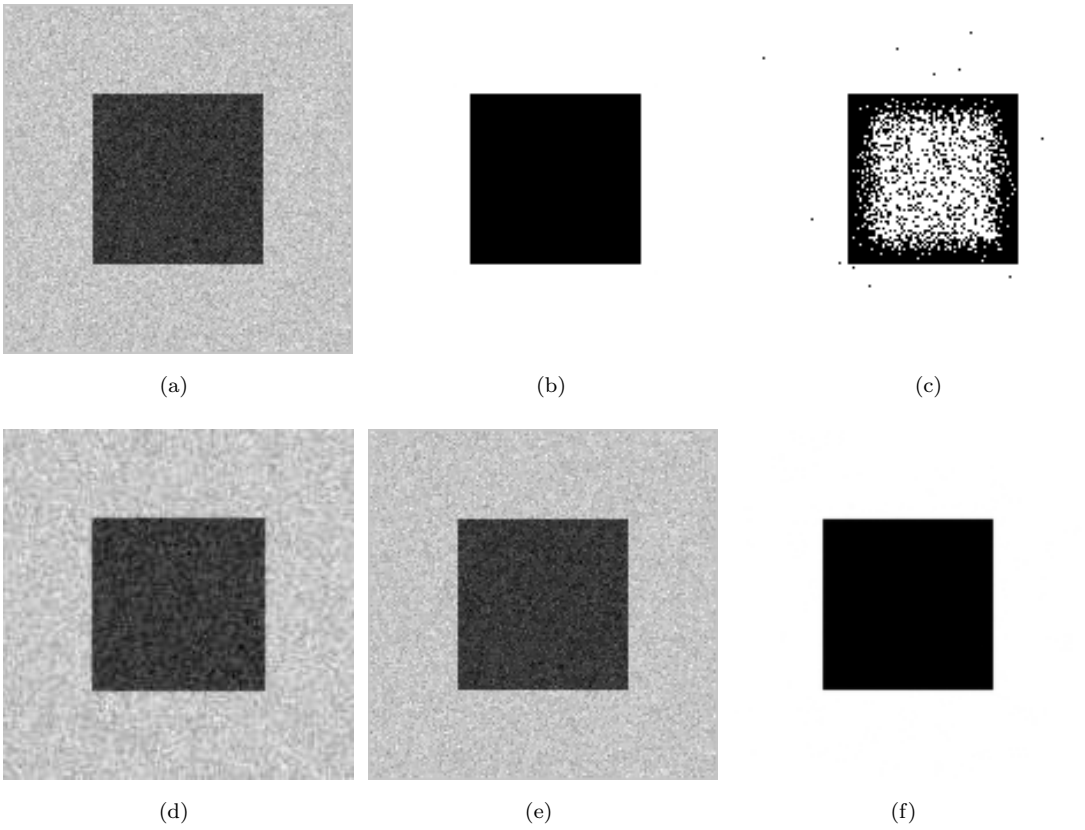
Figure 3.8 illustrates how Otsu's method and Sauvola's method are susceptible to noise. The resulting binarized images are dotted with salt and pepper type noise due to incorrect classification of pixels. Once more, our method is robust to this noise and continues to excel in the binarizing process.

The next example is a photograph taken of George Orwell's novel *Nineteen Eighty-Four* [87]. The print on the reverse side of the page has bled through and the shadow of this text is apparent. This shadow could be perceived as faint text by a naive binarizing process. Otsu's method incorrectly classifies the shadow of the text on the reverse side of the page and Sauvola's method performs better. By comparing our result in Figure 3.9(f) and Figure 3.10(f) it is clear that our method produces similar results without noise present.

Tables 3.1 and 3.2 illustrate the efficacy of our method compared to existing methods. We have included the linear diffusion and Perona-Malik methods as a source of comparing our method with other diffusion based methods. The superiority of our method in this application therefore implies that the inclusion of our source term is necessary in obtaining desirable results.

Table 3.1 measures the number of characters correctly identified by Google's Tesseract [83] optical character recognition engine using the textual information as a ground truth. The method proposed in the paper consistently performs close to optimally and far exceeds competition in the case of noisy images. The *novel extract* image with $10dB$ of gaussian white noise and without noise, Figures 3.9(a) and 3.10(a) respectively, is a prime example of this noise robustness. A further illustration of the effects of noise are given in Table 3.3 where the novel extract is corrupted with various levels of noise and the l_1 error norm with respect to the ground truth is computed.

The second measure compares every pixel in the processed image with a corresponding hand constructed ground truth image. Our method again consistently performs well for all the tested images. One point to highlight is that Sauvola's method and our method achieve a similar level of performance for the *novel extract* image without noise. However, when noise is added to this image the performance of Sauvola's method decreases substantially while our method experiences only a small decrease in performance, this is evident in Table 3.3 for incremental noise levels.



(a) (b) (c)

(d) (e) (f)

FIGURE 3.4: Binarization of an (a) initial noisy block, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result.

3.4 Conclusion

In this paper we have introduced a novel approach to document image denoising and binarization via a modified linear diffusion process. This algorithm is able to achieve excellent results in cases of extreme noise corruption. The proposed method produces results comparable and superior to existing methods. The application of differential calculus to document image binarization is in its infancy and our method has opened a range of potential improvements to the field.

The diffusion process incorporated into our method dynamically changes the intensity of noisy pixels, drawing them toward their neighbours which helps to homogenize noisy areas and reduce the occurrence of outliers. Concurrent to this diffusion process our source term either contributes to or withdraws energy from each pixel, based on its intensity relative to the threshold parameter a.

In Section 3.3 we compare the results of our method with those of some other common binarization techniques. Otsu's method and our proposed method give the best results in the illustrative examples. However, Otsu's method still admits outliers where the

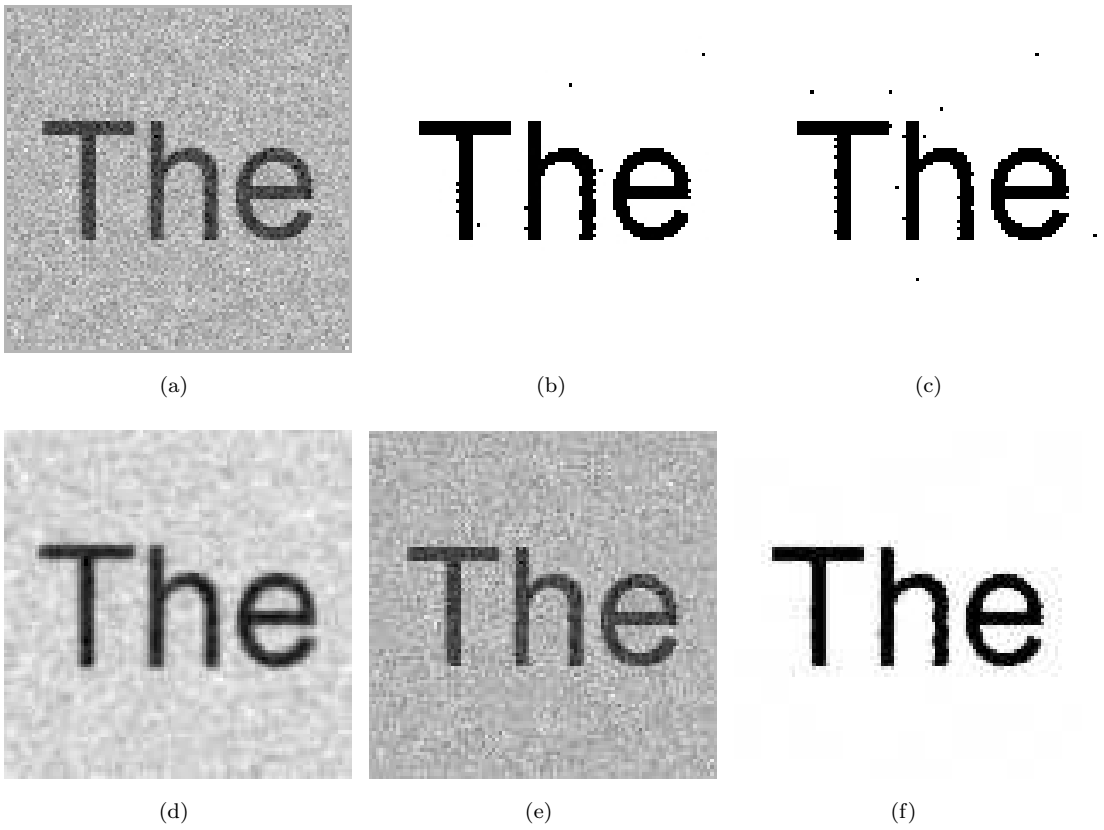


FIGURE 3.5: Binarization of an (a) initial noisy text sample, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result.

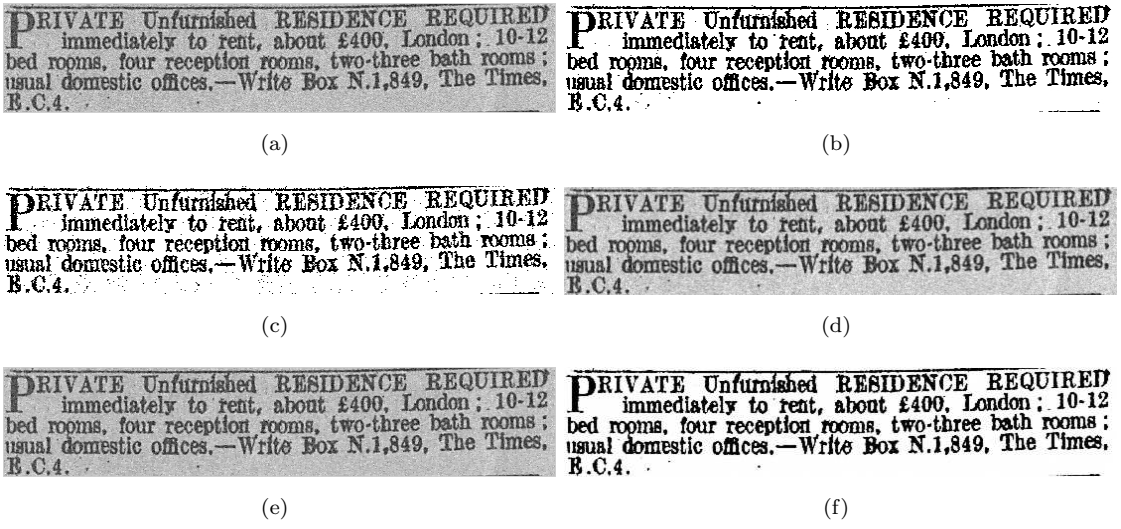


FIGURE 3.6: Binarization of an (a) initial noisy newspaper clip, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result.

algorithm has incorrectly classified pixels. In the cases of document image binarization Otsu's method exhibits an extreme sensitivity to noise. This is attributed to design of



FIGURE 3.7: Binarization of an (a) initial noisy restaurant bill, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result.

the method, in which a global threshold is chosen to binarize an image. In contrast our method, in essence, dynamically adjusts the threshold of each pixel relative to our globally-chosen threshold parameter, a, to achieve a binarized image with very little noise sensitivity and a desirable result.

We have implemented the Crank-Nicolson discretization treating the nonlinear source term explicitly at each step. The source term can be treated implicitly by using several iterations of Newton's method for nonlinear systems at each time step at an additional computational cost.

The inclusion of a heuristic optimization algorithm to automatically determine parameters of the model could be pursued as further work.

Our method was applied to camera-captured images, scanned images, artificially and naturally degraded images yielding excellent results as well as robustness to noise and preservation of textual information.

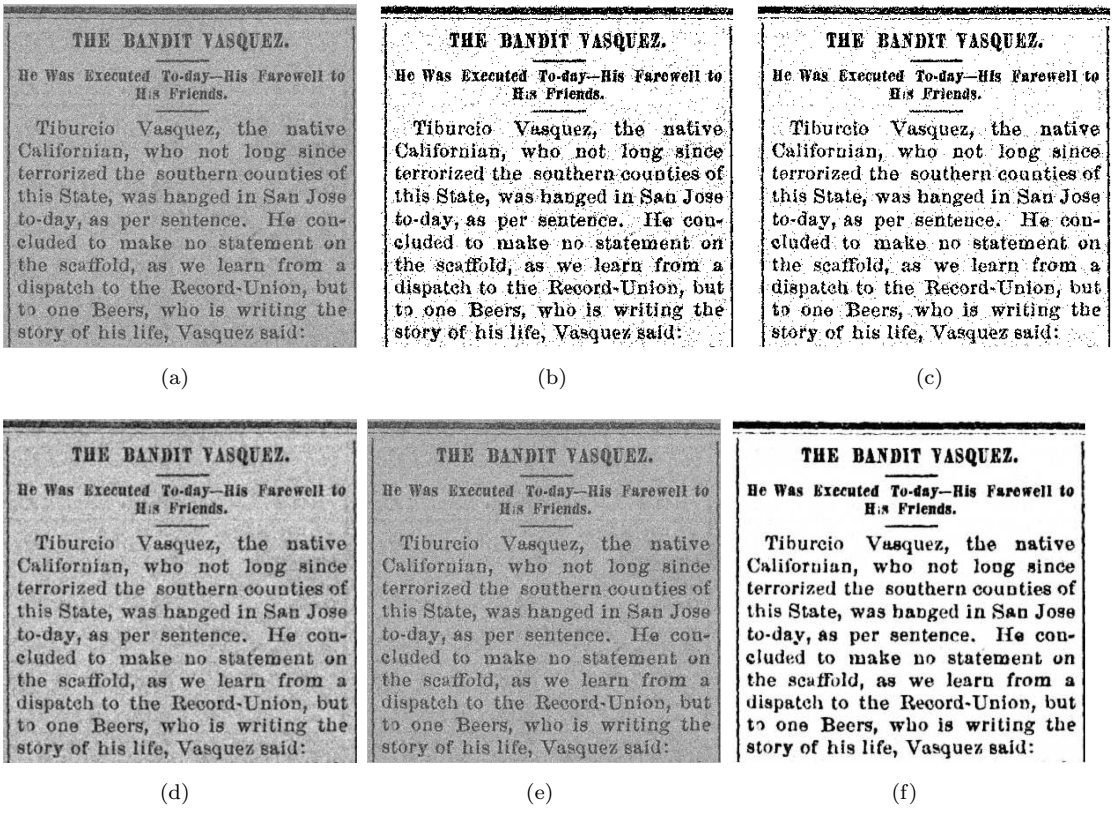


FIGURE 3.8: Binarization of an (a) initial noisy newspaper article, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result.

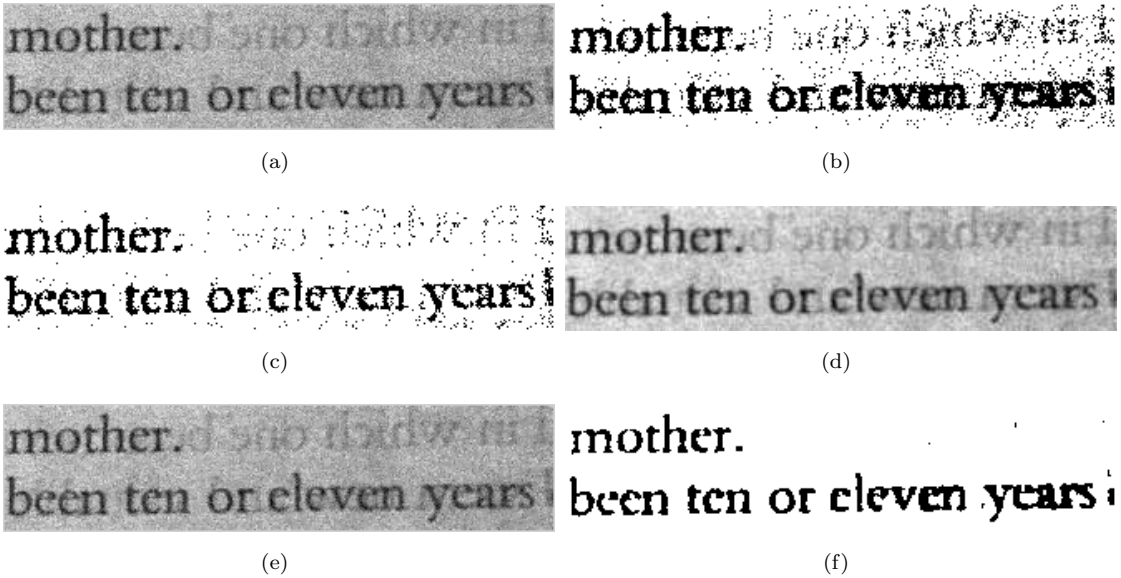


FIGURE 3.9: Binarization of an (a) initial noisy novel extract, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result.

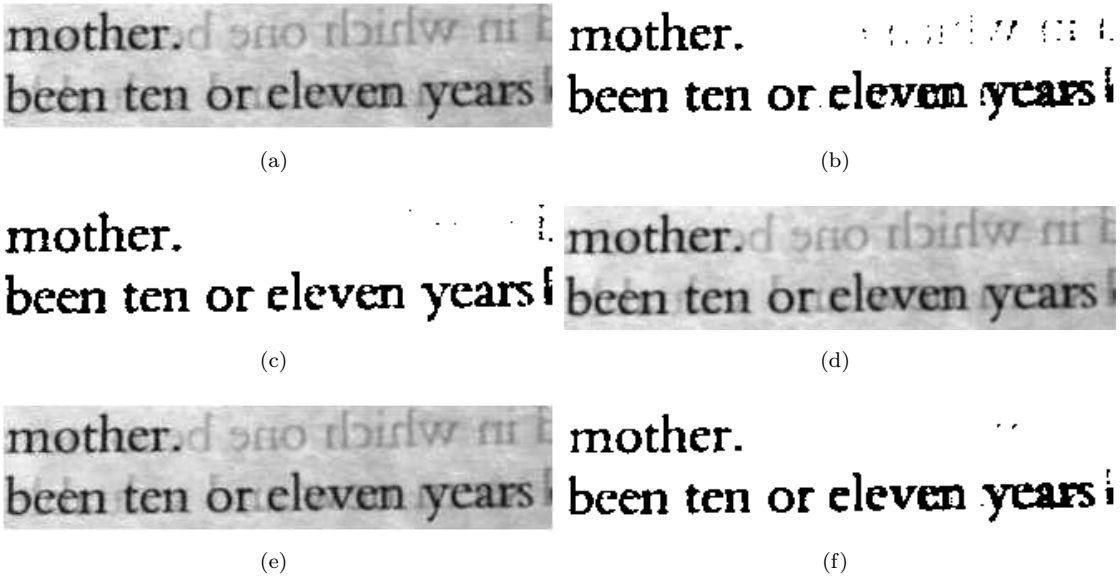


FIGURE 3.10: Binarization of a (a) novel extract, (b) result of Otsu's method, (c) result of Sauvola's method, (d) result of Linear Diffusion method, (e) result of Perona-Malik method and (f) our result.

TABLE 3.1: Table Listing Percentage of Character Correctly Identified.

Image/Method	**Original**	**Otsu**	**Sauvola**	**Diffusion**	**Perona-Malik**	**Our Method**
Newspaper clip	32.09	**67.58**	63.19	34.07	30.77	67.03
Newspaper article	67.04	83.33	94.44	**95.19**	88.52	94.81
Restaurant bill	52.69	39.78	78.49	68.82	60.22	**80.65**
Novel extract	37.04	29.63	70.37	70.37	44.44	**81.48**
Novel without noise	85.19	85.19	**88.89**	85.19	77.78	**88.89**

TABLE 3.2: Table Listing Percentage of Pixels Accurately Binarized.

Image/Method	**Original**	**Otsu**	**Sauvola**	**Our Method**
Newspaper clip	94.20	92.93	94.39	**95.87**
Newspaper article	93.96	93.95	93.69	**95.44**
Restaurant bill	84.43	77.80	96.46	**97.90**
Novel extract	93.09	90.28	95.23	**97.95**
Novel without noise	96.67	96.97	98.14	**98.20**

TABLE 3.3: Table Listing the l_1 Error Norm Between Obtained Image and Ground Truth.

