RECONSTRUCTION AND MOTION ESTIMATION OF SPARSELY SAMPLED IONOSPHERIC DATA

Matthew Philip Foster

A thesis submitted for the degree of Doctor of Philosophy
University of Bath
Department of Electronic and Electrical Engineering

2008

COPYRIGHT

Attention is drawn to the fact that copyright of this thesis rests with its author. A copy of this thesis has been supplied on condition that anyone who consults it is understood to recognise that its copyright rests with the author and they must not copy it or use material from it except as permitted by law or with the consent of the author.

This thesis may be made available for consultation within the University Library and may be photocopied or lent to other libraries for the purposes of consultation.

Matthew Philip Foster

Abstract

This thesis covers two main areas which are related to the mapping and examination of the ionosphere. The first examines the performance and specific nuances of various state-of-the-art interpolation methods with specific application to mapping the ionosphere. This work forms the most widely scoped examination of interpolation technique to date, and includes the introduction of normalised convolution techniques to geophysical data. In this study, adaptive-normalised convolution was found to perform well in ionospheric electron content mapping, and the popular technique, kriging was found to have problems which limit its usefulness.

The second, is the development and examination of automatic data-driven motion estimation methods for use on ionospheric electron content data. Several motion estimation methods are applied to such data, including methods based on optical flow, correlation and boundary-correspondence. Correlation and relaxation labelling based methods are found to perform reasonably, and boundary based methods based on shape-context matching are found to perform well, when coupled with a regularisation stage.

Overall, the techniques examined and developed here will help advance the process of examining the features and morphology of the ionosphere, both during storms and quiet times.

Acknowledgements

I owe thanks to many people for their help and friendship throughout this project: to my peers in the Centre for Space, Oceanic and Atmospheric Science for their friendship; to my supervisor, Dr. Adrian Evans, for his help, interest, encouragement and advice over the past few years; to my partner, Charlie Beldon for her patience and innumerable read-throughs; and to my mother, Caroline and grandfather, Roy for imbibing me with curiosity and always believing in me. Without the help and support of these people I would probably not have started, let alone finished this undertaking.

Finally, I would like to thank the UK Engineering and Pysical Sciences Research Council (EPSRC) for giving me the opportunity to embark on this project.

Contents

List of Figures

List of Tables

List of Acronyms

AC active Contour

ANC adaptive normalised convolution

BSI biharmonic spline interpolation

C/A coarse acquisition

CCC cross correlation coefficient

CME coronal mass ejection

CM curve matching

DD diffusion distance

DoNC differential of normalised convolution

EMD Earth mover's distance

EUV extreme ultra-violet

GNSS global navigation satellite systems

GPS global positioning system

GST gradient square tensors

IGS International GNSS Service

IIR infinite impulse response

IRI International reference ionosphere

IMF Interplanetary Magnetic Field

MCC maximum cross-correlation

ME motion estimation

MIDAS Multi-Instrument Data Analysis System

MM mathematical morphology

NC normalised convolution

NDC normalised differential convolution

NN natural neighbour

NS navigational signals

PSNR peak signal-to-noise ratio

PRN pseudorandom-noise

RBF radial basis function

RL relaxation labelling

RMSE root mean square error

RMS root mean square

ROI regions of interest

SAVD sum of absolute value of differences

SC shape context

SED storm electron density

SIFT scale-invariant feature transform

SSD sum of squared-differences

SSE sum squared error

SVD singular value decomposition

TEC total electron content

TOI tongue of ionisation

TPS thin plate spline

TSM thin sell model

TOA time of arrival

VMF vector-median filter

VHR very-high resolution

List of Symbols

A	Anisotropy
c	Binary confidence map
$\phi(\cdots)$	A radial basis function
$f(\cdots)$	An (interpolated output) image
$g(\cdots)$	A Gaussian filter kernel
γ_1	The skewness of a distribution
γ_2	The kurtosis of a distribution
λ	Eigenvalue or regularisation parameter
ρ	Cross-correlation coefficient
$\otimes$	Convolution
σ_u	Standard deviation along u-axis.
σ^2	Variance
$\overline{x}$ or μ	The mean of a variable, x
$\mathbf{x}$	Position vector (i.e. $\mathbf{x} = [x, y, \ldots]^T$)

Chapter 1

Introduction

Many fields benefit from the introduction of ideas and techniques from other areas, and a great number of advances arise in the *gaps* between areas of work. One of the aims of this thesis is bridging the gap between image processing, computer vision techniques and atmospheric research, specifically in the creation and application ionospheric total electron content (TEC) maps. These maps show the distribution of electrons in the ionosphere, and are used in a wide range of applications, from propagation forecasting and nowcasting[1], to calibration of satellite navigation systems, through to scientific analysis of space weather events.

This work aims to aid some of these applications by introducing the state-of-the-art in interpolation and motion estimation to TEC mapping. To this end, work contained here pertains to the production of accurate TEC maps from sparse data, and the automated tracking of image derived ionospheric features.

As the ionosphere is the medium on which the majority of this work is based, this introductory chapters describes the ionosphere and its storms, considers global positioning system (GPS) based approaches to imaging, and modern motion estimation techniques for sensing its activity during geomagnetic storms.

The introduction concludes with a chapter-by-chapter summary of the remainder of the thesis.

[1] See, for example JPL's real-time TEC mapping page: http://iono.jpl.nasa.gov/latest_rti_global.html

1.1 The Ionosphere

The ionosphere is the region of the atmosphere extending from an altitude of approximately 50 km to over 1000 km. In this region, free electrons can exist for short periods of time and form an electrically conducting plasma. These electrons are liberated when extreme ultra-violet (EUV) light from the Sun ionises neutral atoms in the atmosphere. The resulting plasma has various effects on electromagnetic waves, including delaying or blocking their propagation at certain frequencies.

The process of electron liberation by EUV ionisation is known as photoionisation, and is counteracted by two main processes, both of which act to reduce ionisation. In both of these cases, the rate of recombination is controlled by the number of available neutral atoms, and so varies with altitude. The two process are outlined below.

- The first process is the *recombination* of electrons and ions to form neutral atoms. There are two forms of recombination known as *radiative* and *dissociative*.
  - *Radiative* recombination is most common, and occurs when an electron and an ion recombine directly.
  - *Dissociative* recombination is less common, and involves a more efficient, two stage, process. In the first stage, positive ions interact with various neutral molecules replacing one of the atoms in the molecule. In the second stage, electrons combine with the positively charged molecule just created.
- The second is *attachment*, which occurs at lower altitudes where there are more neutral atoms. This process involves electrons combining with neutral atoms to form negative ions.

Several factors, acting in synchrony, cause the electron density of the ionosphere varies with altitude. These include the density of neutral atoms decreasing with altitude and the fact the intensities of various EUV wavelengths differ across altitudes. The change of electron density with height is known as the vertical electron density profile, and contains several distinct layers, known as D, E, F_1 and F_2, in increasing altitude. During the day high levels of photoionisation ensure that all four layers are present. However, at night, recombination dominates, and the D, E and F_1 layers are almost entirely depleted leaving only the F_2 layer. These layers are illustrated in Fig. 1.1, which shows that the F_2 region is the only layer present at night, and has the highest electron density.

Radio waves propagating through the ionosphere are affected because they cause excitation of plasma. If the wave frequency is less than the *plasma frequency* – the frequency

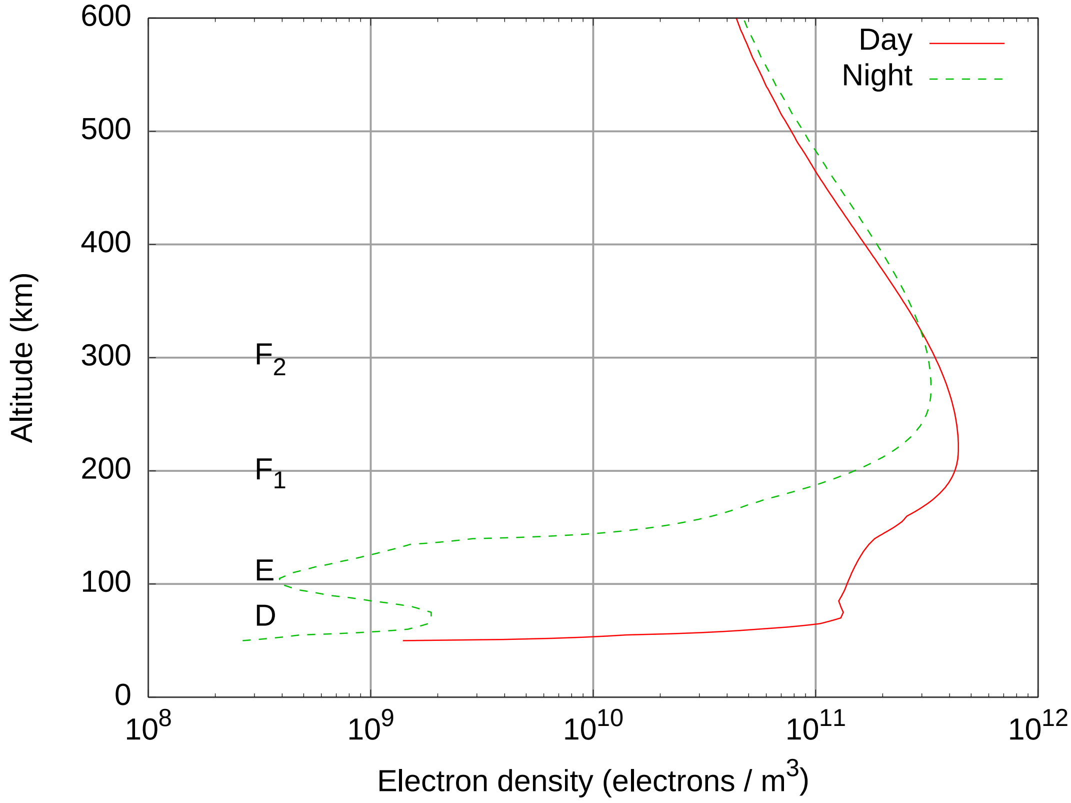


Figure 1.1: Day and night example electron density profiles, generated using IRI2001, for 30 June 2008 at noon and midnight.

at which the electrons and ions in a slab of plasma will oscillate when perturbed – the wave will be re-radiated, otherwise it will be pass through. The plasma frequency is given by $f_N = \sqrt{80.5N}$, where N is the electron density. The *critical frequency* of a layer is the maximum frequency which can be reflected the layer at vertical incidence. The critical frequency, of a layer is given by $f_c \approx 9 \times 10^{-6}\sqrt{N_m}$, where N_m is the maximum electron density of the layer (in electrons per m^3). Critical frequencies in the various layers (which are described below) are denoted f_oE, f_0F_1 and f_0F_2, and such terms are regularly seen in literature pertaining to the ionosphere. In addition to reflection effects, radio waves also experience Faraday rotation when passing through ionised regions. This is equivalent to a time delay which is proportional to the level of ionisation, and inversely proportional to the square of the signal frequency.

1.1.1 Ionospheric Storms

Much of the behaviour of the ionosphere is governed by how the Earth's magnetosphere and the Interplanetary Magnetic Field (IMF) interact and connect. The Earth's magnetosphere is a region of the atmosphere, linked to the top of the ionosphere, which contains a mix of ions and electrons, held in place by the Earth's goemagnetic field and the solar wind. It consists of a long tail, about 70,000 km long, facing away from the Sun, which is swept out by the solar wind. The outer edge of the magnetosphere is known as the *magnetopause*. Outside of this is an area called the *magnetosheath*, which is bounded by the *bow shock*. This is a region where the solar wind velocity drops suddenly, and the magnetic field lines are highly compressed. Fig. 1.2 shows the positions of these regions schematically. Most high energy particles are prevented from entering lower parts of atmosphere by the magnetosphere, making its shielding effects very important for all life on Earth.

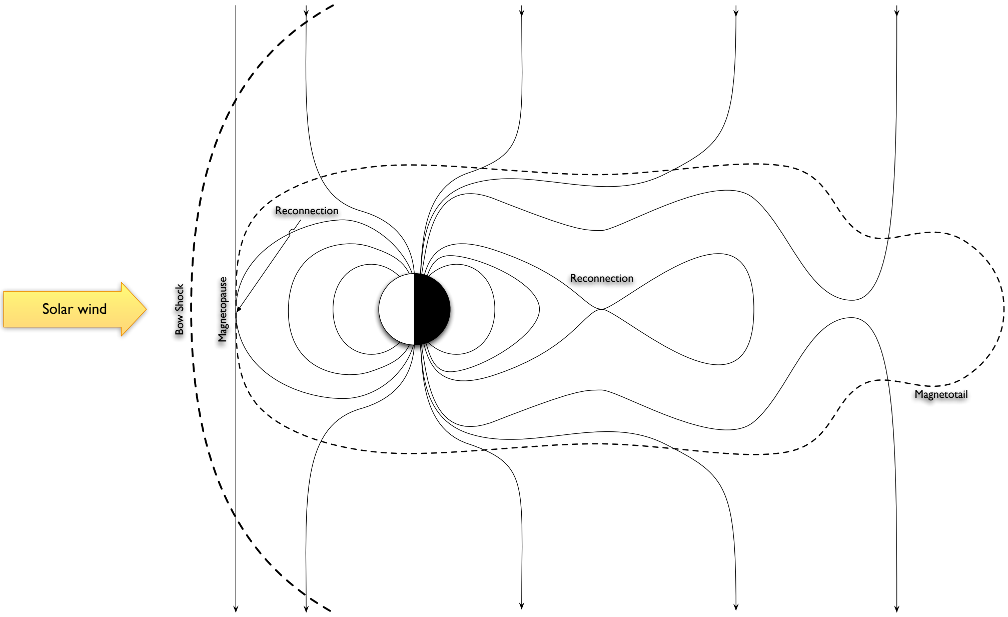


Figure 1.2: A schematic illustrating the Earth's magnetosphere. The solid lines represent magnetic field lines.

The IMF is formed by the steady outflow of solar wind from the Sun carrying the Sun's magnetic field. The Sun's rotation causes the IMF to be shaped like an Archimedean spiral[2]. Intense variation in the Sun's surface magnetic field due to sunspots means that the orientation of the IMF varies with time. The magnitude of the 'vertical' component of the IMF is known as B_z, and its orientation determines whether solar wind plasma can enter the ionosphere. Regions of open magnetic-field lines, known as polar cusps form between the sunward and tailward areas of the geomagnetic field. In the northern polar cusp, the magnetic field is directed towards Earth, and in the southern cusp, the magnetic field lines point away.

When the orientation of IMF is southward, it is able to link with the Earth's field at the polar cusps. As a result, solar wind plasma can be accelerated through the magnetosphere, into the upper ionosphere. When southward B_z coincides with the Sun ejecting very large numbers of particles, due to a solar flare or coronal mass ejection

[2] called the Parker spiral, after Eugene Parker who predicted the Solar wind in the 1950s

(CME), large geomagnetic storms can result. These storms can cause large auroral displays and disrupt power distribution and communications, making monitoring and predicting their occurrence and effects very important.

Geomagnetic storms can result in large quantities of electrons and ions which increase the electron density of the ionosphere. Often, during large storms in the northern hemisphere, regions of enhanced electron density form in patches at mid-latitudes, these then travel across the northern polar cap towards the night-side of the globe in a convective pattern known as a tongue of ionisation (TOI). The path taken across the polar cap depends on the direction of the IMF, with various different convection patters forming under different conditions. These complex convective patterns are the subject of various scientific studies [Foster et al., 2005, Stolle et al., 2005], and make the analysis of the motion of the TOI an interesting and important problem.

Several such large geomagnetic storm events occurred around Halloween of 2003, after several large solar flares. During these events, which were some of the largest ever recorded there were various radiation and geomagnetic effects. These included damage to some 28 satellites, including destruction of two; problems with long range communications and navigation; power outages and cancellation of airline operations. This makes knowledge of the behaviour of geomagnetic storms very important.

1.1.2 Imaging the Ionosphere

Because of the many and varied effects of the ionosphere, knowledge of its morphology and behaviour are of interest to both scientists and engineers. One way of analysing these properties is *imaging*, or mapping, which provides image maps of the distribution of electrons. These maps can be created using several methods, the most common of which make use of data derived from GPS satellites and receivers. These data are then interpolated or inverted to provided full 2- or 3-D fields. Incoherent scatter radars, such as EISCAT can also be used to provided electron density profiles, although spatial coverage is much more limited that with GPS, which can in theory, image the entire ionosphere.

1.1.3 The Global Positioning System

The GPS is a timing and positioning system run by the US Department of Defence. The system is divided into three segments; known as the control, space and user segments. The *control segment* consists of various tracking stations around the world, with the main control centre at Schriever Air Force Base, Colorado, USA. These stations combine measured satellite position data with models in order to precisely compute their positions (ephemeris), and necessary clock corrections. These data are then uploaded to the satellites, for inclusion in navigational signals (NS), which are sent to receiver units.

The *space segment* consists of at least 24 satellites (currently 31) configured such that there are four satellites in six orbital planes, each inclined at 55° to the equatorial plane. This means that at least six satellites are always visible from most places on the surface of the Earth.

Each of the satellites transmits its own NS, containing information on the satellite, clock corrections, ephemeris and other information. This signal is created, and then added to a pseudorandom-noise (PRN) code known as the coarse coarse acquisition (C/A) code. The resulting code is modulated on to carrier wave, known as L_1, creating a spread-spectrum signal which can be used for ranging. A second spread spectrum signal known as L_2 is also transmitted.

Both L_1 and L_2 are modulated by a code known as the precision- (P-) code. A cryptographic key is required to allow use of the P-code. However, many modern receivers make use of L_2 code without decrypting the P-code to improve ranging performance (see below).

The *user segment* consists of GPS receivers and their associated operators, or users. GPS receivers require signals from four satellites in order to compute position in three-dimensions, and time.

The receiver creates a replica C/A code which it correlates with the received signal in order to find the correct time shift for the receivers clock. The receiver clock offset is known as the time of arrival (TOA), or the pseudorange. Once the correct offset is known, the received signal can be despread, and the NS demodulated.

GPS Positioning

By combining the ephemeris data from a given satellite with the pseudorange derived from the C/A, the receiver can fix its position to the surface of a sphere surrounding the satellite. By combining four such measurements, it is possible to fix the position to one unique point – the intersection of the four spheres.

During this process, the receiver's local clock must be continually adjusted, as clock skew can severely bias position measurements. This is done by examining the imaginary

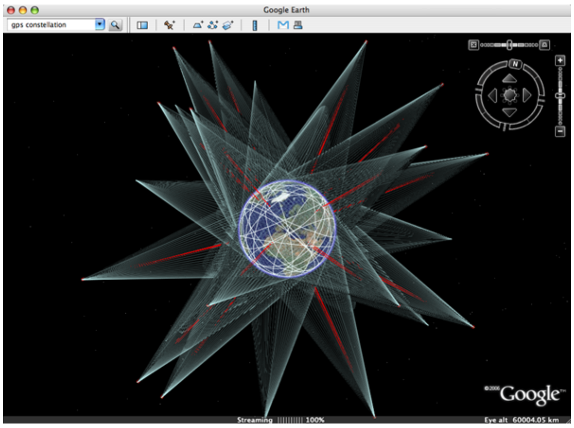


Figure 1.3: A schematic diagram of the GPS constellation created within Google Earth. After http://flickr.com/photos/alexandervandijk/324479060/ (creative commons licensed).

sphere intersections for systematic bias, and altering the clock according to the bias distance. Successive measurements from many satellites can reduce the clock error to negligible amounts.

Error Sources

There are several sources of ranging errors in the GPS. However, the influence of the ionosphere is the largest in magnitude. The ionosphere causes a frequency dependent delay in propagation of L-band signals. This delay varies according to electron concentration along the ray-path, and is typically at a minimum when the satellite is directly overhead. Current GPS handsets are able to reduce ionospheric errors to approximately 10 metres, by using models to estimate range corrections, which (assuming a maximum total electron content of 10^{18} e/m), could be as high as 26 metres for the L_2 band, and 16 metres for L_1 [pp. 294–307 Leick, 1995]. As the delay is frequency dependant, it is possible to make use of a linear combination of both L_1 and L_2's pseudoranges to further reduce the effect of ionospheric delays.

1.1.4 Constructing TEC Maps from GPS Data

A great many fixed position GPS receivers are located around the world, collecting positional data. This can then used for myriad applications, including monitoring tectonic shift and crust strain, as well as for cartography, civil engineering and precision timing. The data recorded by these receivers can also sometimes be used to analyse the delays caused by the ionosphere and other atmospheric regions. Although measurements can be made of many different atmospheric regions and variables, in the case of ionospheric electron density, the delays can be used to derive information on electron concentration.

GPS receivers are mainly situated in areas where population density is high, and as the cost of running stations can be very high, they tend to be found concentrated in more affluent countries. These factors mean that the distribution of receivers is largely random, with a considerably higher measurement density in the Northern hemisphere. The polar regions and oceans are particularly sparsely covered, making it difficult to produce TEC maps over these areas.

The two main methods of constructing maps of ionospheric TEC are tomography [e.g. Mitchell and Spencer, 2003] and interpolation [e.g. Arikan et al., 2007, Meggs et al., 2002].

Tomography

Tomographic methods use ray-tracing to project path measurements onto a grid of voxels. Non-linear inversion techniques can then be used to reconstruct the electron content data at each voxel. Basis functions, and model based interpolation can be also used to help improve output quality. Generally, tomographic inversions can provide high resolution 3-D imagery, but require large amounts of data to do so. This means that in cases when tomography fails, such in historic data and over oceans and polar regions, standard interpolation methods must be employed. Unsurprisingly, there is a very large body of literature surrounding inverse problems and tomography.

Thin-Shell Projection

In order to use standard 2-D interpolation methods, TEC data must first be converted from path measurements to spot values on a fixed height shell. This is known as the thin sell model (TSM) approach, and models the ionosphere as an infinitesimally thin shell, at a given height, normally between 300 and 400 km [Hoffmann-Wellenhof et al., 2001, pp. 102]. The disadvantage of this approach is that information on the vertical structure is completely lost. However, it has to potential to be computationally simple (depending on the interpolation method used, and especially in comparison to large matrix inversions), and allows for analysis of data which are too sparse for other methods to reconstruct. Various methods have been developed for extracting TEC information from the amplitude and phase of GPS signals, e.g. Warnant and Pottiaux [2000], Arikan et al. [2007].

Following initial processing, each path's TEC values were projected onto an infinitesimal shell at a fixed height by calculating the ray's intersection with the shell. With reference

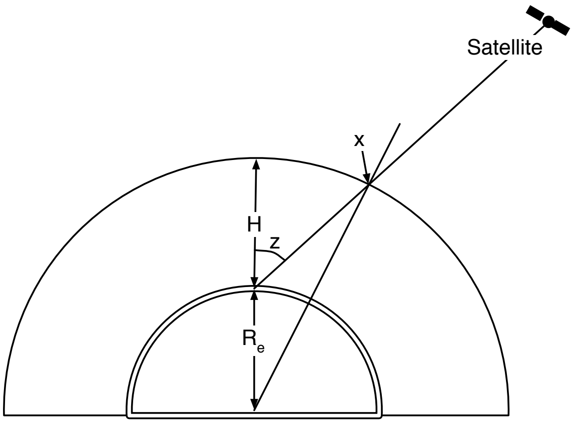


Figure 1.4: Thin Shell Ionosphere Model. R_e is the radius of the Earth, z is the elevation angle from the ground station to the satellite, x is the point at which the path between the satellite and ground station intersects the shell, and H is the height of the shell.

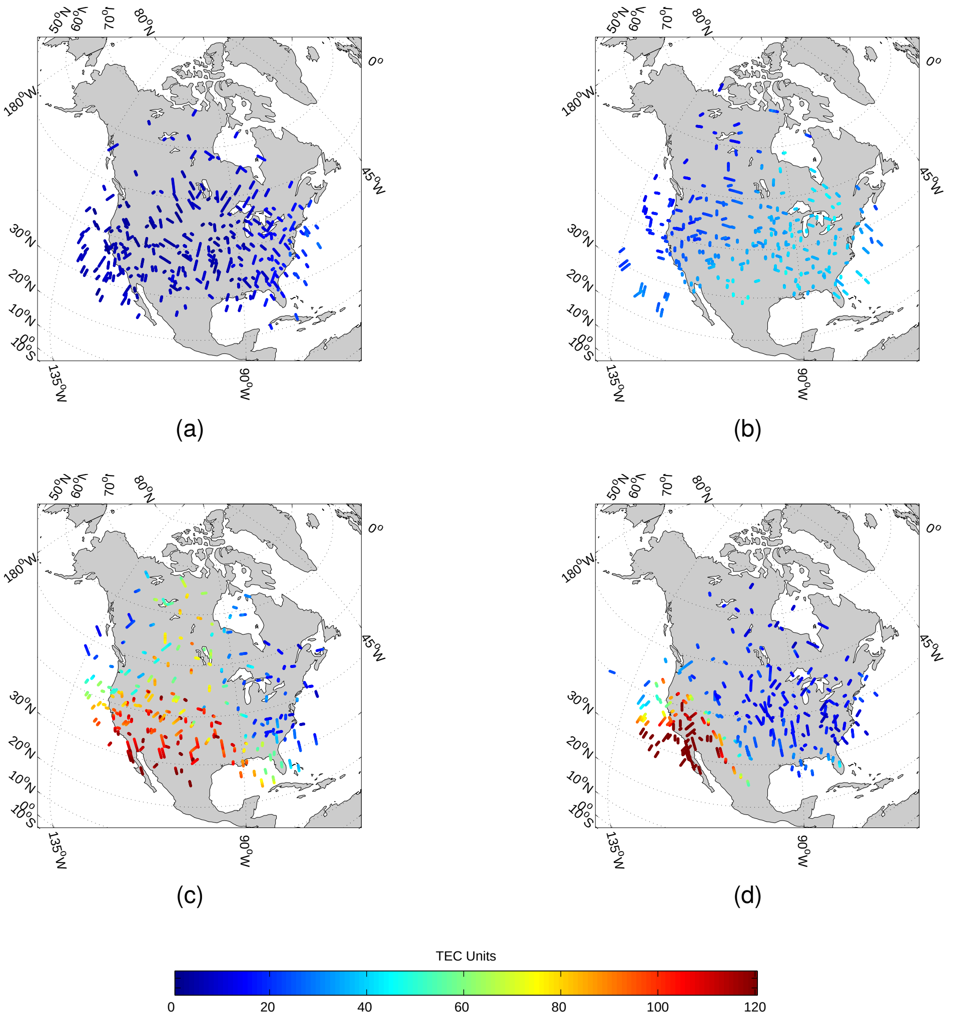
to Fig. 1.4, the function which maps from slant to vertical TEC is then given by

$$F(z) = \left(1 - \frac{R_e \cos(90 - z)}{R_e + H}\right)^{-0.5}, \tag{1.1}$$

where z is the satellite elevation angle (in degrees), H is the height of the shell (400 km in this case) and R_e is the radius of the Earth. Paths with elevation angles <20° were removed because of high levels of error associated with low angles [Mannucci et al., 1999].

Some projected TEC values at 3 hourly intervals over a 12 hour period of the 'Halloween' 2003 storm are shown in Fig. 1.5. Ionospheric storms events, mentioned previously, are caused by Solar interaction and can have wide reaching consequences to communications systems both on Earth and satellite based. These data are typical of the type which are interpolated, or inverted in order to produce TEC maps. These maps may then be used to examine the behaviour of the ionosphere, and are of particular interest during storms.

As the number of paths between GPS ground stations and satellites is relatively low, producing TEC maps is an exercise in reconstruction from sparse data. Recent research has mainly focused on methods such as tomography that provide time-dependent volumetric reconstructions [Mitchell and Spencer, 2003, Pallares et al., 2005]. However, when the data points are too sparsely distributed, these techniques are under-constrained and do not produce meaningful results. In ionospheric studies, problems relating to sparsity are especially prevalent in historic data-sets. For example, in 1992 there were only 25 receiver sites operated by the International GNSS Service (IGS) in the USA [Brockmann and Gurtner, 1996], by 1996 there were over 75, and now there are over 500. Therefore, whilst the issues due to undersampling have largely disappeared for TEC imaging sys-



(a)

(b)

(c)

(d)

Figure 1.5: Example TEC input data for the cross-validation test in Section 3.3.2. All data are from October 30th, 2003 for the time periods (a) 1200-1215 UT, (b) 1600-1615 UT, (c) 2000-2015 UT and (d) 0000-0015 UT on the 31st.

tems utilising modern GPS data, they still remain for older data, and regularly arise in other geoscience applications [Liao et al., July 2007, Gianinetto and Villa, Oct. 2007]. Consequently, interpolation methods still have an important role to play in ionospheric studies. The most commonly used interpolation technique for TEC mapping studies is kriging [Blanch et al., 2002, Stanislawska et al., 2002, Wielgosz et al., 2003]. Although these studies have generally found kriging to perform satisfactorily, in other geophysical applications numerous problems with the kriging method have been reported [e.g. Philip and Watson, 1986]. In addition, there are many other interpolation methods for geophysical data that have received little recent attention from the ionospheric imaging community. Also, and perhaps more importantly, the relative performances of the different interpolation methods have not been assessed.

1.1.5 Tracking Storm Features

During storms, TEC maps can be considered to be composed of three main components: regions of high-electron density, noise and artefacts corresponding to other maxima (and minima), and the remainder of the image which can be considered background. This allows TEC maps to be used for the the examination of changes in electron density during geomagnetic storms: The most obvious high-values feature of note during large storms is the TOI. Tracking and analysing its motion allows the effects of a given storm to be evaluated, and helps to improve scientists' understanding of the behaviour of the Earth's atmosphere and magnetic field under varying conditions. Whilst features are known to move across the pole, *automated empirical* tracking which is not driven by models is an advancement not previously applied to TEC mapping.

In addition, tomography based literature is rich in information with large spatial extent and good representations of morphology, with only limited qualitative descriptions of TOI motion during storms. In contrast, data from sources such as superDARN [Greenwald et al., 1995], provides accurate quantitative motion vectors for specific patches of plasma, with a limited spatial extent.

The automated tracking methods shown here attempt to bridge the gap between these two fields by using tomographic or interpolation based imagery to provide quantitative motion analyses with a large spatial extent.

This thesis presents the results of applying several different motion estimation methods, for the first time, (described in Chapter 5) to ionospheric TEC data in Chapter 6.

1.2 Thesis Overview

The following three chapters of this thesis deal with interpolation, including an introduction to various methods, and two ways of evaluating performance.

Chapter 2 describes sparsity, and then introduces various interpolation methods, including normalised convolution (NC), triangulation based techniques, radial basis function interpolation and kriging.

Chapter 3 deals with how the relative performance of different interpolation methods can be examined in a quantitative fashion, some of these methods are then applied. The methodologies used in this chapter include simulation-validation, for evaluating interpolation on simulated data-fields, and cross-validation, for evaluation using real data.

The cross-validation method is applied to TEC data collected during the geomagnetic storm event that occurred on Halloween of 2003.

Chapter 4 moves on to analysing interpolation methods in a more qualitative fashion. It examines the various artefacts produced by different methods and discusses how error distributions can be examined to provide a wealth of information on how interpolation methods perform in specific cases.

The remaining chapters consider motion estimation techniques, starting with a description of the data used, in Chapter 5. These data consist of TEC maps of the northern polar ionosphere which were created from data collected during the 2003 Halloween Storm. It then discusses methods of motion estimation, including differential analysis, optical flow, template matching, correlation-relaxation labelling and boundary tracking. This chapter also introduces attribute mathematical morphology.

Chapter 6 is an initial study on the use of optical flow, template matching and correlation-relaxation labelling based motion estimation to track the motion of an area of enhanced electron density as it moves during the storm event.

Chapter 7 discusses the use of shape boundaries to infer motion. This chapter charts the development of a two-stage approach which makes use of mathematical morphology (MM) for segmentation, and shape context (SC) matching for motion estimation.