Image/Method	**Original**	**Otsu**	**Sauvola**	**Linear Diffusion**	**Perona-Malik**	**Our Method**
Block	44.7098	2	50	38.5255	44.5255	**1**
The	30.0824	**8**	10	25.8157	30.0196	**8**
Newspaper clip	42.1961	23	17	39.3529	42.4627	**18**
Newspaper article	177.7804	98	117	160.251	177.6196	**74**
Restaurant bill	81.5843	107	82	68.7334	81.698	**18**
Novel extract $0dB$	25.6706	21	9	29.9412	25.5216	**8**
Novel extract $5dB$	29.2902	22	9	31.298	29.2549	**7**
Novel extract $10dB$	31.9725	26	9	30.8275	31.9412	**8**
Novel extract $15dB$	33.7765	38	16	33.4902	33.8392	**11**
Novel extract $20dB$	33.2157	41	17	32.7725	33.0745	**11**
Novel extract $25dB$	37.8	40	22	40.4627	37.5765	**12**
Novel extract $30dB$	38.8471	37	23	34.9647	39.0196	**13**

Chapter 4

A Locally Adaptive, Diffusion Based Text Binarization Technique

"A mathematician, like a painter or poet, is a maker of patterns. If his patterns are more permanent than theirs, it is because they are made with ideas."

Godfrey Harold Hardy

This work has been submitted under:

Jacobs, B. A., and E. Momoniat. "A Locally Adaptive, Diffusion Based Text Binarization Technique." *Pattern Recognition.*

This research proposes an adaptive modification to a novel diffusion based text binarization technique. This technique uses linear diffusion with a nonlinear source term to achieve a binarizing effect. This simple isotropic process is compared to the state-of-the-art Document Image Binarization Contest (DIBCO) contestants and produces remarkable results given the simplicity of the algorithm. Furthermore, the authors show how using a simple discretization scheme allows for the massively parallel implementation of this process.

4.1 Introduction

Document image binarization is a process in which an input image is segmented so that the text is contained in one segment, represented by black, and the background information is contained in the other, represented as white. This has become an important preliminary step in document image analysis and allows optical character recognition (OCR) algorithms to work more effectively.

In the recent years document image binarization has received a substantial effort. A thorough survey of this work is presented in [1–4]. More recently there has been research done on the application of tensor based anisotropic diffusion processes [79–82]. These efforts include performing backward diffusion to undo the effects of ink diffusion on paper [81] which are typically associated with document degradation, as well as applying an anisotropic process to a low quality input image captured by a low cost imaging device or camera phone [82] to obtain a binary image. In stark contrast to these processes the process presented here is extremely simple, isotropic and still effective.

In broadly describing the binarization process we begin by representing our initial data as a two-dimensional matrix, or a height map, where the values at an (x, y) coordinate indicate the height of that pixel. These binarization techniques construct a threshold value such that if we slice the height map at this value we will segment the image in a meaningful way. That is to say, all pixels above the threshold will be classified as foreground and all those pixels below the threshold will be classified as background. Global methods construct this threshold based on information derived from the entire image.

Using a globally defined threshold leads to the inadequacy of thresholding every pixel according to irrelevant information. Locally adaptive methods improve upon this by selecting a threshold based on local information.

The authors recently introduced [88] a novel approach to document image binarization based on a diffusion model with a nonlinear source term. The model can be written as

$$\frac{\partial u}{\partial t} = c_d \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + c_s u(1-u)(u-a), \tag{4.1}$$

subject to

$$u(x, y, 0) = \text{Image}(x, y), \tag{4.2}$$

and Neumann boundary conditions on the image boundary. Here c_d represents the coefficient of diffusion, c_s is the scaling coefficient of the source term, a is a thresholding parameter and Image(x, y) is the initial image data. The first term in equation (4.1) describes a diffusion process and the second describes a thresholding process. The benefits of this model are outlined in [88] but perhaps the reactive behavior of this process is the strongest asset. In [88] we choose the thresholding parameter a globally. In this paper we extend the model by selecting a based on local information making the process locally adaptive.

By localizing the process we are able to process each local sector independently. This means that this process is inherently parallel. Furthermore we discretize our model explicitly so that we may operate on the discrete input data such that each iteration of the method is dependant only on the immediately preceding iteration. Therefore the use of a General Purpose Graphics Processing Unit (GPGPU) arises naturally. GPGPU programming flourishes when an application is massively parallel in nature.

GPUs are becoming more and more a part of any computationally intensive research and the benefits of GPGPU programming are evident in the literature. Cruz et al. [89] give an introduction to GPU architecture and show the efficacy of this technique through application of two fast summation algorithms (fast Gaussian Transform and fast Multipole Method). Su and Xu [90] implement a wavelet-based image denoising algorithm on GPU hardware. The interested reader is directed to [91, 92] for a description of the GPU architecture mechanics. By discretizing (4.1) we may reduce applying this PDE to an image to a series of linear algebraic operations. Krüger and Westermann [93] develop strategies for implementing these linear algebra operators on a GPU. Bolz et al. [94] implement an efficient sparse matrix solver using GPU architecture for solving linear systems, which arise in implicit discretization schemes. Typically large computational costs are involved in the transfer of image data to and from the GPU memory. Fortunately modern GPU hardware has enough memory to allow for a single transmission of data in each direction, without the need to swap in sections of the data set. This makes the problem of an isotropic, explicit scheme even more amenable to GPGPU programming.

In Section 4.2.1 we further substantiate our construction of this model by discussing the merits of a diffusion based model. We then go on to show how the model is localized in Section 4.2.2. A simple implementation of our method is presented and applications of GPGPU are highlighted in Section 4.3. The results in Section 4.4 illustrate the efficacy

of our adaptive method. In the interest of brevity we choose not to compare our method with the extensive range of binarization techniques that exist in the literature. Instead we compute standard performance measures which are used in the Document Image Binarization Competition (DIBCO) series as a benchmark. This allows us to compare our method with the state-of-the-art methods without the need to compare our method with a plethora of new techniques. We do, however, provide some illustrative examples to indicate the performance of our method and subjectively compare with the seminal method of Sauvola [86]. We finally make some concluding remarks in Section 4.5.

4.2 Model Derivation

4.2.1 Diffusion and a Source Term

The effect of image denoising has been well established in the literature and the interested reader is directed to [5, 56–59, 61]. The basis of our model is the well known diffusion equation in two-dimensions.

$$\frac{\partial u}{\partial t} = c_d \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right). \tag{4.3}$$

We affirm the use of a diffusion based equation by means of a simple example. Consider the small window of an image.

$$\begin{pmatrix} 0.82 & 0.93 & 0.87 \\ 0.90 & 0.49 & 0.95 \\ 0.93 & 0.95 & 0.88 \end{pmatrix}$$

If the threshold is determined to be 0.5 then the center pixel would be classified as background and the surrounding pixels as foreground. The diffusion model will circumvent this and draw the center pixel's value toward its neighbors, homogenizing the area and classifying the outlying center pixel as a foreground element. The reactive nature of this model was briefly alluded to in the introduction, but it is this dynamic behavior that empowers the method.

A consequence of diffusion is the blurring of information and while some homogeneity may assist in cementing the binarization process, too much may inhibit it. We therefore

include the Fitzhugh-Nagumo inspired source term

$$c_s u(1-u)(u-a) \tag{4.4}$$

which exhibits binarizing properties and arrive at the model described in equation (4.1). The parameters c_d and c_s may be used to tweak the aggressiveness of each of the processes, diffusion and binarizing respectively.

The source term is intelligently constructed so as to inject energy into a pixel provided its energy is above the specified threshold and below maximum energy, 1. Furthermore the source term will withdraw energy if the pixel's energy is below the specified threshold and above minimum energy, 0. The result of this behavior is that each pixel is binarized either to 1 or to 0. The powerful dynamic of this model is that the fate of any given pixel is not determined only by the initial information but the information contained in its neighbors and the movement incurred by diffusion. A full stability analysis and substantiation of this model is given in [88].

4.2.2 Localizing the Model

The first step in localizing this process is to divide the input image into smaller sub images. The size of this window is dependant on the input image, if the image is relatively homogenous or has a small variance then we may assume a uniform illumination and hence we may use larger windows. If the image has non-uniform illumination it will also have a larger variance, and consequently we choose smaller windows.

By choosing a threshold based on local information we are able to binarize our data more appropriately. If for example we have nonuniform illumination this means that our background is not constant and that locally the threshold between background and foreground is not constant across the image. Consider again a simple example of three vertical stripes. The left and right most stripes are in the background and the center stripe is in the foreground.

$$\begin{pmatrix} 0.52 & 0.63 & 0.53 \\ 0.59 & 0.69 & 0.55 \\ 0.51 & 0.70 & 0.58 \end{pmatrix}$$

This illustrates a case where the foreground is similar in intensity to the background, like black text on a grey page. If we were to choose a global threshold of say 0.5 the entire section would be classified as foreground, contrastingly if we choose the threshold

based on this small window alone perhaps a threshold of 0.6 would be more appropriate.

Given this argument for a local threshold we simply adapt our model so that a is computed based on local information. One caveat to this process is that if our local window becomes too small we may lose perspective of the schematic of text and produce a blocky binarization where the topological significance of the text is lost.

4.3 Implementation: A Massively Parallel Approach

In order to apply our model to discrete data we must discretize equation (4.1). We notate the discrete approximation of $u(x_i, y_j, t_k)$ as $u_{i,j}^k$. Using a standard forward difference approximation for the first derivative term,

$$\frac{\partial u}{\partial t} \approx \frac{u_{i,j}^{k+1} - u_{i,j}^k}{\triangle t}, \tag{4.5}$$

and a central difference approximation for the second derivatives,

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u_{i+1,j}^k - 2u_{i,j}^k + u_{i-1,j}^k}{\triangle x^2}, \tag{4.6}$$

and

$$\frac{\partial^2 u}{\partial y^2} \approx \frac{u_{i,j+1}^k - 2u_{i,j}^k + u_{i,j-1}^k}{\triangle y^2}. \tag{4.7}$$

Applying equations (4.5), (4.6), (4.7) to equation (4.1) we obtain the iterative scheme

$$\begin{aligned} u_{i,j}^{k+1} =& u_{i,j}^k + \\ & \triangle t c_d \left(\frac{u_{i-1,j}^k - 2u_{i,j}^k + u_{i+1,j}^k}{\triangle x^2} \right) + \\ & \triangle t c_d \left(\frac{u_{i,j-1}^k - 2u_{i,j}^k + u_{i,j+1}^k}{\triangle y^2} \right) + \\ & \triangle t c_s u_{i,j}^k \left(1 - u_{i,j}^k\right) \left(u_{i,j}^k - a\right). \end{aligned} \tag{4.8}$$

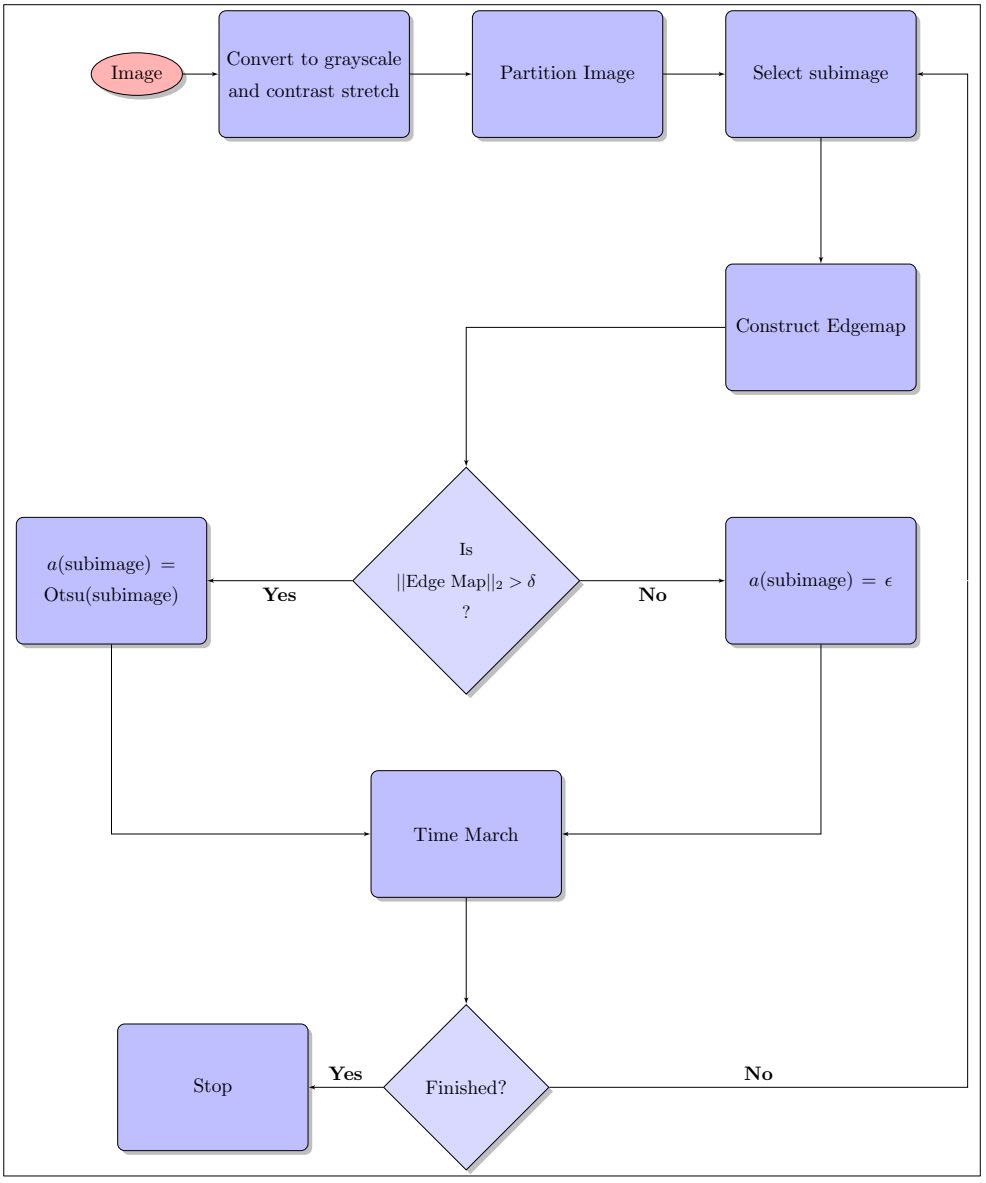
We apply this iterative scheme to every pixel in the input image for $k = 0, 1, \ldots, N$. Recalling the massively parallel nature of this process, we highlight here that the above scheme can be applied to every (i, j) pair simultaneously.

Although the explicit finite difference scheme requires $\triangle t \ll \triangle x^2$ to ensure stability this scheme is parallelizable. Implicit schemes require matrix inversion or solutions to linear

system, which may be optimized but are not inherently parallel therefore computational bottlenecks will occur in the serial processes.

We select a as follows. For each sub image of the input image we construct an edge map and compute the l_2 or energy norm, $\|\cdot\|_2$, of this edge map. The edge map can be found in many ways, here we make use of the Shen-Castan [95] edge detection algorithm. The l_2 norm of this image essentially gives a metric for the 'amount of detail' contained in a sub image. So a sub image that is homogenous will contain almost no edge energy, but a sub image that contains text will have edge energy. If the sub image has a high edge energy we assign a to be the Otsu [84] threshold otherwise we take a to be slightly above the mean of the sub image, so that everything below a is pushed into background.

4.3.1 Algorithm Description



Algorithm 1 Text Binarization via Diffusion Based Model

procedure INPUT(*image*) ▷ Input image to be binarized.
    Convert image to grayscale.
    Partition image into an image array.
    for each subimage in the image array **do**
        EdgeMetric = $\|\text{subimage}\|_2$
        if EdgeMetric $> \delta$ **then** a(subimage) = $Otsu$(subimage)
        else a(subimage) = ϵ
        end if
    end for
    Perform N iterations for every (i, j) pixel. (Done on the GPU.)
end procedure

4.4 Experimental Results

We present some binarized images in Figures 4.1, 4.2 and 4.3 as an illustration of the efficacy of our new method as well as a subjective source of comparison. Our method exhibits a robustness to noise which is evident in Figure 4.2 which has been corrupted with Gaussian noise. We make use of a hand constructed ground truth images for these example as well as the datasets provided by DIBCO 2009 through 2012. The following evaluation measures are used as an objective measure of efficacy. The performance measures are introduced and used in the Document Image Binarization Competitions (DIBCO). We also make use of the data sets provided by the 2009, 2010, 2011 and 2012 [96–99] competitions as a benchmark.

4.4.1 Evaluation Measures

In following the evaluation measures implemented in the DIBCO for state-of-the-art methods we make use of the following measures as described in [96–99], and the interested reader is directed there for a full description of the implementations of these measures.

The F-Measure is a test of accuracy based on both precision and recall, where precision is the proportion of true positives to true positives and false positives and recall is the proportion of true positives to true positives and false negatives. The F-Measure is then a weighted average of precision and recall, and is maximally 100.

The pseudo F-Measure was introduced by Ntirogiannisas et al. [100] and first used as a performance metric in DIBCO 2010 [97]. It is based on the idea that each character has a unique silhouette that can be represented by its skeleton. This method emulates one's natural interaction when identifying characters, by drawing a curve one pixel wide approximately along the character's medial axis. The skeletonized ground truth image is then used in evaluating a method's ability to reconstruct the most important features used in identifying characters. As with the F-Measure, a score of 100 is perfect.

Peak signal-to-noise ratio (PSNR) is a commonly used performance measure of how close one image is to another image. Hence a higher value indicates a higher similarity between two images.

Finally the Distance Reciprocal Distortion Metric (DRD) measures the visual distortion in binary document images. It was introduced in [101] and attempts to correlate the performance of a method with the human visual perception of an image. A lower DRD score indicates a better performance.

4.4.2 Results

We present here a selection of sample images. Typically these results are susceptible to subjective scrutiny. We provide images here for a qualitative source of comparison. We have chosen to compare our method to the seminal method of Sauvola [86] as a means of subjective comparison. While we recognize that Sauvola's method no longer represents the state-of-the-art it does represent a class of adaptive schemes, and particularly a sensitivity to noise that our method exhibits robustness towards. Figure 4.1 illustrates the efficacy of our method on an image with non-uniform illumination typically associated with flash aided photographs. The effects of text bleed-through are tested in Figure 4.2 as well as noise corruption. Finally we show that the proposed method is able to handle sharp shadows in Figure 4.3. These examples have been selected to cover an array of image faults typically associated with text-binarization.

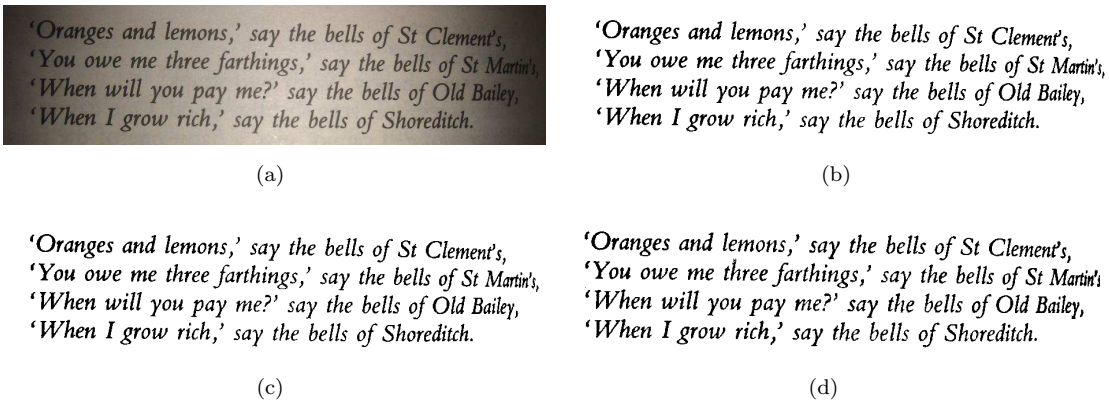


FIGURE 4.1: (a) Sample image of a novel excerpt taken with a mobile device. (b) Ground truth image. (c) Sauvola's binarized result. (d) Our proposed method.

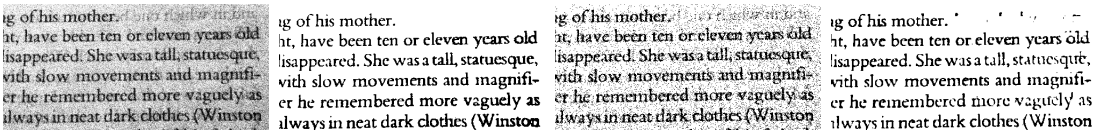
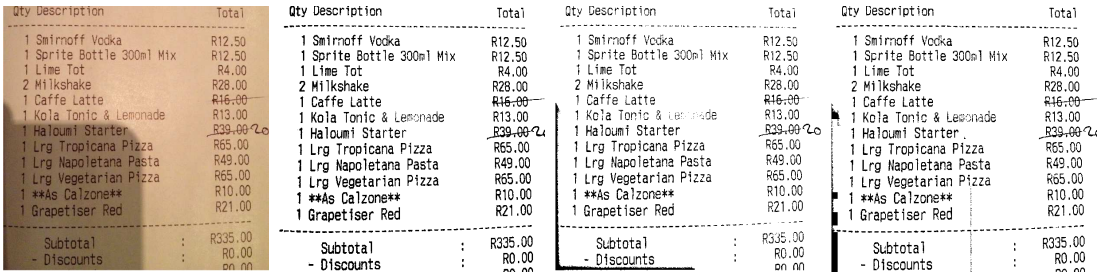


FIGURE 4.2: (a) Sample image of a novel excerpt taken with a mobile device. (b) Ground truth image. (c) Sauvola's binarized result. (d) Our proposed method.

Table 4.1 presents a quantitative measure for the DIBCO datasets and each score calculated as an average over the whole dataset. The best and worst performances of the contestants in the DIBCO are presented in Table 4.2. Our method consistently performs well within the range of participants, and while we do not achieve results better than every method the present method certainly performs admirably among the state-of-the-art methods, most of which are quite sophisticated.



(a) (b) (c) (d)

FIGURE 4.3: (a) Sample image of a restaurant bill taken with a mobile device. (b) Ground truth image. (c) Sauvola's binarized result. (d) Our proposed method.

TABLE 4.1: Table of various performance measures for a variety of datasets.

Dataset	**F-Measure**	**p F-Measure**	**PSNR**	**DRD**
Poem	94.5058	96.5595	18.9591	1.06575
1984 Excerpt	88.8768	90.9416	14.5885	2.40671
Restaurant Bill	87.2407	89.1336	15.9774	2.72693
DIBCO 2009	75.7127	76.4687	13.7456	7.45795
DIBCO 2010	67.6232	68.7829	14.6626	8.01906
DIBCO 2011	77.9079	74.6012	14.6867	6.27623
DIBCO 2012	81.5716	81.4511	16.2618	5.98124

TABLE 4.2: Table indicating the maximal and minimal performance measures of all competition entrants for each year.

	F-Measure		**p F-Measure**		**PSNR**		**DRD**	
Dataset	min	max	min	max	min	max	max	min
DIBCO 2009	35.28	91.24	-	-	9.22	19.66	-	-
DIBCO 2010	57.73	91.78	66.42	95.15	14.29	19.78	-	-
DIBCO 2011	58.74	88.73	54.34	88.02	9.86	17.85	566.194	3.416
DIBCO 2012	75.23	79.41	79.41	95.09	15.94	21.80	23.107	2.660

The results presented in this section clearly illustrate the efficacy of the proposed method for binarizing document images with non-uniform illumination, and high levels of noise typically associated with low quality cameras, such as those found on camera phones and mobile devices. The present method is exceedingly simple and intuitive while still performing desirably. Furthermore the proposed method is novel in its derivation from a partial differential equation as well as the isotropic nature. These features result in an elegant yet simple method that is able to contend well with other methods. Approaching the field of document image binarization from a partial differential equation model opens the field to new and interesting approaches to solving this problem.

4.5 Conclusion

This research has presented an elegant application of differential calculus to document image binarization. We have shown how a physically derived equation may be tailored to produce pleasing results in the field. A simple extension to the model that calculates a adaptively, allows this method to be applied to images with nonuniform illumination as well as high levels of noise and shadows as illustrated.

Furthermore this binarization algorithm enjoys the benefit of being derived from a physical model. This allows one to make use of the extensive numerical tools available for solving partial differential equations. In this work we have employed a simple discretization so that a massively parallel scheme may be exploited, leading to a computationally efficient method. We omit a comparison of computational expense versus a serial implementation of this method in the interest of brevity, however the gains in efficiency of parallel implementations of finite difference schemes are well known, and go so far as to compute real-time solutions of finite difference schemes [102–104].

We have compared our method with those that compete at the highest level in the field and have achieved comparable results based on the DIBCO benchmark. This is compounded with the fact that to the best of the authors' knowledge the current method is the only isotropic method derived from a partial differential equation.

Chapter 5

A Comparison of Two Hybrid Methods for Solving Linear Time-Fractional Partial Differential Equations on a Two-Dimensional Domain

"The mathematician does not study pure mathematics because it is useful; he studies it because he delights in it and he delights in it because it is beautiful."

Henri Poincare

This work has appeared in:

Jacobs, B. A., and C. Harley. "A Comparison of Two Hybrid Methods for Solving Linear Time-Fractional Partial Differential Equations on a Two-Dimensional Domain." *Abstract and Applied Analysis Special Issue: New Trends on Fractional and Functional Differential Equations*. Vol. 2014. Hindawi Publishing Corporation, 2014

A computationally efficient hybridization of the Laplace Transform with two spatial discretization techniques is examined for comparison in accuracy for the solution to time-fractional linear partial differential equations on a bounded domain. The Chebyshev Collocation method is compared with the standard Finite Difference spatial discretization and the absolute error is obtained from the exact solution for select examples as a

measure of performance. Extremely small errors are obtained in the Chebyshev Collocation case under both Dirichlet and Neumann boundary conditions. The hybridization with a transform allows a semi-analytic solution to be obtained that may be evaluated anywhere in time without the need to time-march to a particular point in time.

5.1 Introduction

In recent years fractional derivatives and fractional partial differential equations (FPDEs) have received great attention both in analysis and application (see [11, 32, 105] and references therein). In spite of this there has been very little work done on solving FPDEs on a bounded domain. Agrawal [106] makes use of the Laplace transform and the finite sine transform to obtain an analytic solution to the fractional diffusion-wave equation on a bounded domain. Many authors have applied He's variational iterative method (VIM) [107] to FPDEs with great success. However, like the differential transform method [108] (DTM) and Adomian decomposition method [109](ADM), VIM assumes the FPDE to lie on an infinite domain. To the best of the authors' knowledge, these transform techniques are unable to enforce Dirichlet or Neumann boundary conditions on a bounded domain, and as such we investigate a new methodology to attempt to circumvent this.

The application of diffusion equations to images is abundant, [56–61]. However the application of a time-fractional partial differential equation has not yet been thoroughly examined. Preserving the spatial topography of an image is imperative to maintaining what is deemed to be useful information in that image. It is with this application in mind that we focus on the diffusion equation in two-dimensions, where the introduction of an advection term into the model would propagate information. Similarly considering the time-fractional diffusion equation with $1 < \alpha < 2$ we obtain the diffusion-wave equation, which again has propagational properties. We therefore restrict our choice of α to be on $[0, 1]$. The introduction of a fractional-order derivative raises the question of how to discretize or transform the derivative to produce a form that is amenable to existing techniques.

The Grünwald-Letnikov discretization has been used for numerical schemes for fractional partial differential equations [34–36] and is given by

$$D_t^\alpha u(t) = \lim_{\triangle t \to 0} \frac{1}{\triangle t^\alpha} \sum_{k=0}^{j} \omega_k^\alpha u(t - k\triangle t), \tag{5.1}$$

where

$$\omega_k^{\alpha} = (-1)^k \binom{\alpha}{k}, \tag{5.2}$$

and $M\triangle t = t$. Computationally this discretization becomes extremely expensive for long time simulations as each subsequent step in time is dependant on *every* time step that has preceded it.

The use of a transform method, such as the Laplace transform, allows one to circumvent the problems that arise in the time-domain discretization. However, using the Laplace transform for the fractional order derivative presents the problem of inverting the transform to find a solution. Analytic inversion of the transform is infeasible and hence the numerical scheme presented by Weideman and Trefethen in [110] is put to extensive use. The algorithm presents a method for defining the parameters of the conformal mapping to invert the Laplace transform by evaluating the Bromwich integral using the trapezoidal rule, achieving near optimal results (see [110] and figures therein). In this work we make use of the parabolic contour due to the ease of use and the hyperbolic contour only exhibits a slight improvement in performance over the parabolic contour.

We present here an extension to the work conducted by Jacobs and Harley in [111]. This paper extends the aforementioned work to the general form of a time-fractional parabolic partial differential equation as well as including two types of boundary conditions, dirichlet and neumann.

The to compare finite differences with Chebyshev collocation is largely informed by the similarity in structure of numerical schemes juxtaposed by the dissimilarity in underlying principles. The collocation technique makes use of global interpolants whereas the finite difference interpolates linearly within local neighborhoods. There is an overhead in complexity and computation associated with the Chebyshev collocation however there is a gain in convergence rate and accuracy versus discretization resolution. It is this idea of opposing yet similar approaches that justify the comparison of these two specific discretization methods.
The following section introduces some preliminary definitions followed by a description of the methods employed, including the different cases for boundary conditions in Section 5.3. Section 5.5 presents the results for comparison based on three fundamentally different examples of linear FPDEs. A discussion of the results and their relationship to work beyond this research is presented in Section 5.6 and some concluding remarks are made in Section 5.7.

5.2 Preliminaries

In this work we employ Caputo's definition of a fractional derivative over the Riemann-Louiville derivative due to the fact that the Caputo derivative makes use of the physical boundary conditions, whereas the Riemann-Louiville derivative requires fractional order boundary conditions.

Definition 7. The Riemann-Louiville integral of order $\alpha > 0$ of a function $u(t)$ is

$$J^{\alpha}u(t) = \frac{1}{\Gamma(\alpha)}\int_0^t (t-\tau)^{\alpha-1}u(\tau)d\tau, \quad x > 0. \tag{5.3}$$

Definition 8. The fractional derivative of $u(t)$ according to the Caputo definition with $m-1 < \alpha \leq m$, $m \in \mathbb{N}$, is

$$\frac{\partial^{\alpha}u(t)}{\partial t^{\alpha}} = J^{m-\alpha}D^m u(t) = D_*^{\alpha}u(t) := \frac{1}{\Gamma(m-\alpha)}\int_0^t (t-\tau)^{m-\alpha-1}u^{(m)}(\tau)d\tau. \tag{5.4}$$

If $\alpha = 1$ the Caputo fractional derivative reduces to the ordinary first order derivative. Podlubny [32] illustrates the pleasing property of the Laplace Transform of a Caputo derivative, as can be seen in Eq. (5.5). In our case where $0 < \alpha < 1$ we have,

$$\mathcal{L}\left\{\frac{\partial^{\alpha}u(x,y,t)}{\partial t^{\alpha}}\right\} = s^{\alpha}U(x,y) - s^{\alpha-1}u(x,y,0). \tag{5.5}$$

This property allows one to treat fractional order derivatives algebraically.

Definition 9. The Generalized Mittag-Leffler function of the argument z

$$E_{\alpha,\beta}(z) = \sum_{k=0}^{\infty}\frac{z^k}{\Gamma(k\alpha+\beta)}. \tag{5.6}$$

5.3 Methods: Semi-Discrete Hybrid Transform Method

This section introduces the methodologies used for a two-dimensional FPDE, where the one-dimensional case is a simple reduction of the methods presented here.

Consider the time-fractional differential equation of the form

$$\frac{\partial^{\alpha}u}{\partial t^{\alpha}} = L(u, u_x, u_{xx}, u_y, u_{yy}), \qquad (x,y) \in \Omega \subset \mathcal{R}^2 \tag{5.7}$$

with

$$u(x,y,0) = f(x,y), \tag{5.8}$$

where L is a linear function of its arguments, $\Omega = [-1,1] \times [-1,1]$ to satisfy the domain required by the Chebyshev polynomials and $f(x,y)$ is a functional representation of our image data or a multivariable function. The boundary conditions may be taken to be Dirichlet or Neumann and will be discussed later. We may now apply a Laplace Transform to Eq. (5.7) to obtain,

$$s^{\alpha}U(x,y) - s^{\alpha-1}f(x,y) = \mathcal{L}\left\{L(u,u_x,u_{xx},u_y,u_{yy})\right\}, \tag{5.9}$$

$$s^{\alpha}U(x,y) - s^{\alpha-1}f(x,y) = L(U,U_x,U_{xx},U_y,U_{yy}), \tag{5.10}$$

where

$$\mathcal{L}\left\{u(x,y,t)\right\} = U(x,y,s). \tag{5.11}$$

Boundary conditions may be in the form of Dirichlet conditions,

$$u(-1,y,t) = a, \quad u(1,y,t) = b, \tag{5.12}$$

$$u(x,-1,t) = c, \quad u(x,1,t) = d, \tag{5.13}$$

and hence,

$$U(-1,y) = \mathcal{L}\left\{a\right\}, \quad U(1,y) = \mathcal{L}\left\{b\right\}, \tag{5.14}$$

$$U(x,-1) = \mathcal{L}\left\{c\right\}, \quad U(x,1) = \mathcal{L}\left\{d\right\}. \tag{5.15}$$

Alternatively Neumann boundary conditions give,

$$u_x(-1,y,t) = a, \quad u_x(1,y,t) = b, \tag{5.16}$$

$$u_y(x,-1,t) = c, \quad u_y(x,1,t) = d, \tag{5.17}$$

with,

$$U_x(-1,y) = \mathcal{L}\left\{a\right\}, \quad U_y(1,y) = \mathcal{L}\left\{b\right\}, \tag{5.18}$$

$$U_y(x,-1) = \mathcal{L}\left\{c\right\}, \quad U_y(x,1) = \mathcal{L}\left\{d\right\}. \tag{5.19}$$

The parameters a, b, c and d are potentially functions of the temporal variable and one of the spatial variables, i.e. $a = a(y,t)$. Without loss of generality however, we assume that a, b, c and d are constant.

The spatial components of this model are discretized in two ways - by Cheyshev Collocation and by finite differences.

5.3.1 Chebyshev Collocation

Chebyshev polynomials form a basis on $[-1,1]$ and hence we dictate the domain of our PDE to be Ω. We note here however, that any domain in $\mathbb{R}^2$ can be trivially deformed to match Ω. We discretize our spatial domain using Chebyshev-Gauss-Labatto points,

$$x_i = \cos\left(\frac{i\pi}{N_x}\right), \quad i = 0,1,\ldots,N_x, \qquad y_j = \cos\left(\frac{j\pi}{N_y}\right), \quad j = 0,1,\ldots,N_y. \tag{5.20}$$

Note here that $x_0 = 1$, $x_{N_x} = -1$, $y_0 = 1$ and $y_{N_y} = -1$ indicating that the domain is in essence reversed and one must use caution when imposing the boundary conditions.

Given that our input function or image has been mapped to Ω we may assume that $N_x = N_y$, i.e. we have equal number of collocation points in each spatial direction. We now define a differentiation matrix $D^{(1)} = d_{kl}$,

$$d_{kl} = \begin{cases} \frac{c_k(-1)^{k+l}}{c_l(x_k - x_l)}, & k \neq l, \\ -\frac{x_k}{2(1-x^2{}_k)}, & k = l, \\ \frac{1}{6}\left(2N_x^2+1\right), & k = l = 0, \\ -\frac{1}{6}\left(2N_x^2+1\right), & k = l = N_x, \end{cases} \quad \text{where} \quad c_k = \begin{cases} 2, & k = 0, N_x, \\ 1, & k = 1,\ldots,N_x - 1. \end{cases} \tag{5.21}$$

Bayliss et al. [75] describe a method for minimizing the round off errors incurred in the calculations of the differentiation matrix. Since we write $D^{(2)} = D^{(1)}.D^{(1)}$ we implement the method, described in [75], in order to minimize propagation of round off errors for the second derivative in space.

The derivative matrices in the x direction are

$$\hat{D}_x^{(1)} = D^{(1)}, \qquad \hat{D}_x^{(2)} = D^{(2)}, \tag{5.22}$$

where $D^{(1)}$ is the Chebyshev differentiation matrix of size $(N_x+1)\times(N_y+1)$. Because we have assumed $N_x = N_y$ we derive the pleasing property that $\hat{D}_y^{(1)} = \left(\hat{D}_x^{(1)}\right)^T$ and $\hat{D}_y^{(2)} = \left(\hat{D}_x^{(2)}\right)^T$.

Writing the discretization of Eq. (5.10) in matrix form yields,

$$\hat{D}_x^{(2)}\mathbf{U} + \mathbf{U}\hat{D}_y^{(2)} - s^\alpha \mathbf{U} = -\mathbf{F}, \tag{5.23}$$

where

$$F_{ij} = s^{\alpha-1} f(x_i, y_j). \tag{5.24}$$

By expanding Eq. (5.23) in summation notation we have

$$\sum_{k=0}^{N_x} d_{ik}^{(2)} U(x_k, y_j) + \sum_{k=0}^{N_y} d_{kj}^{(2)} U(x_i, y_k) - s^\alpha U(x_i, y_j) = -s^{\alpha-1} f(x_i, y_j), \tag{5.25}$$

for $i = 0, 1, \ldots, N_x, j = 0, 1, \ldots, N_y$. By extracting the first and last terms in sums we obtain

$$\begin{aligned} &d_{i0}^{(2)} U(x_0, y_j) + d_{iN_x}^{(2)} U(x_{N_x}, y_j) + d_{0j}^{(2)} U(x_i, y_0) + d_{N_y j}^{(2)} U(x_i, y_{N_y}) + \\ &\sum_{k=1}^{N_x-1} d_{ik}^{(2)} U(x_k, y_j) + \sum_{k=1}^{N_y-1} d_{kj}^{(2)} U(x_i, y_k) - s^\alpha U(x_i, y_j) = -s^{\alpha-1} f(x_i, y_j), \end{aligned} \tag{5.26}$$

for $i = 1, \ldots, N_x - 1, j = 1, \ldots, N_y - 1$. We use the form of Eq. (5.26) to impose the boundary conditions.

The solution $\tilde{\mathbf{U}} = \{U(x_1, y_1), U(x_1, y_2), \ldots, U(x_{N_x-1}, y_{N_y-1})\}$, which is the matrix of unknown interior points of $\mathbf{U}$, can be found by solving the system

$$(\tilde{\mathbf{A}} - \mathbf{s}^\alpha \mathbf{I})\tilde{\mathbf{U}} + \tilde{\mathbf{U}}\tilde{\mathbf{B}} = -\tilde{\mathbf{F}}, \tag{5.27}$$

where $\tilde{\mathbf{A}}$ is the matrix of interior points of $\hat{\mathbf{D}}_x^{(2)}$, and $\tilde{\mathbf{B}}$ is the matrix of interior points of $\hat{\mathbf{D}}_y^{(2)}$, so that $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{B}}$ match the dimensions of $\tilde{\mathbf{U}}$. Also

$$\tilde{F}_{ij} = s^{\alpha-1} f(x_i, y_j) + d_{i0}^{(2)} U(x_0, y_j) + d_{iN_x}^{(2)} U(x_{N_x}, y_j) + d_{0j}^{(2)} U(x_i, y_0) + d_{N_y j}^{(2)} U(x_i, y_{N_y}), \tag{5.28}$$

for $i = 1, \ldots, N_x - 1, j = 1, \ldots, N_y - 1$. By using Kronecker tensor products, denoted by $\otimes$, and a lexicographic reordering, or reshaping, of $\tilde{\mathbf{U}}$ and $-\tilde{\mathbf{F}}$ we may write this as

$$\left((\tilde{\mathbf{A}} - \mathbf{s}^\alpha \mathbf{I}) \otimes \mathbf{I} + \mathbf{I} \otimes \tilde{\mathbf{B}}^{\mathbf{T}}\right) \tilde{\mathbf{U}} = -\tilde{\mathbf{F}}, \tag{5.29}$$

which is a linear system that is readily solved.

5.3.1.1 Dirichlet Boundary Conditions

Dirichlet boundary conditions can be imposed directly by substituting Eqs. (5.12) and (5.13) into Eq. (5.26) and collecting all the known terms in $\tilde{\mathbf{F}}$.