1.3 Synopsis

The aims of this thesis are twofold:

1. Examine the performance and specific nuances of various state-of-the-art interpolation methods, with specific application to mapping the ionosphere. To this end, interpolation methods are compared using a simulation methodology and a cross-validation based approach which operates using real data. This work forms the most widely scoped examination of interpolation techniques for TEC mapping to date, and the first application of *normalised convolution* to geophysical data.
2. Develop and examine automated *data-driven* motion estimation methods for use on TEC data. To this end, the results of applying several motion estimation methods to low-resolution TEC images are presented, including optical-flow, template-matching and boundary-correspondence based techniques. Where relevant, field entropy values are specified and reference is made to visual correspondence. This

is the first in-depth examination of automated tracking techniques to ionospheric storm features.

Chapter 2

Interpolation of Scattered Data

Scattered-data interpolation has been studied for many years and in many fields – because of this it has been given many names. The term scattered, for example, is also referred to as 'spatial', or 'multivariate', and the term interpolation is often called 'reconstruction' or, less formally, 'approximation'. An interesting history of interpolation from ancient times is provided by Meijering [2002]. Although its fundamental concepts do not differ, scattered data interpolation is a more recent development. In their study of the mathematical development of multivariate interpolation up to the second half of the 20th century, Gasca and Sauer cite the first modern literature on multivariate interpolation as the works of Borchardt and Kronecker, that appeared in 1860 and 1865 respectively [Gasca and Sauer, 2000].

This chapter considers techniques for interpolating scattered data. A great many different solutions to this fundamentally ill-posed problem have been proposed, and are routinely used. Although the only condition is that the fitted surface passes through the data points, all of these use weighted combinations of the input data to construct output values, Methods of weighting can range from the very simple to the highly elaborate, depending on the technique.

An example of a *simple* case is nearest-neighbour interpolation, which chooses the closest input datum as the output for any given point (essentially setting a weight of one to the nearest input datum, and zero for all others). Fig. 2.1 shows an example field of simulated data, visualised as an elevation map. This field was downsampled by randomly removing 99% of points, and then interpolated using nearest neighbour interpolation. This figure clearly shows the faceted appearance produced by this kind of interpolation. One of the main motivations for using and investigating different interpolation methods is improvement in output quality, and reduction of such artefacts.

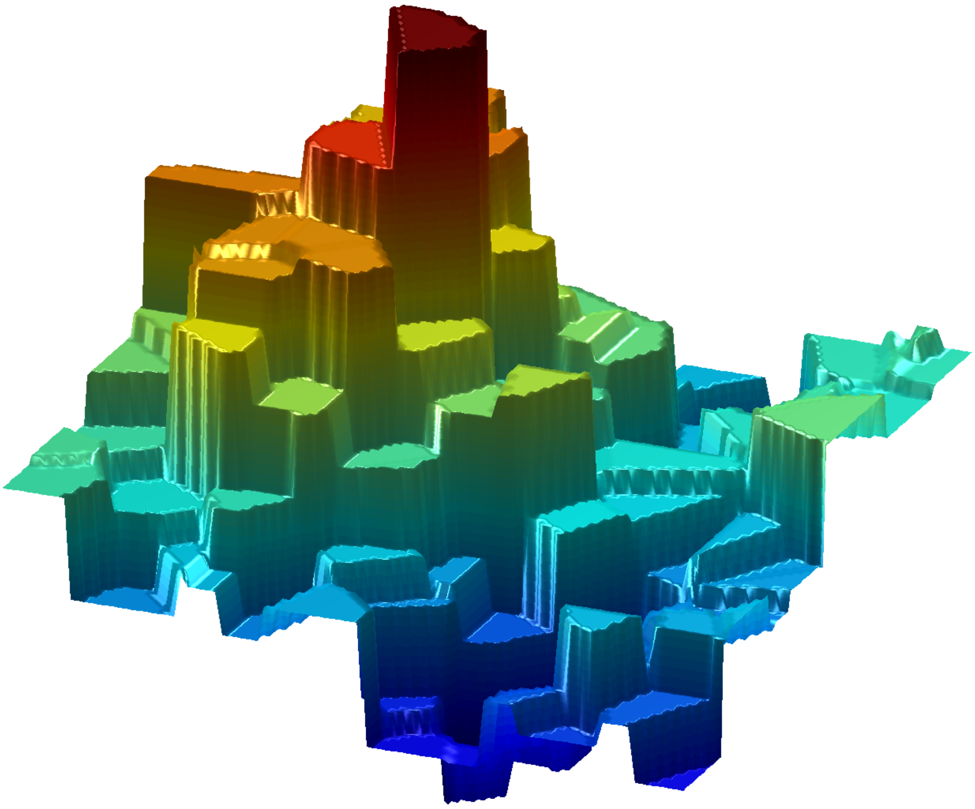


Figure 2.1: An example of nearest neighbour interpolation, illustrating the highly faceted appearance of its' outputs.

Examples of more *complex* cases, are weighing based on area of overlap between the cells of Voronoi diagrams, as used by natural-neighbour interpolation and the use of steered anisotropic Gaussian filters in sympathy with local image features to preserve edge information, as is used by adaptive normalised convolution (ANC) [Pham and van Vliet, 2003].

Interpolation methods differ in many other aspects, including:

- Philosophy: normalised convolution (NC) based techniques use the fact that the *absence* of data, and *zero-valued* data are very different situations to improve output quality of convolution based interpolation methods. In contrast, geostatistical methods, such as kriging use the idea that data can be decomposed into a stochastic part and an autocorrelated part, and estimate these during (or before) interpolation [Cressie, 1990, Trauth, 2006].
- Complexity: methods range in complexity from simple convolution or filtering, through triangulation and Voronoi diagram creation, to solving very large linear systems.

- Output quality: different methods rely on fitted curves and basis functions of differing order, which leads to a range of different qualities of output. Output quality is also heavily constrained by input sparsity.
- Continuity of derivatives: closely related to complexity and output quality is the idea that interpolation methods can try to increase smoothness by ensuring continuous derivatives to different levels. Higher levels of continuity increase complexity and require more input data to compute.
- Intended use: methods which are designed to allow sensitive analysis should try and minimise artefacts in the output, whereas methods designed for real-time operation might cut corners in order to speed up operation.

Interpolation methods can be divided into two categories, *local* and *global*, depending upon the locality of the points which are used to derive a given output point. Local techniques make use of a definition of *locality* to compute output values; only data which fall within a given point's local neighbourhood are used to calculate output values. Global techniques use a weighted sum of *all* data to compute output values and for large numbers of input points an approximation is generally used. When a new datum is added to a globally interpolated field the whole field must be re-calculated whereas, for a locally interpolated field, only those positions within the neighbourhood of the added datum need to be re-calculated. These two points tend to favour the use of local techniques, especially for interactive use.

Before discussing different interpolation methods in more detail, the meaning of scattered data, and how it arises must first be explained. Section 2.1 discusses scattered data and one very important property of scattered data which have been projected into matrices: *sparsity*.

2.1 Scattered Data

Scattered data are data which are spread throughout an object or medium being measured, such that the data may not cover every part of the object evenly. Whilst scattered data-sets often occur by design, it is far more common for them to arise when measurements are made opportunistically, or by proxy. Such data-sets are common in geoscientific research.

A good example of *opportunistic* measurements leading to scattered data is the use of global positioning system (GPS) receivers to calculate electron content along paths between receivers and satellites (see Section 1.1, [page 2]). The number of receivers is largely random (and depends on amongst other things, population density, affluence and the presence of tectonic activity, such as fault-lines). This means that at any given moment, the number of available paths can differ greatly over different areas of the globe, and that the positions of samples is by no means regular.

A good example of *proxy* measurements leading to scattered data is meteor radar measurements of mesospheric winds. These systems use the movement of the trails of ablating meteoroids as tracers for atmospheric motion. This leads to scattered measurements of the motion, as the positions of meteors entering the atmosphere is essentially random.

A good example of *intentionally* scattered data, is the use of methods such as stratified random sampling [Ripley, 2004]. Such schemes are used by scientists to help ease the process of data collection. This technique leads to fields of data where samples are randomly positioned throughout small areas, but the areas themselves are not random. An example where this kind of measurement system might be employed is in the measuring of tree growth [e.g. Ripley, 2004]. Other sampling methods are also available, and a large body of work from the 1950s and 60s deals with the variances introduced by these. See Ripley [2004], and references therein.

A specific case of scattered data considered in this thesis are measurements of electron content made along ray paths between GPS receivers and satellites. Data derived from such measurements can be used for mapping and profiling of the ionosphere.

2.1.1 Sparsity

In terms of a matrix, sparsity, or sparseness, is a measure of how empty it is. Input data sparsity has a large effect on the fidelity of interpolated images. Karvanen and Cichocki [2003] define sparseness as the ℓ_0 norm, divided by the number of elements in the data-set. This is simply the proportion of non-zero values, as the ℓ_0 is defined as:

$$||\mathbf{x}||_0 = \frac{\#\left\{j, x_j \neq 0\right\}}{N} \tag{2.1}$$

where $\mathbf{x}$ is the matrix, with N elements and $\#$ is a function which counts the number of times its content evaluates to *true*. This definition leads to an entirely empty matrix having a sparseness of zero, and a full one having a sparseness of one. As this is in opposition to the definition, this thesis uses $1-||\mathbf{x}||_0$ instead, and uses the term *sparsity*, as opposed to sparseness, for clarity.

Sparsity is only defined for data in matrices, as the notion of ‘elements’ or ‘pixels’ does not exist for scattered data. The sparsity of a given data-set is therefore related to the resolution of the matrix into which it has been projected. Fig. 2.2 demonstrates how changing the resolution of a matrix affects the sparsity, by showing the projection of a small set of scattered data into matrices with various resolutions. The sparsity can be seen to increase as the resolution of the matrix is increased.

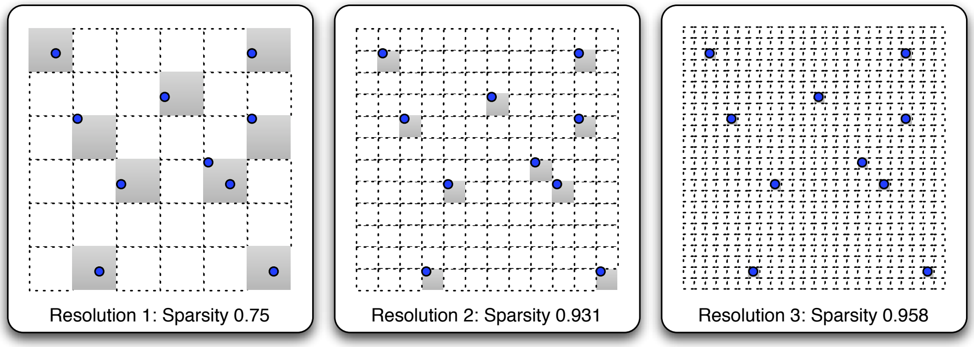

Resolution 1: Sparsity 0.75
Resolution 2: Sparsity 0.931
Resolution 3: Sparsity 0.958

Figure 2.2: The effect that projecting scattered data into different resolution matrices has on sparsity. The small circles represent data points, and grey squares represent matrix elements which take the values of the data.

The effect of sparsity on output errors is discussed in a detailed study applied to both simulated and real data in Chapter 3.

The remainder of this chapter introduces various state-of-the-art and common methods for interpolating scattered data. These include methods based on triangulation, radial basis function (RBF) methods [Carr et al., 1997] and kriging [e.g. Cressie, 1990, Trauth, 2006], as well as a relatively new technique, known as normalised convolution (NC).

2.2 Triangulation Based Interpolation

Triangulations are often used as the basis for interpolation of irregular data – the Delaunay triangulation in particular has some properties which make it particularly useful, and so most triangulations are of this type [e.g. Sugihara et al., 2000].

The result of a Delaunay triangulation is a set of vertices and edges (a graph) which conveniently defines the convex-hull of the data. It also gives a simple definition of locality. The local neighbours of any given point are those which are connected to it by triangulation edges.

The geometric dual of the Delaunay triangulation, known as the Voronoi diagram, also has many useful properties. Its edges form cells around each point, such that the area within each cell is closest to the point at its centre. Cells at the edges of Voronoi diagrams are unbounded. Voronoi edges are always perpendicular to Delaunay edges.

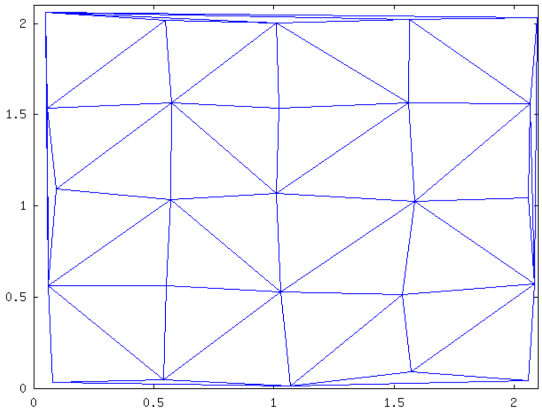


Figure 2.3: An example Delaunay triangulated data, showing how datum may be connected to multiple neighbours.

Once a data-set has been triangulated, each point in the data-set will be connected to several others by triangle vertices (See Fig. 2.3). Given the values of a triangle's nodes (f_i, where $i = 1, 2, 3$), the interpolated value of any point within the triangle, can be found using

$$f(x, y) = \sum_{i=1}^{3} \phi_i(x, y) f_i \tag{2.2}$$

where $\phi_i(\mathbf{x})$ is the interpolating *basis function*, which weights the contributions of the inputs [Watson and Philip, 1984, 1987, Sambridge et al., 1995, Sugihara et al., 2000]. For a simple case, linear interpolation, the basis function can be replaced by a simple first order polynomial,

$$f(x, y) = c_1 x + c_2 y + c_3. \tag{2.3}$$

The coefficients $\mathbf{c} = (c_1, c_2, c_3)$ can then be found by solving $\mathbf{Ac} = \mathbf{f}$ where $\mathbf{f} =$

$(f_1, f_2, f_3)^T$ and $\mathbf{A}$ is a 3×3 matrix of rows with the form $(x_i, y_i, 1)$, where i is the row number.

Higher (and lower) order basis functions can also be used, but require larger numbers of input samples. Zero order triangulation-based interpolation is known as nearest neighbour interpolation, and leads to regions of output values in direct relation to Voronoi cells. Other commonly used schemes include quadratic and cubic interpolation. This process can also be generalised to higher-dimensions, and various implementations are readily available [Barber et al., 1996].

2.3 Natural Neighbour Interpolation

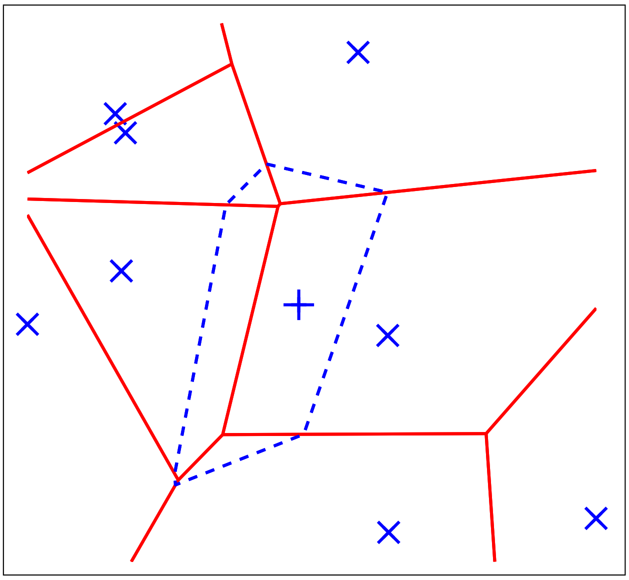


Figure 2.4: Voronoi diagram showing new cell (dotted line) overlapping cells from the original tesselation.

Watson [1985] defines natural neighbours as

> “points which share a common interface, or region, that is equally close to each of the pair, and all other neighbours are no closer”

This means that if circles (or n-spheres in n dimensions) are drawn such that their circumferences pass though $n+1$ or more data points, no data points will be within any of the n-spheres. This is related to the Delaunay triangulation which can be formed by linking the data at points which are on the circumference of each n-sphere. The Delaunay triangulation is not unique when more than $n+1$ points lie on a sphere edge (i.e. when four or more points are coplanar in 2-D).

Once the natural neighbours have been established, the interpolated output value at any point can be determined using a weighted sum of the values at its neighbours. The way in which the weights are determined is best described in terms of Voronoi tessellations, the geometric dual of the Delaunay triangulation, see Fig. 2.4. This method was first suggested by Sibson [1981], Sambridge et al. [1995], Park et al. [2006], Fan et al. [2005], Watson [1985], from the University of Bath, UK, and is sometimes known as Sibson interpolation

For each point where a value is required:

1. Assume the data are already tessellated
2. Re-tessellate the data to include the output point. This adds a new Voronoi cell which overlaps the cells of the natural neighbours of the output point.
3. The contribution from each neighbour is given by the ratio of the area of overlap to the total area of the new cell. These ratios form the basis function $\phi_i(x, y)$ in (2.2).

In terms of the Delaunay triangulation, the basis function $\phi_i(x, y)$ is only non-zero within the circum-circles which pass through the natural neighbour nodes. This means that the operation is local, in the sense that only neighbouring values are used in the interpolation.

2.4 Radial Basis Function Interpolation

RBF interpolation approximates a field of data using a weighted sum of radially symmetrical functions, known as basis functions [e.g. Carr et al., 1997]. One basis function is centred on each input sample, so that any given output point is composed of contributions from each input point. RBF interpolation is therefore considered a global technique. The output at any given point $\mathbf{x}$ is given by

$$f(\mathbf{x}) = p_m(\mathbf{x}) + \sum_{j=1}^{N} \lambda_j \phi(||\mathbf{x} - \mathbf{x}_j||) \tag{2.4}$$

where p_m is a low-order polynomial surface with coefficients $c_0, c_1, \cdots, c_n$ which has been fitted to the data and is only used during linear and thin-plate spline interpolation, ϕ is the basis function whose form is fixed across the field and λ_i is the weight for input $\mathbf{x}_i$. Many different basis functions can be used, with differing performance and order of continuity. Some common functions are shown in Table 2.1 [e.g. Light, 1992, Powell, 1990].

Table 2.1: Example Radial Basis Functions

Name	Equation
Linear	$\phi(r) = r$
Thin-plate Spline	$\phi(r) = r^2 \log r$
Multiquadratic	$\phi(r) = (r^2 + c^2)^{0.5}$
Inverse Multiquadratic	$\phi(r) = (r^2 + c^2)^{-0.5}$
Gaussian	$\phi(r) = e^{-ar^2}$
Biharmonic Spline	$\phi(r) = \lvert r\rvert^2(\log \lvert r\rvert - 1)$

To find values for the weights λ and coefficients c, the linear system

$$\begin{bmatrix} A & P \\ P^T & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ c \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}, \tag{2.5}$$

must be solved, where A is a matrix composed of evaluated basis function values for every possible pair of input values, P is a matrix of homogeneous input coordinates (coordinates with leading ones),

$$P = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & y_n \end{bmatrix}, \tag{2.6}$$

and $[\lambda \; c]^T$ and $f = (f_1, f_2, \ldots, f_n)^T$ are column vectors of weights and input values respectively.

When a polynomial is not being fitted, the output system reduces to

$$A\lambda = f. \tag{2.7}$$

The calculation of the matrix A and solving the linear system described by (2.5) are computationally expensive operations and this has motivated work aimed at decreasing the overall complexity, using techniques such as domain decomposition [e.g. Beatson et al., 2001]. In addition to the basis functions given in Table 2.1, some other interpolation methods can be formulated in terms of RBF interpolation. A well-known example is biharmonic spline interpolation and this is described in more detail below.

2.4.1 Biharmonic Spline Interpolation

A special case of RBF interpolation is biharmonic spline interpolation (BSI). This is a method of finding the minimum curvature surface which passes through a set of points [Sandwell, 1987]. In practice, this method produces results which are indistinguishable to RBF interpolation using thin plate spline (TPS). BSI can be numerically unstable for large numbers of points and, like other cubic methods, has a tendency to drastically overshoot when points are close together. This problem occurs because of the imposed continuity in the surface's derivatives, which makes smoothly varying curves preferable around data points. Therefore, BSI is better suited to the interpolation of highly sparse data. The phenomenon of overshooting is discussed in more detail in Chapter 4.

2.5 Kriging

Kriging was first suggested in the 1960s by D. G. Krige, a South African mining engineer. It was originally developed as a technique for estimating yields of ore deposits from sparsely distributed core samples, but has now been widely applied to many different fields and scenarios (for example, mining, mathematics and classification Boucher et al. [e.g. 2006]), as discussed in Cressie [1990]. One of the main attractions of kriging is its ability to provide a variance estimate for each output point. Kriging and all geostatistical methods operate under the assumption that a process being interpolated or analysed consists of a *stochastic part* and an underlying trend [Matheron, 1973]. The trend may consist of both local and global components. This is Matheron's 'Theory of Regionalized Variables' [Matheron, 1971]. The stochastic component is comprised of both random and autocorrelated parts, where the degree of autocorrelation is a function of distance. This means that points in close proximity are more closely correlated than distant ones. The method considered here is known as *ordinary kriging*, and is designed to work with Guassian data.

The first step in interpolation using kriging is the formation of a semivariogram [e.g. Omre, 1984, Cressie, 1991]. This is a diagram of the spatial dependence of samples and is a function of all possible separations (or lags) and semivariance. The semivariance is defined by

$$\gamma(\mathbf{h}) = 0.5(f(\mathbf{x}) - f(\mathbf{x}+\mathbf{h}))^2, \tag{2.8}$$

where $f(\mathbf{x})$ contains the point values at a given location $(\mathbf{x})$ and $f(\mathbf{x}+\mathbf{h})$ is the point value at a point separated from $\mathbf{x}$ by the lag vector $\mathbf{h}$. For the isotropic two-dimensional case, it is simplest to calculate lags by using a distance metric between points. However,

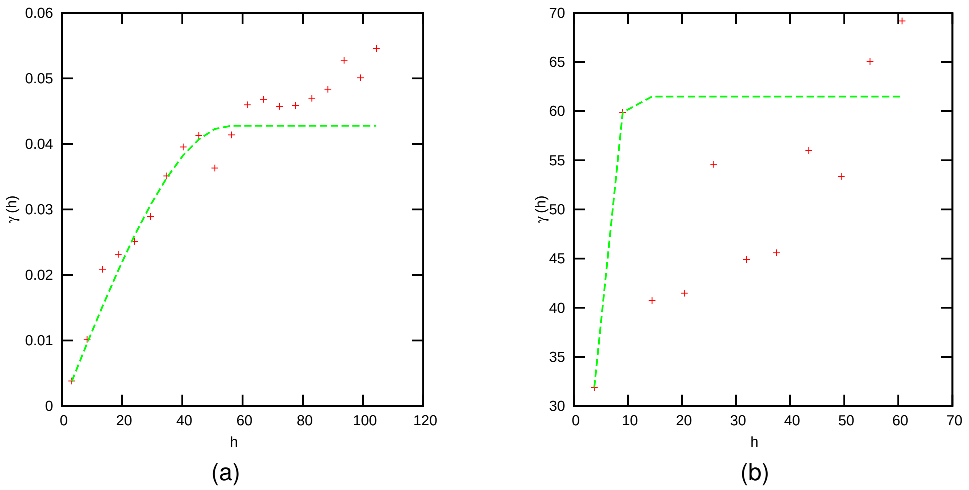


Figure 2.5: Semivariograms (+) and automatically fitted spherical model of (2.9) (dashed lines). (a) Simulated correlated data demonstrating a good fit and (b) degenerate TEC data showing a poor fit.

the number of combinations of sample positions which must be compared quickly becomes very large. For this reason, input coordinates are often binned to reduce the total number of lags.

After the semivariogram has been created, a model must be fitted to it. One example, the spherical semivariogram model is a curve of the form

$$\gamma_{\text{sph}} = \begin{cases} c\left(1.5\left(\frac{h}{a}\right) - 0.5\left(\frac{h}{a}\right)^3\right) & \text{for } 0 \leq h \leq a \\ c & \text{for } h > a, \end{cases} \tag{2.9}$$

where, in geostatistical parlance, c is the 'sill', the value that the semivariogram reaches after its initial rise, and a is the 'range' or length of spatial dependency. This sill is generally close to the variance of the input values. Details of this and other commonly used models can be found in Cressie [1991] and Trauth [2006]. Fig. 2.5a shows a typical semivariogram with a fitted spherical model.

The next, and final, step is the actual kriging process. Kriging uses a weighted average of input points to estimate the value any given output point. The weights are found by minimising the *kriging variance* – the difference between the estimate and the actual input value. An output variance is also directly calculable. As it minimises the variance of the output, kriging is often called the 'best linear unbiased estimator'. However, the variance which is minimised is relative to the semivariogram model, so the results will only ever be as good as the model and, therefore, the semivariogram. For a good, concise description of the kriging process, see Blanch et al. [2002]. It should also be noted that other authors have heavily criticised some of the underlying assumptions behind geostatistics and these issues should be borne in mind when using kriging and related techniques [e.g. Philip and Watson, 1986].

2.6 Normalised Convolution

2.6.1 Introduction

NC techniques are a class of methods which make use of convolution operations for the efficient interpolation and regularisation of image data; that is, data projected into matrices. They differ from other techniques in the fact that they make an implicit distinction between zero-valued and unavailable samples.

Since their initial proposal by Knutsson and Westin [1993], NC techniques have been steadily gaining in popularity, and have been applied to medical imaging see [e.g. Estepar et al., 2003], tensor field regularisation [Westin and Knutsson, 2003], motion compensation [Farneback, 2002], irregular-data fusion [Pham, 2006], and interpolation [Pham and van Vliet, 2003]. This work represents the first foray into the application of NC techniques to the reconstruction of geophysical data.

The most basic form of NC, known as zero-order NC is described by the following equation:

$$f = \frac{f_i \otimes g}{c_i \otimes g} \tag{2.10}$$

where f_i is the input data, c_i is a binary map of input data positions – the 'confidence map', g is a kernel function and $\otimes$ denotes the convolution operator. NC is therefore implemented using simple filtering operations, and so can be very computationally efficient. The term 'normalised' comes from the division by the denominator, which serves to normalise the filtered input data.

2.6.2 Zero-order Normalised Convolution

Zero-order NC is similar in output to linear interpolation. The input image is convolved with a suitably sized filter kernel, in order to determine the contribution to the output by different input data. A confidence map, describing the positions of input points is then convolved with the same filter, in order to give the contribution of the filter kernel to each point in the filtered input. The filtered input is then divided by the filtered confidence map, removing the contribution from the filter, and leaving the interpolated output.

An alternative explanation is that NC produces a local model of the input data using projections onto a set of basis functions. The locality comes from the kernel at each pixel, and the basis function is a polynomial whose order is generally less than two, and is usually zero.

Filters used for NC are usually Gaussians [e.g. Pham and van Vliet, 2003, Pham, 2006], although various other kernels have been used. For example, initial studies [Knutsson and Westin, 1993] used a raised cosine.

Simple zero-order NC is highly efficient, only requiring operations that can be executed extremely quickly with modern computing hardware. It is also intuitively simple, and requires no triangulation or calculation of derivatives. However, filter sizes must be chosen such that they are as small as possible whilst encompassing all input data, or gaps will appear in the output. Using a filter which is larger than necessary will result in over-smoothing in areas where data are closely spaced. In effect, this is discarding data, a proposition which seems counter-intuitive when dealing with irregularly sampled data.

In order to improve output quality of zero-order NC, the filter size can be adapted, and a suitable filter size for each pixel chosen. The simplest possible adaptation of this kind is to vary the filter radii so that the filter used at any given point is related to the distance to the nearest sample. This, and other adaptations are known as ANC.

2.6.3 Higher Order Normalised Convolution

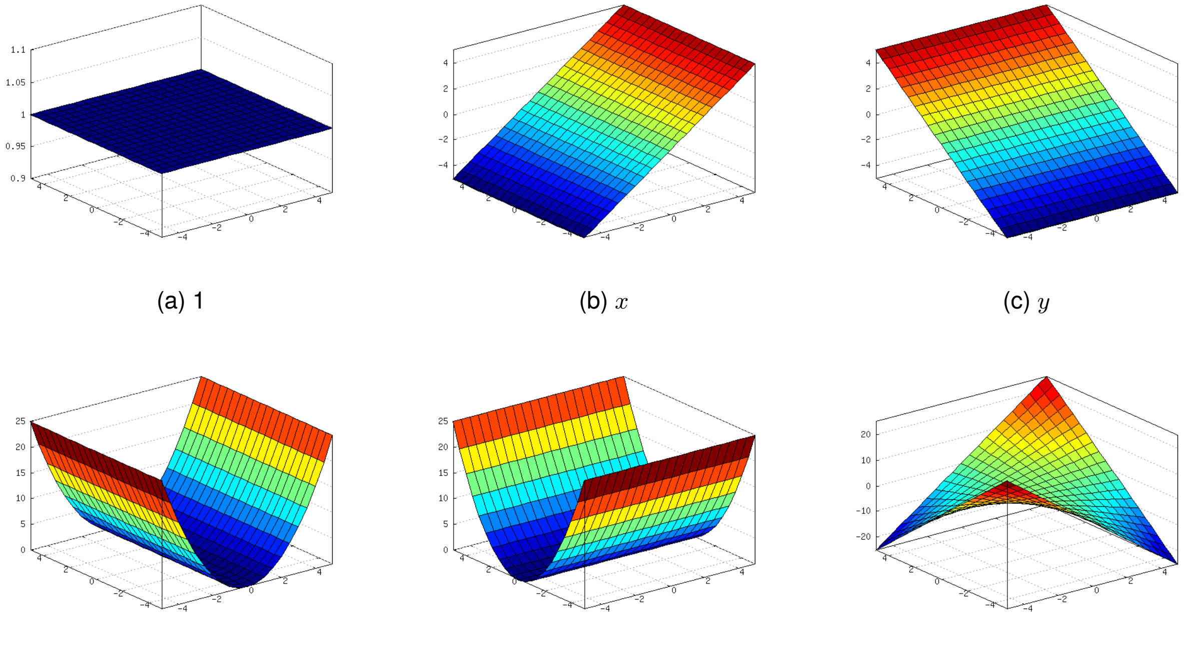
So far, discussion has been limited to NC using constant bases. This section discusses higher order NC, where projection onto higher order bases is carried out, using matrix inversions. Due to the irregularity of sample positions, inversions must be carried out at every output position. For this reason, first- and higher-order NC methods are more computationally expensive than zero-order NC.

First order NC is the simplest improvement over zero-order NC, and uses bi-quadratic basis functions, where zero-order NC uses a constant. Nine-convolutions are required in all: six for the numerator, and three for the denominator. The following equation describes the process mathematically:

$$\begin{bmatrix} f_1 \\ f_x \\ f_y \end{bmatrix} = \left(\begin{bmatrix} g & g.x & g.y \\ g \cdot x & g \cdot x^2 & g \cdot xy \\ g \cdot y & g \cdot xy & g \cdot y^2 \end{bmatrix} \otimes c_i \right)^{-1} \times \left(\begin{bmatrix} g \\ g \cdot x \\ g \cdot y \end{bmatrix} \otimes (c_i \cdot f_i) \right), \tag{2.11}$$

where g, $g \cdot x$, $g \cdot y$, $g \cdot x^2$, $g \cdot xy$, and $g \cdot y^2$ are filter kernels (g) that have been multiplied by surfaces of various orders, this can be seen in Fig. 2.6. The form of (2.11) is similar to (2.10), but involves an inversion, instead of a division, because of the introduction of multiple basis functions. It also outputs f_x and f_y, the first derivatives in the x- and y-directions. The overall process of first-order NC is therefore one of fitting basis functions, and then normalising the output.

Orders of higher than one are possible, but are again more computationally expensive. However, second-order NC would yield second derivatives, which could be useful in some situations, such as edge detection methods using zero crossings.



(a) 1

(b) x

(c) y

(d) x^2

(e) y^2

(f) $x \cdot y$

Figure 2.6: Polynomial basis functions as used by first-order NC

First-order NC provides little improvement over zero-order NC, both of which exhibit performance similar to linear interpolation (but without a faceted appearance) when filters are sized such that there are no output gaps. The main advantage is the fact that gradients are returned in addition to the interpolated image.

First order NC is more computationally expensive that zero-order NC because of the use of multiple basis functions, and matrix inversions. However, in general first-order NC is more sensitive to filter size than zero-order NC. This is illustrated in Fig. 2.8, which shows the root mean square error (RMSE) associated with reconstructing the crane image (Fig. 2.7. A strong negative gradient is visible in the left-hand section of Fig. 2.8b. This section coincides with small filter sizes, demonstrating its high sensitivity to changes in the size of small filters.

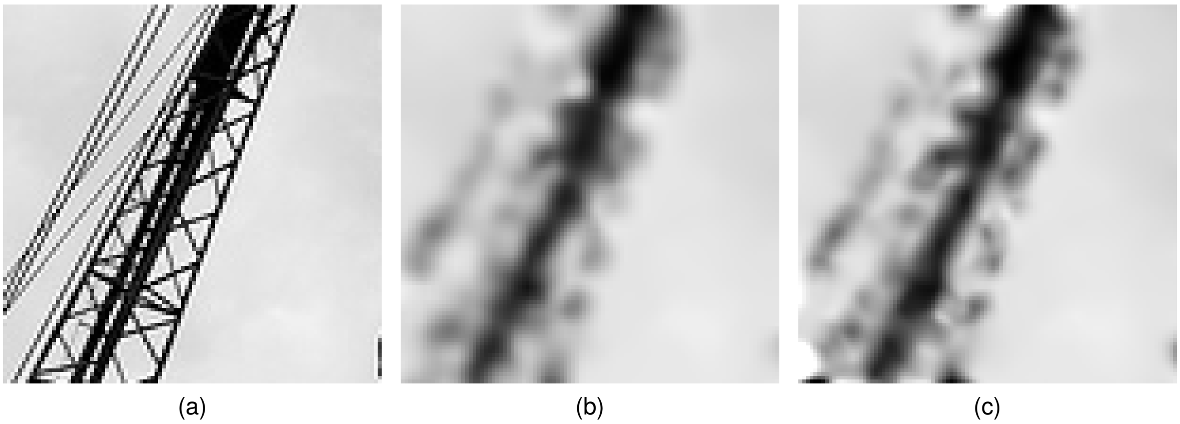
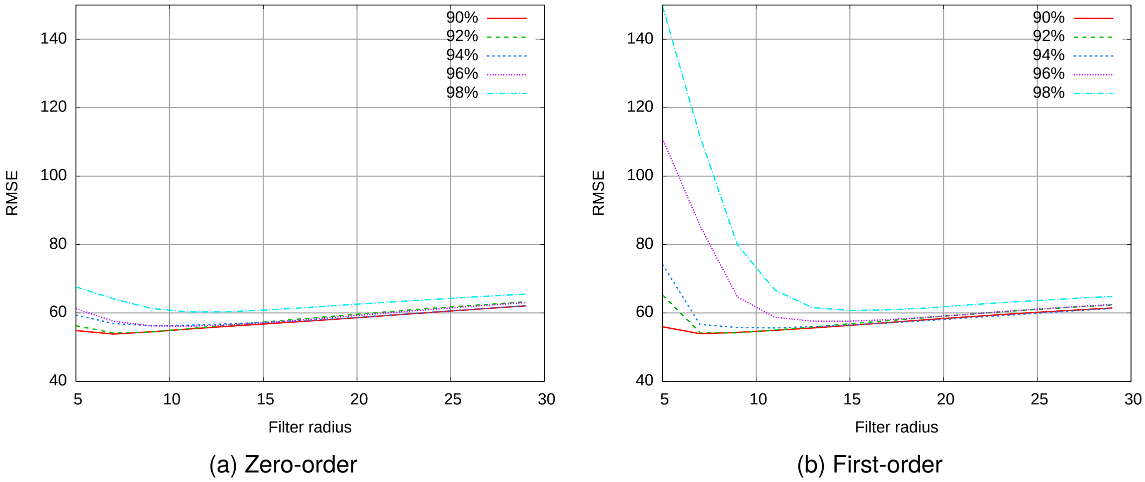


Figure 2.7: (a) Small section of an image of a dockside crane. This was sampled to $\approx 90\%$ sparsity, and reconstructed using (b) zero-order NC, and (c) first-order NC.



(a) Zero-order

(b) First-order

Figure 2.8: Graphs showing RMSE of reconstructions of the crane image from various sparsities and varying filter dimension, using (a) zero- and (b) first-order NC.

2.6.4 Zero Order Adaptive Normalised Convolution

The most simple form of ANC, known as *size adaptive* NC adapts the filters such that the kernel standard deviation at any given point is specified by the Euclidean distance transform [e.g. Hlavac et al., 1999] (or other distance transform) of the input samples.

Alternatively, a more computationally expensive scale-space based method can be used [e.g. Pham, 2006]. This method makes use of a scale-space pyramid to allow estimation of the filter dimension at which a given certainty constant (C) is found. The pyramid is constructed by filtering the confidence map with non-normalised Gaussians of exponentially increasing scales ($\sigma_i = 2^i$, i = -1,0,1,2. . .). Recursive Gaussian filtering [Young and van Vliet, 1995] can be for high-speed operation. The pyramid values at each scale will then increase quadratically, due to the non-normalised filters, and a quadratic regression can be used to estimate the scale at which the desired certainty constant is found. This scale is then used to set the filter sizes at each point, to minimise smoothing effects.

Unfortunately, size adaptation alone does not provide much of an improvement over simple zero-order NC, and in order to further boost performance, alternative methods of adaptation are needed.

As well as adjusting the filter sizes in sympathy with input sample positions, the filters can be adapted according to properties of the image being reconstructed. Using 2-D Gaussians gives three variable parameters, the standard deviations along two axes and filter orientation. By lining filters up with the edges and other features, the output image should contain more similar properties to the input image. However, this requires properties of the sampled image (specifically edge information) to be known, and in non-simulated situations original input images are very rarely available.

For this reason, there are methods of gradient calculation which are based on NC, and are designed to proved edge information in irregularly sampled images. These are described in the following section.

2.6.5 Sparse Data Gradient Estimation

Estimation of gradient in irregularly sampled images requires the image to be interpolated and then differentiated, usually along both the x- and y-axes.

There are two NC based methods of estimating gradient. This first, *first-order NC* was described in section 2.6.3 [Pham, 2006], and produces gradients as a product of its normal process. The second, *differential of normalised convolution (DoNC)* uses differentiated zero-order NC, and is described below.

Differential of Normalised Convolution

The DoNC method is formed by applying derivative operators to the NC equation (2.10), in two directions. This gives the following equations for the x-axis Westin and Knutsson [2003], Piroddi and Petrou [2004]:

$$\Delta_x\left(\frac{D(x,y)}{N(x,y)}\right) \equiv \frac{D_x(x,y)\times N(x,y) - N_x(x,y)\times D(x,y)}{N^2(x,y)}, \qquad (2.12)$$

where:

$$D_x(x, y) = x.g(x, y) \otimes f_i(x, y), \tag{2.13}$$

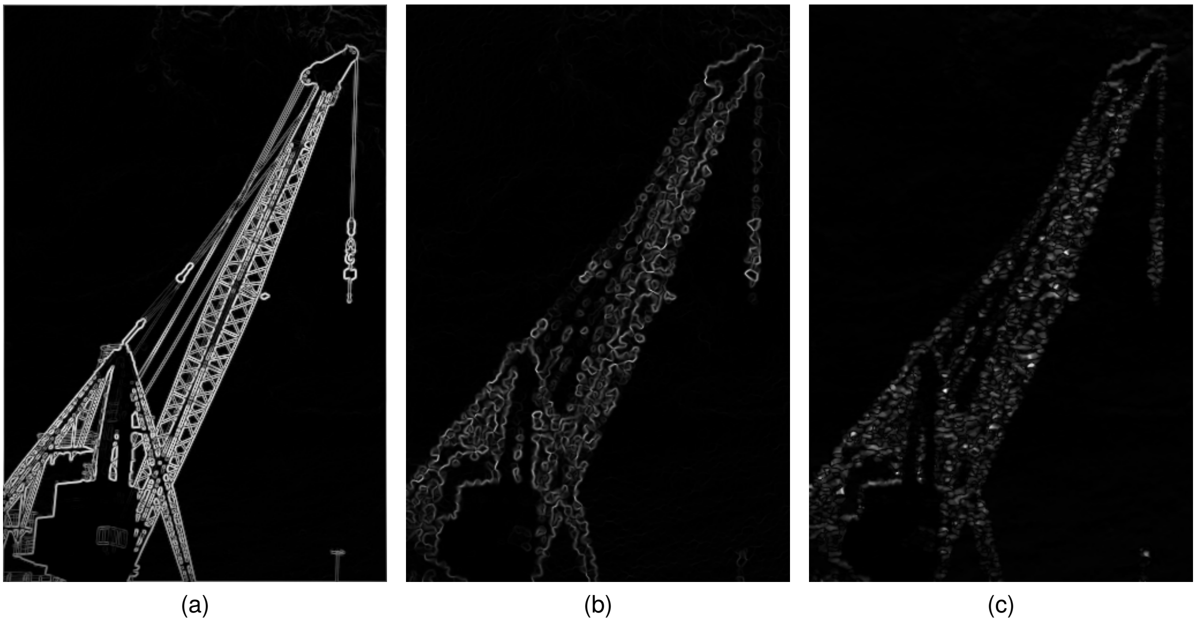
and:

$$N_x(x, y) = x.g(x, y) \otimes c_i(x, y). \tag{2.14}$$

In the (2.13) and (2.14), $x \cdot g(x, y)$ is an edge enhancement filter which could be any arbitrary (normally Gaussian) filter multiplied by a variable x. This effectively tilts the filter relative to the x-axis.

The same process is also extended to the y-axis, and both processes are applied to sampled inputs, they can be used to provide the edge vector $[\Delta_x, \Delta_y]^T$.

As discussed above, the filters used for DoNC should be chosen such that there are no gaps in the output, but whilst attempting to minimise over-smoothing. Fig. 2.9b shows an example of DoNC and first order NC applied to an image sampled to $\approx 90\%$ sparsity, both used the same size Gaussian filters (11 with $\sigma \approx 3$). For comparison, the Sobel operator has also been applied to the original image.


(a)

(b)

(c)

Figure 2.9: Images illustrating the difference between (a) Sobel edge detection on an unsampled original image, (b) DoNC and (c) first-order NC. Both images were reconstructed using 11 filters with $\sigma \approx 3$ from an image from which 90% of samples had been removed.

Gradients from First-order NC

Gradients calculated using first-order NC are similar to those produced by DoNC, but can be slightly more accurate for some images, such as those with a large amount of high-frequency content [de Jong et al., 1998]. The lower computational complexity, and similar performance of DoNC suggest that it is a sensible choice in practical situations where speed is important.

2.6.6 Adaptive Normalised Convolution

Once estimates of image gradients are available, they can be analysed in order to provide information on image structure, including local energies, orientations and anisotropies. These properties describe gradient, edge direction and how non-uniformly varying a give image is.

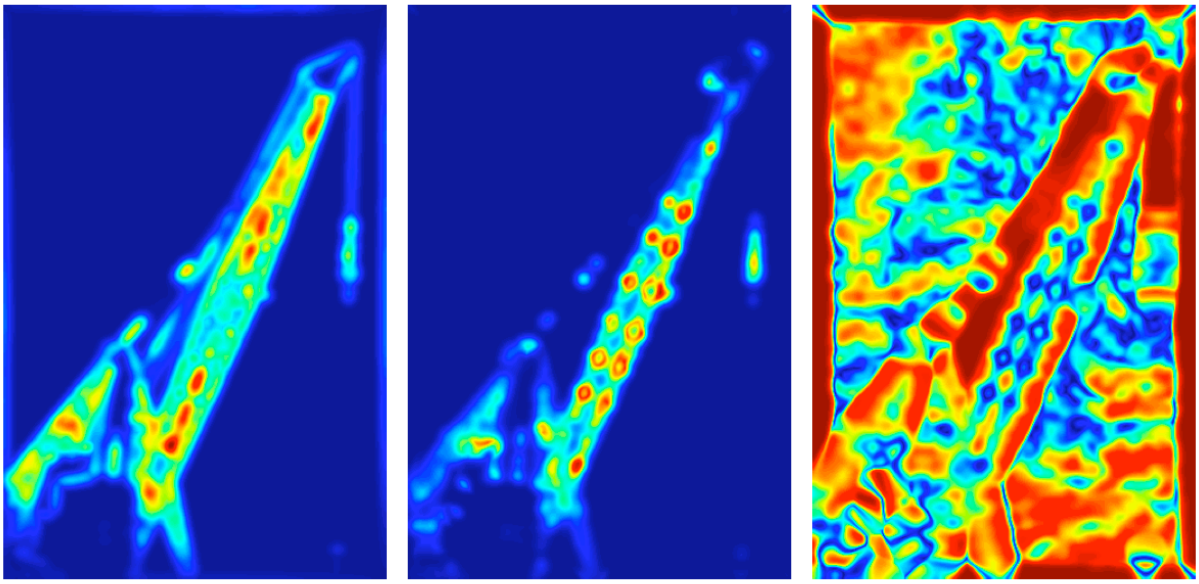
These properties can then be used to set the size and orientation of the NC filters in order to maintain features, and high frequency detail, whilst ensuring that no gaps appear in the output. This process has been used in the past by Nitzberg and Shiota [1992], Almansa and Lindeberg [2000], and specifically adapted for use in ANC by Pham and van Vliet [2003].

Kass and Witkin [1987] describe that fact that anisotropy can be detected by examining local power spectra, and noting that high-frequency energy will tend to lie perpendicular to the direction of flow. It is then suggested that orientation-selective linear filters can be used to detect this clustering energy, and hence the anisotropy of the local area. van Vliet and Verbeek [1995] develop this idea by using smooth local-gradient measures to form matrices known as gradient square tensors (GST), which are approximately equivalent to covariance matrices of the gradients, and are given by:

$$GST = \begin{bmatrix} g_x^2 & g_{xy} \\ g_{xy} & g_y^2 \end{bmatrix}. \tag{2.15}$$

The effect of smoothing is the localisation of the matrix, so that when its eigenvectors are calculated, they correspond to the local area. Each pixel will have an associated GST, and calculating these gives the various pieces of information, which are summarised in Table 2.2.

The two most useful of these products are the anisotropy (A, see Table 2.2 and Fig. 2.10),



(a) λ_1 (b) λ_2 (c) A

Figure 2.10: Eigenvalues and anisotropy. (a) and (b) show the largest and smallest eigenvalues of the crane image and (c) the anisotropy. Images were generated using smoothed gradient vectors from Sobel operators.

λ_1	largest eigenvalue
λ_2	smallest eigenvalue
$A = 1 - \frac{\lambda_2}{\lambda_1}$	anisotropy (consistency of local orientation)
φ_1	local gradient direction (direction associated with λ_1)
φ_2	local orientation (direction associated with λ_2)
$\lambda_1 + \lambda_2$	local energy

Table 2.2: GST Products

and φ_2, the ‘local direction’, which is given by:

$$\varphi_2 = tan^{-1}\left(\frac{g_{xy}}{\lambda_2 - g_y^2}\right). \tag{2.16}$$

These measures are then used to set the size of the filters at any given point using:

$$\sigma_u = C(1-A)^{\alpha}\sigma_a \tag{2.17}$$

and

$$\sigma_v = C(1+A)^{\alpha}\sigma_a \tag{2.18}$$

where u and v denote filter axes, and σ_u and σ_v correspond to Gaussian standard deviations along those axes. These three parameters $(\sigma_u, \sigma_v, \varphi_2)$ can then be used to set the size and rotation of the filter used at any given output point Fig. 2.11 shows how these parameters relate to a rotated Gaussian kernel.

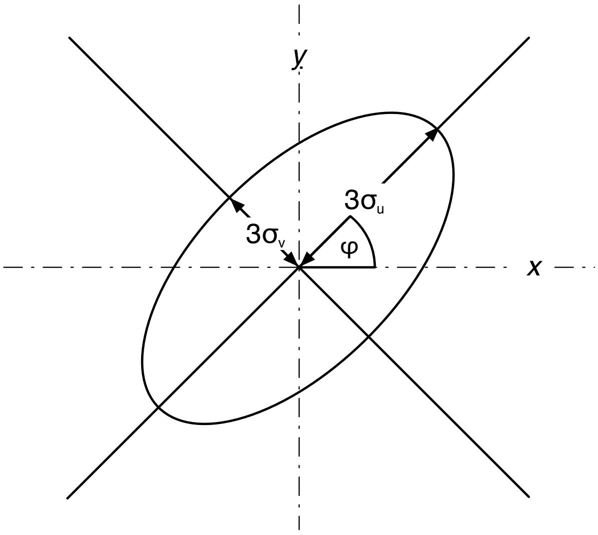


Figure 2.11: A schematic diagram showing shape, and salient parameters of a rotated 2-D Gaussian. The oval indicates 3σ confidence limits in each axis, which corresponds to a 99.7% confidence interval, and is a good place to truncate filters.

The GST process can also be used to extract information on local curvature, which can then be used to warp filters [van Ginkel et al., 1999].

2.6.7 Implementation

Various points in the above processes require filtering with Gaussian kernels for either smoothing, or in the case of NC localisation of output contributions. Multiple filtering stages can quickly lead to high computation times, making it prudent to investigate methods of speeding up these processes.

Two types of filtering process are heavily used by ANC. These are isotropic and anisotropic filtering, and both have their own sets of speed improvements. These are dealt with separately below.

Isotropic Filtering

Generation of 2-D Gaussian filter kernels can be sped up appreciably by using the property of separability, which allows symmetrical functions to be split down into orthogonal components. This is possible when a filter kernel can be written as the output product of two vectors. For example, consider the Sobel kernel:

$$\begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{2.19}$$

Using this, one can quickly generate 2-D Gaussian filters by convolving two vectors:

$$g(u, v; \sigma_u, \sigma_v) = \frac{1}{\sqrt{2\pi}\sigma_u} exp\left\{-\frac{1}{2}\frac{u^2}{\sigma_u^2}\right\} \otimes \frac{1}{\sqrt{2\pi}\sigma_v} exp\left\{-\frac{1}{2}\frac{v^2}{\sigma_v^2}\right\} \tag{2.20}$$

Alternatively, since convolution is associative, this property can be used to quickly filter images by using two 1-D processes, instead of a single 2-D process.

The image is first filtered in one dimension, using one kernel, to give an intermediate image. This image is then filtered in the other dimension, using the other kernel. The output is exactly the same as if a more complex 2-D filtering process had been used, but is much faster thanks to the lower complexity.

Complexity of filtering operations which can be decomposed drops from M^2P^2 multiplies and additions to $M^2(2P)$. Where M is the image dimension, and P is the filter dimension. Therefore, the relative speedup is a factor of $\frac{M^2}{2P}$.

A great many filters can be separated into orthogonal components, using the following algorithm:

1. Perform the singular value decomposition (SVD) of the filter kernel. This will give three values: U (output basis vectors), S (singular values) and V (input basis vectors). The can be recombined to reconstruct the original kernel using:

   $$K = USV^* \tag{2.21}$$

   Where $*$ denotes the conjugate matrix transpose.

2. Take the rank of the singular values. This corresponds to the number of independent rows in S. If rank$(S) = 1$, then the filter kernel is separable.

3. Two separate the kernel, two 1-D vectors are formed by multiplying U and V by the square root of the non-zero value from S:

   $$\begin{aligned} K_u &= U\sqrt{s}, \\ K_v &= V\sqrt{s}. \end{aligned} \tag{2.22}$$

Gaussian filtering can also be carried out using recursive or infinite impulse response (IIR) filters, for further speed improvements [Young and van Vliet, 1995].

Anisotropic Filtering

Filtering images using steered, anisotropic filters is a computationally expensive operation, as convolution with a different filter kernel is required at each output point. As with isotropic filtering, speed improvements can be made to anisotropic filters by using approximate filter separability, as described by Geusebroek et al. [2002].

This process uses two filtering stages, one parallel to the x-axis, and another along the direction of rotation. The first filter stage uses the following filter:

$$g_x(x, y) = w_o f(x, y) + \sum_{i=1}^{\lfloor N/2 \rfloor} w_i (f(x - i, y) + f(x_i, y)) \tag{2.23}$$

Where N is the size of the Gaussian filter, whose weights are given by $w_o \dots w_N$. The filter standard deviation is:

$$\sigma_x = \frac{\sigma_u \sigma_v}{\sqrt{\sigma_u^2 \cos^2 \theta \sigma_v^2 \sin^2 \theta}} \tag{2.24}$$

In this equation, θ is the filter orientation and σ_u and σ_v are the standard deviations of the filter before rotation. The second pass of the filter operates on the intermediate output of the first, using linear interpolation to calculate off-grid values, and has the following form:

$$\begin{aligned} g_\theta(x, y) \quad = \quad & w_o g_x(x, y) + \\ & \sum_{j=1}^{\lfloor M/2 \rfloor} w_j \{ a(g_x(\lfloor x - j/\mu \rfloor, y - j) + \\ & g_x(\lfloor x + j/\mu \rfloor, y + j) + \\ & (1 - a)(g_x(\lfloor x - j/\mu \rfloor - 1, y - j) + \\ & g_x(\lfloor x + j/\mu \rfloor + 1, y + j))\} \end{aligned} \tag{2.25}$$

Where $\mu = \tan \varphi$ is the direction along which this filtering operation occurs. This term arises from the equation for the Gaussian standard deviation for this pass, which is:

$$\sigma_\phi = \frac{1}{\sin \phi} \sqrt{\sigma_u^2 \cos^2 \theta \sigma_v^2 \sin^2 \theta}, \tag{2.26}$$

φ (and μ) are found using:

$$\mu = \tan\varphi = \frac{\sigma_u^2 \cos^2\theta \sigma_v^2 \sin^2\theta}{(\sigma_u^2 - \sigma_v^2)\cos\theta\sin\theta}. \tag{2.27}$$

As is the case for other separable filters, the overall complexity depends on the size of the filter being used. This process can also be extended to use recursive filters for further speed increases [Geusebroek et al., 2002].

Fig. 2.12 diagrammatically shows the whole ANC process, including various filtering stages which can be used in order to tune the process for different types of image. Finally, Fig. 2.13 shows an example image, including before and after sampling, and after reconstruction using ANC.

2.7 Interpolating TEC Data

This chapter has introduced variously commonly used and state-of-the art interpolation methods. These methods have ranged from ubiquitous triangulation based methods, such as the cubic and linear methods, to more obscure methods such as natural neighbour. It also included RBF interpolation, kriging and ANC.

As discussed in Chapter 1.1, information on the electron content of the ionosphere can be collected using the GPS, by examining the phase and amplitude changes which occur in paths between transmitting satellites and ground based receivers. These data can then be processed in order to create maps of the ionospheric TEC.

Chapter 3 examines the performance of the algorithms described in this chapter, using both simulated and real data, and quantitative techniques known as simulation- and cross-validation. This study forms the most complete examination of interpolation methods for TEC mapping of its kind, and the first application of NC to geophysical data.

The interpolation schemes evaluated represent a broad cross-section of those in common use. Specifically they are:

- Triangulation based (nearest neighbour) — Section 2.2
- Natural neighbour — Section 2.3
- Radial basis function — Section 2.4

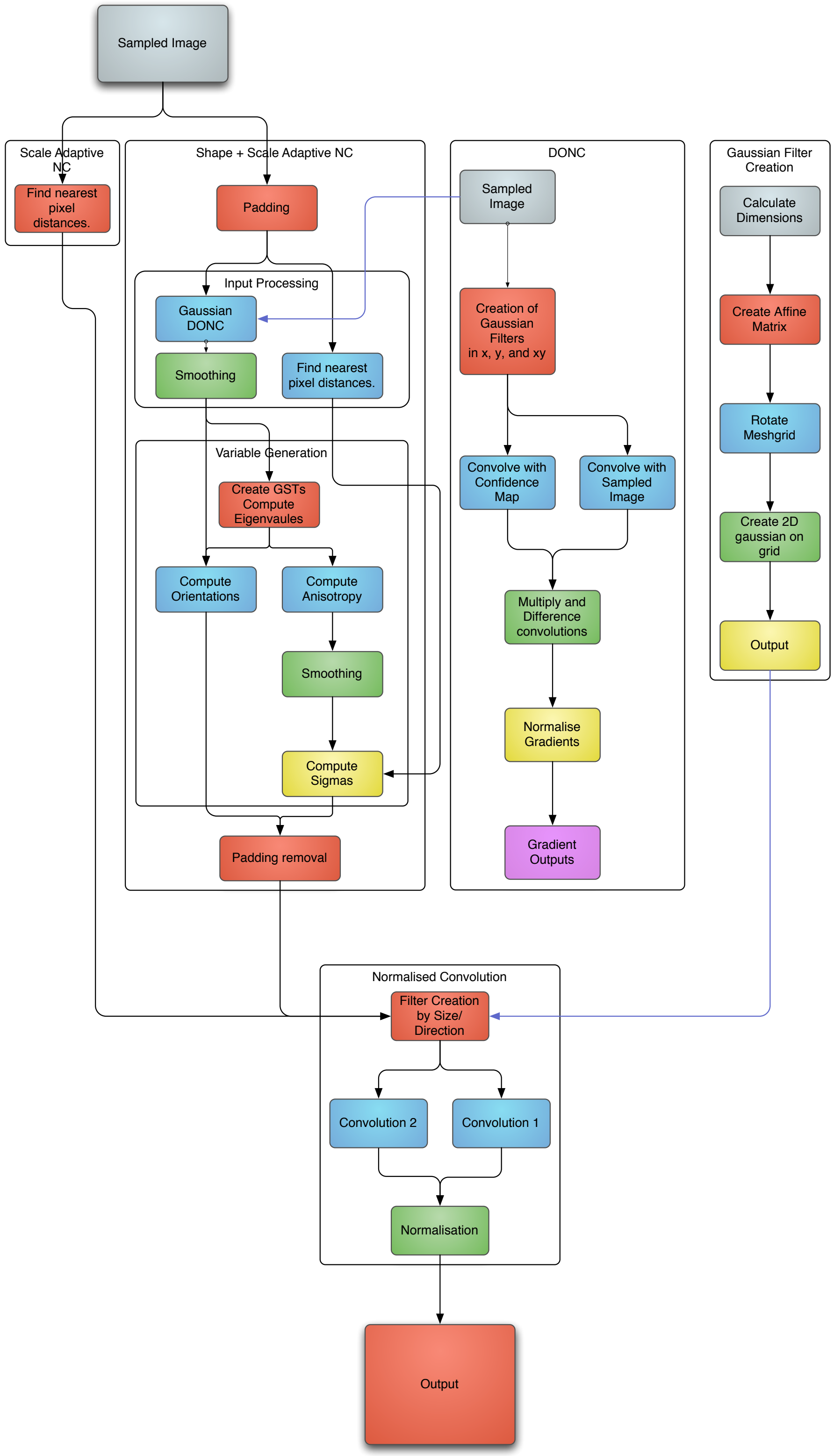
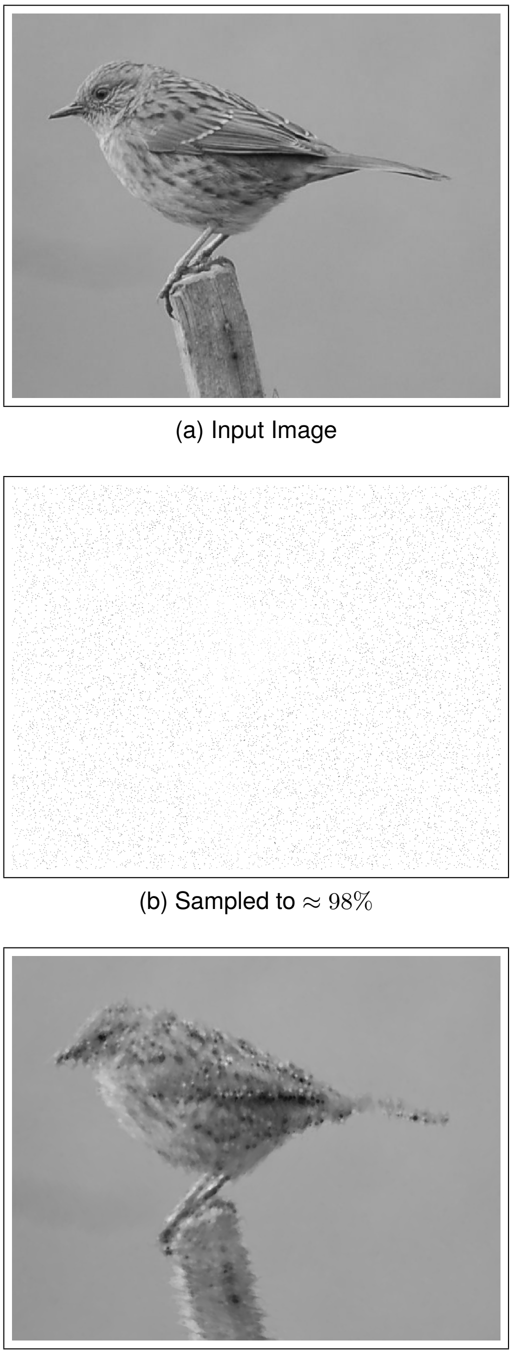


Figure 2.12: A Flow diagram showing the overall ANC process.

- Biharmonic spline — Section 2.4.1
- Ordinary kriging — Section 2.5.

Of the list above, only ordinary kriging, RBF interpolation and BSI are considered truly global techniques. Natural neighbour, nearest neighbour and triangulation based interpolation all use a neighbourhood defined by the Delaunay triangulation of the input data coordinates.



(a) Input Image

(b) Sampled to $\approx 98\%$

(c) ANC Output

Figure 2.13: A photograph of a dunnock (a), which has been randomly sampled to $\approx 98\%$ (b), and then reconstructed using ANC (c).

Chapter 3

Performance Evaluation of Interpolation Techniques for TEC Mapping

3.1 Introduction

The overall aim of this chapter is to assess the performance of currently available multivariate interpolation techniques for ionospheric total electron content (TEC) mapping. The need to establish the relative performance of scattered data interpolation schemes has been recognised and partly addressed in the past, see for example Franke [1982]. However, only very specific case studies exist involving more up to date methods [Blanch et al., 2002, Mahdian et al., 2001, Rauth, 1998]. This study considers, for the first time, the specific application of such techniques to TEC mapping and presents the results of a comprehensive quantitative evaluation, using both simulated data and real ionospheric electron content measurements. Schemes evaluated include those previously used for TEC mapping (e.g. kriging), interpolation methods commonly used in other fields (such as interpolation based on Voronoi tessellations and radial basis functions), and schemes in use for other geophysical applications (such as natural neighbour interpolation). We further propose the application of adaptive normalised convolution (ANC) to the problem of TEC mapping and quantify its performance in comparison with extant techniques. ANC is a recently proposed interpolation scheme that has found application to the reconstruction of data with varying spatial frequency content, orientation and anisotropy. As these properties are also found in TEC images, their reconstruction using ANC appears an attractive proposition. The proposed objective evaluation scheme enables the benefits conferred by ANC to be quantitatively assessed.

The quantitative evaluation methodology is outlined in Section 3.3 and used to determine the performance of the interpolation schemes described in Chapter 2. Finally, discussion and conclusions are presented in Section 3.4.

3.2 Evaluation Methodologies

The evaluation methodologies used here correspond loosely to the best practices given by Thacker et al. [2008]. They suggest that greater understanding of algorithms will allow the successful development of more generally-robust systems, which will then help drive the algorithm's expansion and acceptance into other fields.