5.3.1.2 Neumann Boundary Conditions

Neumann boundary conditions given by Eqs. (5.14) are discretized as

$$\frac{\partial U}{\partial x}\left(x_{N_x}=-1,y_j\right)\approx\sum_{k=0}^{N_x}d_{N_xk}U(x_k,y_j)=\mathcal{L}\left\{a\right\},\tag{5.30}$$

$$\frac{\partial U}{\partial x}\left(x_0=1,y_j\right)\approx\sum_{k=0}^{N_x}d_{0k}U(x_k,y_j)=\mathcal{L}\left\{b\right\}.\tag{5.31}$$

Similarly for Eqs. (5.15)

$$\frac{\partial U}{\partial y}\left(x_i,y_{N_y}=-1\right)\approx\sum_{k=0}^{N_y}d_{kN_y}U(x_i,y_k)=\mathcal{L}\left\{c\right\},\tag{5.32}$$

$$\frac{\partial U}{\partial y}\left(x_i,y_0=1\right)\approx\sum_{k=0}^{N_y}d_{k0}U(x_i,y_k)=\mathcal{L}\left\{d\right\}.\tag{5.33}$$

By extracting the first and last terms in the sum, the discretizations can be written as

$$\begin{pmatrix} d_{N_x0} & d_{N_xN_x} \\ d_{00} & d_{0N_x} \end{pmatrix}\begin{pmatrix} U(x_0,y_j) \\ U(x_{N_x},y_j) \end{pmatrix}=\begin{pmatrix} \mathcal{L}\left\{a\right\}-\sum_{k=1}^{N_x-1}d_{N_xk}U(x_k,y_j) \\ \mathcal{L}\left\{b\right\}-\sum_{k=1}^{N_x-1}d_{0k}U(x_k,y_j) \end{pmatrix}\tag{5.34}$$

and

$$\begin{pmatrix} d_{0N_y} & d_{N_yN_y} \\ d_{00} & d_{N_y0} \end{pmatrix}\begin{pmatrix} U(x_i,y_0) \\ U(x_i,y_{N_y}) \end{pmatrix}=\begin{pmatrix} \mathcal{L}\left\{c\right\}-\sum_{k=1}^{N_y-1}d_{kN_y}U(x_i,y_k) \\ \mathcal{L}\left\{d\right\}-\sum_{k=1}^{N_y-1}d_{k0}U(x_i,y_k) \end{pmatrix}\tag{5.35}$$

The solutions to these linear systems are then substituted into Eq. (5.26).

5.3.2 Finite Difference Discretization

Below we make use of the following finite difference formulae

$$\frac{\partial U}{\partial x}\approx\frac{1}{2\triangle x}\left(U_{i+1,j}-U_{i-1,j}\right),\tag{5.36}$$

$$\frac{\partial U}{\partial y}\approx\frac{1}{2\triangle y}\left(U_{i,j+1}-U_{i,j-1}\right),\tag{5.37}$$

$$\frac{\partial^2 U}{\partial x^2}\approx\frac{1}{\triangle x^2}\left(U_{i+1,j}-2U_{i,j}+U_{i-1,j}\right),\tag{5.38}$$

$$\frac{\partial^2 U}{\partial y^2}\approx\frac{1}{\triangle y^2}\left(U_{i,j+1}-2U_{i,j}+U_{i,-1j}\right),\tag{5.39}$$

where $\triangle x=2/N_x$ and $\triangle y=2/N_y$.

5.3.2.1 Dirichlet Boundary Conditions

Discretizing Eq. (5.10) using a standard central-difference scheme and writing in matrix notation we deduce

$$(\tilde{\mathbf{C}} - \mathbf{s}^{\alpha}\mathbf{I})\tilde{\mathbf{U}} + \tilde{\mathbf{U}}\tilde{\mathbf{D}} = -\tilde{\mathbf{G}}, \tag{5.40}$$

where $\tilde{\mathbf{C}}$ and $\tilde{\mathbf{D}}$ are tridiagonal matrices corresponding to the finite difference differential matrix with dimension $(N_x - 1, N_y - 1)$ and

$$\tilde{\mathbf{G}}_{ij} = s^{\alpha-1} f(x_i, y_j), \qquad i = 1, \ldots, N_x - 1, j = 1, \ldots, N_y - 1. \tag{5.41}$$

We write this as a linear system to be solved as follows

$$\left(\left(\tilde{\mathbf{C}} - \mathbf{s}^{\alpha}\mathbf{I}\right) \otimes \mathbf{I} + \mathbf{I} \otimes \tilde{\mathbf{D}}^{\mathbf{T}}\right) \tilde{\mathbf{U}} = -\tilde{\mathbf{G}}. \tag{5.42}$$

The boundary conditions, Eqs. (5.12) and (5.13), can be enforced directly onto the matrix $\mathbf{U}$, where the interior points of $\mathbf{U}$ are $\tilde{\mathbf{U}}$.

5.3.2.2 Neumann Boundary Conditions

By discretizing the boundary conditions Eq. (5.14) and Eq. (5.15) using a standard central difference scheme we derive

$$U_{-1,j} = U_{1,j} - 2\triangle x \mathcal{L}\{a\}, \tag{5.43}$$

$$U_{N_x+1,j} = U_{N_x-1,j} + 2\triangle x \mathcal{L}\{b\}, \tag{5.44}$$

$$U_{i,-1} = U_{i,1} - 2\triangle y \mathcal{L}\{c\}, \tag{5.45}$$

$$U_{i,N_y+1} = U_{i,N_y-1} + 2\triangle y \mathcal{L}\{d\}. \tag{5.46}$$

Including the above conditions in the matrix $\mathbf{C}$ and $\mathbf{D}$ each with dimension (N_x+1, N_y+1) we can write the entire system as

$$(\mathbf{C} - \mathbf{s}^{\alpha}\mathbf{I})\mathbf{U} + \mathbf{U}\mathbf{D} = -\mathbf{G}, \tag{5.47}$$

where

$$\mathbf{G}_{ij} = s^{\alpha-1} f(x_i, y_j), \qquad i = 0, \ldots, N_x, j = 0, \ldots, N_y. \tag{5.48}$$

This may be solved as a linear system by writing

$$\left((\mathbf{C} - \mathbf{s}^{\alpha}\mathbf{I}) \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{D}^{\mathbf{T}}\right) \mathbf{U} = -\mathbf{G}. \tag{5.49}$$

5.4 Analysis

5.4.1 Solvability

We have reduced all four cases to a system of linear systems, presented in Eqs. (5.29), (5.42) and (5.49). A system $\mathbf{Mx} = \mathbf{b}$ has a unique solution, $\mathbf{x} = \mathbf{M}^{-1}\mathbf{b}$, provided the matrix $\mathbf{M}$ has an inverse $\mathbf{M}^{-1}$ that exists. In the following analysis we denote the length in one dimension of the respective matrix $\mathbf{M}$ by L, since this length is scheme dependent.

5.4.1.1 Finite Difference Scheme

The finite difference structure allows us to put to use the well known condition that a positive-definite matrix is invertible. Therefore we derive a criterion on s that ensures the matrix $\mathbf{M}$, for the finite difference scheme, is always positive-definite. The structure of the matrix on the right hand side of Eqs. (5.42) and (5.49) defines this criterion as

$$|m_{ii}| > \sum_{\substack{k=1 \\ k \neq i}}^{L} |m_{ik}|, \tag{5.50}$$

which reduces to, in the pathological case,

$$|4 + s^{\alpha}| > 2. \tag{5.51}$$

In approximating the Bromwich integral by the trapezoidal rule we truncate the limits of integration from $(-\infty, \infty)$ to $[-N_t, N_t]$. The contour path of the integral is defined to be a parabola, denoted as s above, in the complex plane with a minimum proportional to the truncation parameter N_t and inversely proportional to the final time we are integrating to, t_1. This means that provided the ratio N_t/t_1 is sufficiently large the parabola will traverse the complex space avoiding the lower bound required by the condition (5.51). Hence the finite difference scheme is solvable provided one chooses parameters for the evaluation of the Bromwich integral that satisfy the above condition. An example of the above condition is illustrated in Figure 5.1 where the quarter-circle of radius 2 represents the lower bound required by the above condition.

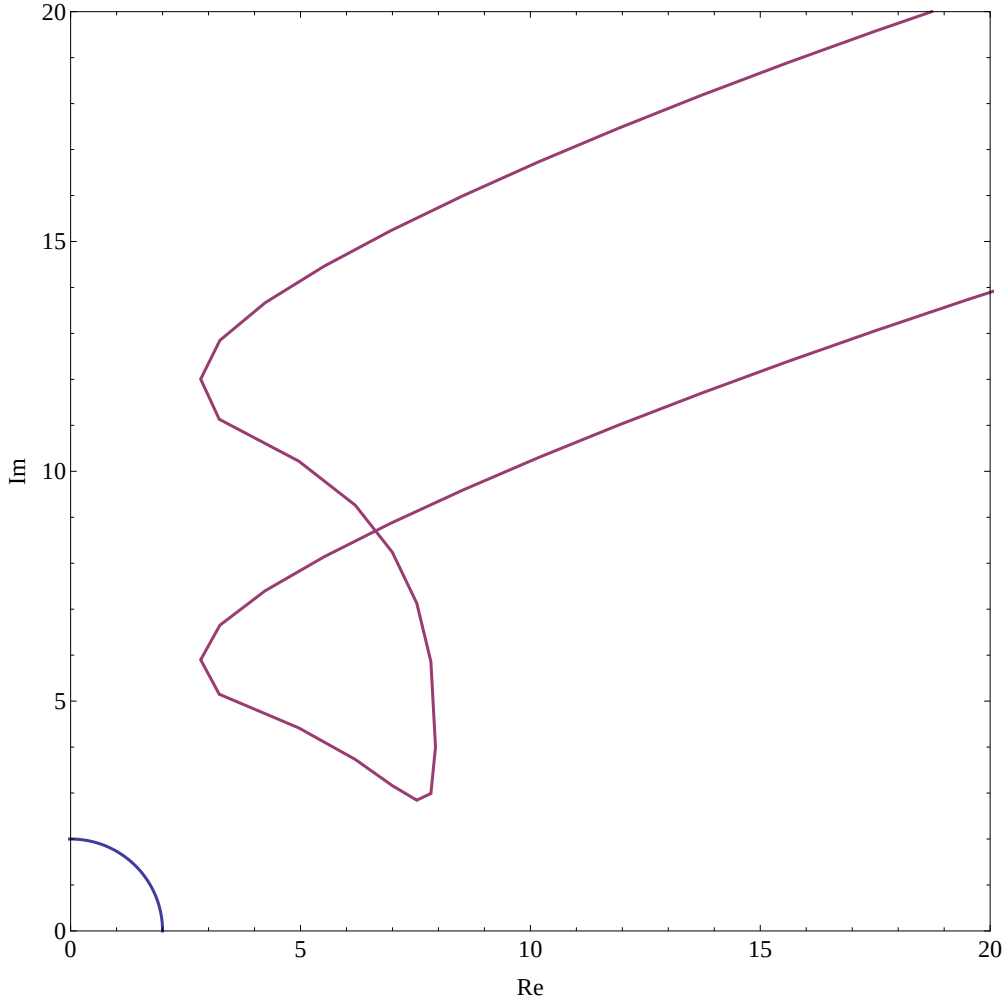


FIGURE 5.1: Integration Contour in the Real-Imaginary Plane Non-Intersecting with the lower bound for $\alpha = 0.75$.

5.4.1.2 Chebyshev Collocation

In the case of Chebyshev collocation the solvability criterion is a little more difficult to satisfy, and hence we are only able to derive a necessary criterion that our choice of inversion parameters must satisfy.

Proposition 1. If $\mathbf{M}$ is an irreducible diagonally dominant matrix for which $|m_{ii}| > \sum_{\substack{k=1 \\ k \neq i}}^{L} |m_{ik}|$ for at least one i, then $\mathbf{M}$ is invertible [112].

The proof to the above proposition may be found in [112]. A matrix is irreducible if it is no reducible. The definition of a reducible matrix is given below.

Definition 10. A square $n \times n$ matrix $\mathbf{M} = m_{ij}$ is called reducible if the indices $1, 2, ..., n$ can be divided into two disjoint nonempty sets $i_1, i_2, \ldots, i_\mu$ and $j_1, j_2, \ldots, j_\nu$ (with $\mu+\nu = n$) such that

$$m_{i_a j_b} = 0$$

for $a = 1, 2, ..., \mu$ and $b = 1, 2, ..., \nu$.

All that is left is to show that our matrix $\mathbf{M}$ is always irreducible and satisfies Proposition 1 for some i. We note that the differentiation matrices are constructed to be dense and hence, $\mathbf{M}$ is trivially irreducible. Once again the structure of Eq. (5.29) dictates the condition which is required for the matrix $\mathbf{M}$ to satisfy Proposition 1. This condition, given by

$$|m_{ii}| > \sum_{\substack{k=1 \\ k \neq i}}^{L} |m_{ik}|, \tag{5.52}$$

reduces to

$$|2d_{ii}^{(2)} - s^\alpha| > \sum_{\substack{k=1 \\ k \neq i}}^{L} 2|d_{ik}|. \tag{5.53}$$

The above condition is easy to verify in practice. However due to the dependence of the derivative matrix on the resolution parameters N_x and N_y it is difficult to write a general condition in closed form. Appropriate choices of parameters in the inversion resolution ensure the scheme is solvable as described in the previous section. Given the derivative matrix for Chebyshev collocation is full the right-hand side of the solvability condition (5.53) produces a lower bound with a much larger magnitude than in the finite difference case moreover the lower bound grows with N_x. Examples of contours are given for $N_x = 5$ in Figure 5.2 and for $N_x = 20$ in Figure 5.3 illustrating the adherence of a well constructed contour to the solvability condition (5.53).

5.4.2 Accuracy

5.4.3 Finite Difference Scheme

Theoretically the accuracy of the method is well-known to be $\mathcal{O}(\triangle x^2) + \mathcal{O}(\triangle y^2)$ [112]. The errors obtained in Section 5.5 concur with theoretical bounds.

5.4.4 Chebyshev Collocation

The work of Breur and Everson [113] and Don and Solomonoff [114] both present arguments on the accuracy of Chebyshev collocation. In practice the accuracy of the

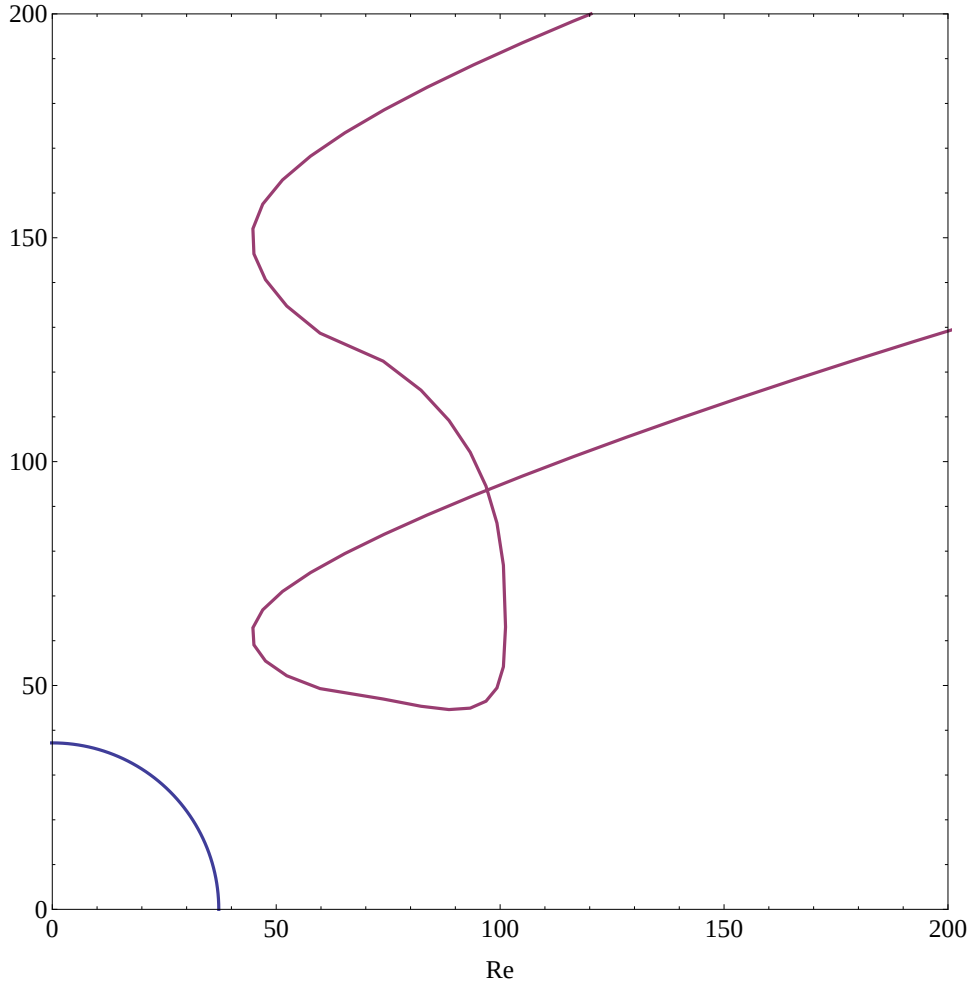


FIGURE 5.2: Integration Contour in the Real-Imaginary Plane Non-Intersecting with the lower bound for $\alpha = 0.75$ and $N_x = 5$.

method is measured by numerically differentiating a function and comparing the numerical derivative with the analytic result. The results obtained in the current work are consistent with the results presented in [113, 114] obtaining very good errors for small values of N_x and N_y. The errors in Chebyshev collocation tend to increase rather drastically for very large values of N_x contradicting the typical rule of thumb. The aforementioned work explains that while the truncation error decreases as resolution increases, the round-off errors accumulate dramatically and dominate.

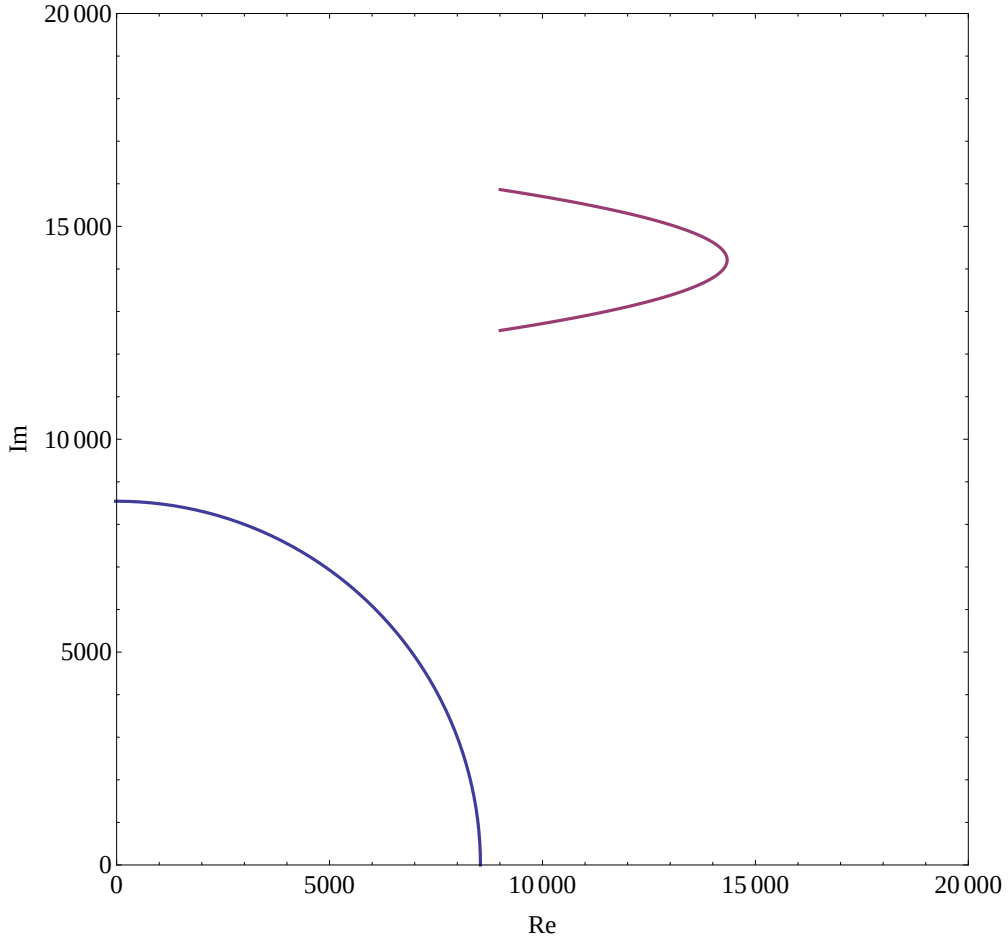


FIGURE 5.3: Integration Contour in the Real-Imaginary Plane Non-Intersecting with the lower bound for $\alpha = 0.75$ and $N_x = 20$.