The guidelines given are intended to help parameterise new algorithms, improve understanding of internal operation and reduce brittleness. This will improve ease of integration into larger systems, such that it can be carried out without fear of failure, and *with* knowledge of inherent limitations.

Whilst interpolation methods are not necessarily considered *computer-vision* algorithms. The type of understanding and parameterisation is nevertheless extremely useful and surprisingly underused.

Several key questions are given by Thacker et al. [2008] which can be used to assess the current state of the test methodologies and best practices for algorithmic testing. These are listed (reordered) and considered below, in the context of interpolation of sparse ionospheric TEC data:

1. **How is testing currently performed?**

   Testing of TEC mapping is generally simulation based, for example Meggs et al. [2002] tests the accuracy of reconstructing a simple model ionosphere using thin-shell and 3-D inversions. Whilst not strictly related to interpolation, this is the generally accepted approach to assessing methods of TEC mapping.

   Arikan et al. [2007] uses the IRI 2001 model to test a TEC mapping and regularisation method, which is essentially validating reconstructed *real* data with semi-empirical model data.

   Samardjiev et al. [1993], used the CCIR model to compare f_0F_2 results[1]). However, in many cases modelled data are far smoother than the actual phenomena being

[1] The CCIR model is now part of the International Reference Ionosphere (IRI) model.

modelled, which leads to anomalous results. In particular, results tend to be biased towards favouring techniques which produce artificially smooth outputs.

2. **Is there a data set for which the correct answers are known?**

   No standard set of real TEC slant measurements with associated correct reconstructed outputs is available. Meggs et al. [2002] make comparisons between slant TEC, path measurements, simulated data. However there are no data-sets for which ground truths are directly available.

   Simulated data may be sampled and then reconstructed. The reconstructions may then be compared with the original simulated data. This technique will be referred to as *simulation-validation*, and discussed in more detail in Section 3.2.1.

   Cross-validation [e.g. Chen et al., 2008, Blanch, 2003, Kohavi, 1995] is a methodology which allows the partitioning of data into separate sets which can then be used for both reconstruction and validation. Partitioning can be done using various different block sizes, depending on data availability. This is discussed in more detail in Section 3.2.2.

3. **Are there any strawman algorithms?**

   There are no specifically agreed ‘strawman’ algorithms, although several of the most commonly used methods, such as linear and cubic triangulation based interpolation would make obvious choices. Code implementing these and various other methods is readily available (see Chapter 2).

4. **What should we be measuring to quantify performance? What metrics are used?**

   Error values and associated statistics can be specified in either appropriate units, or as a proportion of total values. Errors can also be bounded as appropriate.

5. **Are there experiments which show algorithms are stable and work as expected?**

   Simulated data may be sampled and reconstructed, although exhaustive testing may be impossible due to the ill-posed nature of the interpolation problem.

   In addition some known pathological problems exist for certain algorithms, such as overshooting in cubic and higher-order methods. Knowledge of these artefacts and when they occur can help inform decisions regarding which method to use, and when.

   The distributions of error values gathered using simulation-validation (and less easily, cross-validation) may be examined for biases and skewness. The estimation and analysis of interpolation error distributions and artefacts is discussed in more detail in Chapter 4.

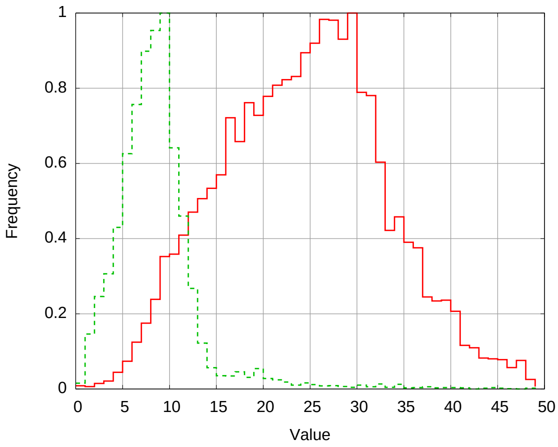


Figure 3.1: Normalised histograms of the two types of simulated correlated data described in section 3.3.1. Multivariate (solid) and univariate (dashed).

3.2.1 Simulation-Validation

Simulation-validation is a method of analysing interpolation performance using simulated data. In this geophysical case, it is reasonable to assume that data being interpolated will exhibit smooth spatial variations, that is, there will be autocorrelation.

Simulated autocorrelated data can be created by following the methodology detailed by Omre [1984], who describes the creation of two kinds of simulated field, designed with testing Kriging in mind.

The first type may be produced by generating fields of normally-distributed random data (by using a random-number-generator), which can then filtered using a pillbox or ball shaped kernel. The filtering process introduces autocorrelation with a lag distance dependent on the filter radius. This data has a multivariate Gaussian distribution which is considered ideal for ordinary kriging.

The second method generates univariate data with an approximately log-normal distribution by filtering fields of uniformly distributed random data. Multi-normality is then removed by examining 5×5 neighbourhoods, around each point, and randomly selecting from the ten highest values. Finally, the natural logarithm of the each data point is calculated. Histograms illustrating typical distributions generated by these two methods can be seen in Fig. 3.1.

Innumerable other methods of generating simulated data exist, such as plasma fractals[2]. However, for the sake of brevity, only the methods of Omre [1984] are used in the case study below.

[2] See, for example: `http://www.ic.sunysb.edu/Stu/jseyster/plasma/`

The *validation* part of this process consists of appropriately sampling the simulated data, for example, by uniform random sampling to an appropriate level of sparsity. The sampled field can then be reconstructed and the original and output images subtracted. Error measures such as root mean square error (RMSE), peak signal-to-noise ratio (PSNR) or sum of squared-differences (SSD) can then be calculated using the output 'difference' image. Error measures can also be given as a proportion of the input image values. The image can also be examined visually.

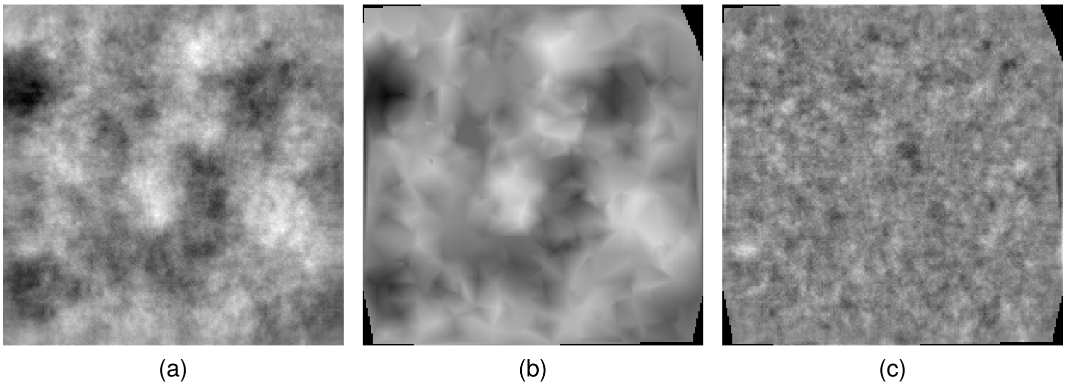


Figure 3.2: Example images demonstrating simulation validation. (a) shows a simulated autocorrelated field. (b) shows the result of uniform down-sampling by 97.99%, and interpolating using linear triangulation based interpolation. (c) shows the difference between (a) and (b). The RMSE (not including pixels outside of the convex hull, shown in black) was 14.2.

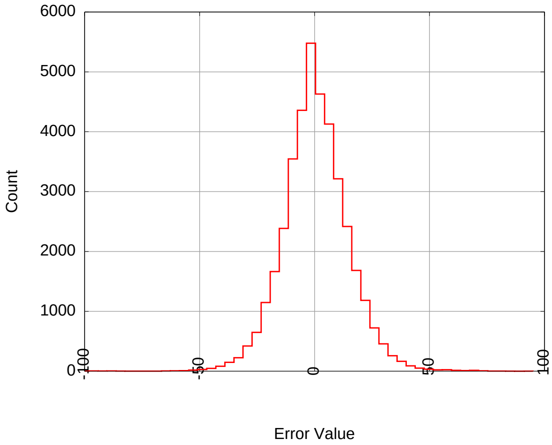


Figure 3.3: Error histogram calculated from Fig. 3.2c.

3.2.2 Cross-Validation

Cross-validation is a statistical technique designed to allow testing with the same dataset that is used for some process or methodology. Typically, cross validation is used for training classifiers, in which training data are divided into two classes, one of which is then used for training, and the other testing the classifier.

The use of cross-validation for interpolation is very similar, although the generation of many different data sparsities requires the data set to either be repeatedly divided, or divided into into multiple classes (a process known as *k*-fold cross-validation [e.g. Chen et al., 2008]).

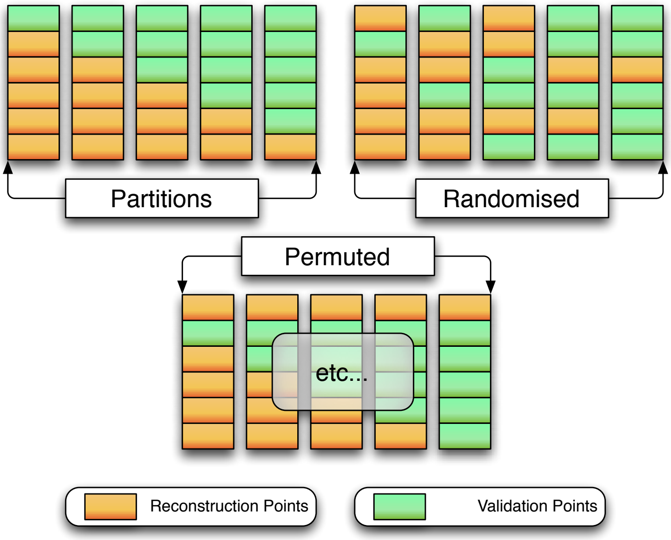


Figure 3.4: Permuting blocks for use in cross-validation.

To apply k-fold cross-validation to the evaluation of interpolation techniques, it is necessary modify the method of class separation. This allows fields with different sparsities to be created. In the case of scattered data, this can be done using the following algorithm:

First, randomise the list of data and partition the list into several blocks with equal sizes. Then:

1. Interpolate the data in the first block, using the methods being tested.
2. Calculate the sparsity of the image to which this data gives rise (see Section 2.1.1, page 17).
3. Difference the output data with the data in the remaining blocks and compute appropriate error measures (such as the RMSE).
4. Repeat this process using the first and second block, then the first, second and third block, etc. until the desired range of sparsities has been examined, or no blocks remain.

If multiple subsets of data are available, for example time slices, in the case of TEC data, this process can be repeated for each slice, and the results averaged by binning the sparsity values. Partitions could also be chosen randomly, or permuted to generate more results if necessary. However, it should be noted that using blocks for both simulation *and* validation at very similar sparsities should be avoided where possible.

To ensure the significance of the results, validation results should only be used where a reasonable number of validation positions are available. This can be aided by setting the block size to at least 30 points, thus ensuring a sensible minimum.

3.3 Interpolation Performance Evaluation

The interpolation methods tested and examined in this study are a mixture of commonly available implementations and custom written code (see Chapter 2). From the triangulation-based class of techniques, the nearest neighbour, linear and cubic interpolation available in MATLAB 2007a's `griddata` function were selected [The Mathworks, Inc, 2007]. The natural neighbour interpolation code was that available from Sakov [2005]. The radial basis function (RBF) interpolation of Carr et al. [1997] was implemented using both linear and multiquadratic bases. No domain decomposition was used due to its complexity and the fact that the fields being interpolated were relatively small. The biharmonic spline algorithm used was the `v4` algorithm in MATLAB's `griddata`, which uses the algorithm of Sandwell [1987]. The kriging method used is known as *ordinary kriging* and works with isotropic, normally distributed data. The implementation evaluated was based on code given in Trauth [2006], with some modifications. In particular, a spherical model was chosen as it represents a good trade-off between the complexity associated with models with a high degree of freedom and the poor performance exhibited by simpler functions such as the linear model. In tests the spherical model was found to perform well with both simulated and TEC data. To enable the unsupervised reconstruction of TEC fields, the spherical model was automatically fitted to the semivariogram using a least-squares method. The ANC interpolation technique implemented was the zero-order scheme described in Section 2.6.6 (page 31). The kernel used was a two-dimensional Gaussian, whose size and orientation were set using (2.16)–(2.18). The gradient square tensors (GST) measures were constructed using gradients obtained from normalised differential convolution (NDC). To reduce the complexity, the efficient decomposition technique that provides a close approximation for rotated Gaussians proposed by Geusebroek et al. [2002] was used in the final filtering stage. The Euclidean distance was used for all techniques requiring a distance metric.

The techniques to be evaluated were applied to both simulated and real TEC data. As simulated data provides ground truth values, it has the advantage of allowing analysis of residual errors to be calculated at every point in the output field. In addition, parameters such as the of the input sparsity can be carefully controlled. It should be noted, however, that the performance of any interpolation method can vary considerably with the statistics of the input data, and therefore the results gained through simulation are not necessarily indicative of the general performance. Therefore, the ultimate test of reconstruction techniques remains their application to real data. To this end, the interpolation methods are applied to TEC data from the much studied October 2003 ionospheric storm.

3.3.1 Simulated-Validation Results

The interpolation techniques were used to reconstruct each type of the simulated data from sparsities ranging from 95% to 99% in steps of 1%, and 99% to 99.9% in steps of 0.2%. The sampling was carried out by thresholding uniform pseudo-random numbers, so the percentage of remaining samples is not necessarily the same as the requested value. Plotted results show the actual sparsity obtained. Each generated data field was sampled 30 times at each sparsity, and then reconstructed with each interpolation method. The number of reconstructions was set to 30 to minimise computation time, whilst ensuring the statistical significance of the results.

The RMSE between the original and reconstructed data outputs were calculated and averaged over the 30 reconstructions, see Figs. 3.5 and 3.6. In both of these figures, the RMSE values were normalised by dividing by the average value of the data being interpolated to give the RMSE as a proportion in which, for example, a value of 0.1 corresponds to a 10% error. For clarity, the results for the two worst performing techniques, linear and nearest neighbour interpolation, were removed.

The RMSE performance of all interpolation techniques increases with sparsity for both types of simulated data. Overall, the RMSE for the multivariate data increases from around 10% at a sparsity of 95% to 25-30% at a sparsity of 99.6%. The performance for the univariate data at the corresponding sparsities is better, increasing from approximately 7% to 15% at sparsities of 95% and 99.6% respectively.

The worst performing technique for both the univariate and multivariate simulated data is biharmonic spline interpolation (BSI), which has a consistently higher RMSE than other schemes. Although kriging is the best performer for many sparsities, its error is dramatically increased at sparsities $> 99.3\%$ and at certain other lower sparsities, probably as a result of failing to correctly fit to the semivariogram. This is significant as the errors in these cases are up to 4 times those of the other techniques. The ANC performance at all sparsities is 1-2% worse than the best performing techniques. Cubic interpolation generally performs well and the overall best performer is natural neigh-

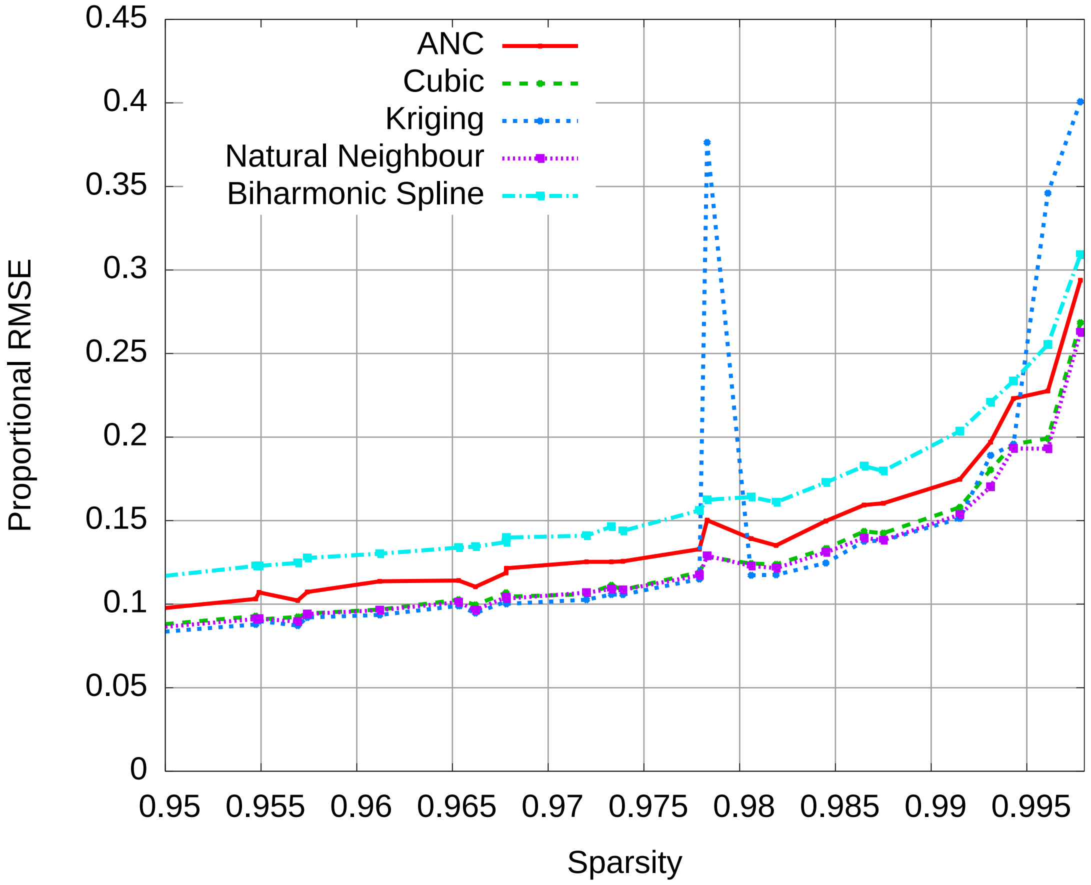


Figure 3.5: Proportional RMSE as a function of sparsity for simulated multivariate correlated data reconstruction.

bour interpolation, which exhibits no anomalous behaviour, whilst maintaining good performance throughout.

This process described above was repeated whilst altering the size of the pillbox used to impose auto-correlation of the simulated data from 10 to 50 in steps of 10. Overall, this has little effect on the RMSE of the reconstructions, with the exception of kriging, whose implicit assumptions about data auto-correlation are violated when the lag distance is small. In both the univariate and multivariate cases natural neighbour interpolation performed best with respect to changing radius of correlation (see Figs. 3.7 and Fig. 3.8). This is because its performance is based on data position rather than value.

In addition to providing overall error values, simulated data allows for analysis of residual errors at every point in the field. In all cases the residual errors exhibited Gaussian distributions with means very close to zero, showing the interpolation techniques have negligible bias. A typical result of this type of analysis is shown in Fig. 3.3.

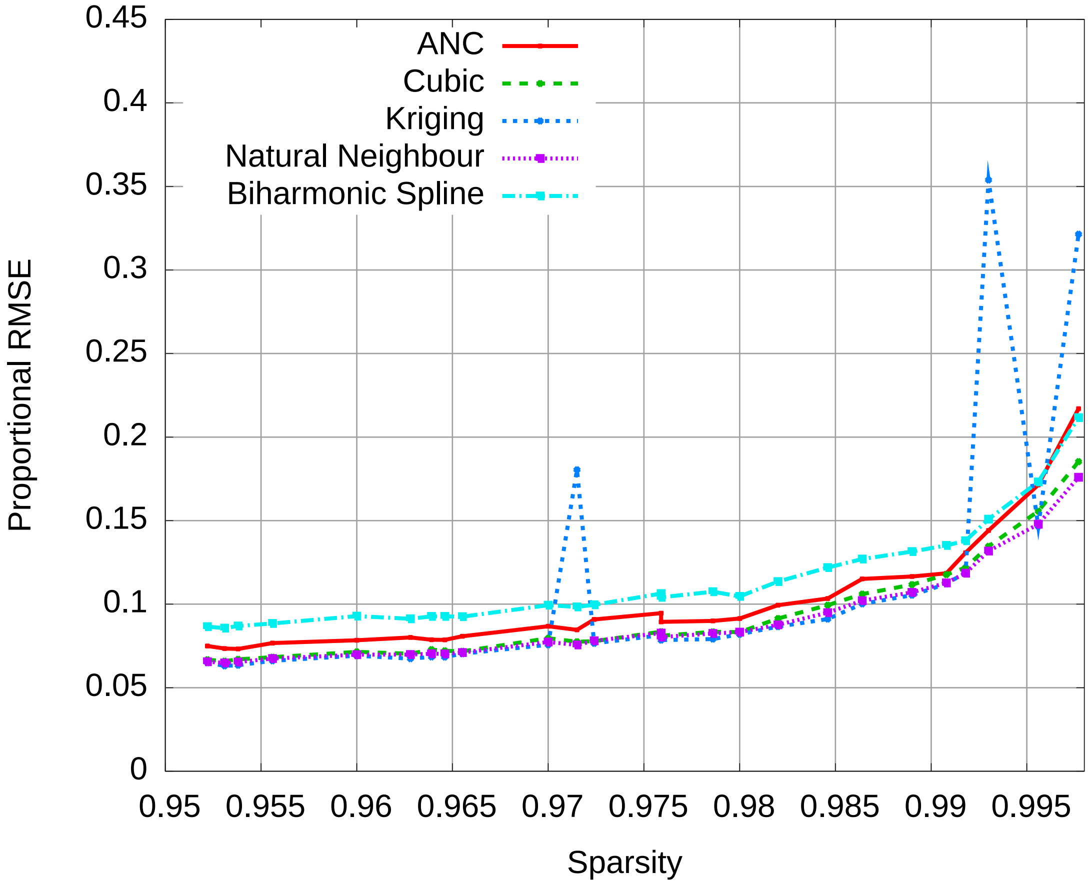


Figure 3.6: Proportional RMSE as a function of sparsity for simulated univariate correlated data reconstruction.

3.3.2 TEC Cross-Validation Results

The sources of data used to test TEC reconstruction are approximately 80 global positioning system (GPS) measuring stations lying within 20–70° N, and 70–130° W. This corresponds to a coverage of most of North America. Whilst more sites were available at that time, not all sites were used, as the main aim of this chapter is to examine interpolation during high sparsity cases. The time period over which data were drawn was from noon to midnight on October 30th, 2003 – the peak of the 'Halloween Storm' [Hernandez-Pajares et al., 2005]. Data were considered stationary within 15 minute intervals, and projected onto a 'thin shell' for reconstruction (see Section 1.1.4, page 8). The thin shell used covered the same area as the ground stations and had latitudinal and longitudinal resolutions of 0.5°, giving rise to fields of size 101 × 121 pixels. As each ground-based receiver station can see approximately 6 satellites at any one time, there are around 500 paths associated with the 80 measuring stations.

Care was also taken to ensure that the average TEC values of the points used for validation were similar in magnitude to those being used for the reconstructions, to

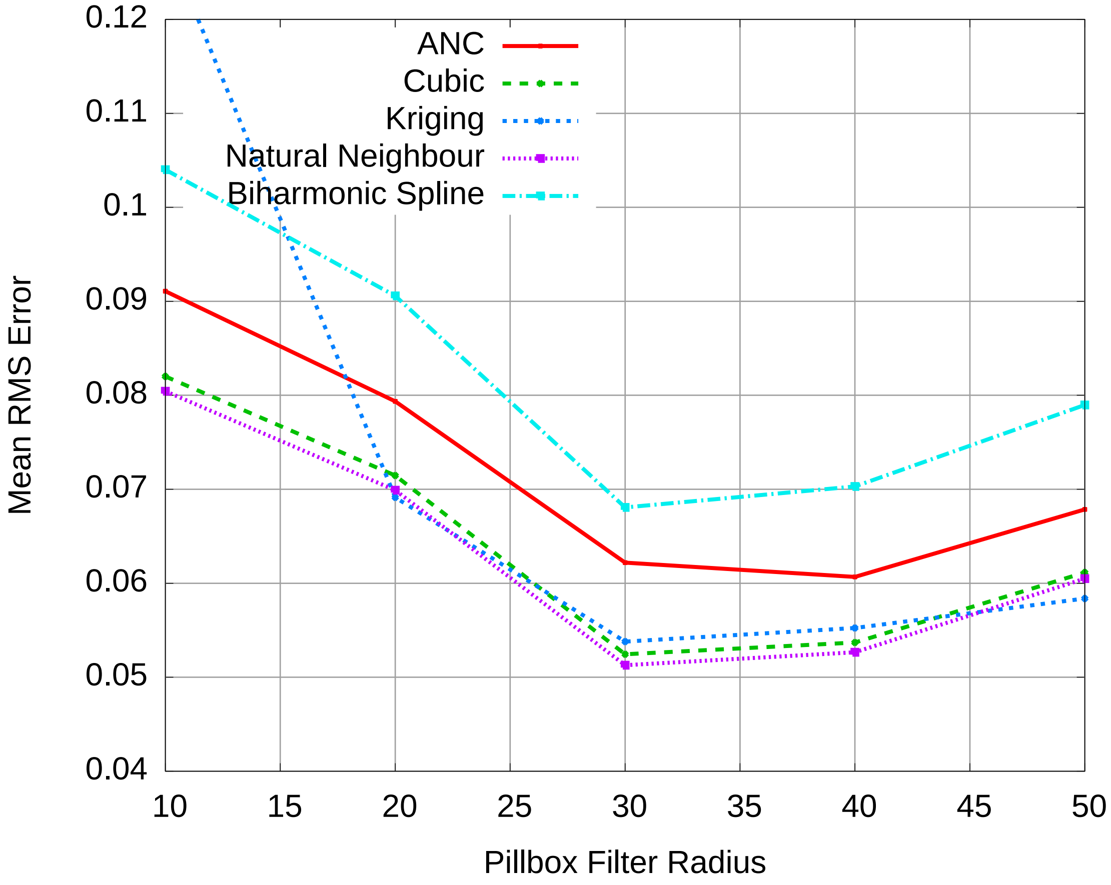


Figure 3.7: Proportional RMSE as a function of radius for simulated multivariate correlated data reconstruction.

avoid biasing the output values.

Fig. 3.9 shows the electron content results binned into 40 sections, and averaged across each section for the interpolation techniques evaluated in the previous section. As before, the RMSE have been divided by the average field value to produce proportional RMSE results that are more directly comparable with the simulated results. The lower sparsity limit of 0.9825% is higher than for the simulated data as this is the greatest density that can be achieved with the available data points. Compared to the simulated data, the increase in error with sparsity is less marked for all techniques.

Once again, BSI was the overall worst performing technique. The inconsistency of the errors across the range of sparsities associated with kriging that was exhibited in the simulated results is also evident. For example, at a sparsity of 99.3% its RMSE is over 4 times that of the best performing technique. Cubic interpolation performed consistently with approximately average results.

Natural neighbour interpolation again performed well but, unlike the results for sim-

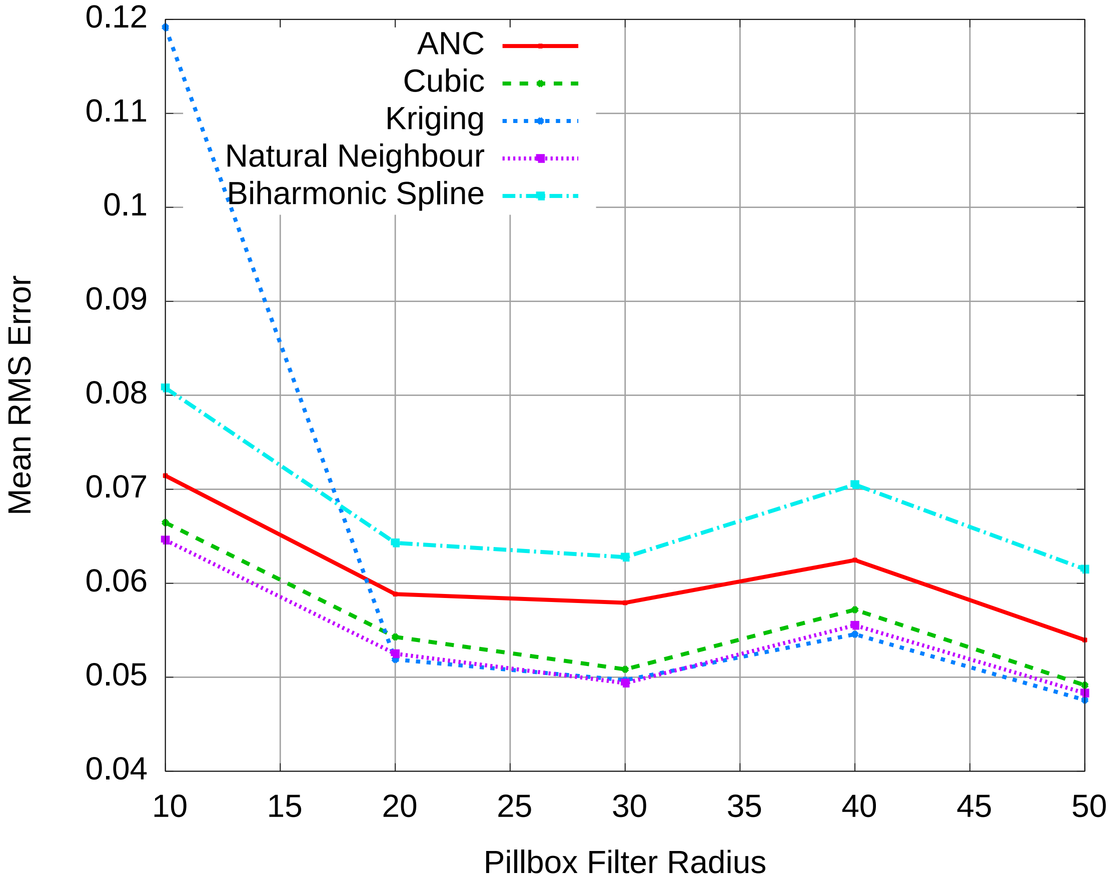


Figure 3.8: Proportional RMSE as a function of radius for simulated univariate correlated data reconstruction.

ulated data, its performance is matched by that of ANC. In fact, over the range of sparsities the proportional RMSE produced by ANC is, on average, 0.08% less than the equivalent natural neighbour results, and also showed a slightly lower variance. In comparison, the average RMSE produced by cubic and kriging interpolation were 0.91% and 1.57% worse than ANC, respectively. The inconsistency of kriging is reflected by a variance approximately 10 times higher than the other techniques.