5.5 Results

5.5.1 Example 1: One-Dimensional Time-Fractional Diffusion Equation with Homogenous Neumann Boundary Conditions

We consider first the time-fractional diffusion equation in one dimension

$$\frac{\partial^\alpha u}{\partial t^\alpha} = \frac{\partial^2 u}{\partial x^2}, \tag{5.54}$$

subject to the boundary conditions

$$u_x(-1,t) = 0, \qquad u_x(1,t) = 0, \tag{5.55}$$

and initial condition

$$u(x,0) = \cos\left(\frac{\pi}{2}(x+1)\right). \tag{5.56}$$

The results obtained by finite differences and Chebyshev collocation were compared to the exact solution given by Kazemi and Erjaeea [115] as

$$u(x,t) = \cos\left(\frac{\pi}{2}(x+1)\right)\left(1 - t^{\alpha} E_{\alpha,1+\alpha}(-t^{\alpha})\right), \tag{5.57}$$

where $E_{\alpha,\beta}(z)$ is the Generalized Mittag-Leffler function of the argument z. We select $\alpha = 0.8$ in line with [115], however experimental results indicate that the methods are robust for virtually any value of $0 < \alpha \leq 1$. We note here that the domain was originally $[0, \pi]$ and hence a linear transformation in the spatial variables is required to map the domain to $[-1, 1]$, which is in accordance with the domain of the Chebyshev polynomials. The errors in the finite difference and Chebyshev schemes are tabulated below in Table 5.1. Figures 5.4 and 5.5 illustrate the diminishing error incurred in the

N_x	**Error in Finite Difference**	**Error in Chebyshev Collocation**
5	0.009927020087821425	0.0009403962041890646
10	0.002471078174960284	$2.5800430680789077\times10^{-8}$
15	0.0010985435173019864	$9.547918011776346\times10^{-15}$
20	0.0006179865325516287	$8.992806499463768\times10^{-15}$

TABLE 5.1: Maximum absolute error in the presented method's solution of the problem described by Example 1 at time $t = 0.5$.

process of inverting the Laplace transform. Inversion of the Laplace transform, even in the analytic case, can lead to a singularity at $t = 0$. This arises in the numerical inversion and results in a relatively large error in the present schemes near $t = 0$. However this error diminishes rapidly and our schemes obtain an accurate solution.

5.5.2 Example 2: Diffusion Advection Equation with Dirichlet Boundary Conditions

This example considers the time-fractional diffusion-advection equation in one dimension

$$\frac{\partial^{\alpha} u}{\partial t^{\alpha}} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial u}{\partial x}, \tag{5.58}$$

subject to

$$u(-1,t) = e^{-1}, \qquad u(1,t) = e^{-1}. \tag{5.59}$$

and

$$u(x,0) = e^{-x^2}. \tag{5.60}$$

To the authors' knowledge no exact solution exists for the time-fractional diffusion-advection equation. Comparing the present methods with `NDSolve` in `Mathematica 9`

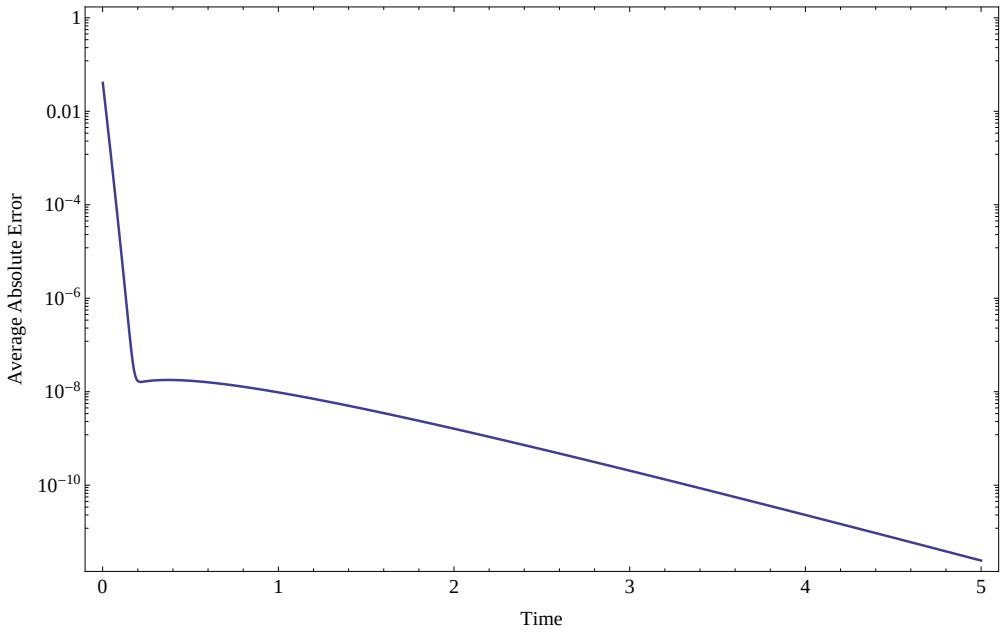


FIGURE 5.4: Log Plot Illustrating the Diminishing Average Error in the Chebyshev Collocation Scheme with Time for Example 1.

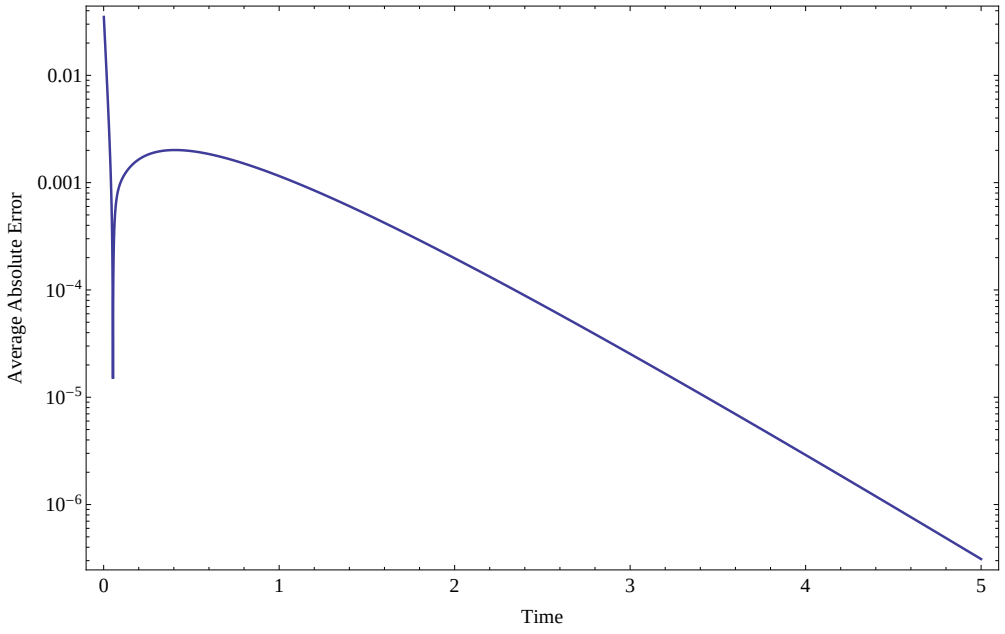


FIGURE 5.5: Log Plot Illustrating the Diminishing Average Error in the Finite Difference Scheme with Time for Example 1.

yields satisfactory results for $\alpha = 1$, but no solution can be found for fractional α. We instead compare our solutions in the Laplace domain, where we obtain an exact solution to the transformed equation using `DSolve` in `Mathematica 9`. This allows one to compare the performance of the present methods for various values for α. The errors obtained, for $\alpha = 0.7$, are presented in Table 5.2. Given that these errors are valid in the transform domain we note that the numerical error of $\mathcal{O}(8.12^{-N_t})$ is incurred upon

inversion of the Laplace transform, as presented by Weideman and Trefethen in [110] and where N_t is typically 50.

N_x	**Error in Finite Difference**	**Error in Chebyshev Collocation**
5	0.0002846591036554467	0.000031320115677323235
10	0.0003424383347704913	$5.981351275385904\times10^{-9}$
15	0.00028950142395197005	$5.763522471780025\times10^{-12}$
20	0.0002476113730374256	$5.538704044016907\times10^{-12}$

TABLE 5.2: Maximum absolute error in the presented method's solution of the problem described by Example 2 in the Laplace domain at $s = 50$.

5.5.3 Example 3: Two-Dimensional Time-Fractional Diffusion Equation with Homogenous Dirichlet Boundary Conditions

We now consider the two-dimensional time-fractional diffusion equation

$$\frac{\partial^\alpha u}{\partial t^\alpha} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}, \tag{5.61}$$

with boundary conditions

$$u(-1, y, t) = 0, \quad u(1, y, t) = 0, \tag{5.62}$$

$$u(x, -1, t) = 0, \quad u(x, 1, t) = 0, \tag{5.63}$$

and initial condition

$$u(x, y, 0) = \sin(\pi(x + 1))\sin(\pi(y + 1)), \tag{5.64}$$

so that the boundary conditions are consistent with the initial condition. The parameter α is taken to be 0.8. Momani [116] gives an exact solution as

$$u(x, y, t) = \sin(\pi(x + 1))\sin(\pi(y + 1))E_\alpha(-2t^\alpha) \tag{5.65}$$

where $E_\alpha(z) = E_{\alpha,1}(z)$ is the Mittag-Leffler function of order α. The efficacy of these methods for fractional order derivatives is illustrated in Table 5.3.

5.6 Discussion

The results above strongly advocate the use of Chebyshev collocation as a spatial discretization method given the rapid error reduction with increasing spatial resolution.

N_x	**Error in Finite Difference**	**Error in Chebyshev Collocation**
5	0.038120869340055985	0.0064580077147820825
10	0.009341572386976915	$8.103172060291985\times10^{-7}$
15	0.004521845750043552	$3.937461467984349\times10^{-12}$
20	0.002567981542938025	$6.517009154549669\times10^{-13}$

TABLE 5.3: Maximum absolute error in the presented method's solution of the problem described by Example 3 at time $t = 0.5$.

These hybrid methods present a robust way in which one can solve linear time-fractional partial differential equations on a bounded domain with Neumann or Dirichlet boundary conditions, particularly given discrete initial data.

Chebyshev collocation presents extremely small errors when compared to the exact solution. We use numerical experiments as substantiation of the method for applying discrete initial conditions where an exact solution may not exist.

The efficiency of the Laplace transform within the context of this hybridized method over a time-marching scheme is three-fold. First we are able to treat the fractional-order derivative algebraically. The error incurred in the temporal dimension is only attributed to the evaluation of the Bromwich integral and furthermore, this error drops off rapidly with increased resolution as illustrated in [110]. Finally the solution obtained is semi-analytic in the sense that it is a function of time so that we may evaluate our solution at any time rather than needing to march to that time. The Grünwald-Letnikov discretization presented in equation (5.1) is an example of a time-marching scheme. The computational time required for a long time solution via the Grünwald-Letnikov discretization is enormous, due to the fractionality being dependant on every time step that precedes the current time. Moreover every time step incurs a truncation error, so that the further the solution marches the greater the error, contrastingly the present method's error diminishes as time evolves. As a counter-point, if one were seeking a solution after a very short time, then a time-marching scheme may be better suited.

5.7 Conclusions

This research has presented a numerical experimental comparison between the standard finite difference method and the Chebyshev collocation method as a means of spatial discretization when hybridized with the Laplace transform. These methods enjoy the benefits of an exact transform in temporal variable and furthermore allow one to easily and efficiently deal with a fractional order derivative, at the cost of numerically inverting

the Laplace transform.

The goal of these methods is to apply a fractional order diffusion equation to an image on a bounded two-dimensional domain. The use of a discretization is therefore unavoidable given that the initial condition may in fact be discrete.

The solution to the discretized equations are found by writing a two-dimensional system of size $N_x \times N_x$ and $N_y \times N_y$ as a one-dimensional system of size $N_x N_y \times N_x N_y$. While this is more computational expensive it does exhibit an elegance in construction. An alternative approach would be to implement an Alternating-Direction Implicit (ADI) type scheme [37], where each dimension is acted on in turn rather than at once.

Due to the Laplace transform being a linear operator this method is not suitable for nonlinear problems, nor is it applicable to FPDEs with both fractional spatial derivatives and fractional temporal derivatives.

We have shown a hybrid method combing an analytic transform and a spatial discretization can be extremely effective at solving linear FPDEs on a two-dimensional bounded domain with Dirichlet or Neumann boundary conditions.

Chapter 6

Application of Hybrid Laplace Method to Nonlinear Time-Fractional Partial Differential Equations via Quasi-Linearization

"The life of a mathematician is dominated by an insatiable curiosity, a desire bordering on passion to solve the problems he is studying."

Jean Dieudonne

This work has been submitted under:

Jacobs, B. A., and C. Harley. "Application of Hybrid Laplace Method to Nonlinear Time-Fractional Partial Differential Equations via Quasi-Linearization." *Discrete and Continuous Dynamical Systems - Series B.*

This work considers a hybrid solution method for the time-fractional diffusion model with a cubic nonlinear source term in one and two dimensions. Both Dirichlet and Neumann boundary conditions are considered for each dimensional case. The hybrid method involves a Laplace transformation in the temporal domain which is numerically inverted and Chebyshev collocation is employed in the spatial domain due to its increased accuracy over a standard finite difference discretization. Due to the fractional-order

derivative we are only able to compare the accuracy of this method with `Mathematica`'s `NDSolve` where possible, however a detailed discussion of the merits and shortcomings of the proposed hybridization is presented. An application to image processing via a finite-difference discretization is included in order to substantiate the method.

6.1 Introduction

This work examines the performance of a hybrid Laplace Transform - Chebyshev collocation technique applied to the time-fractional diffusion equation in two-dimensions with a nonlinear source term:

$$u_{t^\alpha} = c_d \left(u_{xx} + u_{yy}\right) + c_s u(1-u)(u-a) \tag{6.1}$$

where

$$u_{t^\alpha} = \frac{\partial^\alpha u}{\partial t^\alpha}, \quad u_{xx} = \frac{\partial^2 u}{\partial x^2} \quad \text{and} \quad u_{yy} = \frac{\partial^2 u}{\partial y^2}. \tag{6.2}$$

This model is explored subject to both Dirichlet and Neumann boundary conditions on the bounded domain $\Omega = [-1,1] \times [-1,1]$ to satisfy the domain required by the Chebyshev polynomials with initial condition $u(x,y,0) = f(x,y)$. This method benefits from the analyticity of the Laplace transform and efficient numerical inversion of this transform, an accurate discretization approach through Chebyshev collocation and a convergent linearization technique, which results in a robust method for solving nonlinear time-fractional partial differential equations on a bounded domain. Following the method detailed by Jacobs and Harley in [111, 117] we also employ a finite-difference discretiaztion in applying this method to images. This is elaborated on further in Section 6.5.

Recently fractional derivatives and fractional partial differential equations (FPDEs) have received great attention both in analysis and application (see [11, 32, 105] and references therein). Despite this large effort, very little attention has been paid to solving FPDEs on a bounded domain through transformation techniques. Agrawal [106] makes use of the Laplace transform and the finite sine transform to obtain an analytic solution to the fractional diffusion-wave equation on a bounded domain. Other techniques such as the variational iteration method, Adomian decomposition method and differential transform method have all been applied to fractional partial differential equations but with a focus on unbounded domains.

The linear diffusion model has been applied in many different fields including image processing [56–61]. However, to the best of the authors' knowledge, the application

of a time-fractional partial differential equation has not yet been thoroughly examined within such a context. It is from the standpoint of applying a fractional partial differential equation to an image that leads us to examine the time-fractional diffusion equation in two-dimensions. In a recent paper Jacobs and Momoniat [88] show that the diffusion equation with this nonlinear source term is able to binarize a document image with great success. In extending this concept to fractional order derivatives, we seek to preserve the symmetry of the diffusion equation. Values of $\alpha \in (0, 1]$ resolve the diffusion equation to sub-diffusion, which preserves symmetry and exhibits some interesting dynamics which are discussed later on in this work. Values of $\alpha > 1$ introduce a transport effect which then breaks symmetry. It is for this reason that we impose the restriction, $\alpha \in (0, 1]$.

In this work we make use of the Laplace transform which allows us to handle the fractional-order derivative in an algebraic way, and incur no error in doing so. Inversion of the Laplace transform is however difficult to obtain analytically. We therefore make extensive use of the numerical inversion procedure described by Weideman and Trefethen in [110] which defines a contour of integration that maps the domain of the Bromwich integral from the entire complex space to the real space, from which we can approximate this integral with a trapezoidal rule. The authors present two contours, of which we select the hyperbolic contour due to its superior performance.

With a robust method for inverting the Laplace transform we may then hybridize the transform with a discretization technique. The Laplace transform of the temporal variable avoids the need for a time-marching scheme as well as reducing the fractional order derivative to an algebraic expression. The transformed model is then discretized by use of Chebyshev Collocation due to its superior performance to finite differences as illustrated in [111, 117]. The resulting system is solved and the transform inverted to obtain a semi-analytic solution, continuous in time and discrete in space.

In the following section we present some preliminary results which are put to use throughout the paper. In Section 6.3 the implemented methods are described, including the different cases for boundary conditions. Section 6.4 presents the solutions obtained by the proposed method as well as an error comparison with `Mathematica`'s `NDSolve` for the one and two dimensional cases of our model as well as both Dirichlet and Neumann boundary conditions. In Section 6.5 we provide a real-world application of this method and model in the form of document image binarization, illustrating the ability to obtain a useful result with the present method when coupled with the finite-difference discretization. A discussion of the results and their relationship to work beyond this research is presented in Section 6.6 and some concluding remarks are made in Section 6.7.

6.2 Preliminaries

In this work we employ Caputo's definition of a fractional derivative over the Riemann-Louiville derivative due to the fact that the Caputo derivative makes use of the physical boundary conditions, whereas the Riemann-Louiville derivative requires fractional order boundary conditions.

Definition 11. The Riemann-Louiville integral of order $\alpha > 0$ of a function $u(t)$ is

$$J^{\alpha}u(t) = \frac{1}{\Gamma(\alpha)}\int_0^t (t-\tau)^{\alpha-1}u(\tau)d\tau, \quad x > 0. \tag{6.3}$$

Definition 12. The fractional derivative of $u(t)$ according to the Caputo definition with $m-1 < \alpha \leq m$, $m \in \mathbb{N}$, is

$$\frac{\partial^{\alpha}u(t)}{\partial t^{\alpha}} = J^{m-\alpha}D^m u(t) = D_*^{\alpha}u(t) := \frac{1}{\Gamma(m-\alpha)}\int_0^t (t-\tau)^{m-\alpha-1}u^{(m)}(\tau)d\tau. \tag{6.4}$$

If $\alpha \in \mathbb{Z}$ the Caputo fractional derivative reduces to the ordinary derivative or integral. Podlubny [32] illustrates the pleasing property of the Laplace Transform of a Caputo derivative, as can be seen in equation (6.5). In our case where $0 < \alpha < 1$ we have,

$$\mathcal{L}\left\{\frac{\partial^{\alpha}u(x,y,t)}{\partial t^{\alpha}}\right\} = s^{\alpha}U(x,y) - s^{\alpha-1}u(x,y,0). \tag{6.5}$$

This property allows one to treat fractional order derivatives algebraically.

Definition 13. The Generalized Mittag-Leffler function of the argument z

$$E_{\alpha,\beta}(z) = \sum_{k=0}^{\infty}\frac{z^k}{\Gamma(k\alpha+\beta)}. \tag{6.6}$$

6.3 Methods

This section introduces the methodologies used for the two-dimensional model previously presented. We may write our model as

$$c_d u_{xx} + c_d u_{yy} + c_s(a+1)u^2 - c_s a u - c_s u^3 - u_{t^{\alpha}} = \Phi(u_{xx}, u, u_{t^{\alpha}}) = 0. \tag{6.7}$$

The quasi-linearization technique can be viewed as a generalized Newton-Raphson method in functional space. An iterative scheme is constructed creating a sequence of linear equations that approximate the nonlinear equation (6.7) and boundary conditions. Furthermore, this sequence of solutions converges quadratically and monotonically [76–78].

The quasi-linear form is given by

$$c_d u_{xx}^{n+1} + c_d u_{yy}^{n+1} + X_1^n u^{n+1} + X_2^n u_{t^\alpha}^{n+1} = X_3^n, \tag{6.8}$$

where n indicates the index of successive approximation. Moreover, $u(x,y,t)^n$ is known entirely at the explicit index n. The coefficients are given by

$$X_1^n = \frac{\partial \Phi}{\partial u} = c_s 2(a+1)u^n - c_s a - 3c_s(u^n)^2, \tag{6.9}$$

and

$$X_2^n = \frac{\partial \Phi}{\partial u_{t^\alpha}} = -1. \tag{6.10}$$

If $\underline{u} = (u_{xx}, u_{yy}, u, u_{t^\alpha})$ indexed by j then

$$\begin{aligned} X_3^n &= \sum_j \underline{u}_j^n \frac{\partial \Phi}{\partial \underline{u}_j}, \\ &= \left[u_{xx}\frac{\partial \Phi}{\partial u_{xx}} + u_{yy}\frac{\partial \Phi}{\partial u_{yy}} + u\frac{\partial \Phi}{\partial u} + u_{t^\alpha}\frac{\partial \Phi}{\partial u_{t^\alpha}} \right]^n, \\ &= \left[c_d u_{xx} + c_d u_{yy} + 2(a+1)u^2 - au - 3u^3 - (u_{t^\alpha}) \right]^n, \\ &= \left[c_d u_{xx} + c_d u_{yy} + (a+1)u^2 - au - u^3 - (u_{t^\alpha}) \right]^n + \left[(a+1)u^2 - 2u^3 \right]^n. \end{aligned} \tag{6.11}$$

Since the first term above satisfies the equation (6.7) it is replaced with 0 giving

$$X_3^n = \left[(a+1)u^2 - 2u^3 \right]^n. \tag{6.12}$$

Equation (6.8) can now be transformed by the Laplace Transform, a linear operator, to obtain

$$c_d U_{xx}^{n+1} + c_d U_{yy}^{n+1} + X_1^n U^{n+1} - s^\alpha U^{n+1} = \frac{X_3^n}{s} - s^{\alpha-1} f(x,y), \tag{6.13}$$

where

$$U(x,y,s) = \mathcal{L}\left\{ u(x,y,t) \right\}. \tag{6.14}$$

Equation (6.13) may be discretized by Chebyshev Collocation, described in the following section.

6.3.1 Chebyshev Collocation

Chebyshev polynomials form a basis on $[-1,1]$ and hence we dictate the domain of our PDE to be $\Omega = [-1,1]^n$ where n indicates the number of spatial dimensions, which in this case is $n = 2$. We note here however, that any domain in $\mathbb{R}^2$ can be trivially deformed to match Ω. We discretize our spatial domain using Chebyshev-Gauss-Labatto

points,

$$x_i = \cos\left(\frac{i\pi}{N_x}\right), \quad i = 0, 1, \ldots, N_x, \qquad y_j = \cos\left(\frac{j\pi}{N_y}\right), \quad j = 0, 1, \ldots, N_y. \tag{6.15}$$

Given this choice of spatial discretization we have $x_0 = 1$, $x_{N_x} = -1$, $y_0 = 1$ and $y_{N_y} = -1$ indicating that the domain is in essence reversed and one must exercise caution when imposing the boundary conditions.

In mapping our domain to Ω we may assume that $N_x = N_y$, i.e. we have equal number of collocation points in each spatial direction. We now define a differentiation matrix $D^{(1)} = d_{kl}$,

$$d_{kl} = \begin{cases} \frac{c_k(-1)^{k+l}}{c_l(x_k - x_l)}, & k \neq l, \\ -\frac{x_k}{2(1-x^2{}_k)}, & k = l, \\ \frac{1}{6}\left(2N_x^2 + 1\right), & k = l = 0, \\ -\frac{1}{6}\left(2N_x^2 + 1\right), & k = l = N_x, \end{cases} \quad \text{where} \quad c_k = \begin{cases} 2, & k = 0, N_x, \\ 1, & k = 1, \ldots, N_x - 1. \end{cases} \tag{6.16}$$

Bayliss et al. [75] describe a method for minimizing the round off errors incurred in the calculations of higher order differentiation matrices. Since we write $D^{(2)} = D^{(1)}.D^{(1)}$ we implement the method, described in [75], in order to minimize propagation of round off errors for the second derivative in space.

The derivative matrices in the x direction are

$$\hat{D}_x^{(1)} = D^{(1)}, \qquad \hat{D}_x^{(2)} = D^{(2)}, \tag{6.17}$$

where $D^{(1)}$ is the Chebyshev differentiation matrix of size $(N_x + 1) \times (N_y + 1)$. Because we have assumed $N_x = N_y$ we derive the pleasing property that $\hat{D}_y^{(1)} = \left(\hat{D}_x^{(1)}\right)^T$ and $\hat{D}_y^{(2)} = \left(\hat{D}_x^{(2)}\right)^T$.

Writing the discretization of equation (6.13) in matrix form yields,

$$\hat{D}_x^{(2)}\mathbf{U}^{n+1} + \mathbf{U}^{n+1}\hat{D}_y^{(2)} + X_1^n\mathbf{U}^{n+1} - s^\alpha\mathbf{U}^{n+1} = \frac{X_3^n}{s} - \mathbf{F}, \tag{6.18}$$

where

$$F_{ij} = s^{\alpha-1}f(x_i, y_j). \tag{6.19}$$

By expanding equation (6.18) in summation notation we have

$$\sum_{k=0}^{N_x} d_{ik}^{(2)} U^{n+1}(x_k, y_j) + \sum_{k=0}^{N_y} d_{kj}^{(2)} U^{n+1}(x_i, y_k) + [X_1]_{i,j}^n U^{n+1}(x_i, y_j) - s^\alpha U^{n+1}(x_i, y_j) = \frac{[X_3]_{i,j}^n}{s} - s^{\alpha-1} f(x_i, y_j), \tag{6.20}$$

for $i = 0, 1, \ldots, N_x, j = 0, 1, \ldots, N_y$. By extracting the first and last terms in sums we obtain

$$d_{i0}^{(2)} U^{n+1}(x_0, y_j) + d_{iN_x}^{(2)} U^{n+1}(x_{N_x}, y_j) + d_{0j}^{(2)} U^{n+1}(x_i, y_0) + d_{N_y j}^{(2)} U^{n+1}(x_i, y_{N_y}) + \sum_{k=1}^{N_x-1} d_{ik}^{(2)} U^{n+1}(x_k, y_j) + \sum_{k=1}^{N_y-1} d_{kj}^{(2)} U^{n+1}(x_i, y_k) + [X_1]_{i,j}^n U^{n+1}(x_i, y_j) - s^\alpha U^{n+1}(x_i, y_j) = \frac{[X_3]_{i,j}^n}{s} - s^{\alpha-1} f(x_i, y_j), \tag{6.21}$$

for $i = 1, \ldots, N_x - 1, j = 1, \ldots, N_y - 1$. We use the form of equation (6.21) to impose the boundary conditions.

The solution $\tilde{\mathbf{U}} = \{U(x_1, y_1), U(x_1, y_2), \ldots, U(x_{N_x-1}, y_{N_y-1})\}$, which is the matrix of unknown interior points of $\mathbf{U}$, can be found by solving the system

$$(\tilde{\mathbf{A}} - \mathbf{s}^\alpha \mathbf{I})\tilde{\mathbf{U}} + \mathbf{X_1^n} \circ \tilde{\mathbf{U}} + \tilde{\mathbf{U}}\tilde{\mathbf{B}} = \frac{\mathbf{X_3^n}}{s} - \tilde{\mathbf{F}}, \tag{6.22}$$

where $\tilde{\mathbf{A}}$ is the matrix of interior points of $\hat{\mathbf{D}}_x^{(2)}$, and $\tilde{\mathbf{B}}$ is the matrix of interior points of $\hat{\mathbf{D}}_y^{(2)}$, so that $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{B}}$ match the dimensions of $\tilde{\mathbf{U}}$. We also use $\circ$ to denote the Hadamard product between two matrices. Also

$$\tilde{F}_{ij} = s^{\alpha-1} f(x_i, y_j) + d_{i0}^{(2)} U(x_0, y_j) + d_{iN_x}^{(2)} U(x_{N_x}, y_j) + d_{0j}^{(2)} U(x_i, y_0) + d_{N_y j}^{(2)} U(x_i, y_{N_y}), \tag{6.23}$$

for $i = 1, \ldots, N_x - 1, j = 1, \ldots, N_y - 1$.

6.3.1.1 Dirichlet Boundary Conditions

Boundary conditions may be in the form of Dirichlet conditions,

$$u(-1, y, t) = a, \quad u(1, y, t) = b, \tag{6.24}$$

$$u(x, -1, t) = c, \quad u(x, 1, t) = d, \tag{6.25}$$

and hence,

$$U(-1,y) = \mathcal{L}\{a\}, \quad U(1,y) = \mathcal{L}\{b\}, \tag{6.26}$$

$$U(x,-1) = \mathcal{L}\{c\}, \quad U(x,1) = \mathcal{L}\{d\}. \tag{6.27}$$

The parameters a, b, c and d are potentially functions of the temporal variable and one of the spatial variables, i.e. $a = a(y,t)$. We assume that a, b, c and d are constant.

Dirichlet boundary conditions can be imposed directly by substituting equations (6.24) and (6.25) into equation (6.21) and collecting all the known terms in $\tilde{\mathbf{F}}$.

6.3.1.2 Neumann Boundary Conditions

Alternatively Neumann boundary conditions give,

$$u_x(-1,y,t) = a, \quad u_x(1,y,t) = b, \tag{6.28}$$

$$u_y(x,-1,t) = c, \quad u_y(x,1,t) = d, \tag{6.29}$$

with,

$$U_x(-1,y) = \mathcal{L}\{a\}, \quad U_y(1,y) = \mathcal{L}\{b\}, \tag{6.30}$$

$$U_y(x,-1) = \mathcal{L}\{c\}, \quad U_y(x,1) = \mathcal{L}\{d\}. \tag{6.31}$$

Neumann boundary conditions given by equations (6.26) are discretized as

$$\frac{\partial U}{\partial x}(x_{N_x} = -1, y_j) \approx \sum_{k=0}^{N_x} d_{N_x k} U^n(x_k, y_j) = \mathcal{L}\{a\}, \tag{6.32}$$

$$\frac{\partial U}{\partial x}(x_0 = 1, y_j) \approx \sum_{k=0}^{N_x} d_{0k} U^n(x_k, y_j) = \mathcal{L}\{b\}. \tag{6.33}$$

Similarly for equations (6.27)

$$\frac{\partial U}{\partial y}(x_i, y_{N_y} = -1) \approx \sum_{k=0}^{N_y} d_{k N_y} U^n(x_i, y_k) = \mathcal{L}\{c\}, \tag{6.34}$$

$$\frac{\partial U}{\partial y}(x_i, y_0 = 1) \approx \sum_{k=0}^{N_y} d_{k0} U^n(x_i, y_k) = \mathcal{L}\{d\}. \tag{6.35}$$

By extracting the first and last terms in the sum, the discretizations can be written as

$$\begin{pmatrix} d_{N_x 0} & d_{N_x N_x} \\ d_{00} & d_{0N_x} \end{pmatrix} \begin{pmatrix} U^n(x_0, y_j) \\ U^n(x_{N_x}, y_j) \end{pmatrix} = \begin{pmatrix} \mathcal{L}\{b\} - \sum_{k=1}^{N_x-1} d_{0k} U^n(x_k, y_j) \\ \mathcal{L}\{a\} - \sum_{k=1}^{N_x-1} d_{N_x k} U^n(x_k, y_j) \end{pmatrix} \tag{6.36}$$

and

$$\begin{pmatrix} d_{0N_y} & d_{N_y N_y} \\ d_{00} & d_{N_y 0} \end{pmatrix} \begin{pmatrix} U^n(x_i, y_0) \\ U^n(x_i, y_{N_y}) \end{pmatrix} = \begin{pmatrix} \mathcal{L}\{d\} - \sum_{k=1}^{N_y-1} d_{k0} U^n(x_i, y_k) \\ \mathcal{L}\{c\} - \sum_{k=1}^{N_y-1} d_{kN_y} U^n(x_i, y_k) \end{pmatrix} \tag{6.37}$$

The solutions to these linear systems are then substituted into equation (6.21).

6.4 Results

In this section we consider only the results obtained by Chebyshev collocation due to the enormous increase in accuracy obtained over the finite-difference method that was presented by the authors in [111, 117].

6.4.1 Example 1

The first example we consider

$$u(x, 0) = e^{-x^2}, \tag{6.38}$$

with Dirichlet boundary conditions consistent with this initial condition,

$$u(-1, t) = u(1, t) = e^{-1}. \tag{6.39}$$

Table 6.1 below illustrates the maximum absolute error between the solution obtained by the present method and the solution obtained by `NDSolve` in `Mathematica 9` for $\alpha = 1$. To the best of the authors' knowledge no exact solution exists for the fractional case, hence no comparison can be made. However, we present Figure 6.1 which illustrates the

N_x	**Error vs** `NDSolve`
5	0.029145584018683945
10	0.00015607552473428932
15	0.00014992336248420557
20	0.00014949081096204964
25	0.00014948899958276735
30	0.00014948859870700382

TABLE 6.1: Maximum absolute error in the presented method's solution of the problem described by Example 1 at time $t = 0.5$.

behaviour of the mid-point of the solution as time evolves for various values of α.

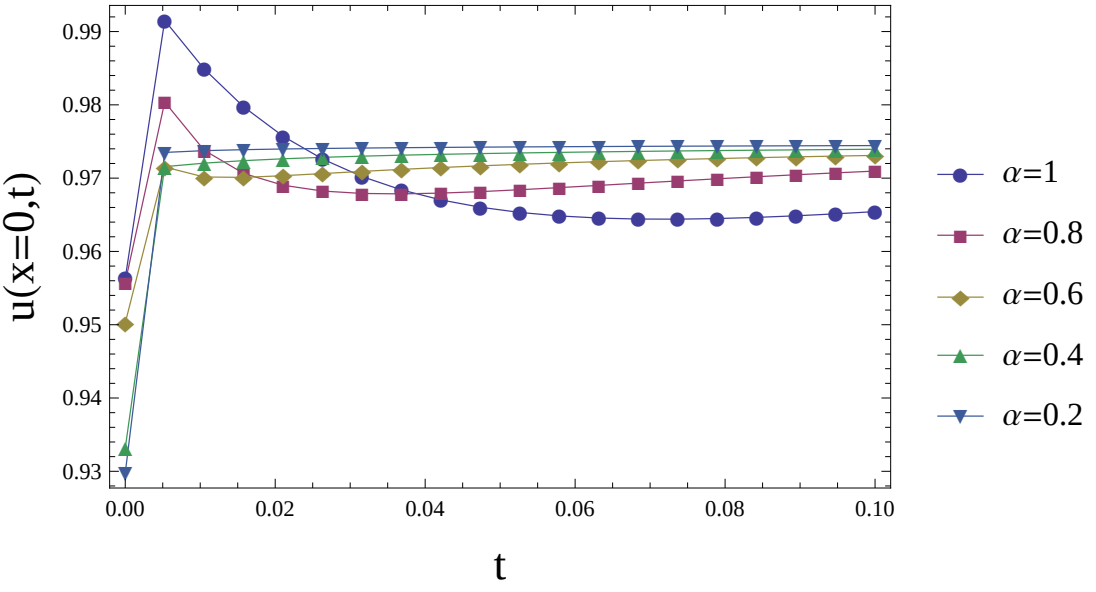


FIGURE 6.1: Plot of $u(x = 0, t)$ for various values of α.

6.4.2 Example 2

We now consider the one-dimensional case with initial condition

$$u(x, 0) = e^{-(x+1)} \tag{6.40}$$

and Neumann Boundary conditions

$$u_x(-1, t) = 1 \tag{6.41}$$

$$u_x(1, t) = \frac{-1}{e^2} \tag{6.42}$$

Once again our solution converges with N moving from 5 to 10 but does not continue to improve. Table 6.2 presents the errors in our method when compared to `NDSolve` for $\alpha = 1$. We present the behaviour of the mid-point $u(x = 0, t)$ as time evolves for different values of α in Figure 6.2.

6.4.3 Example 3

This example considers the two-dimensional case with initial condition

$$u(x, y, 0) = (x - 1)(x + 1)(y - 1)(y + 1) \tag{6.43}$$

N_x	**Error vs** `NDSolve`
5	0.18888552094993993
10	0.06402298769357995
15	0.06716933046894086
20	0.06886100702660541
25	0.0685232103205754
30	0.06782287252683938

TABLE 6.2: Maximum absolute error in the presented method's solution of the problem described by Example 2 at time $t = 0.5$.

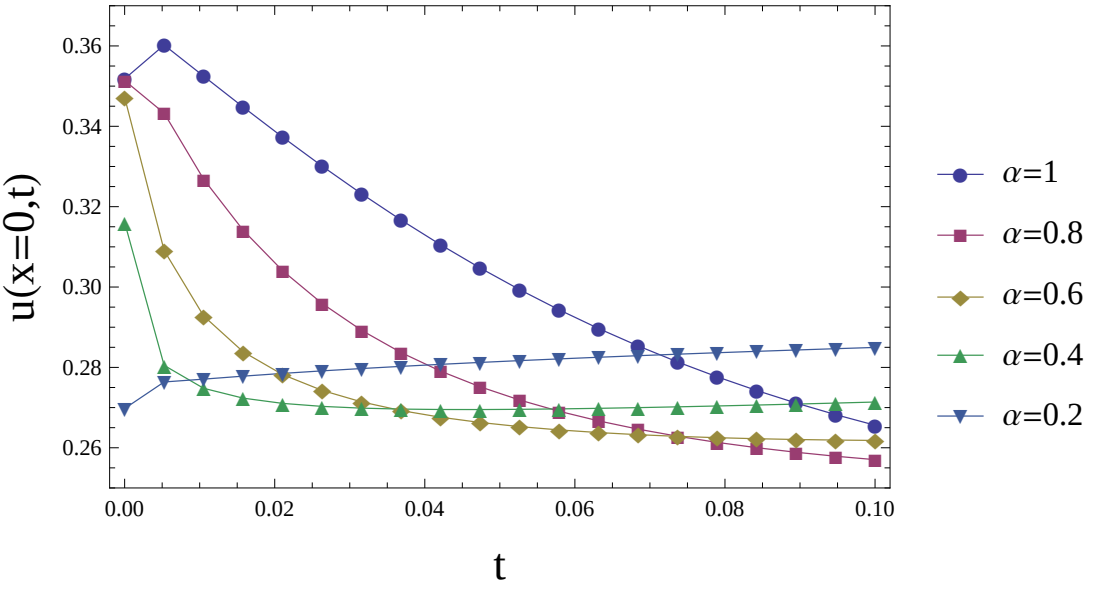


FIGURE 6.2: Plot of $u(x = 0, t)$ for various values of α.

and consistent Dirichlet boundary conditions described as

$$u(-1, y, t) = u(1, y, t) = u(x, -1, t) = u(x, 1, t) = 0. \tag{6.44}$$

We present here the errors in the present method when compared to `NDSolve`. Figure 6.3 describes the evolution of the mid-point in two-dimensions with time for various α values.

6.4.4 Example 4

Finally we consider the case of a two-dimensional initial condition with Neumann conditions,

$$u(x, y, 0) = \cos(x)\cos(y) \tag{6.45}$$

$N_x = N_y$	**Error vs** `NDSolve`
5	0.000050581035499991776
10	0.000064787185355778 35
15	0.000063102490737803 43
20	0.000064787396424136 41
25	0.00006424152346392 26
30	0.000064787396398447 85

TABLE 6.3: Maximum absolute error in the presented method's solution of the problem described by Example 3 at time $t = 0.5$.