To illustrate the results produced by some of the different interpolation techniques, Fig. 3.10 presents example reconstructions produced by ANC and kriging for two sets of input data from Fig. 1.5c. The input data in each case was 25% of the available GPS path signals, giving a sparsity of approximately 98.8%. For both sets of input data, kriging and ANC have produced slightly different results. The proportional RMSE for each reconstruction can be found using the remaining 75% of the data. For this case the average proportional RMSE is 0.0524 TEC Units and the difference between the kriging and ANC errors is less than 1%. Fig. 3.10 also shows that the set of input data used produces more significant differences in the output fields than the choice of interpolation method. This observation underlies the benefit of the cross-validation evaluation procedure which removes any sensitivity to choice of input data by the

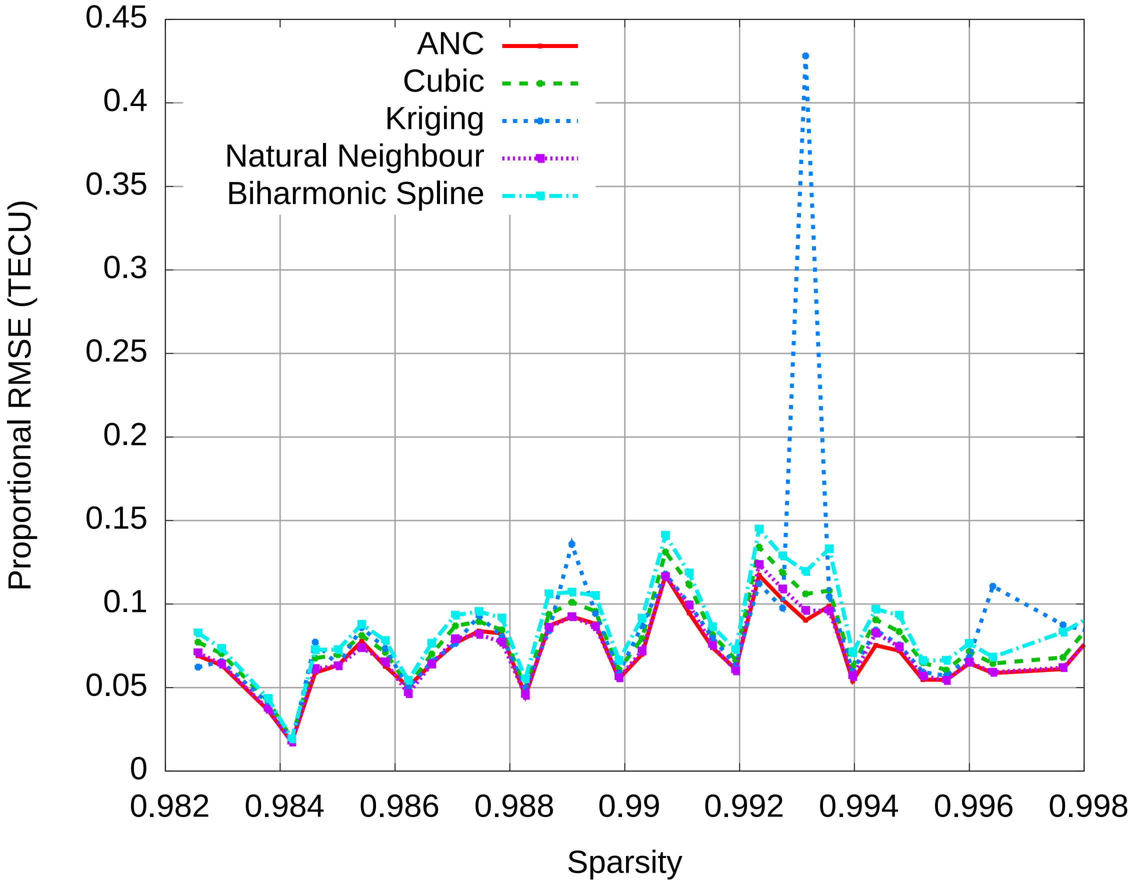
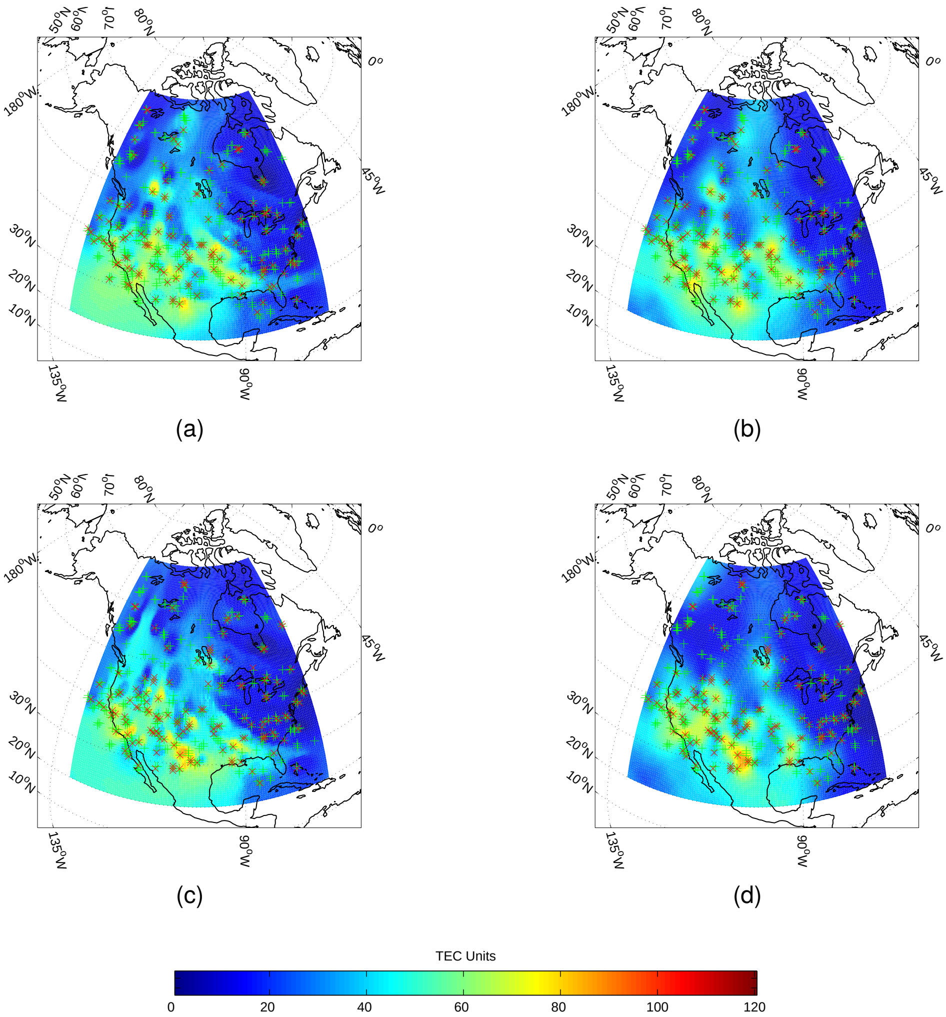


Figure 3.9: Proportional RMSE as a function of sparsity for TEC data from GPS path measurements. Errors were calculated using the cross-validation method described in Section 3.2.2).

averaging the results within a given range of sparsities.

3.4 Discussion and Conclusions

In the literature, kriging has been the interpolation method of choice for producing TEC maps of the ionosphere, as well as for other geophysical applications Blanch [2003], Stanislawska et al. [2002], Wielgosz et al. [2003]. However, to date there has been little in the way of evidence to support its adoption over other interpolation schemes. This chapter has sought to address this issue by performing a comprehensive quantitative evaluation of kriging and a selection of other interpolation methods currently in use. To this end, an evaluation methodology that uses both simulated and TEC data has been proposed. With simulated data, error values can be calculated at all output points. For TEC reconstructions this is not possible and, instead, an evaluation using cross-validation was performed. Considering the overall performance for both simulated and TEC data, the following remarks about the individual interpolation techniques can be



(a)

(b)

(c)

(d)

Figure 3.10: Example output images produced by ANC and kriging using two sets of the input data from Fig. 1.5c. Each set consisted of 25% of the available data ($\times$) and the remaining data ($+$) used to calculated the proportional RMSE. (a) and (b) Set 1 results (sparsity 98.94%) produced by ANC and kriging respectively. (c) and (d) Set 2 results (sparsity 98.79%) produced by ANC and kriging respectively. The proportional RMSE values are (a) 0.0640, (b) 0.0533, (c) 0.0503, and (d) 0.0421 TEC Units.

made.

Triangulation-based techniques are widely used in computer graphics applications. The best performing of these, cubic interpolation, has a relatively low complexity and its performance is, in many cases reasonably close to that of the best performing, more complex methods. Therefore, for ionospheric applications where a small loss in accuracy can be sacrificed for a faster run-time, it is a reasonable choice of technique.

Although the kriging scheme used in this investigation performs well at many sparsities, it exhibits a very large variance for both the simulated and TEC data. This variance is due to spikes where the proportional RMSE is excessively high. Two main stages of kriging-based interpolation are the construction of the semivariogram and the fitting of a suitable model, and both of these are sensitive to the settings of their various parameters. This is one of the main reasons why it is often recommended that kriging is implemented as an interactive process, as opposed to an automatic one. When the semivarogram model being used fails to accurately fit the experimental semivarigram, the output of the interpolation will be poor. Fig. 2.5b shows an example, degenerate semivariogram where the data have a high variance at all lags, a breach of the fundamental assumption of high autocorrelation at low lags. In these cases the fitted model poorly matches the actual data, resulting in an output field with a high error. Although, in theory, it may be possible to automatically detect degenerate semivariograms and try to find a more suitable one to fit to, the procedure is very complex and, for the tests performed here, the level of sophistication required would be far beyond that required by the other techniques being evaluated.

Natural neighbour interpolation performs well across all data types and sparsities. It is the best performing method for both types of simulated data and is only surpassed by ANC on the TEC data. The main drawback of natural neighbour interpolation is that it is complicated to implement, and there are few modern reference implementations available. However, if it were more widely known, and its performance recognised, this situation could change.

While there are many performance features that are common for both the simulated and TEC reconstructions there are also some significant differences, such as the change in RMSE with sparsity, and the relative performance of individual interpolation techniques. This suggests that in testing interpolation methods simulations should only be used if they are demonstrably very similar to the real data to be interpolated. If this is not the case, the data-driven cross-validation methodology demonstrated here is ideal for testing the performance of interpolation schemes using only real data. The major difference between the simulated and TEC data is that of anisotropy and this helps explain why the relative performance of ANC, a technique that copes well with anisotropy, was dramatically improved for the TEC reconstruction. Indeed, given that ANC was the best performing technique for the TEC data, these results suggest that ANC and natural neighbour interpolation should be the methods of choice for ionospheric reconstructions as they offer an error performance that is better and more consistent than kriging.

The following chapter discusses the examination of interpolation error distributions and biases, as mentioned in Section 3.2, above.

Chapter 4

Interpolation Artefacts and Error Distributions

The examination of artefacts and the measurement of error distributions are two ways of *sanity checking* interpolation methods, and gathering knowledge of their specific nuances. This idea was introduced in Section 3.2, (on page 41), and is further described here, with examples.

4.1 Artefacts

Artefacts are a very interesting and important aspect of interpolation. They can differ greatly across different methods, and appear in widely varying situations. This section examines different methods with a view to identifying different artefacts, and examining where and when they occur. Such knowledge can be very useful in a wide variety of situations, such as when using sensitive analysis methods, or when closely examining interpolated outputs.

From an alternative angle, knowledge of different artefacts can aid in gaining an intuitive understanding of how different interpolation methods work, and should therefore help prospective users to choose the technique most suited to their data-sets, and applications.

In this case, the word *artefact* can refer to any characteristic of an interpolated output field which has been introduced by the interpolation technique used to create it. Examples of artefacts commonly seen are peaks, concave slopes and overshooting edges. This definition is deliberately loose, as interpolation is arguably the introduction of artefacts around a sparse set of data points. As interpolation is ill-posed, there is an infinite number of available outputs, all of which are interpolated versions of the input data, the vast majority of which are completely inappropriate. Of the remaining (miniscule fraction) of outputs deemed acceptable, only some will be appropriate in any given case. For this reason, any specific notable feature of any interpolation method can be considered an *artefact*, and so the following discussion will be kept fairly broad in scope.

Fig. 4.3 shows some example data from the Shuttle Radar Topography Mission. These data have been downsampled by approximately 99%, and then reconstructed using various interpolation methods. This figure illustrates artefacts the produced, as well as giving a general *feel* for the outputs produced by the different methods.

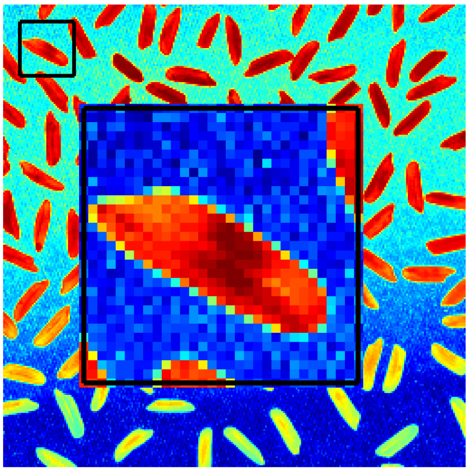
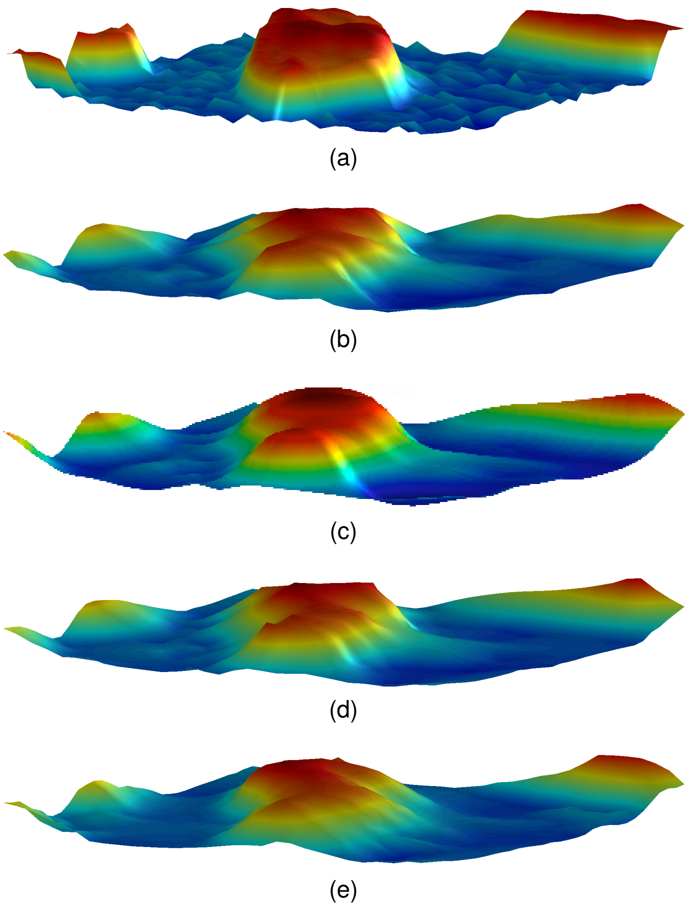


Figure 4.1: A greyscale image of rice grains, displayed using false colour, with the single grain used in Fig. 4.2.

Fig. 4.1 shows an image of rice grains, and illustrates a small section which was sampled and interpolated to produce Fig. 4.2. In this figure, various interpolated versions of (a) (which was first randomly sampled to $\approx 99\%$) are shown. Whilst there are no *major* differences between the surfaces, they do illustrate the subtle differences between the different methods, particularly in the large flat regions.

4.1.1 Triangulation Based Linear Interpolation

Linear triangulation based interpolation operates by fitting flat surfaces to Delaunay triangulated x- and y-coordinates, with heights defined by the z-coordinates. For this reason, data interpolated by this kind of interpolation can appear to be highly faceted,



(a)

(b)

(c)

(d)

(e)

Figure 4.2: Examples of a section of an image of rice grains. The original image, displayed as a displacement map (a) was sampled to a sparsity of $\approx 99\%$, and is followed by (b) linear, (c) RBF TPS,(d) NN and (e) ANC interpolated versions of the sampled image.

especially when interpolating particularly sparse data.

Linear interpolation based on triangulation is analogous to fixing triangular plates together over a frame whose vertices are the data points, and with edges following those of the triangles. There is not necessarily any continuity across any derivatives of the edges, and therefore the surfaces can appear highly jagged. However, because no continuity of derivatives is enforced all points in the interpolated output will lie within the surfaces defined by the triangulation. This means that linear interpolation of this kind never contains overshoots, which can be an advantage in some situations, and absolutely essential in others. The main downside is the faceting effect introduced by this method.

4.1.2 Linear RBF Interpolation

Linear radial basis function (RBF) interpolation uses a basis function which is varies linearly with the distance from input datum. This yields results which are similar to

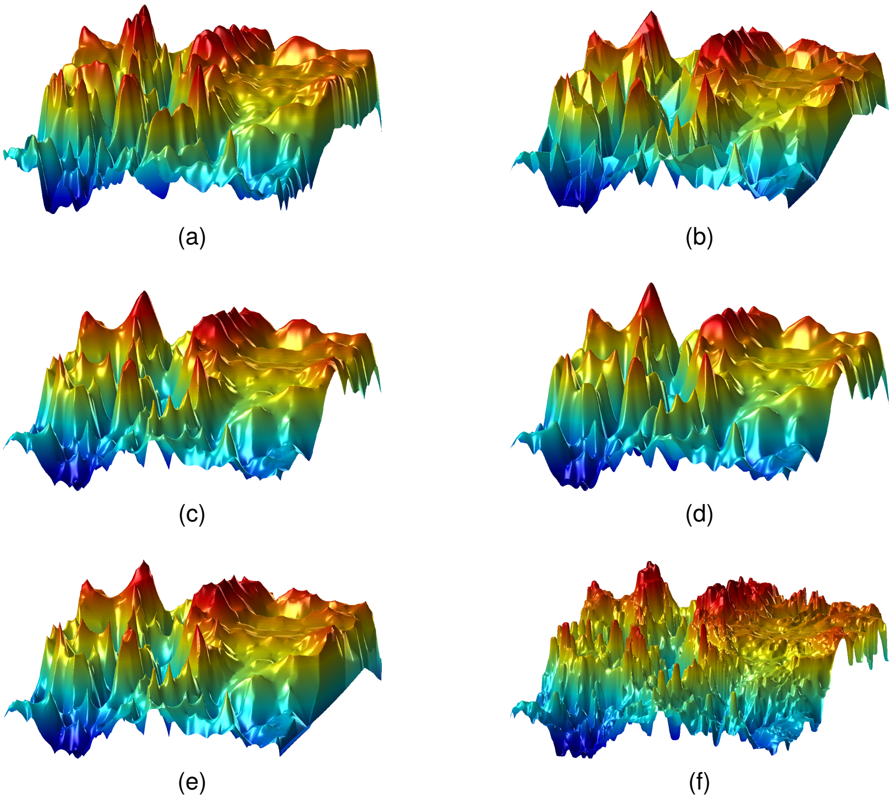


Figure 4.3: Example elevation data from the SRTM, shown as a false colour surface. (a) is the original input data, which was sampled to $\approx 99\%$ sparsity, and reconstructed by (b) was interpolated using linear triangulation based interpolation and (c) was interpolated using linear RBF interpolation., (d) TPS RBF interpolation, (e) NN interpolation, (f) ANC.

those produced by linear triangulation-based interpolation, but which do not contain the faceted appearance. The trade-off here is that RBF interpolation is a global interpolation method, which means it is computationally expensive, especially when compared to triangulation based methods. For small data-sets where overshoots and faceting effects are undesirable, this interpolation method is ideal.

4.1.3 Cubic Interpolation

Cubic polynomial surfaces are very commonly used for interpolation. This is because their surfaces are smooth and their creation is less computationally expensive and requires less input datum than other higher order surfaces. These factors mean that cubic interpolation methods are among the most commonly used of all interpolation methods, and are the default in many software packages. This makes knowing about potential artefacts all the more important.

Figs. 4.2c and 4.3d were interpolated using thin plate spline (TPS) RBF interpolation, a cubic-order method which was introduced in Section 2.4, on page 21. This produces results which are almost identical to biharmonic spline interpolation (BSI), and is characterised by smooth, isotropic surfaces. A TPS is a kind of cubic spline, and as such produces similar outputs to triangulation based cubic interpolation. These outputs are all characterised by smooth surfaces, and a tendency to produce values which overshoot around extrema. This is because cubic interpolation methods attempt to maintain continuous first and second derivatives at all points. Fig. 4.4 shows a minor example of the kind of overshoot and undershoot that can be caused by cubic interpolation. The area in the top left shows a small and steep peak, caused by the two close points after the leading low valued point. On the right, a similar undershoot with a larger horizontal extent appears for the same reason.

A good physical analogy to spline based interpolation is imagining the image having been interpolated using metal sheets, which are able to bend a certain amount, and whose joins with other plates must be continuous to at least the first derivative. This physical analogy is the idea around which splines were originally designed.

Other interpolation methods such as the venerable Akima method also Akima [1978], Ripley [2004], aim to reduce these overshoots [1] and succeed in suppressing them. However, algorithms such as this are no longer commonly used in modern software and tend only to be used in applications where legacy code is heavily relied upon, such as in the interpolation of precipitation information. Chen et al. [2008], for example compares three commonly used interpolation methods for interpolation global rainfall. The methods tested include Shepard [1968], which is based on a modified inverse distance weighting function, designed when computational methods for irregular interpolation was first being heavily investigated, in the late 1960s.

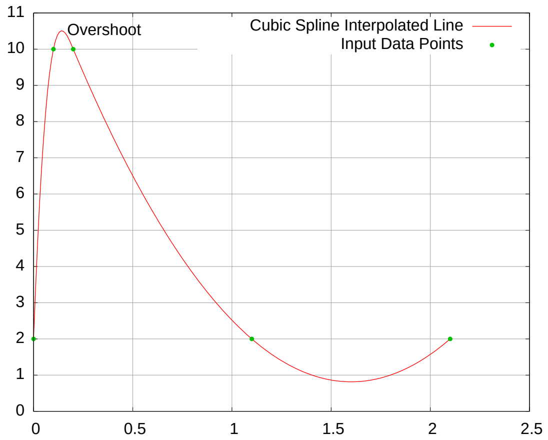


Figure 4.4: Examples of overshoot (and undershoot) in cubic interpolated data.

[1] which are themselves a well documented and analysed problem [see e.g. Fried and Zietz, 1973, Maeland, 1988]

4.1.4 Natural Neighbour Interpolation

In common with many other methods, natural neighbour (NN) again uses the Delaunay triangulation. However, it differs from all other methods in that it uses the ratio of overlapping areas to determine the weighting of data points. This results in surfaces which vary more smoothly than linear interpolation based on triangulation. The most obvious artefact caused by NN is the fact that it tends to produce sharp points near the input datum.

Physically, the surfaces produced by NN are similar in appearance to that of a heavy rubber sheet, stretched over and attached to input points. The sheet is more heavily stretched near input points.

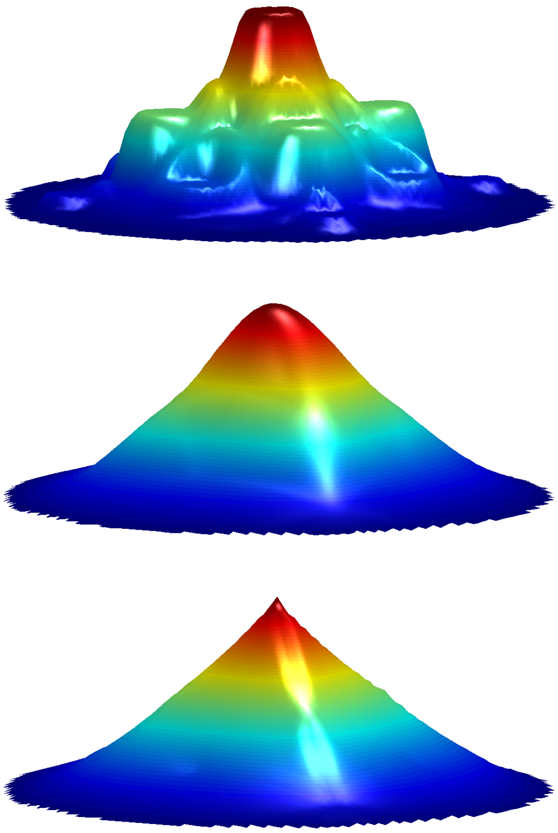


Figure 4.5: Example interpolated pyramids, demonstrating the pointy effect that NN causes (bottom), along with the same data interpolated with cubic interpolation based on triangulation (middle), and ANC with undersized filters (top).

4.1.5 Adaptive Normalised Convolution

The adaptive normalised convolution (ANC) method (see Section 2.6, page 25), uses convolution with rotated and scaled filters to perform interpolation. It produces outputs which take account of image anisotropy, and whose appearance is somewhere between natural-neighbour and TPS interpolation. The main artefact that ANC produces is a kind of stepping effect, which occurs when the filters used are too small to accurately capture the spatial variation of the sampled image. In this case, the output images appear similar to the topmost image of Fig. 4.5, which has a stepped appearance. This kind of artefact can be prevented by increasing the minimum size of the filters used. Errors such as this are fairly common in adaptive techniques where the adaptation algorithm is unable to handle data whose properties do not match those for which it was designed.

4.2 Interpolation Error Distributions

One very effective way of examining interpolation methods for possible problems is to create a histogram of the errors between an interpolated output, and a simulated full-field input. This can also be attempted using a cross-validation style method (see Chapter 3), although often this will not yield enough error measurements for the creation of a full histogram.

The histogram should describe an approximate Gaussian distribution centred on zero. If the input data are reasonably dense, then the bins around zero will likely have considerably higher values than would be expected in a standard Gaussian, due to large numbers of zero (and close to zero) valued errors. Long tails are also fairly common in real data due to noise which cannot be effectively interpolated.

Once the distribution of errors has been calculated, the first three statistical moments of the distribution can be used to examine the output for biases and skewed output, as well as the calculation confidence limits on the errors.

In ascending order the moments can be used as follows:

1. The first standardised moment of a distribution is its mean. If this is not zero, then there is a systematic bias in the reconstruction being examined. These results should almost certainly be discarded. Fortunately, seriously biased interpolated outputs only rarely occur in practice, making testing common interpolation methods largely unnecessary.

$$\mu = \overline{x} = \frac{1}{N}\sum_{n=1}^{N} x_i. \tag{4.1}$$

2. The second standardised moment is the variance. This is useful for characterising the spread of error values. Taking the value of the distribution at the positions indicated by the standard deviation gives a 68.26% confidence limit. Confidence limits are discussed in more detail below. Sample variance may be calculated using:

$$\sigma^2 = \frac{1}{N}\sum_{n=1}^{N}(x_i - \overline{x})^2. \tag{4.2}$$

3. The third moment describes the 'skewness', or asymmetry of the distribution. A non zero skewness, is indicative of a tendency for the interpolation method to under-, or over-estimate the output values which do not lie on input datum. Asymmetry in distributions is known as 'skew', and is defined as the third-standardised moment of a given distribution. The sample skewness is estimated using:

$$\gamma_1 = \frac{\frac{1}{N}\sum\limits_{n=1}^{N}(x_i - \overline{x})^3}{(\sigma_2)^{3/2}}. \tag{4.3}$$

   Where μ_3 is the third moment about the mean, and σ is the standard deviation.

4. The fourth moments is the 'kurtosis', which describes how outlier-prone a distribution is. Standard Gaussian distributions have a kurtosis of 3; distributions which are more outlier prone have higher kurtosis values, and those which are less outlier prone have lower vales. Many definitions of kurtosis subtract 3 to make the kurtosis of the normal distribution equal to zero. This is known as *excess kurtosis*. High kurtosis values in error distributions are an indicator that there may be fairly small numbers of large errors. If this is the case, an examination of confidence bounds to ensure that the performance is acceptable may be carried out. Fig. 4.6 shows plots of three distributions: Gaussian, which has an excess kurtosis of 0, Logistic, which has an excess kurtosis of 1.2, and Wigner semicircle which has an excess kurtosis of -1. These curves show that the kurtosis measures how much of the distribution lies close to the mean.

   The sample excess kurtosis is given by:

$$\gamma_2 = \frac{1}{N(\sigma^2)^2}\sum_{n=1}^{N}(x_i - \overline{x})^4 - 3. \tag{4.4}$$

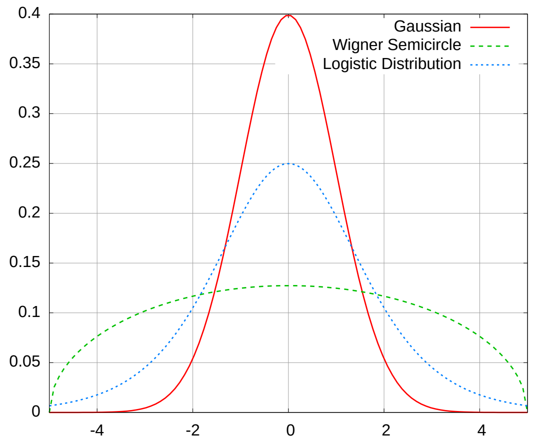
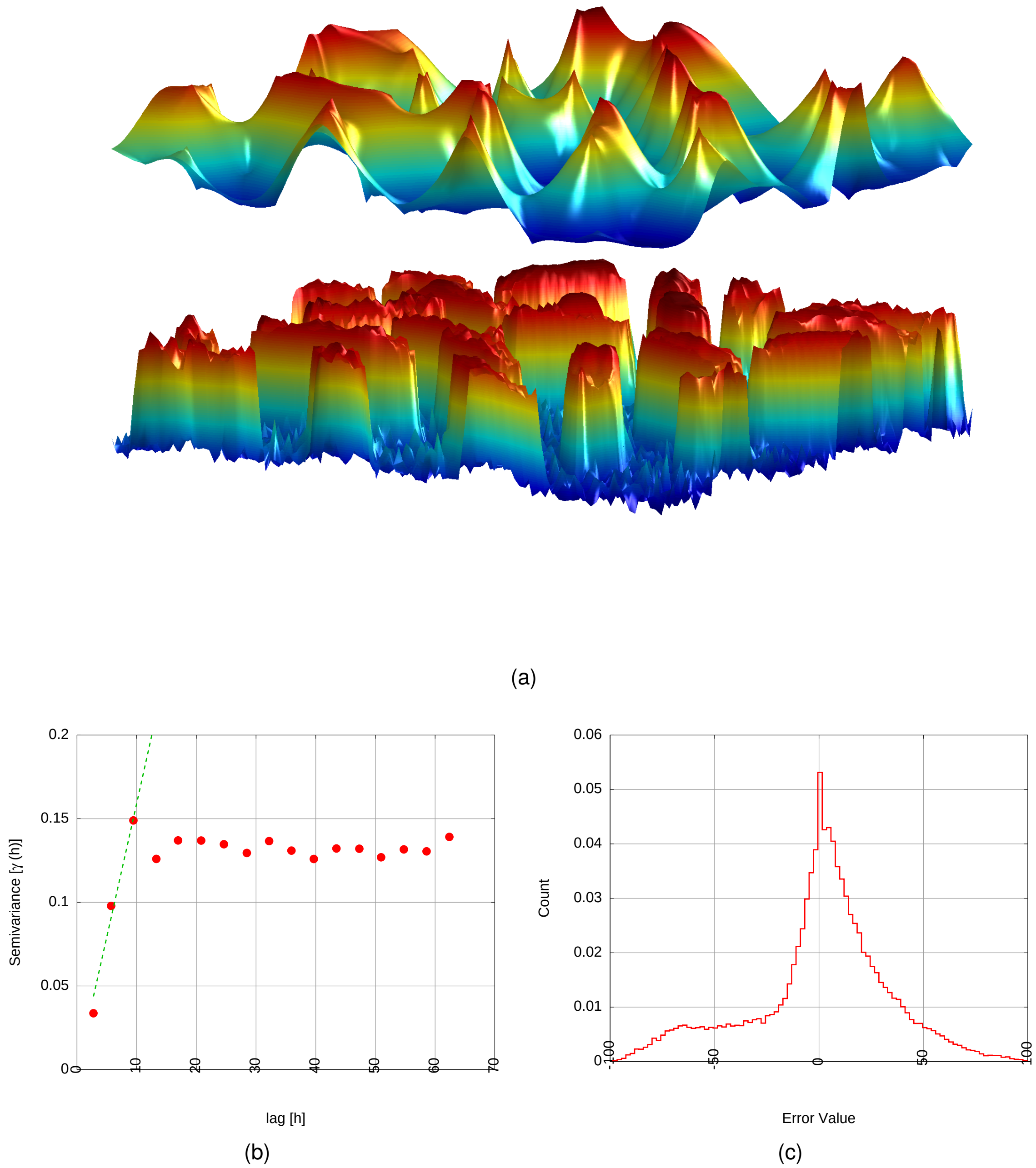


Figure 4.6: Various distributions with differing kurtosis values, including Gaussian (excess kurtosis 0), Logistic (excess kurtosis 1.2) and Wigner semicircle (excess kurtosis -1).