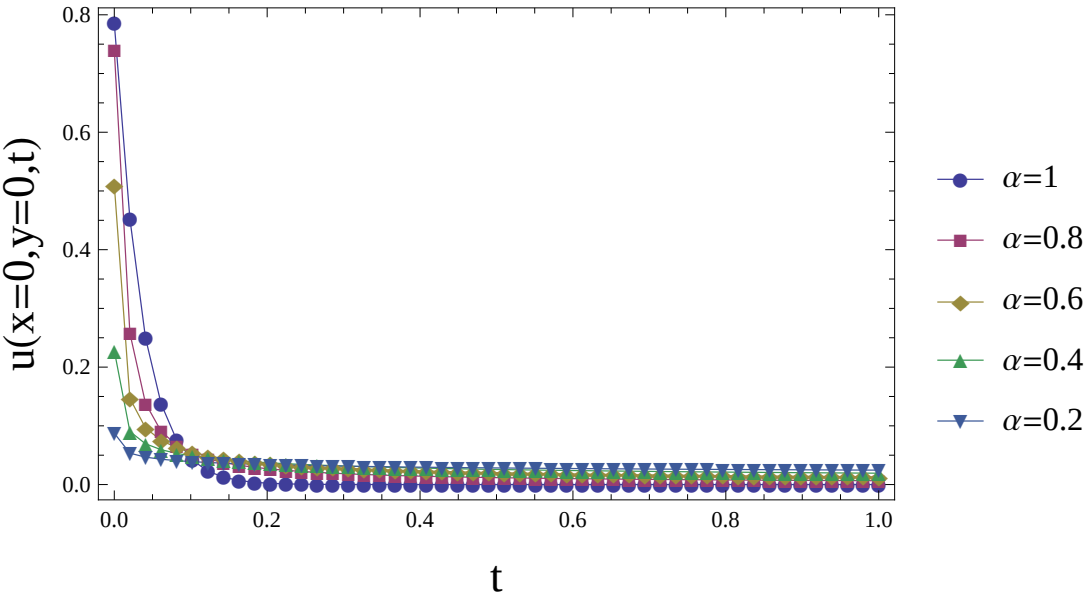


FIGURE 6.3: Plot of $u(x = 0, y = 0, t)$ for various values of α.

and

$$u_x(-1, y, t) = \sin(1)\cos(y) \tag{6.46}$$

$$u_x(1, y, t) = -\cos(1)\cos(y) \tag{6.47}$$

$$u_y(x, -1, t) = \cos(x)\sin(1) \tag{6.48}$$

$$u_y(x, 1, t) = -\cos(x)\sin(1). \tag{6.49}$$

We present here the errors in the present method when compared to `NDSolve`. Figure 6.4 describes the evolution of the mid-point in two-dimensions with time for various α values.

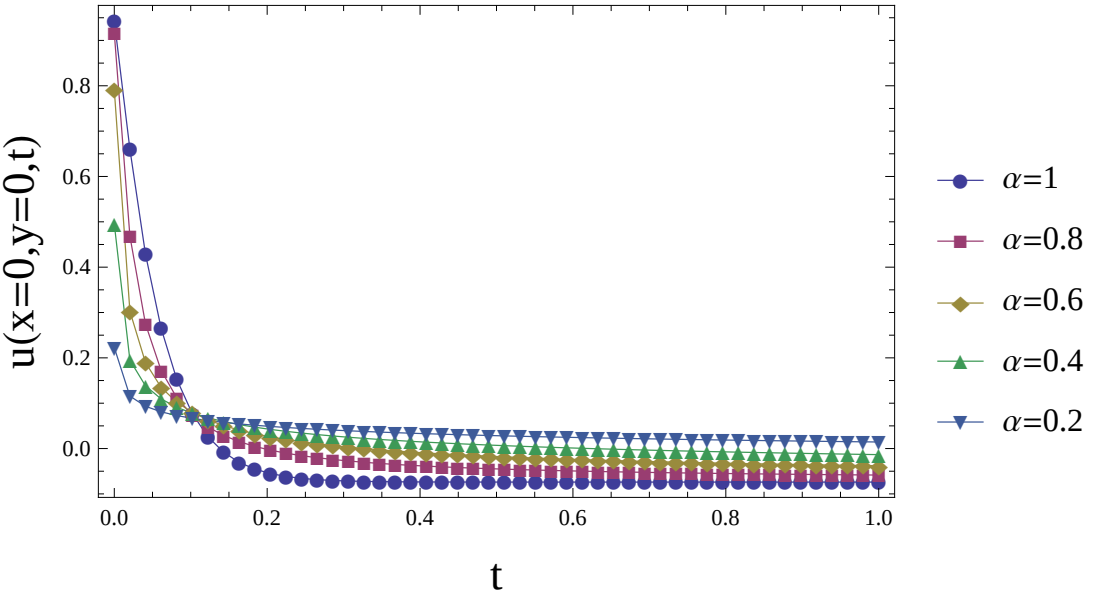


FIGURE 6.4: Plot of $u(x = 0, y = 0, t)$ for various values of α.

$N_x = N_y$	**Error vs** `NDSolve`
5	0.00875011619006949
10	0.002380333748528196
15	0.002658963151695559
20	0.0027658847054590208
25	0.002819390822498269
30	0.0028497574705675377

TABLE 6.4: Maximum absolute error in the presented method's solution of the problem described by Example 4 at time $t = 1$.

6.5 Image Processing Application

Despite the impressive accuracy obtained by Chebyshev collocation for small values of N_x the method suffers from severe round-off error for values of $N_x > 100$ due to finite-precision arithmetic [113, 114]. In applying this scheme to an image, the dimensions of the input image dictate the resolution of the scheme and in most cases the input image dimensions will exceed 100. This forces us to employ a finite difference discretization scheme instead of the Chebyshev scheme as described by Jacobs and Harley in [117]. Jacobs and Momoniat [88] present the results of applying the integer order model to an image for the purpose of document image binarization. We present here some examples of the results obtained using the hybrid-transform method. The power of the hybrid-transform method lies in the ability to immediately determine the solution at any point in time as opposed to needing to iterate to a given point in time. Figure 6.5 illustrates the solutions obtained, given an input image, at varying points in time. Figure

6.6 shows the effects of the fractional term on the resulting image, wherein the steady state has been altered as well as the transient phase being executed much more quickly. This mirrors the effects illustrated in the examples above, Figures 6.1, 6.2, 6.3 and 6.4, where the steady state of the equation is altered with changing values of α as well as the equation reaching a steady state more rapidly. This suggests the order of the derivative may be used as an extra dimension of control in image processing, where a smaller α value actually brightens the background of the image, it also reduces diffusivity of the model by altering the steady state away from a uniformly diffused image.

If we take for example $N_x = N_y = 100$ then the stability criteria for a standard time-marching scheme would be $\triangle t = (1/100)^{2/\alpha}$. As α decreases from 1, $\triangle t$ becomes so small that the number of iterations required to attain a final time of $t = 0.003$ is unfeasibly large. This further justifies the use of a hybrid transform method in the application of image processing.

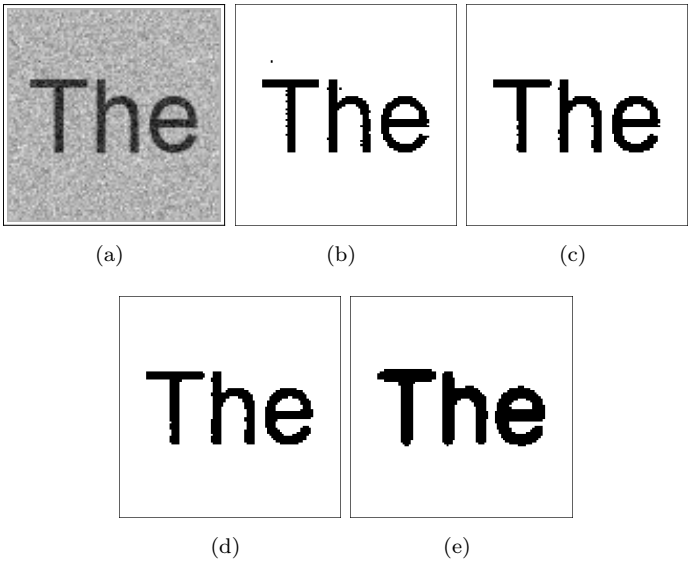
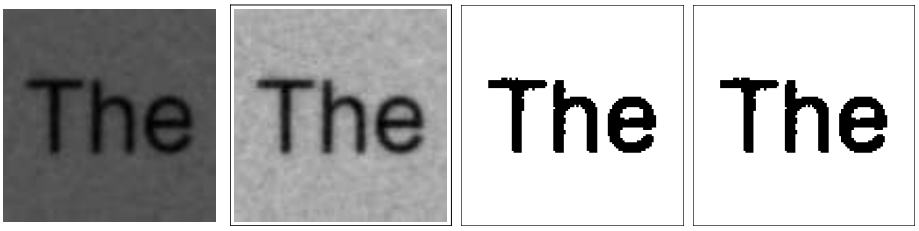


FIGURE 6.5: Figures depicting results obtained at increasing time using the current method and $\alpha = 1$. (a) The original image, (b) Binarized original image at $t \approx 0$, (c) Processed binary image at $t = 0.003$, (d) Processed binary image at $t = 0.008$, (e) Over diffused image at $t = 0.01$.

6.6 Discussion

The results above indicate a strong convergence to a solution with increasing N. The similar results obtained for $N > 5$ are attributed to the comparison being drawn between two numerical methods which may be different from the true exact solution. Results



(a) (b) (c) (d)

FIGURE 6.6: Figures depicting results obtained at increasing time using the current method and $\alpha = 0.2$. (a) The original image, (b) Processed image at $t = 0.01$, (c) Processed binary image at $t = 0.001$, (d) Processed binary image at $t = 0.01$

obtained in [111] indicate that when the present method is compared with an exact solution the accuracy increases dramatically with increasing N. Despite this, the present method attains results similar to that of an industry standard, `NDSolve`, which is encouraging.

Figures 6.1, 6.2, 6.3 and 6.4 indicate the dynamic behaviour of the sub-diffusive process. Interestingly the diffusive process becomes increasingly more aggressive as α decreases from 1 to 0, however the final 'steady-state' achieved by these processes is typically less severe than the standard diffusion model. In image processing terms we could achieve a semi-diffused result extremely quickly by choosing α to be small, rather than using $\alpha = 1$ and diffusing over a longer period.

In both the one and two-dimensional cases, the Neumann boundary conditioned problems result in an accuracy two orders of magnitude worse than that of the Dirichlet cases. In the linear case where an exact solution exists the accuracy of the present method increases dramatically with increasing N [117], however due to the comparison being drawn with another numerical method, `NDSolve`, the free ends affect the comparative solution, rather than reinforcing a condition on the boundary as in the Dirichlet case.

6.7 Conclusions

In addition to obtaining a high accuracy the present method is robust for $0 < \alpha \leq 1$ and is in fact exact in the transformation from a time-fractional partial differential equation to partial differential equation or differential equation depending on the dimensionality. The errors incurred are then discretization error and numerical inversion of the Bromwich integral which is $\mathcal{O}(10.2^{-N})$ [110].

The application of this method to images introduces a computational hurdle since the resolution of the image dictates the spatial discretization resolution. After the transformation one must solve a linear system that is of the same dimension as the input image. This is exacerbated by the use of Chebyshev collocation, since the derivative matrices are full and not tri-diagonal or banded as they are when using a standard finite-difference scheme, where the Thomas algorithm can be put to use as in [76–78]. In such case a numerical method should be implemented to improve the computational time required for large input data. The Chebyshev collocation method is also known to have large round off errors for large problems, $N_x > 100$, due to finite-precision [113, 114]. Alternatively, for large systems a finite difference discretization may be used to speed up computational time at the cost of accuracy in the solution. We have shown for example the images obtained by employing the hybrid-transform method to illustrate that the method does produce the expected results in a real-world application.

We have presented a method that is robust for a time-fractional diffusion equation with a nonlinear source term for one and two-dimensional cases with both Dirichlet and Neumann boundary conditions. We have shown through numerical experiments that in the case of $\alpha = 1$ our solution obtains a result similar to that of `NDSolve` in `Mathematica 9` and extends trivially to fractional-order temporal derivative of order $\alpha \in (0, 1]$. The application to image processing substantiates this method for real world problems, highlighting the effectiveness of the fractional ordered derivative as a further dimension of control which has a dramatic effect both on the transition time of the model as well as the final state attained. Within the realm of image processing this opens new avenues of research allowing more control over physically derived processes to construct methods that are physically substantial.

Although this work has just begun to uncover the efficacy of applying FPDEs to images the depth of understanding that belies the field of fractional order problems can add enormously to the field of image processing and other data driven sciences. Moreover the numerical analysis and computational hurdles associated with this deep extension of PDEs also opens new avenues of research. Due to the subjectivity of images and image processing there is a certain difficulty associated with developing and applying new techniques within this framework. However this research goes a little way toward developing effective tools making use of image processing as an illustrative platform.
To the best of the authors' knowledge this hybridization of the Laplace transform, Chebyshev Collocation and quasi-linearization scheme has not yet been applied to FPDEs in one or two dimensions. This collaboration yields a method that is accurate and robust for time-fractional derivatives.

Chapter 7

Conclusion

"Mathematics is music for the mind; music is mathematics for the soul."

Anonymous

This work explored the application of partial differential equations and fractional partial differential equations to images and investigated effective methods of solving and applying these.

7.1 Document Image Binarization

In Chapters 3 and 4 we introduced a novel approach to document image binarization through a simple isotropic process with excellent results. The performance of our method was compared with the state-of-the-art global and adaptive methods. A thorough survey of the field was performed and a comparison of the existing binarization processes is given in [1–4]. Using the results from these reviews we selected the best performing methods for comparison. The isotropic nature of the scheme lends itself to being parallelizable and a massively parallel implementation was described in Chapter 4. Furthermore we were able to localize the method to further generalize the types of input images that were permissible resulting in a method that is extremely robust to noise and non-uniform illumination and produces results in a computationally efficient manner.

Our method also entertains some novelty in being based on a partial differential equation as opposed to a set of algorithmic rules. This helps us to substantiate the efficacy of our method given the physicality of the model. We are also able to draw from a vast literature on analysis to provide strict bounds on performance and stability.

In order to measure the efficacy of our method we are able to employ a number of industry standard performance metrics. Perhaps most appropriate of these measures is the l_1 norm which when comparing binary images calculates the number of pixels that differ between two images. In addition to this we employ the F-Measure, pseudo F-Measure, the Distance Reciprocal Distortion Metric and finally the Peak signal-to-noise ratio, all of which indicate a good score of our method. These methods were introduced and used as standard measures in the document image binarization competions (DIBCO) from 2009 to 2012 [96–99].

Although our method is not the best method in every situation and for all images, an unreasonable request, it is competitive at the highest level. Moreover the novelty of this method is perhaps the most intriguing aspect as well as the sheer simplicity, both in its conception as well as its execution. It is because of these qualities that we chose to generalize the model to fractional order as a means of introducing additional control over the process and striving toward an even more effective method.

7.2 Fractional Order Partial Differential Equations

Fractional partial differential equations present some computational difficulties. In broad terms if we consider, for example, the sub-diffusive process that is said to express some diffusive memory and is interpreted through the Grünwald-Letnikov [11, 32] discretization as an infinite sum of previous time steps. In essence what this is saying is: the evolution of this equation is not only dependent on the current state of the media but also on *every* state that has previously occurred. The "short-memory" principle that Podlubny [32] discusses indicates that states that are sufficiently long ago have no bearing on what may occur next. Computationally this is conducted by storing many more states than just the current state and hence as time evolves computation becomes more and more expensive.

It is this that supports the transformation of the temporal variable. The Laplace transform allows one to treat the fractional derivative algebraically and only in the inversion of this transform are temporal bounds imposed. We therefore implicitly take into account all states in time. The result of the hybridization of a transform method and a spatial discretization is the reduction of a multi-dimensional fractional partial differential equation to the solution of a system of algebraic equations, linear or nonlinear depending on

the original equation. The resulting solution is a time-valued discrete function where the solution may be evaluated at any point in time directly, without the need to march to this time. If the final time required is extremely small say $t = 0.001$ then with a scheme resolution of 100 the stability requirement of $\triangle t \leq 0.25\triangle x^2$ requires $\triangle t = 0.000025$ requiring a mere 40 steps in time to achieve $t = 0.001$. However in the fractional case this stability requirement behaves as per $\triangle t \leq \left(0.25\triangle x^2\right)^{1/\alpha}$. With $\alpha = 0.2$ the number of steps required to reach $t = 0.001$ is 1.024×10^{24}, substantially more than 40, making the usefulness of a transform method apparent.

The Laplace transform was used in conjunction with both a standard finite difference discretization as well as a Chebyshev collocation method as a means of discretizing the spatial variables in one and two dimensions. The implementation of these methods was done under both Dirichlet and Neumann boundary conditions on a finite domain, requiring subtle but fundamental changes in the approach. Chapter 5 showed how extremely accurate solutions may be obtained through the use of the Chebyshev-Laplace hybrid technique. However due to the use of finite-precision the differentiation matrices, that are dense, incur massive round-off errors as the size of the matrices exceed about 100. This eliminates the Chebyshev method from image processing applications since the resolution of the input image dictates the dimension of the scheme. Finally though, the use of a finite difference-Laplace hybrid method is plausible due to the structure of the differentiation matrices. The effects of a fractional order model applied to images was then investigated in Chapter 6.

Due to the linearity of the Laplace transform the semi-discrete schemes described in Chapter 5 could not be applied directly to the time-fractional Fitzhugh-Nagumo equation due to the cubic nonlinear source term. By employing the quasi-linearization scheme we were able to apply the hybrid schemes to not only the linear case, but also the nonlinear case subject to a discontinuous initial condition or an image. Once the methods were in place the effects of the fractional-order nonlinear model on images was investigated.

Introducing the time-fractionality alters the steady state of the equation, as well as the transient time to obtain the steady state. In many cases reducing the order of differentiation below one dramatically reduces the transient time and this effect is increased as the order tends to zero. Altering the steady state has dramatic effects on a resulting image. This introduces a new dimension of control allowing one to brighten or darken the image by changing α in a dynamic way, rather than directly increasing or decreasing each pixel's value. Furthermore by altering the steady state of the equation, the fractional order drives an image away from a completely diffused state, as is the integer order case. This means, in terms of image processing, that the structure of the image

remains intact allowing for very fine control over details such as degree of diffusion, as well as aggressiveness of the binarizing source term.

This research opens up new avenues of approach in the field of image processing. New fractional order models may be implemented and their effects analyzed. By introducing the extra dimension of control via the fractional ordered derivative we are able to dramatically change the effects of a model on the image as well as the transient time. We have also derived an effective method for implementing such a model, in both the linear and nonlinear case, that allows the practitioner to immediately jump to the state in time rather than being required to march to that time, as is needed in many traditional approaches.

7.3 Conclusion

This work aimed to derive systematic methods for the solution of two-dimensional diffusion equations with a nonlinear source term for an arbitrary order derivative in time under different boundary conditions. The application of the time-fractional diffusion equation to images via the presented method is prohibitively expensive due to the resulting algebraic systems being large, based on the resolution of the input image, and needing to be solved multiple times. However we did show that the presented method was effective at producing accurate solutions to the two-dimensional problem under different boundary conditions with a relatively small number of spatial discretization points.

The integer order case produced excellent results and ultimately culminated in an effective process for document image binarization compared with the state-of-the-art. We proved this method to be stable under certain conditions as well as presenting a stable efficient method that is massively parallelizable and implemented on GPUs.

Bibliography

[1] J. He, Q. D. M. Do, A. C. Downton, and J. H. Kim. A comparison of binarization methods for historical archive documents. *Proceedings of the 2005 Eight International Conference on Document Analysis and Recognition*, 1:538 – 542, 2005.

[2] M. Sezgin and B. Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 13(1):146 – 165, 2004.

[3] O. D. Trier and T. Taxt. Evaluation of binarization methods for document images. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 17(3):312 – 315, 1995.

[4] P. W. Palumbo, P. Swaminathan, and S. N. Srihari. Document image binarization: Evaluation of algorithms. *Society of Photo-Optical Instrumentation Engineers*, 697:278 – 285, 1986.

[5] J. Malik P. Perona. Scale space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal.*, 12:629 – 639, 1990.

[6] M. Bertalmio, A. L. Bertozzi, and G. Sapiro. Navier-stokes, fluid dynamics, and image and video inpainting. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–355. IEEE, 2001.

[7] M. Ebihara, H. Mahara, T. Sakurai, A. Nomura, A. Osa, and H. Miike. Segmentation and edge detection of noisy image and low contrast image based on a reaction-diffusion model. *The Journal of the Institute of Image Electronics Engineers of Japan*, 32:378 – 385, 2003.