4.2.1 An Error Skew Case Study

As a further example of the skew problem mentioned above in Point 3, Fig. 4.7 shows a sample image (Fig. 4.1) and its reconstruction with kriging (a), the semivariogram (b), and the error histogram (c). This histogram shows that the technique is slightly skewed in this case, indicating that output values tend to be under-estimates.

To examine why this is, this section considers how kriging operates. Kriging works in several stages, the first of which is the estimation of an *experimental semivariogram*, which describes the spatial autocorrelation of the data being interpolated. This then has a model fitted to it, which is then used as a basis function for a global interpolation. In this case, the image has patches with high levels, followed closely by patches with low levels (the patches are actually grains of rice), because each grain of rice has a different orientation, the level of autocorrelation in the image varies significantly across the image, which is also highly anisotropic. This means that the semivariogram is unable to fully capture the spatial variation of the image, which in turn leads to a poorly fitting semivariogram model, which culminates in a poor interpolated output. The main consequence of the poorly fitting model is that the basis function chosen for the interpolation leads to concave surfaces in the output, which causes a large proportion of the output positions to contain under estimates than overestimates, this is evidenced by the longer left hand tail in Fig. 4.7c. Problems with kriging are also discussed in Section 3.4, page 52.



(a)

(b)

(c)

Figure 4.7: (a) (top) a reconstructed version of the sampled image (bottom), (b) the semivariogram for the sampled version of the sampled version of the bottom image in (a). (c) a histogram showing the distribution of error values between the images in (a).

4.2.2 Confidence Limits

Error distributions can be used to calculate confidence limits on output errors. These can be calculated using the standard distribution of errors, or as absolute error bounds are probably more useful, the histogram of absolute errors could be used to calculate

Table 4.1: Absolute error percentiles for 95% sparsity.

Percentile	ANC	Nat. Neigh	Nearest Neigh.	Linear	Cubic	BSI	Kriging
50	0.0309	0.0274	0.0345	0.0276	0.0275	0.0268	0.0264
90	0.0872	0.0772	0.1038	0.0790	0.0785	0.0762	0.0748
95	0.1088	0.0961	0.1305	0.0990	0.0974	0.0951	0.0929
99	0.1546	0.1383	0.1861	0.1462	0.1413	0.1369	0.1319

the limits.

Fig. 4.8 shows an example normalised histogram with the 90, 95 and 99% confidence limits plotted as vertical lines (from left to right). Other rank statistics, such as the median (the 50th percentile) can also be calculated if desired, and the process can also be carried out using the normal error histogram, or histogram of other error measures, such as root mean square error (RMSE) or sum squared error (SSE).

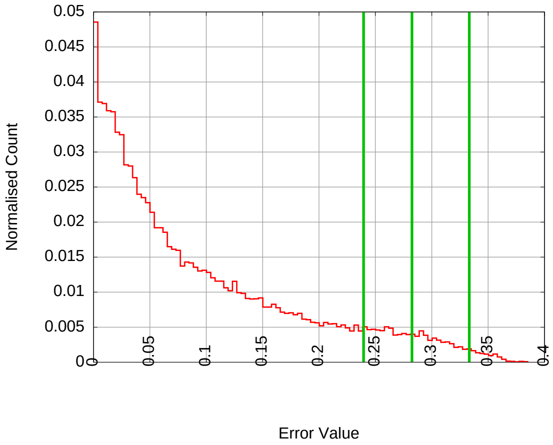
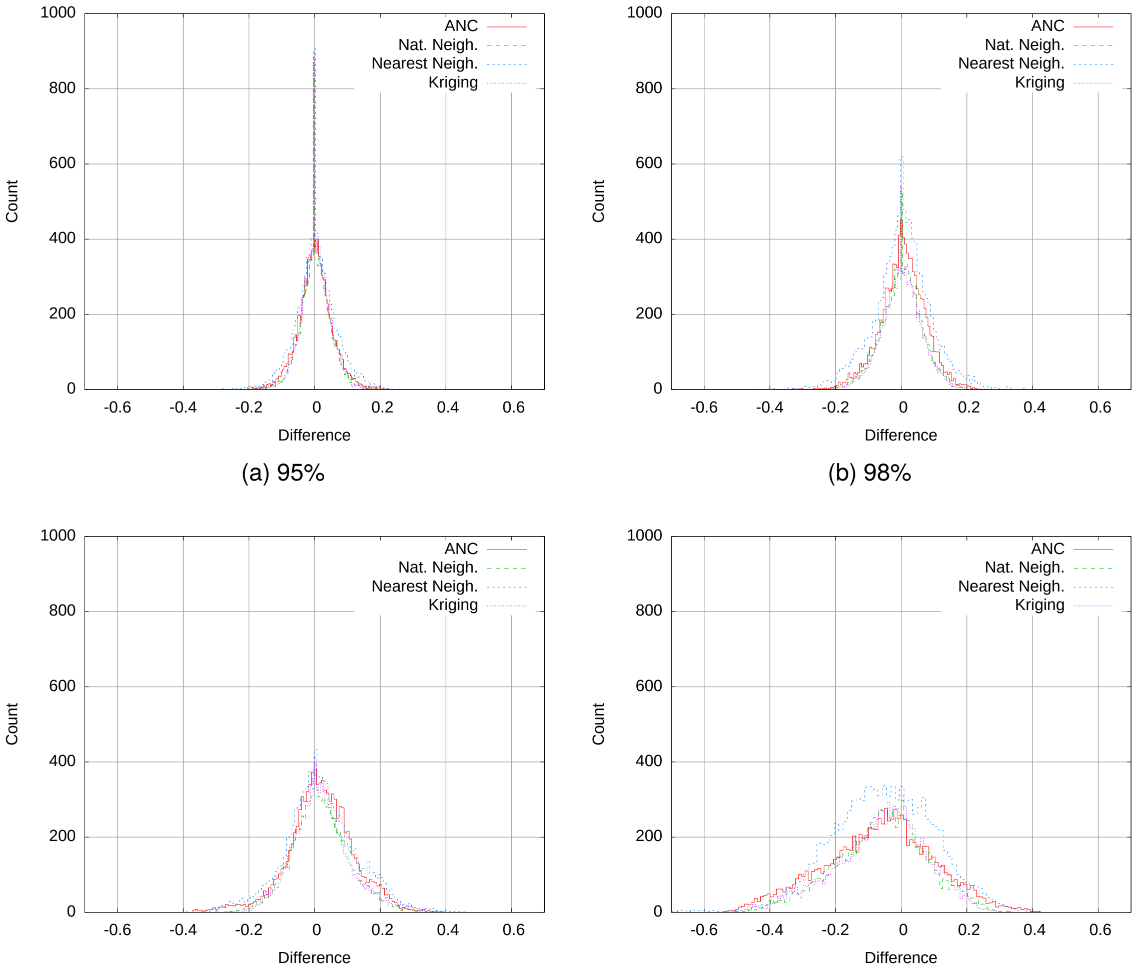


Figure 4.8: An example normalised histogram of absolute errors, with 90, 95 and 99% confidence limits (left to right)

Fig. 4.9 shows some example histograms, created by interpolating simulated data as described and used in Section 3.3.1, on page 47. A single field generated in this way was downsampled to various sparsities and then reconstructing using the interpolation methods used in Section 3.3.1. Histograms were then greated by subtracting the input and output data. These plots clearly show that the error histograms become wider as sparsity increases, as one would expect. A single input field was used to ensure the central limit theorem did not force the results to appear Gaussian, and for illustration.

Tables 4.1, 4.2, 4.3 and 4.4 show percentiles which were calculated from the absolute value of the errors found in the simulation.

These tables and figures clearly show how the histograms widen with increasing sparsity. They also show that the performance of different techniques is comparable. In Table 4.1, for example, the 99th percentile values for BSI, cubic and linear triangulation



(a) 95%

(b) 98%

(c) 99%

(d) 99.8%

Figure 4.9: Error distributions calculated by interpolating downsampled simulated data, differencing the inputs and output and creating histograms. The sparsities used to generate the histograms are below the graphs.

Table 4.2: Absolute error percentiles for 98% sparsity.

Percentile	ANC	Nat. Neigh	Nearest Neigh.	Linear	Cubic	BSI	Kriging
50	0.0434	0.0394	0.0495	0.0406	0.0404	0.0389	0.0376
90	0.1160	0.1046	0.1483	0.1082	0.1088	0.1064	0.1018
95	0.1428	0.1285	0.1847	0.1323	0.1341	0.1309	0.1255
99	0.1952	0.1750	0.2598	0.1794	0.1822	0.1733	0.1666

Table 4.3: Absolute error percentiles for 99% sparsity.

Percentile	ANC	Nat. Neigh	Nearest Neigh.	Linear	Cubic	BSI	Kriging
50	0.0608	0.0545	0.0652	0.0550	0.0537	0.0502	0.0527
90	0.1765	0.1511	0.1944	0.1626	0.1538	0.1274	0.1515
95	0.2181	0.1872	0.2326	0.2008	0.1887	0.1557	0.1876
99	0.3002	0.2463	0.3118	0.2627	0.2480	0.2147	0.2781

based interpolation and natural neighbour are all very similar. The worst performer by far is nearest neighbour. Table 4.4, shows similar trends although the performance

Table 4.4: Absolute error percentiles for 99.8% sparsity.

Percentile	ANC	Nat. Neigh	Nearest Neigh.	Linear	Cubic	BSI	Kriging
50	0.1164	0.0929	0.1110	0.0948	0.0962	0.0834	0.0919
90	0.3108	0.2523	0.2685	0.2400	0.2379	0.2167	0.2615
95	0.3700	0.3074	0.3217	0.2854	0.2854	0.2674	0.3224
99	0.4547	0.3978	0.4501	0.3623	0.3557	0.3327	0.4240

gap has narrowed, and ANC performs worst in terms of the 99th percentile. Examining Fig. 4.9d reveals that this is probably because the distribution of errors for nearest neighbour has a low kurtosis, meaning that most errors are grouped around an approximately zero mean. All of the plotted error distributions appear to be approximately Gaussian, and are centred close to zero, indicating that the methods work well on this type of simulated data.

4.3 Conclusions

This chapter has described some of the artefacts which can occur when using various interpolation methods. It has also showed how error distributions can be used to examine the behaviour of these methods, and discussed and demonstrated the use of percentiles to specify error performance.

The main conclusion that can be drawn from this chapter is that all of the interpolation methods behave well, and have symmetrical distributions which are centred on zero (see Fig. 4.9). Simple tests done on the total electron content (TEC) data in Chpater 3 provided similar results.

The following chapters changes topic to the estimating the motion of a feature known as a tongue of ionisation (TOI), which appears during ionospheric storms and moves across the north polar region.

Chapter 5

Tongue of Ionisation Motion Estimation

This chapter describes several motion estimation techniques with a view to evaluating their suitability to estimating motion in polar ionospheric total electron content (TEC) data which contains storm features, such as tongue of ionisation (TOI).

First, in Section 5.1, the data being examined are described, example frames and a schematic diagram are presented to illustrate the spatial extent of the data. Section 5.4 then introduces some of the assumptions which are common to all motion estimation techniques. Section 5.5 describes *differential analysis*, one of the most fundamental techniques, and the precursor of the subject of Section 5.6, *optical flow*. Section 5.7 introduces *template matching* and *correlation-relaxation labelling*, two related techniques. (These are examined in more detail in Section 6.2.) Section 5.8 introduces correspondence based motion estimation techniques. This includes tracking based on *points of interest*, as well as *boundary tracking* which is discussed in more detail in Section 5.9. Finally, the ability of the techniques to operate on TEC data is evaluated in Section 5.11.

5.1 Data Sources

The chief source of data considered in this, and later chapters on motion estimation (6 and 7), was gathered by global positioning system (GPS) receivers during the 'Halloween Storm', which occurred during the 29—31 October 2003. This was a large geomagnetic storm, with peak activity that occurred between about 20:00 and 23:00 UTC, over North America. During this time, a large region of storm electron density (SED) developed over mainland USA, and moved northward over Canada and through the polar region to the night-side of the Earth. This storm caused a great many operational problems and failures for satellites as well as many problems for land-based communications and power systems.

The available data consist of images created using the Multi-Instrument Data Analysis System (MIDAS) software from the Department of Electronic and Electrical Engineering, at the University of Bath [Spencer and Mitchell, 2007]. These images were generated by extracting integrated electron measurements from paths between GPS receivers and satellites, and performing a 4-D tomographic pseudo-inversion. The region covered by the inversion is a 3-D grid extending radially in an approximate square of dimension 96° around the north geographic pole, such that the voxels have a horizontal resolution of 4° and vertical resolution of 40 km, from an altitude of 100 to 1600 km. TEC maps were created by taking radial line integrals through the grid, giving a 2-D image with a resolution of 25×25 pixels. Fig. 5.1 shows the extent and location of the mapped area, as well as an example of actual data.

Fig. 5.2 shows some example frames of these data. The images displayed are false colour contour maps, and thus represent highly upsampled image data. The actual pixel positions are illustrated in Fig. 5.1. Other figures in this style (in the following chapters) are visualised in the same manner.

The low resolution of this data is the chief source of difficulty in estimating the motion of the TOI directly from image data. The images under examination contain data ranging in value from close to zero to ≈ 250 TECU[1], meaning that a compact representation using 8-bit unsigned integers is possible. This is advantageous because many image processing techniques, such as mathematical morphology are able to operate more efficiently on integer classes.

Also of note, is the fact the motion occurring in the latter part of the sequence is confusing and hard to ascertain *by eye*. This suggests that all motion-estimation techniques will have difficulty analysing the motion in this section; what is hard for humans is likely to be very hard for computer vision and image processing systems.

Previous studies of TOI motion using tomographic images have provided qualitative descriptions from upsampled images, rather than vector fields. In contrast, systems such as superDARN [Greenwald et al., 1995] provide vectors in small patches. The following chapters aim to develop a methodology for providing quantitative descriptions of the motion field, using tomographic images.

[1] 1 TECU or TEC unit, corresponds to an electron concentration of 10^{16} electrons per m^2

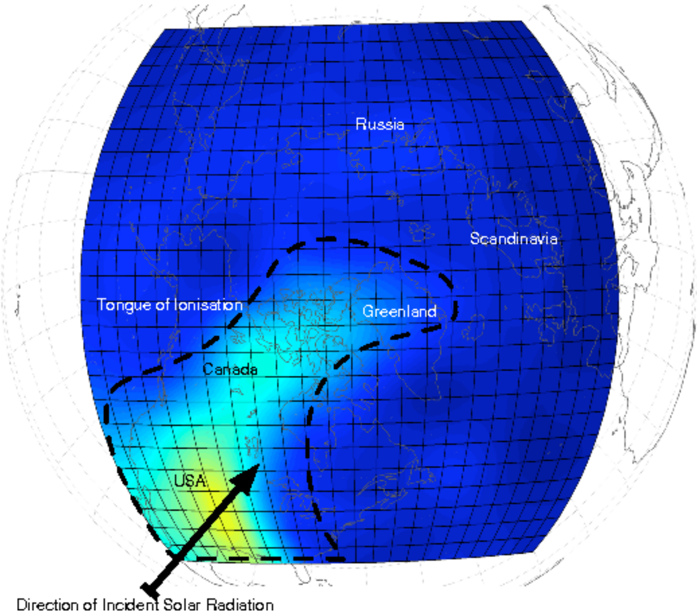


Figure 5.1: The physical location and coverage of the TEC data considered here. The dashed line marks the TOI and various regions and countries are marked, along with the direction of incident solar radiation and material.

5.2 Model Based Approaches

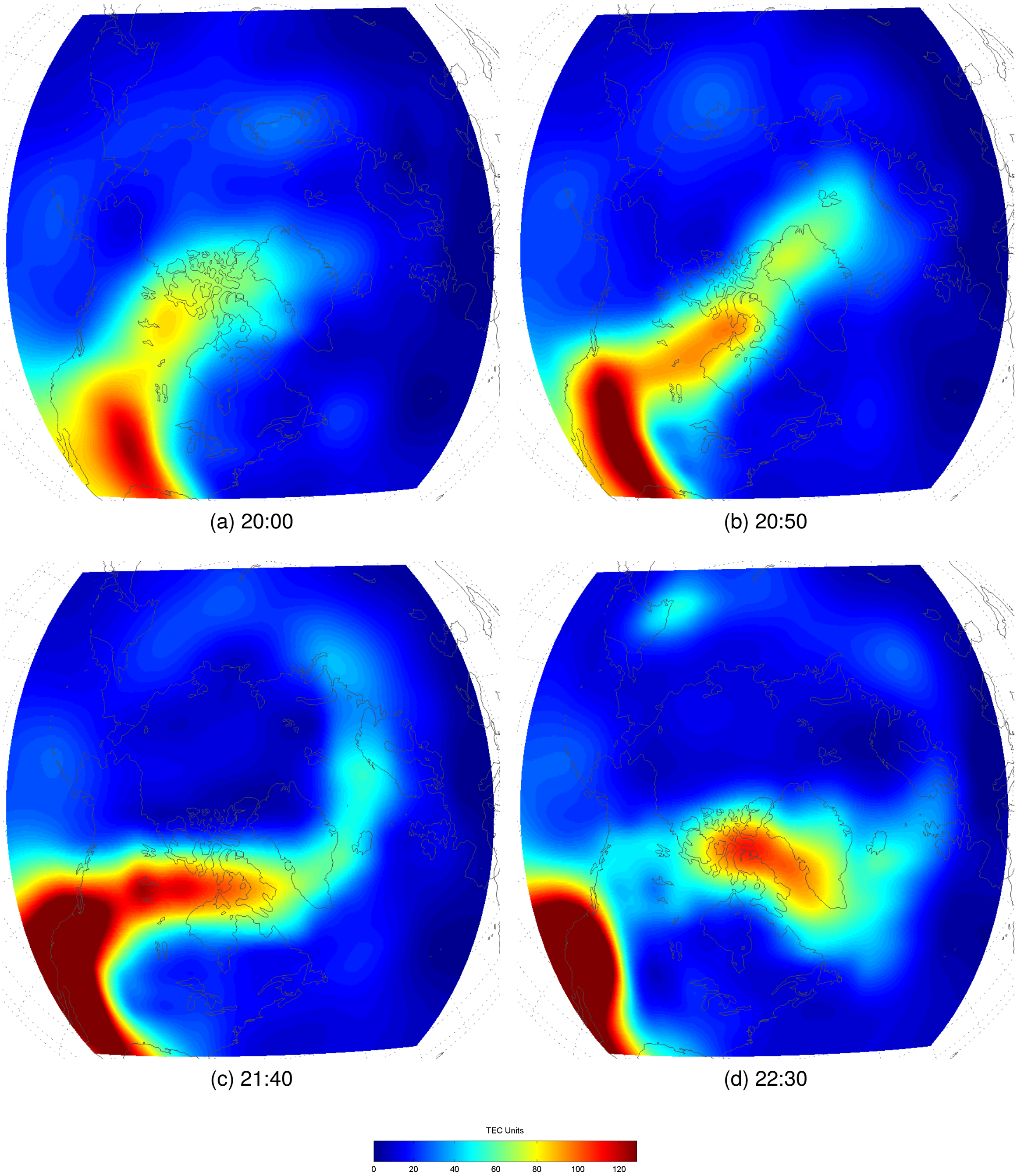
Traditionally, analysis of TOI motion has been carried out using models [e.g. Alexeev et al., 2007], as opposed to making use of image-derived vectors. For example, Fig. 5.3 shows plots of the data being considered here with overlaid modelled $\mathbf{E}\times\mathbf{B}/B^2$ velocity vectors. Here, $\mathbf{E}$ is the electric field, and $\mathbf{B}$ is the geomagnetic field [e.g. Ruohoniemi and Baker, 1998]. These vectors were calculated using the Weimer [1995] electric field along with B_y, B_z and solar wind velocity measurements from the Advanced Composition Explorer satellite data[2]. These allow the electric field to be modelled with time, which when combined with an estimate of the magnetic field allow plasma velocity estimates to be made [Spencer and Mitchell, 2007].

Fig. 5.3 clearly show a simple precessing two-cell convective pattern. Within this pattern, the TOI moves through the centre of the two cells, which are surrounded by concentric rings of vectors. Various different convection patterns can occur, depending on the orientation and strength of the components of the Interplanetary Magnetic Field (IMF) [e.g. Cowley, 1998]. However, this pattern is one of the most common.

It is also interesting to note that in the latter part of the sequence, some vectors switch direction as the convection pattern rotates with the direction of the Sun.

The vectors generated by this simple model are not necessarily indicative of the motion

[2] See: http://www.srl.caltech.edu/ACE/ASC/



(a) 20:00

(b) 20:50

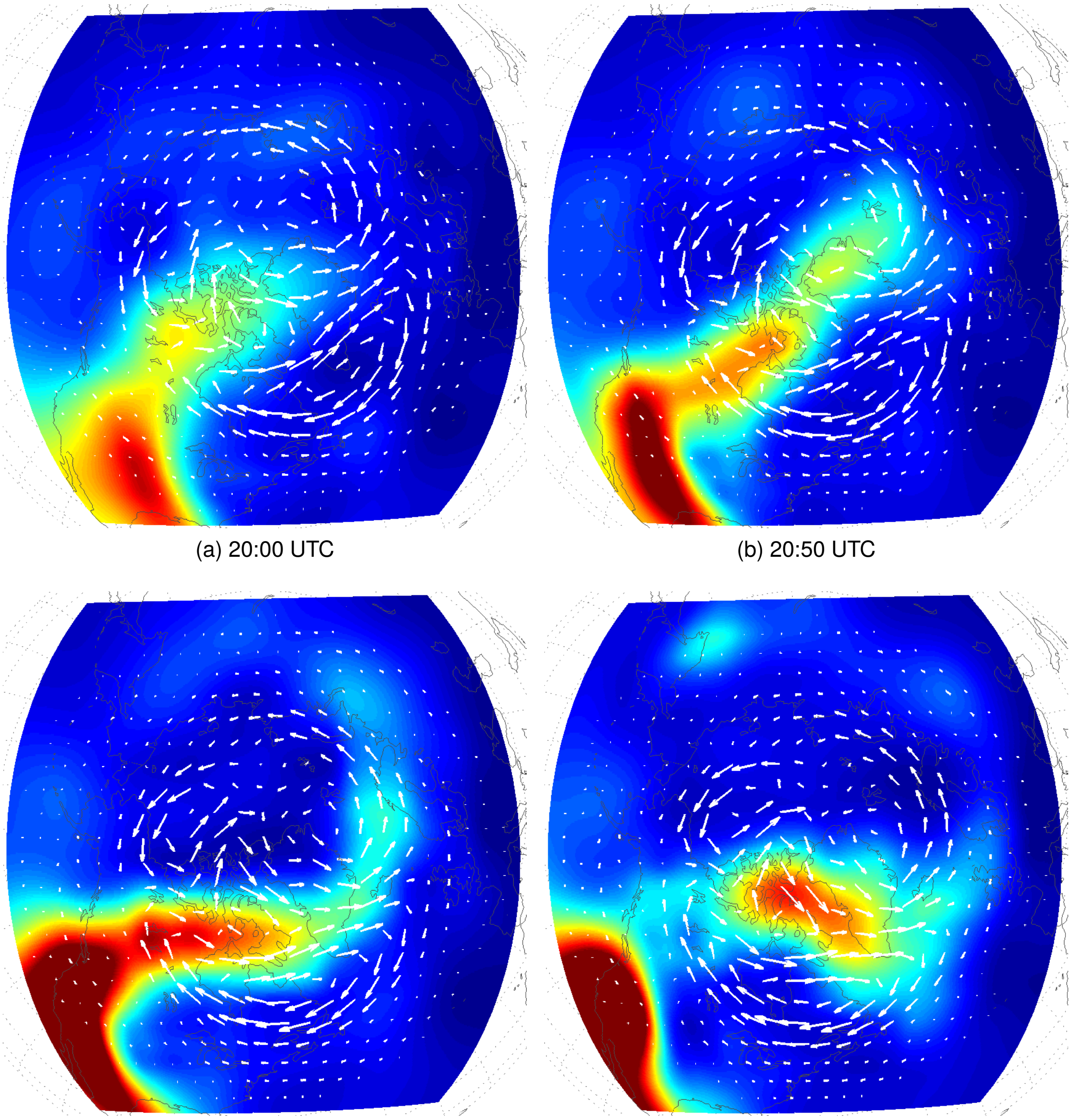
(c) 21:40

(d) 22:30

Figure 5.2: Example false-colour frames from the images sequence from Halloween 2003. Images (a)—(d) are each separated in time by 50 minutes, and are up-sampled by two. The colour-scale is shown below. These figures are vastly upsampled contour plots, for visualisation purposes.

of the TOI, since they model the electron drift velocity over a very large area.

Other methods of analysing TOI motion include model/assimilation methods, such as described by Bust and Crowley [2007], although this method provides *trajectories* rather than vector field, requires large amount of data to fuse, and being based on three different assimilation and modelling stage is likely to be computationally expensive.


(a) 20:00 UTC

(b) 20:50 UTC

(c) 21:40 UTC

(d) 22:30 UTC

Figure 5.3: Plots of modelled $E \times B$ vectors.

5.3 Aims and Objectives

The aim of the remainder of this chapter is to describe various motion estimation methods, and examine their potential suitability for estimation the motion of features within Section 5.1. The following chapters will then examine the performance of these potential methods using the data.

Unlike clouds, which can be used to measure winds, the movement of the TOI is not simply to be used as a tracer for measuring an underlying velocity field, but is *itself* the actual object of interest. This means that the modelled vectors, described above, have limited utility for comparisons outside of the TOI.

Unfortunately, there are no ground truths available with which to validate the vector outputs (a future extension of this work could be the use of vectors from superDARN, although the availability of vectors during storms is low). This means that the overall aim of this process is the extraction of vectors which show *good visual correspondence* with the motion in the image sequence (see Appendix C). As examination of the sequence reveals, and previous sections suggest, the motion is complex, particularly towards the end.

Analysis of the motion of the TOI tends to rely on either vector data from radar, which lacks information on electron content, *or* TEC images, which leads to analysis that is heuristic and qualitative. The aim of this study is to bridge the gap between these two areas by providing quantitative motion analysis based only on TEC maps.

5.4 Assumptions

As with most practical systems, assumptions must be made in order to allow object motion to be estimated and analysed using images. The main assumptions made by most motion estimation systems are given in Hlavac et al. [1999], and include:

- The motion of any given object does not exceed some *maximum velocity*. This means that if the object's position in one frame is known, it must be within a circle of radius $v_m.dt$, which is centred on the last known position, in the next. Where v_m is the maximum velocity of the object, and dt is the time between frames.
- The acceleration of objects is small.
- The motion of objects is *similar*, meaning that points belonging to one object move similarly.
- There is *mutual correspondence* between object points in each frame. The main exception is object occlusion, which can mean that parts of an object present in one frame may not be present in following frames.

The third of these is known as the *velocity smoothness constraint* and is the cornerstone of optical flow techniques. This means that object motion should vary smoothly across an image: a very reasonable assumption (similar to the theory of regionalised variables, see Section 2.5) which arises from the fact that images tend have regional autocorrelation which may be characterised by measuring autocorrelation distance, or using a semivariogram. If this autocorrelation is not present, an image is essentially random noise. This does not necessarily hold where multiple objects are moving in different directions, which would be expected to cause discontinuities in the motion field.

It should be noted that the final two assumptions do not necessarily hold unless the objects being analysed are rigid. Nevertheless, these are the fundamental assumptions upon which techniques such as template matching (see below) are based.

The two main approaches to motion estimation are optical flow, of which differential analysis is a precursor, and correspondence based methods, in which similarity metrics are used to identify objects across as they move across frames. Methods of this type include template matching and boundary correspondence based techniques. The remainder of this chapter discusses these methods, starting with differential analysis in Section 5.5.

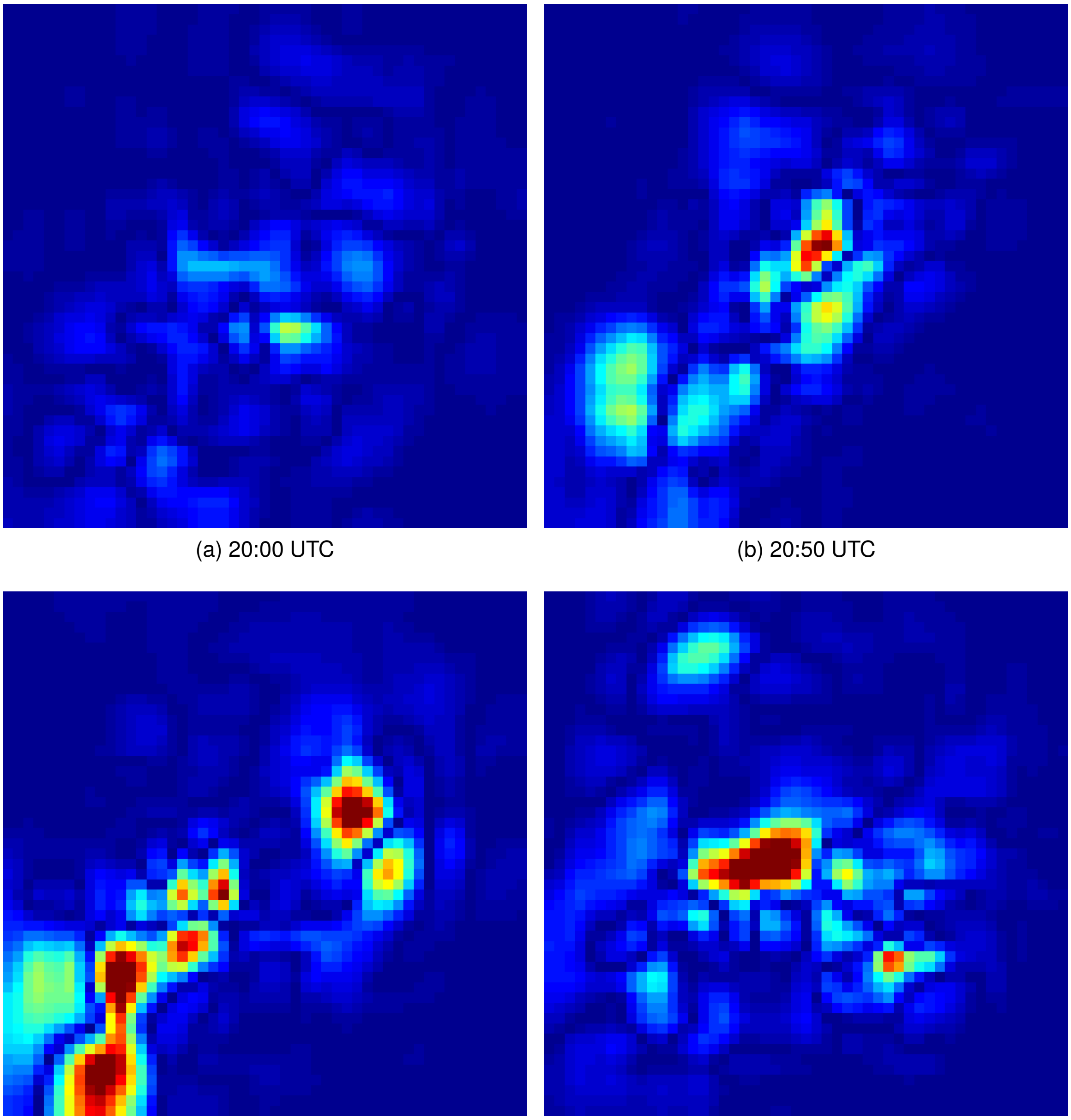
5.5 Differential Analysis

Subtracting images acquired at different times allows motion to be *detected* in a very simple and direct fashion. Given two frames $f_1(x, y)$ and $f_2(x, y)$, motion between them can be detected to give a binary image $d(x, y)$, by subtracting the two images and thresholding the result:

$$\begin{aligned} d(x,y) &= 0 \quad if \quad \text{if}|f_1(x,y) - f_2(x,y)| \leq \epsilon \\ &= 1 \quad \text{otherwise.} \end{aligned} \tag{5.1}$$

Where ϵ is a small positive constant. This allows detection of any motion relative to the background (provided images are registered correctly) but cannot distinguish between different motion directions. Cumulative difference images (essentially moving averages) can be used to acquire more information, but require a stationary 'background' to allow for its proper removal.

Fig. 5.4 shows an example of differential motion applied to four frames of TEC data (shown in Fig. 5.2). For purposes of illustration these have not had a threshold applied and have a scaled colour map. In all of these cases, whilst some motion *would* be detected, this method cannot accurately capture the motion of the TOI and gives the highest values in regions outside of the areas where motion can most easily be seen.


(a) 20:00 UTC

(b) 20:50 UTC

(c) 21:40 UTC

(d) 22:30 UTC

Figure 5.4: Absolute differences between frames at different times in the TEC data sequence.

This is because of the change in noise and background between the images is too high Similarly, there is a change in average value through the sequence, making a static threshold ineffective.

This technique is commonly improved by detecting moving edges Nixon and Aguado [2008]. However, in this case edge detection operators, such as the Sobel operator, which typically uses a 3×3 (or larger) kernel, are too large relative to the scale of the image to provide meaningful results. Applying a simple Sobel magnitude edge detector to the early TEC frames results in two lines, each approximately half the width of the main features, running up either side. This is in contrast to the desired output, which would ideally form a line around the feature. Edge detection is also problematic as differential methods have a tendency to enhance noise.

The use of both spatial and temporal gradients allows differential analysis to reliably detect slow moving edges and some weak edges. This is the conceptual leap used by optical flow techniques, to allow motion *estimation*, rather than just *detection*. Optical flow techniques are described in Section 5.6, below.

5.6 Optical Flow

An optical flow field is a velocity field which represents the motion of points across an image. In this thesis, the term is used to denote a specific class of techniques which aim to calculate motion fields by differentiating in time and space, and applying smoothness constraints on the produced flow fields. The main additional assumption on which optical flow computation is based, is that an object point's brightness is constant over time. This is the *constant brightness* criterion. Optical flow's derivation [see e.g. Nixon and Aguado, 2008] uses this to give the motion constraint equation:

$$u\nabla x + v\nabla x + \nabla t = 0 \tag{5.2}$$

Where $\nabla x = \frac{\delta \mathbf{P}}{\delta x}$, $\nabla y = \frac{\delta \mathbf{P}}{\delta y}$, $\nabla t = \frac{\delta \mathbf{P}}{\delta t}$, u and v are the horizontal and vertical *optical flow* components, and $\mathbf{P}$ is an input image image.

This equation describes how an image changes in time or *moves*, and shows that 'optical flow' or the motion of each pixel, plus the rate of spatial intensity produce this effect together. To extract the optical flow, this equation can be solved by estimating ∇_x, ∇_y and ∇_z using image derivatives. However, this problem is ill-posed, since many values of u and v will satisfy (5.2).

Instead, solutions which minimise the error (ec) and the integral of the rate of change of flow along both axes (es) must be sought. In addition, a smoothness constraint can be applied to ec, because of the assumption that neighbouring points move with similar velocities. This leads to the minimisation of:

$$e = \lambda \times ec + es \tag{5.3}$$

Where λ is a regularisation parameter, or smoothness constraint. This can then be implemented in discrete form using first-order differences and solved in an iterative process. Gradients can be calculated by examining 2×2 pixel, or larger neighbourhoods (or using scale-space methods). However, in the context of these small images, larger neighbourhood sizes will not be as effective. Also, using even-sized neighbourhoods will result in a slight offset to the estimates.

Altering λ, the regularisation parameter, in (5.3) from small to large values increases the weighting of the smoothness constraint over the brightness constraint. Tuning this parameter can increase field smoothness, although care must be taken not to over-smooth the output field, as this will remove detected motion. This is especially true in cases where the motion is expected to be small. (5.3) can be modified to include other terms as required.

There are many different approaches to optical flow motion estimation, for example others based only on first derivatives, extensions using second derivatives, and various modifications to the smoothness constraints. See [Barron et al., 1994] for a more complete review, including evaluations of the methods' performances.

5.7 Template Matching and Correlation-Relaxation Labelling

Where images are noisy, region based motion estimation methods may prove to be more appropriate motion estimators than those based on image derivatives [Barron et al., 1994]. In these cases, region matching methods, such as template matching may be used instead.

Template and block matching arose from the need to of find *template* images, or small objects, in other images [Nixon and Aguado, 2008, Gonzalez and Woods, 2001].

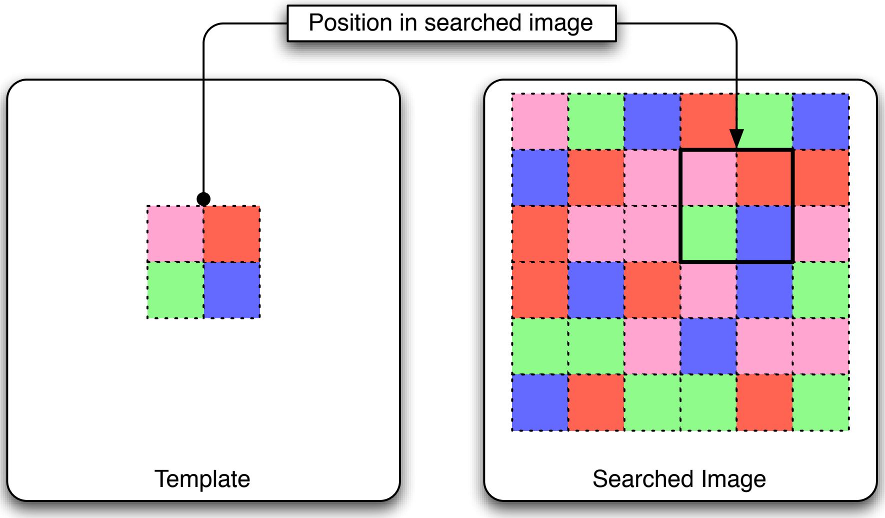


Figure 5.5: A schematic example of template matching

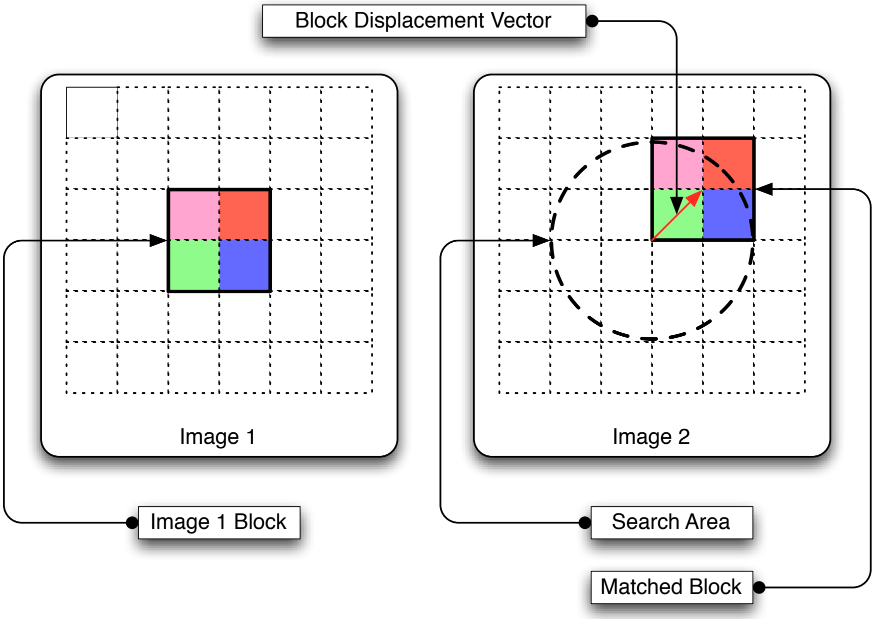


Figure 5.6: The template matching process. Image two is 'searched' for values similar to those within the current template in image one. The result is a vector describing the displacement necessary to shift the template in image one to its position in image two.

This can by carried out by sliding the template image throughout the source and using a similarity or dissimilarity measure (such as cross-correlation, or sum of squared-differences (SSD)) to measure compatibility between the two. In a more general case, rotation and scale must often also be considered, although in motion analysis these are not usually necessary when the motion between frames is small.

As described above, template matching is a technique which may be used for estimating motion between images by comparing blocks in one image, with blocks at various different offsets in a second image. For the purpose of estimating motion, a *block* is simply a square patch of pixels, and the offsets and comparisons are essentially searching for patches of pixels in the second image which are similar to the block in the first. Once this has been carried out, the motion estimate is simply the block-offset which is deemed to most appropriately represent the apparent motion between the image. This is often simply the shift to the best matching block, but can also be adjusted using some additional filtering or regularisation.

In this section, the first image will be referred to as the *source* image, and the second will be known as the *destination*. Fig. 5.6 illustrates the (simplified) process of template matching.

5.7.1 Similarity and Dissimilarity Measures

Similarity measures establish the correspondence (or correlation) between source and destination blocks. The choice of measure depends on computational cost, and performance, and, as with most problems, some trade-offs must be made. The most common similarity and dissimilarity measures are described below, in approximate order of complexity:

- The sum of absolute value of differences (SAVD) is a *dissimilarity* measure, defined as the sum of the absolute value of differences between corresponding pixels in a source block f and destination block g.

  $$SAVD(f,g) = \sum_i \sum_j |f(i,j) - g(i,j)| \tag{5.4}$$

  SAVD can be made invariant to changes in average intensity, by subtracting the mean of from each block before performing the subtraction.

- The SSD is also a dissimilarity measure, and is very similar to the SAVD. It is defined as:

  $$SSD(f,g) = \sum_i \sum_j (f(i,j) - g(i,j))^2 \tag{5.5}$$

  Although the mean intensity is commonly subtracted, as described above.

- The normalised cross correlation coefficient (CCC) is widely regarded as the most effective measure, and is a *similarity* measure, returning 1 when blocks are identical, and -1 when they are *orthonormal*. The normalised CCC differs from the standard cross-correlation in the subtraction of means, and division by variances. This is advantageous as the resulting equation is invariant to changes in mean intensity and standard-deviation and is fairly resistant to noise.

  $$\rho(f,g) = \frac{\sum_i \sum_j \left((f - \overline{f}) \times (g - \overline{g})\right)}{\sqrt{(f - \overline{f})^2} \times \sqrt{(g - \overline{g})^2}}. \tag{5.6}$$

  Where $\overline{f}$ and $\overline{g}$ are the mean values of f and g respectively.

- Ordinal measures [Evans, 1999, 2000b, Bhat et al., 1998] are similarity metrics based on relative rank of intensity values. They were not examined in this study, as using CCC based template matching has been found to be more effective.

After the searches and correlations have been performed for a given block, the output will be a set of displacement vectors with a associated CCC for each vector. The most simple systems use the vector associated with maximum CCC as the motion estimate for a given block, a process known as the maximum cross-correlation (MCC) method. However, when using additional smoothing stages, a more sophisticated approach is to use the top n vectors. As an additional analysis step, the CCC values can be plotted again the vector components as a 2-D surface. These *correlation surfaces* are often noisy multi-modal, and have indistinct peaks, which can make choosing the best vectors difficult [Anandan, 1989, Evans, 2000a]. This is especially true when using remotely sensed data which is characterised by a lack of texture and contrast, and a tendency to contain non-rigid objects. Therefore, relaxation-labelling is used to regularise the motion fields in this work.

5.7.2 Increasing Field Density

Template matching aims to find parts in a destination image which match a given template from a source image. For the purposes of motion estimation, it is simple and convenient for this template to be a square block of pixels in the source image which are searched for in the second image. This process is known as block matching and is a subset of the more general technique of template matching.

As described above, the block matching procedure aims to find matches between blocks in two images. The are several ways of laying out the blocks in the source image, including *non-overlapping* and *overlapping* blocks.

Performing motion estimation with *non-overlapping* blocks gives one motion vector for each block. This can lead to a motion field with few vectors if the images being examined are small, or the blocks large.

Denser motion fields can be estimated by either decreasing block sizes, or by overlapping the image blocks instead of tiling them. A disadvantage of using smaller blocks is that it increases the output noise in the vector field, as smaller blocks reduce the statistical significance of matches. Blocks can be overlapped by shifting adjacent blocks by an amount which is less than the block dimension, with the natural limit being a one pixel shift. Using *overlapping* blocks gives more vectors and a smoother output field than the non-overlapping case due to increased correlation between neighbouring blocks. The disadvantage is a correspondingly higher computational cost due to a larger number of comparisons.

The main weakness of correlation-based methods is their poor performance in situations where texture is lacking. This is a particular problem in the sequence being considered here, as the vast majority of the area of the images corresponds to regions with very low energy and texture. Additionally, a large proportion of each image is background noise. As a result, a large proportion of the vectors are essentially useless. To help mitigate this problem, threshold values can be applied to based on block statistics. By ignoring blocks which do not meet specified conditions (such as variance), motion estimates can be limited to being made in areas in which they are appropriate.

5.7.3 Sub-pixel Block Matching

Motion between frames does necessarily occur in integer pixel increments. This is because the images being examined are sampled representations of a continuous object or area. For this reason, it is useful to be able to measure motion with *sub-pixel* accuracy. This is done by up-sampling the images being examined, before performing block matching. However, up-sampling the images can result in the introduction of interpolation artefacts. To reduce these, the up-sampling factor should be kept low (< 4, if using bi-cubic interpolation). Care should also be taken when up-sampling to ensure that the desired values lie on the same grid as the input samples. This will ensure that the actual input values are included in the output. Additionally, there is no need to up-sample the source frame, which can instead be padded with zeros. Fig. 5.7 shows the process diagrammatically. The output of sub-pixel accuracte motion estimation systems is a vector field where the vector-resolution is $\frac{1}{S}$ (where S is the up-sampling rate), and where the maximum displacement is equal to the search radius.

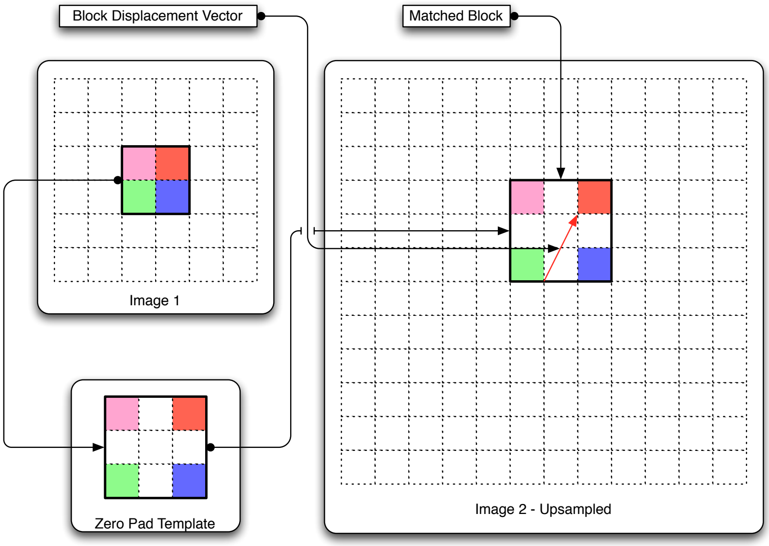


Figure 5.7: A diagram illustrating sub-pixel template matching

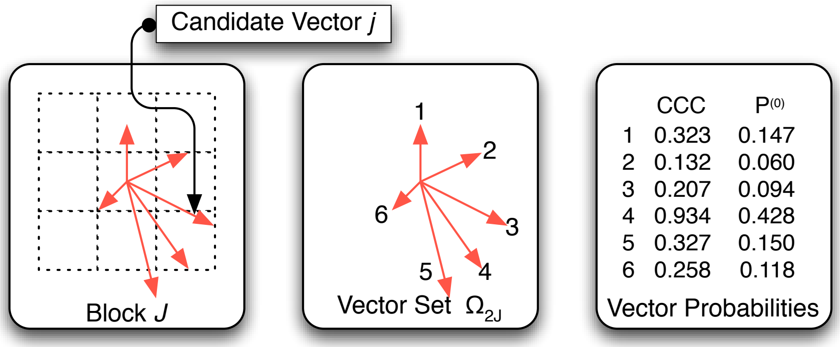


Figure 5.8: Terms used in the relaxation labelling equations (5.7)—(5.10).

5.7.4 Relaxation Labelling

The MCC method has the tendency to produce noisy vector fields because the correlation surfaces it returns are often multi-modal (as described above). To mitigate this problem, additional smoothness constraints may be applied. This section introduces an effective vector-regularisation method known as relaxation labelling (RL), which is based on probabilistic scene labelling [see e.g. Hlavac et al., 1999].

Relaxation labelling, or probabilistic relaxation, is a method of regularising or smoothing vector fields whilst ensuring spatial consistency. It is commonly used when examining remotely-sensed images of non-rigid physical phenomena and where inconsistently moving fields would be unphysical [Evans, 2006, 2000a, Wu et al., 1997]. In video compression for example, a smooth vector field compresses better than a disparate one, because the entropy is lower. Similarly, when tracking clouds, a smooth vector field may be more realistic than a noisy one, as small scale motion is often the result of mismatched areas.

The relaxation process works as follows. Some of the terms used below are illustrated graphically in Fig 5.8:

1. Each block (J) will have various vectors (j) associated with it, and each pair of vectors will have an associated CCC. Vectors which have a low CCC should be discarded.
2. The remaining vectors' CCCs should be normalised, such that the sum across any given block's CCC values is one. These are now considered to be probabilities. The initial probability for block J, and vector j, is denoted $P^{(0)}(J \rightarrow j)$, and is

calculated as follows:

$$P^{(0)}(J \to j) = \frac{\rho(J \to j)}{\sum\limits_{\lambda \in \Omega_{2J}} \rho(J \to \lambda)} \tag{5.7}$$

Where Ω_{2J} is the set of all candidate vectors for a block J.

3. The probabilities are updated using the non-linear relaxation formula (5.8). This formula uses several nested support functions to judge a given vector's similarity with its neighbours in an iterative fashion. These similarity measures are then used to update the probability attached to each vector. The vector from each block with the highest probability attached is chosen as the output.

$$P^{(n+1)}(J \to j) = \frac{P^{(n)}(J \to j)Q(J \to j)}{\sum\limits_{\lambda \in \Omega_{2J}} P^{(n)}(J \to \lambda)Q(J \to \lambda)} \tag{5.8}$$

Where $Q(\cdot)$ is the following compatibility measuring function, which judges a vector j's compatibility against those in neighbouring blocks.

$$Q(J \to j) = \prod_{I \in G_j} \sum_{i \in \Omega_{2J}} P^{(n)}(I \to I)R(I, J, i, j) \tag{5.9}$$

The output from this will depend on both the probabilities of the neighbouring block's vectors, and the output of the function $R(\cdot)$, is the mutual information measure, which depends on vector similarity, and has the following form:

$$R(I, J, i, j) = \exp\left(-|\Delta x_{I,i} - \Delta x_{J,j}|\right) \cdot \exp\left(-|\Delta y_{I,i} - \Delta y_{J,j}|\right) D(I, J) \tag{5.10}$$

Where $x_{I,i}$ and $x_{J,j}$ are the x-components of the displacement vectors i and j in blocks I and J respectively. σ is a parameter that controls the convergence during the iterative procedure of updating the probabilities. In practice, it is usually set to 1. $D(I, J)$ is a function which returns the relative distance between the blocks I and J, given by:

$$D(I, J) = \max\left(0, D_0 - D_{I,J}\right). \tag{5.11}$$

In (5.11), $D_{I,J}$ is the sum of the horizontal and vertical distances between the blocks, normalised to the block dimension. The form used for (5.10) and (5.11) are that of Evans [2000a]. Others may also be used, provided they measure vector similarity in some way. Previous studies, such as Wu [1995] have used polar comparisons arguments in place of the Cartesian form used here.

4. This process is repeated, until the probabilities are stable to within a given absolute value, or for a given number of iterations.

The output field is then generated by choosing the vector with the highest attached probability from each block, and should be a fairly smooth and consistent field.

In some cases, it may be appropriate to add a 'no-match' category in the above process. This is an additional label, indicating that no appropriate vector could be found in a given neighbourhood. As RL aims to improve the smoothness of the field, and an empty field is totally smooth, using a no-match category [see e.g. Wu, 1995] has a tendency to cause mass migrations towards this label, especially with noisy geophysical data. For this reason, using a conditional vector-median post filter may be more germane.

5.7.5 Vector-median Filtering

Vector-median filters remove vectors which the relaxation labelling stage could not regularise, and work by switching vectors with the median of their local neighbours if their magnitudes differ by more than a set amount.

As described by Astola et al. [1990], vector median filters have various useful properties such as a zero impulse response, edge preservation, minimisation of blurring and *no* introduction of new vector values.

The vector median is defined (following the nomenclature in Astola et al. [1990]) as

$$\mathsf{x}_{vm} \in \mathsf{x}_i | i = 1, \ldots, N \tag{5.12}$$

and for $j = 1, \ldots, N$

$$\sum_{i=1}^{N} ||\mathsf{x}_{vm} - \mathsf{x}_i||_2 \leq \sum_{i=1}^{N} ||\mathsf{x}_j - \mathsf{x}_i||_2 \tag{5.13}$$

where x_{vm} denotes the median of a vector x, and $|| \cdot ||_2$ denotes the ℓ_2 norm. Other norms could also be used instead. This can be computed simply by computing the distance from each vector to every other, then summing. The vector which minimises the distance to all others is then chosen as the vector median. If there is no unique minimum value, any satisfying vector may be chosen.

5.7.6 Correlation-Relaxation Labelling

Correlation-relaxation labelling is the extension of template matching using the CCC by the addition of a relaxation-labelling stage, as described above. The process may be further augmented with a vector-median post-filter

The CCC has been shown to be particularly suitable for use with RL. For example, a good comparison of (many of) the similarity measures in Section 5.7.1 was made by Wu et al. [1997], who concluded that whilst the SSD and SAVD are much faster to compute than the CCC, the CCC produces better quality vectors. If these vectors are to be used in a relaxation labelling stage, the number of iterations necessary is lower than when using other similarity measures, which results in faster processing times.

This type of motion estimation has found use in many diverse areas, especially in cases where the temporal resolution between images is too low for optical flow techniques to be effective. Template matching and correlation-relaxation labelling have been widely applied for estimating the motion of non-rigid remotely sensed data. For example, Dransfeld et al. [2006] applied maximum cross-correlation based motion estimation to ocean surface imagery, and found that alone it was unable to provide locally consistent fields which were representative of small-scale ocean currents. Bellerby [2006] used template matching with a scaled correlation measure as the first stage in constructing a mesh to represent a topological transformation. Evans [2006] applied correlation-relaxation labelling to multichannel MSG cloud imagery and found that it gave locally consistent dense vector fields. Evans [1999], also showed the effectiveness of relaxation labelling when used to ordinal measures (based on rank-statistics). These previous results suggest that template matching and correlation-relaxation labelling may be effectively applied to TEC images.

5.7.7 Vector Field Masking

When using the CCC to estimate image motion, areas of image which contain little texture lead to bad matches. This is because flat, or almost flat areas match one-another very well. Areas of this type contain little in the way of information, and can be considered *background*. Using a suitable mask to disable such areas is an effective way of removing mitigating the problems they cause, whilst limiting motion estimates to areas in which they are useful. One example of an effective masking procedure is the use of a variance filter, with a threshold. By tuning the threshold, it is possible to create a mask which disables areas with low variances. These areas correspond to background regions.

5.8 Correspondence Based Techniques

Sparse velocity fields can be estimated by using the correspondence of *points of interest*. The definition of points of interest is fairly broad, and includes both feature detector outputs and object properties, such as shape boundaries.

The *feature* based process process works by detecting points of interest using operators such as the Moravec operator or scale-invariant feature transform (SIFT) features [Lowe, 1999]. These operators generally search for corners (or intersecting straight edges). The points returned from two or more adjacent frames are then used to estimate inter-frame motion, using, for example a probabilistic relaxation framework [e.g. Hlavac et al., 1999], or rectangular assignment [Kuhn, 1955]. The result of this process will be a set of correspondences between detected points of interest which can be converted to motion vectors by simply subtracting the vectors. This class of techniques is useful in large images where the time interval is too long for optical flow techniques to be effective. Due to the fact that most feature detectors, such as SIFT, require multiple filtering operations at different scales, this method is unsuitable for use on the small images used in the studies in this thesis.

The *object property* based process may make use of various different feature properties, such as object centroids or shape boundaries. The use of shape boundaries for motion estimation is considered below.

5.9 Boundary Tracking

Boundary tracking is a broad class of techniques which aim to estimate motion by segmenting object boundaries and then solving correspondences between boundary points across frames. They typically require two stages, *segmentation* and *boundary matching* or *motion extraction*. The segmentation stage extracts the shapes from which boundaries will be traced, and the matching stage calculates correspondences between points on shape boundaries. Subtracting these correspondences gives rise to motion estimates, which can be further processed if necessary.

5.10 Snakes

Snakes, or active contours [Kass et al., 1988] are a method of iterativly finding objects with an image. A common analogy is with the deflation of a balloon around an object, once a certain amount of air has been let out of the balloon, its boundary will describe a hull around the object [e.g. Nixon and Aguado, 2008]. Snakes model this process mathematically, using a cost function (known as an energy functional) to try and match image features, such as edge intensity and brightness, whilst minimising length and

Method	Suitable?	Reasoning
Differential Detection		This method cannot extract vector directions, and instead detects the presence of motion.
Optical Flow	✓	Derivative based approaches can be ineffective in low spatial / temporal resolution images. Nevertheless, this method may prove reasonably effective.
Correlation-Relaxation Labelling	✓	Provided sufficient texture is present region based techniques should work fairly well. Relaxation labelling will help smooth output vectors, increasing resilience against noise.
Feature extraction techniques		Feature extraction methods are ineffective in very low-resolution images, due to a lack of texture and the inability to use multi-scale methods.
Boundary Tracking	✓	Provided boundaries can be accurately extracted, they can be arbitrarily resampled, allowing high-resolution vector estimates along shape boundaries.

Table 5.1: Summary of suitability of motion estimation techniques for use on TEC data.

curvature. Snakes are commonly used to find and track shape boundaries and are usually represented as splines. Splines are good boundary descriptors, which can be also used to infer motion. However, when applied to the TEC images used here, snakes were found to be unsuitable. This is because of the lack of texture in the images, and their low resolution.

5.11 Summary and Suitability for TOI Motion Estimation

Of the methods presented above, several have been demonstrated to be unsuitable for use on low resolution TEC images, and problems have been identified with others. Table 5.1 summarises the described techniques, identifying those which are suitable for use on TEC data, and which will be further examined.

The following chapters will examine these methods in turn, starting with optical flow and correlation-relaxation labelling, in Chapter 6, then considering boundary matching techniques in Chapter 7.

Chapter 6

Motion Estimation using Optical Flow and Correlation-Relaxation Labelling

This chapter discusses the application of optical flow and template matching based techniques to estimating the motion of the tongue of ionisation (TOI) in total electron content (TEC) images. As these methods are data-driven, they are largely de-coupled from the underlying physics and have the potential to provide a set of techniques for analysing specific storm events that is not reliant on physical models [Weimer, 1995, Bilitza, 2001].

Optical flow was introduced in Section 5.6, on page 76 and template matching and correlation-relaxation labelling were introduced in Section 5.7, on page 77. In the following case studies, field entropy values will be specified, as a proxy for field smoothness. These entropy values can be used to assess improvements made to the various methods, and will allow their relative performances to be compared.

Entropy values may be calculated for the vector fields using the motion encoding method of H.263. The entropy values specified here are the number of bits per vector needed for lossless encoding, as specified by Rijkse [1996]. Entropy is a natural measure of vector field smoothness, as smoother vector fields are inherently more compressible.

6.1 Motion Estimation Using Optical Flow

This section examines the application of simple gradient based optical flow (see Section 5.6, page 76) to estimating the motion of ionospheric storm enhancements, as described in Section 5.1, on page 68. It should be noted that the field of optical flow is very large, and the reference implementation used here is not necessarily indicative of *all* methods.

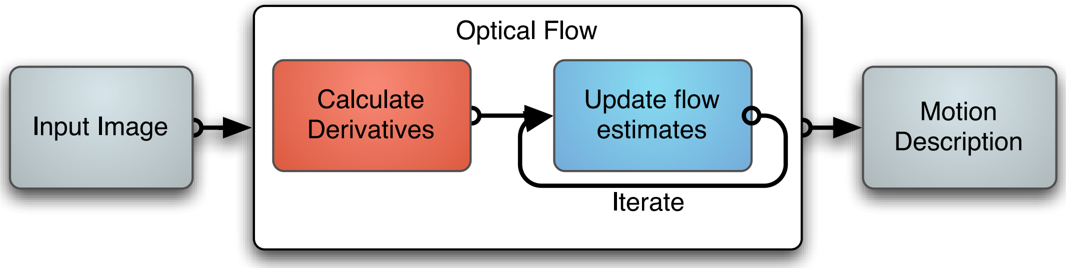


Figure 6.1: The processing pipeline for the optical flow process.

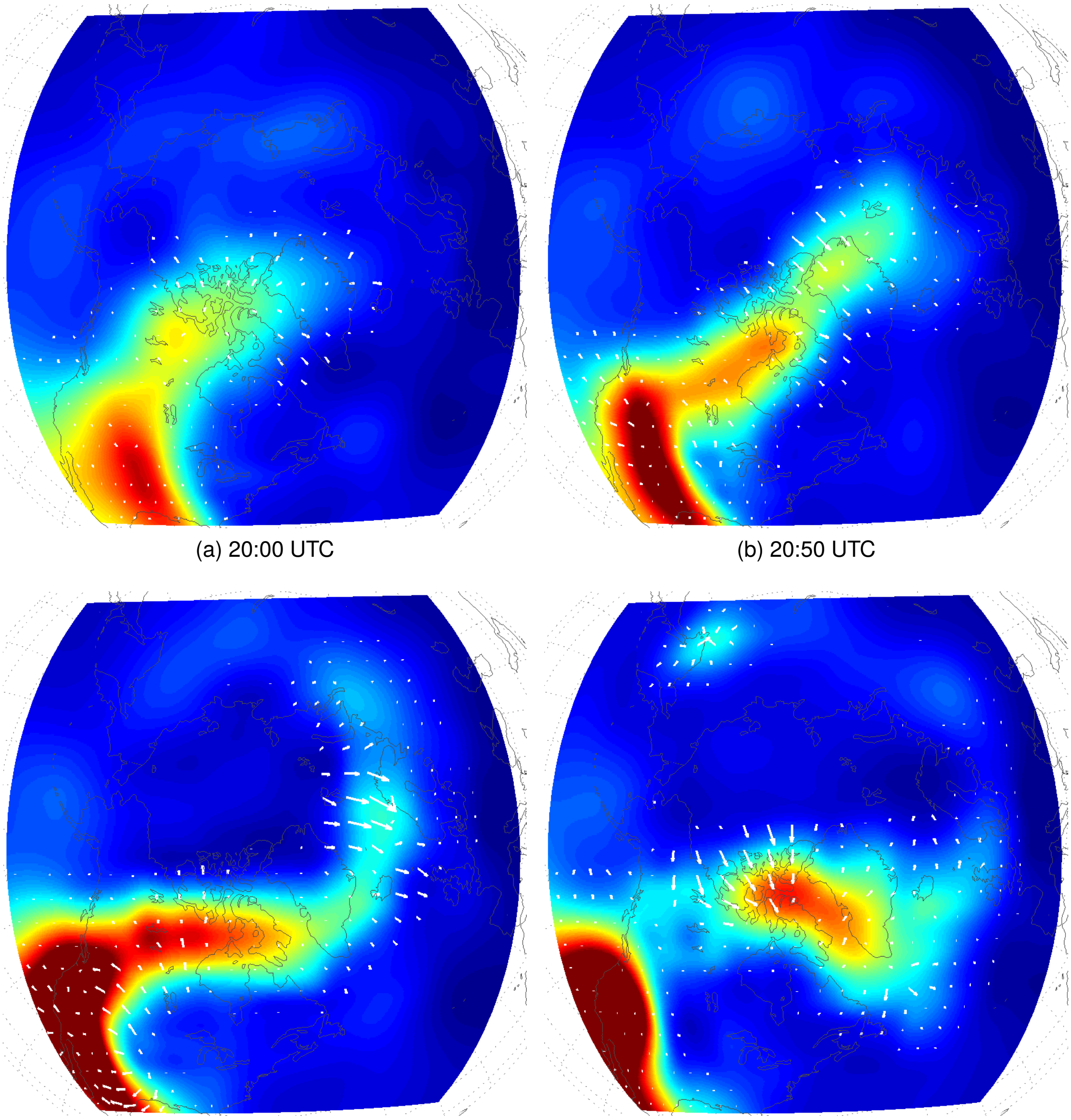
The implementation used in this chapter was based on a modified version of the Matlab code given in Nixon and Aguado [2008] and as such these results are representative of a modern optical-flow algorithm. Fig. 6.1 shows the optical flow process diagrammatically.