[8] R. Fitzhugh. Impulse and physiological states in models of nerve membrane. *Biophys Journal*, 1:445 – 466, 1961.

[9] J. S. Nagumo, S. Arimoto, and S. Yoshizawa. An active pulse transmission line simulating nerve axon. *Proc. IRE*, 50:2061 – 2071, 1962.

[10] G. W. Leibnitz. Letter from hanover, germany, september 30, 1695 to g. a. l'hopital. *Leibnizen Mathematische Schriften*, 2:301 – 302, 1962.

[11] K. B. Oldham and J. Spanier. *The Fractional Calculus.* Academic Press, Inc., 1974.

[12] G. Jumarie. Fractional master equation: non-standard analysis and liouville riemann derivative. *Chaos, Solitons and Fractals*, 12:2577 – 2587, 2001.

[13] G. Jumarie. Fractional partial differential equations and modified riemann liouville derivative new methods for solution. *Journal of Applied Mathematics and Computing*, 24(1 - 2):31 – 48, 2007.

[14] G. Jumarie. Fourier's transform of fractional order via mittag-leffler function and modified riemann-liouville derivative. *Journal of Applied Mathematics and Informatics*, 26(5 - 6):1101 – 1121, 2008.

[15] G. Jumarie. Modified riemann-liouville derivative and fractional taylor series of nondifferentiable functions further results. *Computers and Mathematics with Applications*, 62:1367 – 1376, 2006.

[16] G. Jumarie. On the solution of the stochastic differential equation of exponential growth driven by fractional brownian motion. *Applied Mathematics Letters*, 18: 817 – 826, 2005.

[17] G. Jumarie. On the representation of fractional brownian motion as an integral with respect to $(dt)^a$. *Applied Mathematics Letters*, 18:739 – 748, 2005.

[18] R. Hilfer. *Applications of Fractional Calculus in Physics.* World Scientific, 2000.

[19] S. Havlin and D. Ben-Avraham. Diffusion in disordered media. *Advances in Physics*, 51(1):187 – 292, 2002.

[20] W. Chen and S. Holm. Physical interpretation of fraction diffusion-wave equation via lossy media obeying frequency power law. *Physics Review Letters (under preperation),*, 2003.

[21] C. Cattaneo. Sulla conduzione del calore. *Atti Semin. Mat. Fis. della Universita di Modena*, 3:3, 1948.

[22] A. Compte and R. Metzler. The generalized cattaneo equation for the description of anomalous transport processes. *Journal of Physics A: Mathematical and General*, 30:7277 – 7289, 1997.

[23] R. Metzler and J. Klafter. The random walk's guide to anamalous diffusion: A fractional dynamics approach. *Physical Reports*, 339:1 –77, 2000.

[24] R. Metzler and J. Klafter. The restaurant at the end of the random walk: recent developments in the description of anaomalous transport. *Journal of Physics A: Mathematical and General*, 37:161 – 208, 2004.

[25] Q. Zeng, H. Li, and D. Liu. Anomalous fractional diffusion equation for transport phenomena. *Communications in Nonlinear Science and Numerical Simulation*, 4 (2):99–104, 1999.

[26] D. A. Benson, S. W. Wheatcraft, and M. M. Meerschaert. Application of a fractional advection-dispersion equation. *Water Resources Research*, 36(6):1403–1412, 2000.

[27] J. He. Approximate analytical solution for seepage flow with fractional derivatives in porous media. *Computer Methods in Applied Mechanics and Engineering*, 167 (1):57–68, 1998.

[28] R. Metzler, W. G. Glöckle, and T. F. Nonnenmacher. Fractional model equation for anomalous diffusion. *Physica A: Statistical Mechanics and its Applications*, 211(1):13–24, 1994.

[29] V. Gafiychuk, B. Datsko, and V. Meleshko. Mathematical modeling of time fractional reaction–diffusion systems. *Journal of Computational and Applied Mathematics*, 220(1):215–225, 2008.

[30] B. Y. Datsko and V. V. Gafiychuks. Mathematical modeling of fractional reaction-diffusion systems with different order time derivatives. *Journal of Mathematical Sciences*, 165(3):392–402, 2010.

[31] B. Bonilla, M. Rivero, L. Rodríguez-Germá, and J. J. Trujillo. Fractional differential equations as alternative models to nonlinear differential equations. *Applied Mathematics and Computation*, 187(1):79–88, 2007.

[32] I. Podlubny. *Fractional Differential Equations*, volume 198 of *Mathematics in Science and Engineering*. Academic Press, 1999.

[33] I. Podlubny. Geometric and physical interpretation of fractional integration and fractional differntiation. *Fractional Calculus and Applied Analysis*, 5(4):367 – 386, 2002.

[34] M. M. Meerschaert and C. Tadjeran. Finite difference approximations for fractional advection-dispersion flow equations. *Journal of Computational and Applied Mathematics*, 172:65 – 77, 2004.

[35] S. B. Yuste and L. Acedo. An explicit finite difference method and a new von neumann-type stability analysis for fractional diffusion equations. *SIAM Journal on Numerical Analysis*, 42(5):1862 – 1874, 2005.

[36] C. Tadjeran and M. M. Meerschaert. A second-order accurate numerical method for the two-dimensial fractional diffusion equation. *Journal of Computational Physics*, 220:813 – 823, 2007.

[37] G. Birkhoff, R. S. Varga, and D. Young. Alternating direction implicit methods. *Advances in computers*, 3:189–273, 1962.

[38] K. Diethelm, N. J. Ford, and A. D. Freed. A predictor-corrector approach for the numerical solution of fractional differential equations. *Nonlinear Dynamics*, 29(1-4):3–22, 2002.

[39] A. Mohebbi, M. Abbaszadeh, and M. Dehghan. A high-order and unconditionally stable scheme for the modified anomalous fractional sub-diffusion equation with a nonlinear source term. *Journal of Computational Physics*, 2013.

[40] M. Cui. Compact alternating direction implicit method for two-dimensional time fractional diffusion equation. *Journal of Computational Physics*, 231(6):2621–2633, 2012.

[41] M. M. Meerschaert, H. P. Scheffler, and C. Tadjeran. Finite difference methods for two-dimensional fractional dispersion equation. *Journal of Computational Physics*, 211(1):249–261, 2006.

[42] S. Momani and Z. Odibat. Numerical comparison of methods for solving linear differential equations of fractional order. *Chaos, Solitons & Fractals*, 31(5):1248–1255, 2007.

[43] V. E. Lynch, B. A. Carreras, D. del Castillo-Negrete, K.M Ferreira-Mejias, and H. R. Hicks. Numerical methods for the solution of partial differential equations of fractional order. *Journal of Computational Physics*, 192(2):406–421, 2003.

[44] Z. Odibat and S. Momani. Numerical methods for nonlinear partial differential equations of fractional order. *Applied Mathematical Modelling*, 32(1):28–39, 2008.

[45] A. Ostermann and M. Roche. Runge-kutta methods for partial differential equations and fractional orders of convergence. *Mathematics of computation*, 59(200): 403–420, 1992.

[46] A. El-Kahlout, T. O. Salim, and S. El-Azab. Exact solution of time fractional partial differential equation. *Applied Mathematical Sciences*, 2(52):2577–2590, 2008.

[47] R. K. Saxena, A. M. Mathai, and H. J. Haubold. Solutions of the fractional reaction equation and the fractional diffusion equation. In *Proceedings of the Third UN/ESA/NASA Workshop on the International Heliophysical Year 2007 and Basic Space Science*, pages 53–62. Springer, 2010.

[48] A. Arikoglu and I. Ozkol. Solution of fractional differential equations by using differential transform method. *Chaos, Solitons & Fractals*, 34(5):1473–1481, 2007.

[49] A. Cetinkaya, O. Kıymaz, and J. Camlı. Solutions of nonlinear pdes of fractional order with generalized differential transform method. In *International Mathematical Forum*, volume 6, pages 39–47, 2011.

[50] G. C. Wu. A fractional variational iteration method for solving fractional nonlinear differential equations. *Computers & Mathematics with Applications*, 61(8):2186–2190, 2011.

[51] Z. Odibat and S. Momani. Application of variational iteration method to nonlinear differential equations of fractional order. *International Journal of Nonlinear Sciences and Numerical Simulation*, 7(1):27–34, 2006.

[52] Z. Odibat and S. Momani. A generalized differential transform method for linear partial differential equations of fractional order. *Applied Mathematics Letters*, 21 (2):194–199, 2008.

[53] S. Momani and Z. Odibat. A novel method for nonlinear fractional partial differential equations: Combination of dtm and generalized taylor's formula. *Journal of Computational and Applied Mathematics*, 220(1):85–95, 2008.

[54] H. P. Chu and C. L. Chen. Hybrid differential transform and finite difference method to solve the nonlinear heat conduction problem. *Communications in Nonlinear Science and Numerical Simulation*, 13(8):1605–1614, 2008.

[55] H. Jafari and V. Daftardar-Gejji. Positive solutions of nonlinear fractional boundary value problems using adomian decomposition method. *Applied Mathematics and Computation*, 180(2):700–706, 2006.

[56] F. Catté, P. L. Lions, J. M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM Journal of Numerical Analysis*, 29: 182 – 193, 1992.

[57] F. C. Liu and Y. Ha. Noise reduction based on reaction-diffusion system. *Proceedings of the 2007 International Conference on Wavelet Analysis and Pattern Recognition*, pages 831 – 834, 2007.

[58] G. H. Cottet and L. Germain. Image processing through reaction combined with nonlinear diffusion. *Mathematics of Computation*, 61:659 – 673, 1993.

[59] J. Kacur and K. Mikula. Solution of nonlinear diffusion appearing in image smoothing and edge detection. *Applied Numerical Mathematics*, 17:47 – 59, 1995.

[60] L. Alvarez, J. M. Morel, and P. L. Lions. Image selective smoothing by nonlinear diffusion. *SIAM Journal of Numerical Analysis*, 29:845 – 866, 1992.

[61] S. Morfu. On some applications of diffusion processes for image processing. *Physics Letters A*, 373:2438 – 2444, 2009.

[62] Y. L. You and M. Kaveh. Fourth-order partial differential equations for noise removal. *Image Processing, IEEE Transactions on*, 9(10):1723–1730, 2000.

[63] J. Bai and X. C. Feng. Fractional-order anisotropic diffusion for image denoising. *IEEE Transactions on Image Processing*, 16(10):2492 – 2502, 2007.

[64] R. Abdur and A. I. B. Md. Ismail. Numerical studies on two-dimensional schrodinger equation by chebyshev spectral collocation method. *U.P.B. Scientific Bulletin, Series A*, 73(1):101 – 110, 2011.

[65] Z. Avazzadeh, M. Heydari, and G. B. Loghmani. A comparison between solving two dimensional integral equations by the traditional collocation method and radial basis functions. *Applied Mathematical Sciences*, 5(23):1145 – 1152, 2011.

[66] M. Javidi and A. Golbabai. Spectral collocation method for parabolic partial differential equations with neumann boundary conditions. *Applied Mathematical Sciences*, 1(5):211 – 218, 2007.

[67] A. H. Khater, R. S. Temsah, and M. M. Hassan. A chebyshev spectral collocation method for solving burgers' -type equations. *Journal of Computational and Applied Mathematics*, 222:333 – 350, 2008.

[68] L. N. Trefethen. *Spectral Methods in MATLAB*. Society for Industrial and Applied Mathematics, 2000.

[69] D. Funaro and D. Gottlieb. A new method of imposing boundary conditions in pseudospectral approximations of hyperbolic equations. *Mathematics of computation*, 51(184):599–613, 1988.

[70] N. Bressan and A. Quarteroni. Analysis of chebyshev collocation methods for parabolic equations. *SIAM journal on numerical analysis*, 23(6):1138–1154, 1986.

[71] F. Costabile and A. Napoli. Stability of chebyshev collocation methods. *Computers & Mathematics with Applications*, 47(4):659–666, 2004.

[72] A. Bueno-Orovio, D. Kay, and K. Burrage. Fourier spectral methods for fractional-in-space reaction-diffusion equations. *Journal of Computational Physics*, 2012.

[73] M. Caputo. Linear models of dissipation whose q is almost frequency independent. *Geophysical Journal of the Royal Astronomical Society*, 13(5):529 – 539, 1967.

[74] D. W. Peaceman and Jr. H. H. Rachford. The numerical solution of parabolic and elliptic differential equations. *Journal of the Society for Industrial and Applied Mathematics*, 3(1):28 – 41, 1955.

[75] A. Bayliss, A. Class, and B. J. Matkowsky. Roundoff error in computing derivatives using the chebyshev differentiation matrix. *Journal of Computational Physics*, 116: 380 – 383, 1994.

[76] P. J. Singh, S. Roy, and I. Pop. Unsteady mixed convection from a rotating vertical slender cylinder in an axial flow. *International Journal of Heat and Mass Transfer*, 51(5-6):1423–1430, mar 2008. ISSN 00179310.

[77] P. M. Patil, S. Roy, and I. Pop. Unsteady mixed convection flow over a vertical stretching sheet in a parallel free stream with variable wall temperature. *International Journal of Heat and Mass Transfer*, 53(21-22):4741–4748, oct 2010. ISSN 00179310.

[78] P. M. Patil and S. Roy. Unsteady mixed convection flow from a moving vertical plate in a parallel free stream: Influence of heat generation or absorption. *International Journal of Heat and Mass Transfer*, 53(21-22):4749–4756, oct 2010. ISSN 00179310.

[79] F. Drira, F. Lebourgeois, and H. Emptoz. Document images restoration by a new tensor based diffusion process: Application to the recognition of old printed documents. *International Conferenece on Document Analysis and Recognition*, 1: 321 – 325, 2009.

[80] F. Drira, F. Lebourgeois, and H. Emptoz. A new pde-based approach for singularity-preserving regularization: Application to degraded characters restoration. *International Journal of Document Analysis and Recognition*, 15(3):183 – 212, 2012.

[81] R. F. Moghaddam and M. Cheriet. Low quality document image modeling and enhancement. *International Journal of Document Analysis and Recognition*, 11 (4):183 – 201, 2009.

[82] Z. Mahani, J. Zahid, S. Saoud, M. Rhabi, and A. Hakim. Text enhancement by pdes based methods. *Image and Signal Processing*, pages 65 – 76, 2012.

[83] R. Smith. *Tesseract OCR*, 2012.

[84] N. Otsu. A threshold selection method from grey-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-9(1):62 – 66, 1979.

[85] W. Niblack. *An introduction to digital image processing.* Strandberg Publishing Company, 1985.

[86] J. Sauvola and M. Pietikainen. Adaptive document image binarization. *Pattern Recognition*, 33:225 – 236, 2000.

[87] G. Orwell. *Nineteen Eighty-Four.* Secker and Warburg, 1949.

[88] B. A. Jacobs and E. Momoniat. A novel approach to text binarization via a diffusion-based model. *Applied Mathematics and Computation*, To appear, 2013.

[89] F. A. Cruz, S. K. Laytona, and L. A. Barba. How to obtain efficient gpu kernels: An illustration using fmm and fgt algorithms. *Computer Physics Communications*, 182:2084 – 2098, 2011.

[90] Y. Su and Z. Xu. Parallel implementation of wavelet-based image denoising on programmable pc-grade graphics hardware. *Signal Processing*, 90:2396 – 2411, 2010.

[91] K. Fatahalian. Overview: Making sense of gpu architectures. *ACM SIGGRAPH*, 10:723 – 728, 2008.

[92] J. Owens. Gpu architecture overview. *ACM SIGGRAPH*, 10:723 – 728, 2007.

[93] J. Krüger and R. Westermann. Linear algebra operators for gpu implementation of numerical algorithms. *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH*, 22(3):908 – 916, 2003.

[94] J. Bolz, I. Farmer, E. Grinspun, and P. Schröoder. Sparse matrix solvers on the gpu: conjugate gradients and multigrid. *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH*, 22(3):917 – 924, 2003.

[95] J. Shen and S. Castan. An optimal linear operator for edge detection. *Proc. CVPR*, 86:109 – 114, 1986.

[96] B. Gatos, K. Ntirogiannis, and I. Pratikakis. Icdar 2009 document image binarization contest (dibco 2009). In *Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on*, pages 1375–1382. IEEE, 2009.

[97] I. Pratikakis, B. Gatos, and K. Ntirogiannis. H-dibco 2010-handwritten document image binarization competition. In *Frontiers in Handwriting Recognition (ICFHR), 2010 International Conference on*, pages 727–732. IEEE, 2010.

[98] I. Pratikakis, B. Gatos, and K. Ntirogiannis. Icdar 2011 document image binarization contest (dibco 2011). In *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, pages 1506–1510. IEEE, 2011.

[99] I. Pratikakis, B. Gatos, and K. Ntirogiannis. Icfhr 2012 competition on handwritten document image binarization (h-dibco 2012). In *Frontiers in Handwriting Recognition (ICFHR), 2012 International Conference on*, pages 817–822. IEEE, 2012.

[100] K. Ntirogiannis, B. Gatos, and I. Pratikakis. An objective evaluation methodology for document image binarization techniques. In *Document Analysis Systems, 2008. DAS'08. The Eighth IAPR International Workshop on*, pages 217–224. IEEE, 2008.

[101] H. Lu, A. C. Kot, and Y. Q. Shi. Distance-reciprocal distortion measure for binary document images. *Signal Processing Letters, IEEE*, 11(2):228–231, 2004.

[102] Lauri Savioja. Real-time 3d finite-difference time-domain simulation of low-and mid-frequency room acoustics. In *13th Int. Conf on Digital Audio Effects*, volume 1, page 75, 2010.

[103] Jens Krüger and Rüdiger Westermann. Linear algebra operators for gpu implementation of numerical algorithms. In *ACM Transactions on Graphics (TOG)*, volume 22, pages 908–916. ACM, 2003.

[104] Athanasios Antoniou, Konstantinos I Karantasis, Eleftherios D Polychronopoulos, and John A Ekaterinaris. Acceleration of a finite-difference weno scheme for large-scale simulations on many-core architectures. In *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, page 525, 2010.

[105] G. H. Gao, Z. A. Sun, and Y. N. Zhang. A finite difference scheme for fractional sub-diffusion equations on an unbounded domain using artificial boundary conditions. *Journal of Computational Physics*, 231(7):28652879, 2012.

[106] O. P. Agrawal. Solution for a fractional diffusion-wave equation defined in a bounded domain. *Nonlinear Dynamics*, 29:145 – 155, 2002.

[107] J. He. Variational iteration method for delay differential equations. *Communications in Nonlinear Science and Numerical Simulation*, 2(4):235 – 236, 1997.

[108] J. K. Zhou. *Differential transformation and its applications for electrical circuits.* Huazhong University Press, Wuhan, China, 1986.

[109] G. Adomian. A review of the decomposition method in applied mathematics. *Journal of mathematical analysis and applications*, 135(2):501–544, 1988.

[110] J. A. C. Weideman and L. N. Trefethen. Parabolic and hyperbolic contours for bromwich integral. *Mathematics of Computation*, 76(259):1341 – 1356, 2007.

[111] B. A. Jacobs and C. Harley. A comparison of two hybrid methods for applying the time-fractional heat equation to a two dimensional function. *AIP Conference Proceedings*, 1558:2119 – 2122, 2013.

[112] J. W. Thomas. *Numerical Partial Differential Equations.* Springer, 1999.

[113] K. S. Breuer and R. M. Everson. On the errors incurred calculating derivatives using chebyshev polynomials. *Journal of Computational Physics*, 99(1):56–67, 1992.

[114] W. S. Don and A. Solomonoff. Accuracy and speed in computing the chebyshev collocation derivative. *SIAM Journal on Scientific Computing*, 16(6):1253–1268, 1995.

[115] M. Kazemi and G. H. Erjaee. Analytical and numerical solutions of different parabolic heat equations presented in the form of multi-term fractional differential equations. *Iranian Journal of Science and Technology, IJST A*, 3:185192, 2011.

[116] S. Momani. Analytical approximate solution for fractional heat-like and wave-like equations with variable coefficients using the decomposition method. *Applied Mathematics and Computation*, 165:459 – 472, 2005.

[117] B. A. Jacobs and C. Harley. A comparison of two hybrid methods for solving linear time-fractional partial differential equations on a two-dimensional domain. *Submitted to Abstract and Applied Analysis Special Issue: New Trends on Fractional and Functional Differential Equations*, 2013.