6.1.1 Results

To examine the performance of the gradient based optical flow technique, the TEC image sequence was processed using various regularisation parameters (λ), ranging from 0.1 to 10. Ten iterations were used in all cases to ensure stable results.

Figs. 6.2, 6.3, and 6.4 show the results from processing the images using regularisation parameters of 0.1, 1 and 5 respectively. The main thing to note, with regards to these images, is their high degree of similarity.

Table 6.1 shows the average vector entropy values for output fields with different regularisation parameters. The first set shows the median, mean and standard deviations of field entropy values and the second shows the result of masking the same vector field using a thresholded a standard deviation filter ($\sigma = 4$), as in Section 6.2 These values are similar in magnitude to those in Table 6.2 (on page 99), although generally higher in this case. This indicates that the vector fields generated are less smooth than those produced by correlation-relaxation labelling.



(a) 20:00 UTC

(b) 20:50 UTC

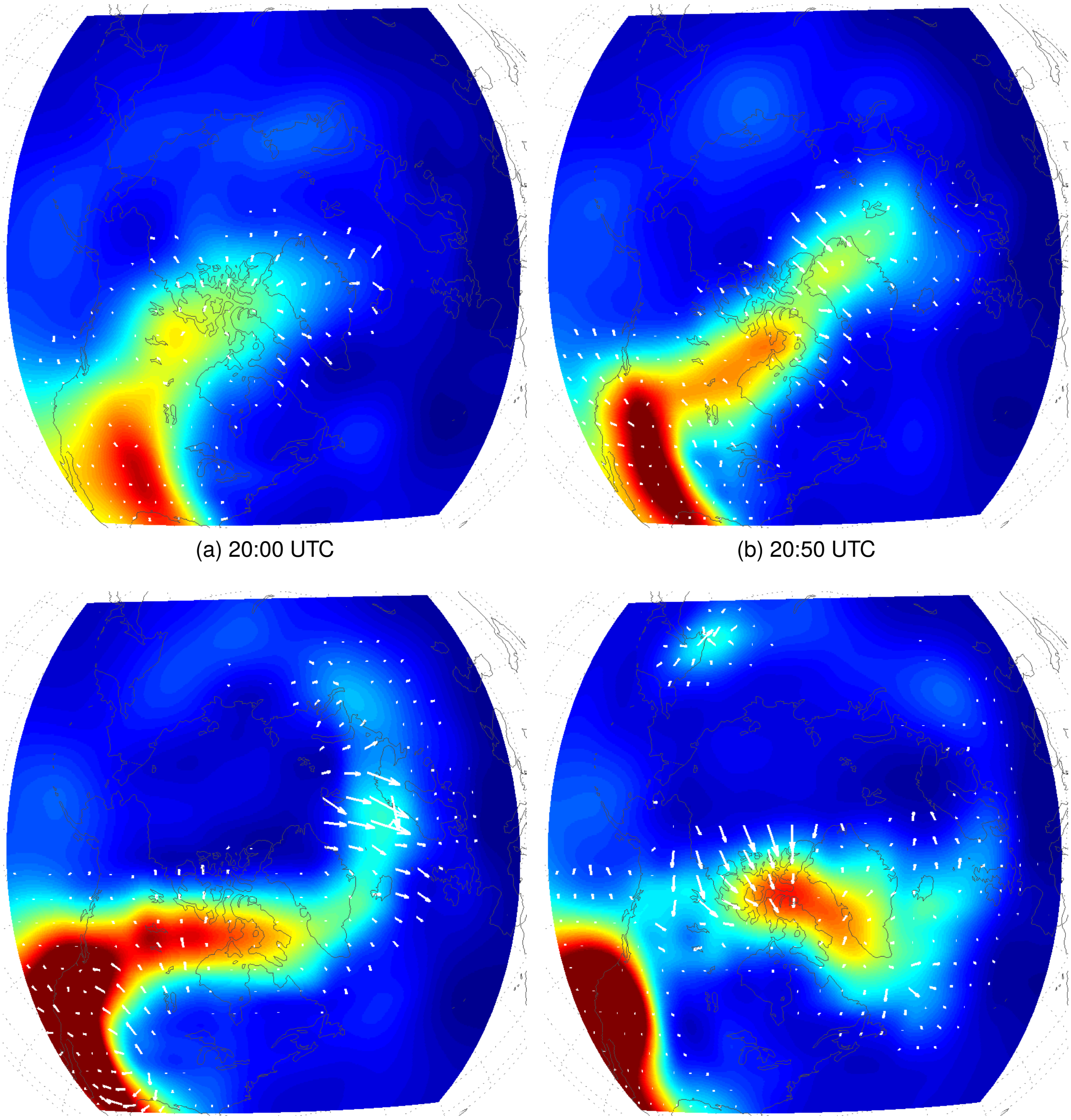
(c) 21:40 UTC

(d) 22:30 UTC

Figure 6.2: Vectors produced using optical flow processing on the TEC image sequence, with $\lambda = 0.1$ and ten iterations.

As table 6.1 shows, changing the regularisation parameter has little effect on the entropy values, and results in an increase in entropy of approximately 30% from the lowest to the highest in the masked case, and less in the standard.

Masking the vector field decreases the entropy values by approximately 2.5 times and decreases the standard deviations of the entropies by a factor of approximately 5. This suggests that removing the 'background' regions is an effective addition to the technique.



(a) 20:00 UTC

(b) 20:50 UTC

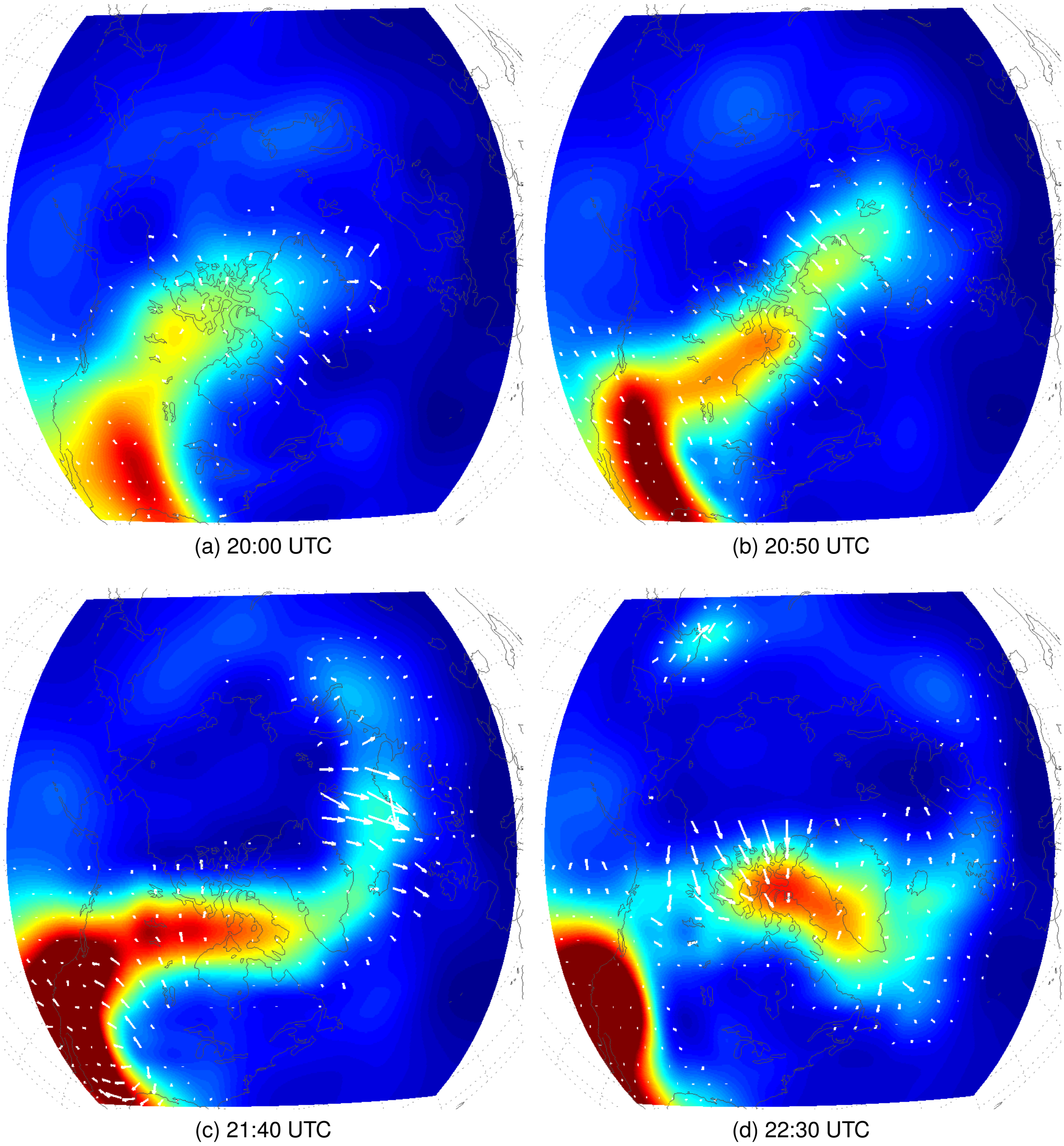
(c) 21:40 UTC

(d) 22:30 UTC

Figure 6.3: Vectors produced using optical flow processing on the TEC image sequence, with $\lambda = 1$ and ten iterations.

Visual comparison of these results with Fig. 5.3 (on page 72) shows similar vectors at the start of the sequence, although the vectors towards the end are somewhat confused. Also notable is the fact that in the last frame of each set (d), the general direction of flow is downwards. This is in contrast to the modelled vectors which show an upward motion in that part of the frame.

The vector fields in Figs. 6.2, 6.3, and 6.4 all show a tendency towards over-smoothing, even when the lowest regularisation parameter was used. Also notable is the propensity for vectors to appear to spread outward around 'bumps'. This can be seen in Fig. 6.3d,



(a) 20:00 UTC

(b) 20:50 UTC

(c) 21:40 UTC

(d) 22:30 UTC

Figure 6.4: Vectors produced using optical flow processing on the TEC image sequence, with $\lambda = 5$ and ten iterations.

in the top-left of the image.

6.1.2 Optical Flow Conclusions

The gradient method of calculating optical flow has been shown to perform adequately for estimating the motion of TEC enhancements. Unfortunately, the results for the last section of the sequence have proven to be complex and lacking in consistency. This is exemplified by Fig. 6.2d – 6.4d, and was discussed in Section 5.1 on page 68.

Table 6.1: Optical flow vector entropies

	λ	median H	mean H	σ
Standard	0.1	5.759	5.765	0.017
	0.2	5.773	5.778	0.018
	0.3	5.763	5.771	0.018
	0.4	5.782	5.791	0.030
	0.5	5.763	5.772	0.023
	0.6	5.769	5.778	0.026
	0.7	5.777	5.783	0.022
	0.8	5.774	5.781	0.024
	0.9	5.779	5.784	0.021
	1.0	5.782	5.785	0.020
	2.0	5.775	5.783	0.026
	5.0	5.783	5.800	0.045
	10.0	5.803	5.813	0.037
Masked	0.1	3.088	3.088	0.003
	0.2	3.226	3.227	0.004
	0.3	3.180	3.180	0.003
	0.4	3.176	3.174	0.008
	0.5	3.168	3.168	0.005
	0.6	3.229	3.228	0.007
	0.7	3.216	3.216	0.005
	0.8	3.266	3.266	0.006
	0.9	3.290	3.292	0.006
	1.0	3.314	3.312	0.006
	2.0	3.308	3.308	0.010
	5.0	3.290	3.291	0.007
	10.0	3.333	3.333	0.012

In general, the calculated entropy values are higher for optical flow than for the correlation-relaxation labelling results (see Section 6.2). This suggests that the fields are less smooth, something which visual comparisons corroborate. However, it should also be noted that optical flow techniques produce vector fields with floating-point magnitudes, a factor which will definitely increase entropy.

Future work in this area would benefit from the application of more recent advancements to optical flow techniques, such as scale-space estimation. This would allow the detection of larger displacements which might more accurately capture the motion of the features in the end of the sequence. However, care would need to be taken not to use filters which were too large, due to the small size of the images.

6.2 Motion Estimation Using Correlation-Relaxation Labelling

This section is concerned with the development of data-driven approaches to estimating the motion of TOI within TEC maps, using template matching techniques. A schematic outline of the process is shown below, in Fig. 6.5.

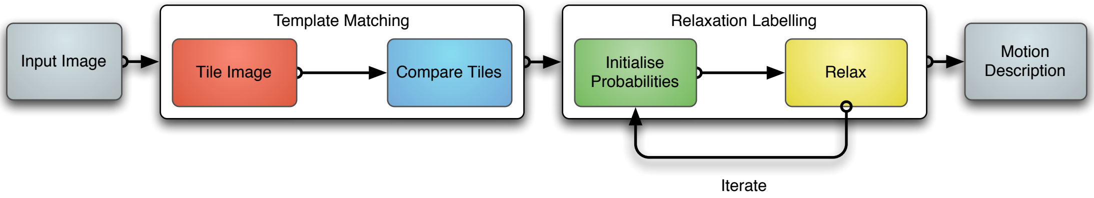
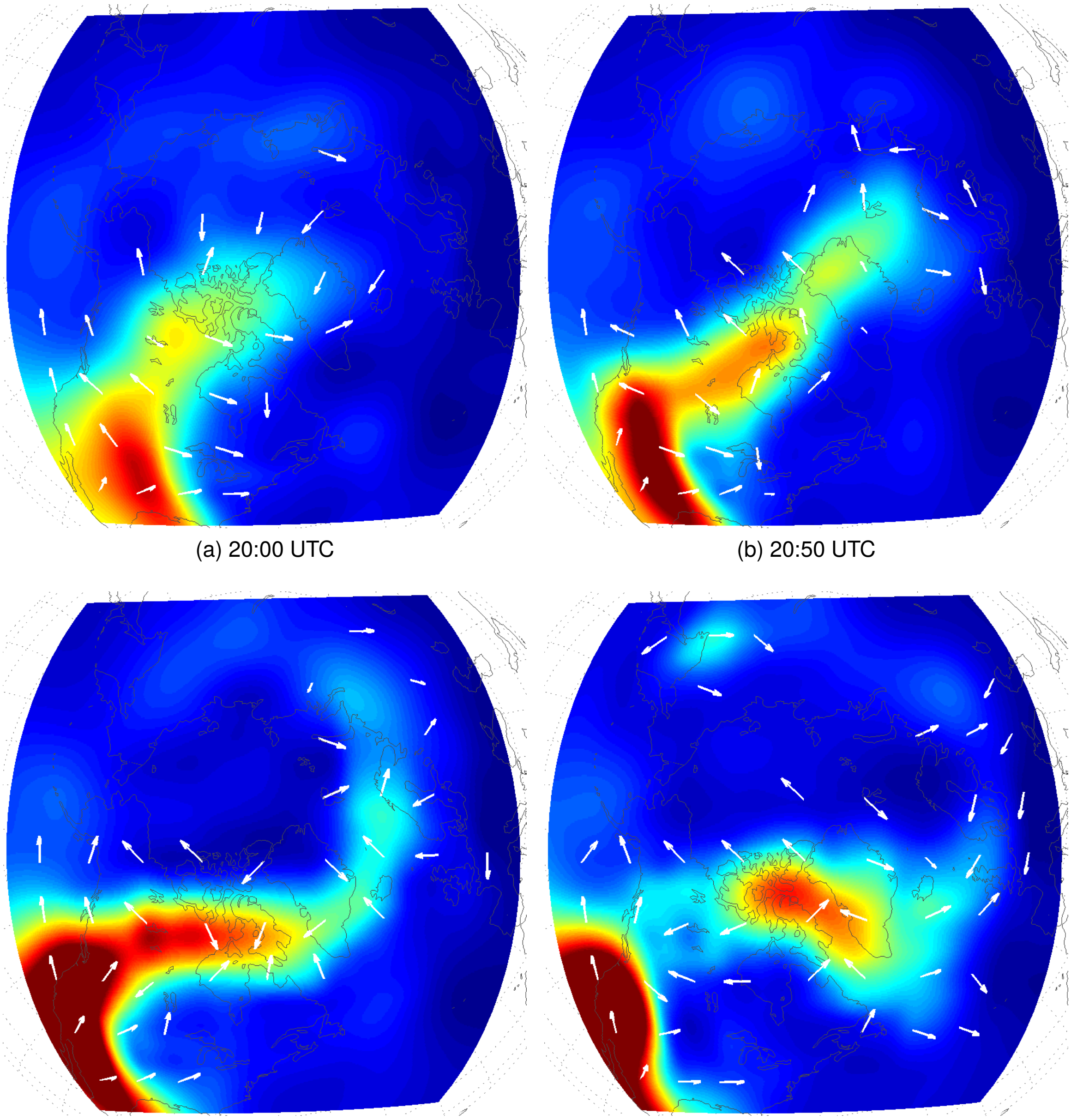


Figure 6.5: The processing pipeline for the correlation-relaxation labelling process.

Template matching (described in Section 5.7, on page 77) using the maximum cross-correlation (MCC) method has been used widely to estimate motion in many geoscience applications, for example, deriving cloud motion vectors [Schmetz et al., 1993] and estimating sea surface currents [Dransfeld et al., 2006].

To successfully apply MCC-based motion estimation techniques to ionospheric TEC images, two main problems must first be overcome: (1) the non-rigid motion of the TOI and (2) the low-resolution of the TEC images.

The low resolution TEC images are characterised by limited textural content. This is a problem which manifests in smooth images which lack the high frequency information that is requisite for successful template matching. Limited texture results in correlation surfaces that are smooth and contain many similar cross correlation coefficient (CCC) values. These problems mean that accurately discriminating between the true motion and anomalous matches is very difficult. Although the relaxation labelling process can overcome this to some degree, when the quality of the input vectors is low it can only select the least bad′ vector from the candidate set. To fully overcome this problem, the application of a vector-median filter (VMF) to the relaxed motion field is proposed [Astola et al., 1990].


(a) 20:00 UTC

(b) 20:50 UTC

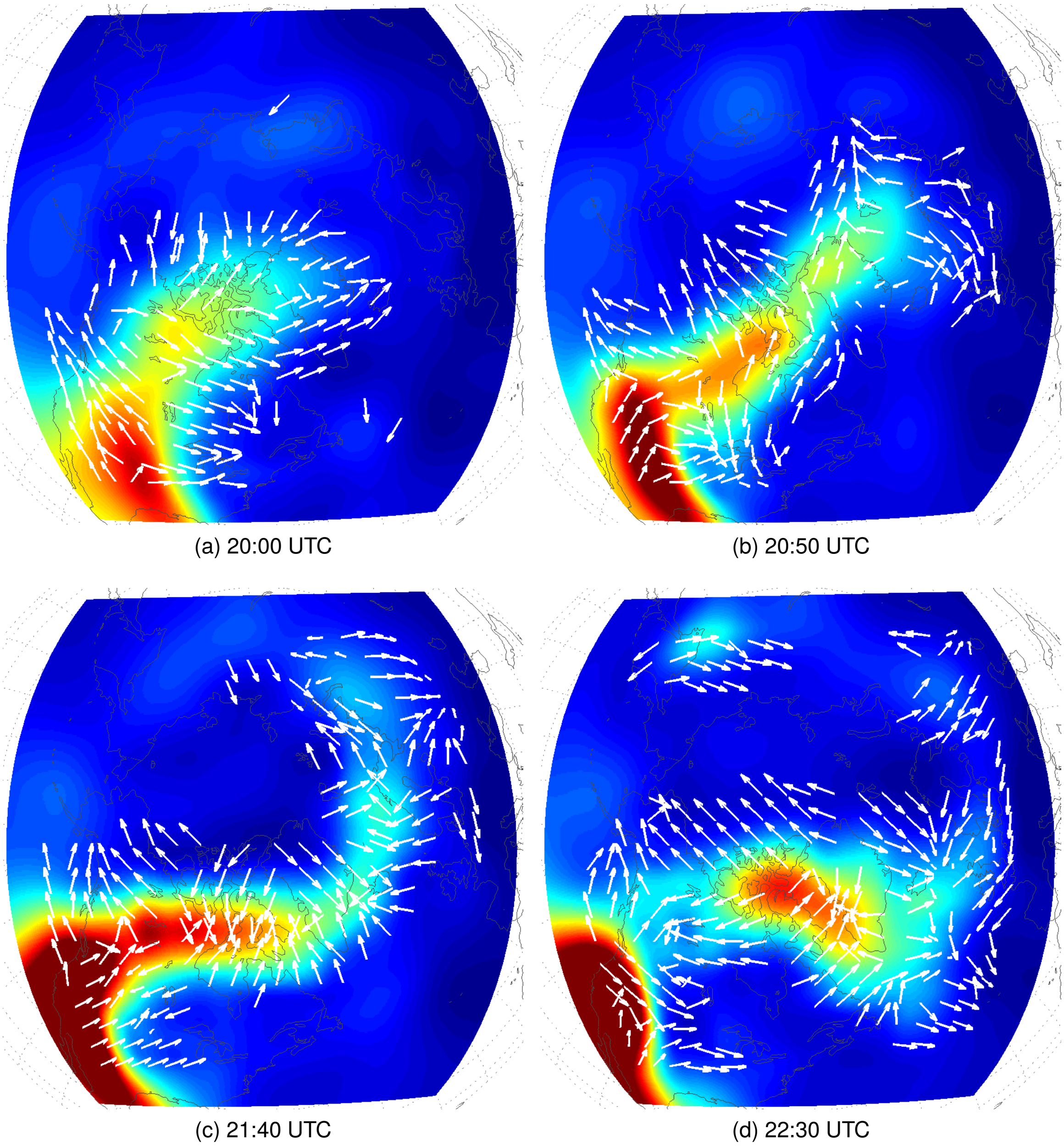
(c) 21:40 UTC

(d) 22:30 UTC

Figure 6.6: MCC motion vectors for non-overlapping 5×5 blocks.

6.2.1 Experimental Results

The effectiveness of the various motion estimation methods was evaluated using the TEC image sequence described in Section 5.1 (on page 68). All experiments used 5×5


(a) 20:00 UTC

(b) 20:50 UTC

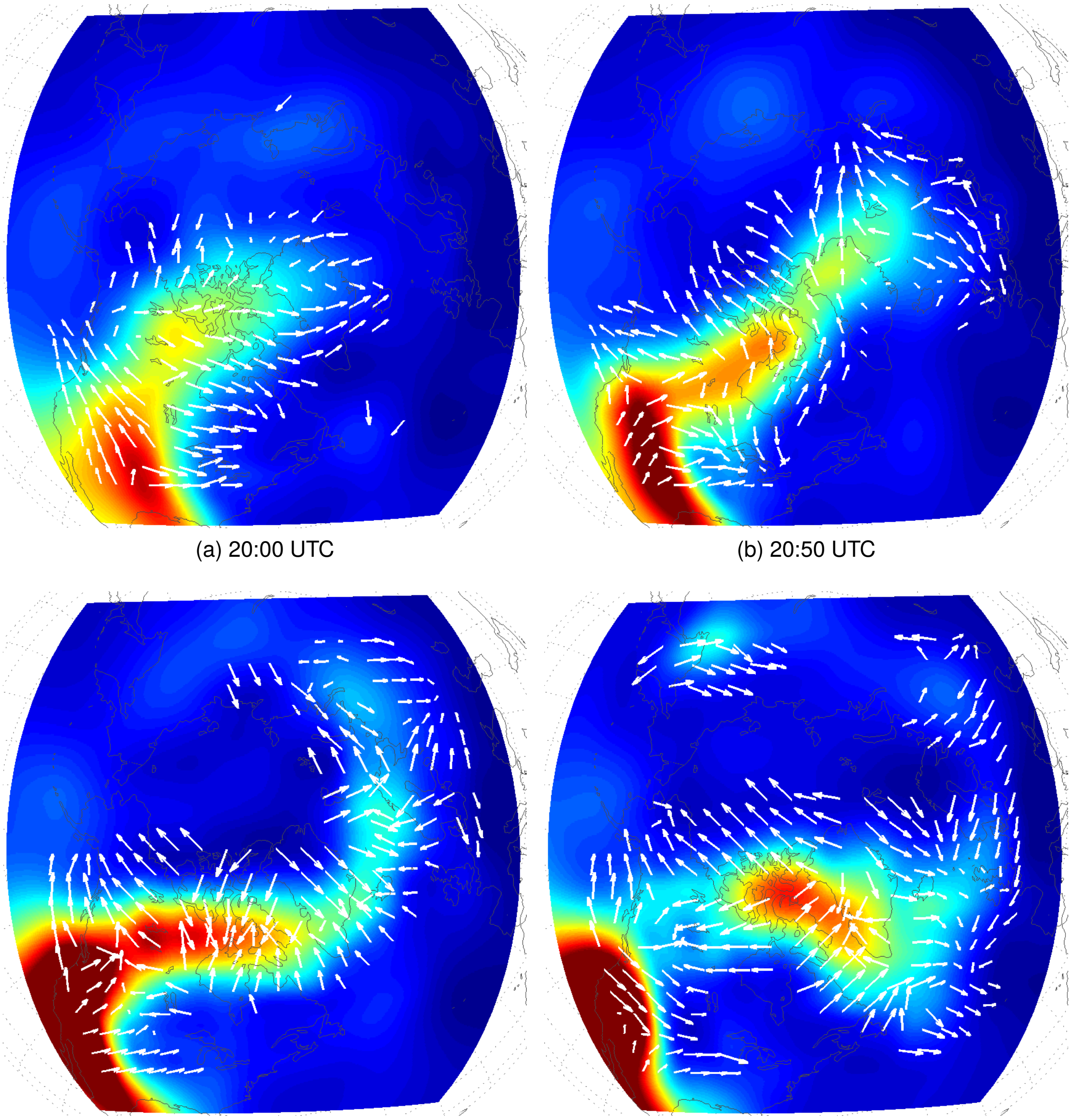
(c) 21:40 UTC

(d) 22:30 UTC

Figure 6.7: MCC motion vectors for overlapping 5×5 blocks. For clarity, the vectors are subsampled before displaying.

blocks and a search radius 5 pixels. Motion vectors were found to a half-pixel accuracy which gives rise to a spatial resolution of 1°. Templates in the image background were not considered for matching, these were determined by thresholding the variance of the templates, such that $\sigma^2 > 16$. Finally, for display purposes, the vector fields were all down-sampled by a factor of three, allowing easier comparisons to be made throughout, and between, the images in the sequences.

Fig. 6.6 shows the motion fields produced using the MCC method with non-overlapping blocks on the four images from the TEC sequence shown in Fig. 5.2. The vectors


(a) 20:00 UTC

(b) 20:50 UTC

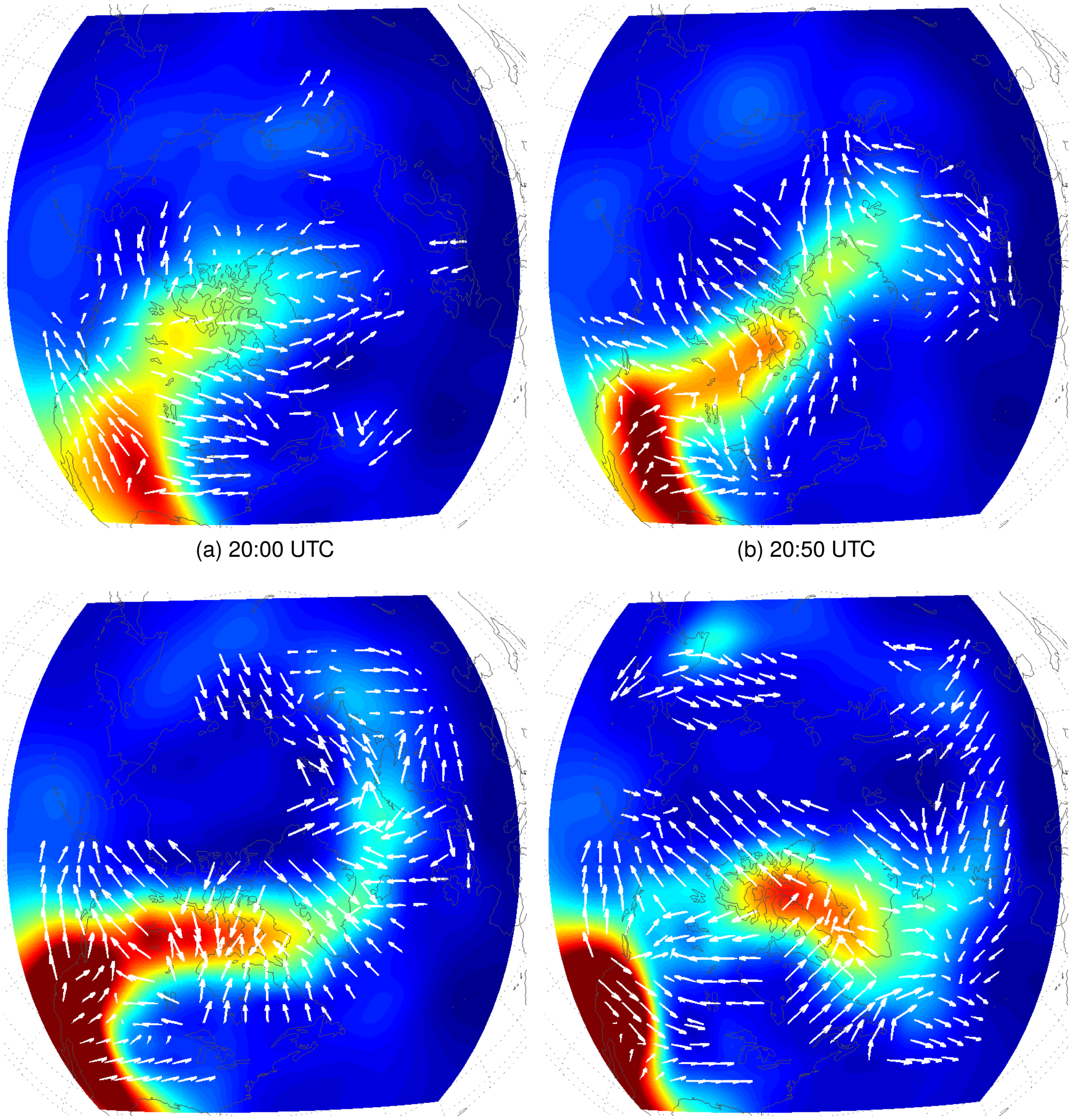
(c) 21:40 UTC

(d) 22:30 UTC

Figure 6.8: Motion vectors after relaxation labelling for overlapping blocks. For clarity, the vectors are subsampled before displaying.

produced by this method are sparse and show some local inconsistencies. The sparsity of the output can be reduced by applying the MCC method to overlapping blocks, see Fig. 6.7. However, whilst this improves the visualisation of the motion of the TOI, there are still many instances of locally inconsistent vectors which show a flow that is not physically realisable. This clearly demonstrates that the MCC method alone is unable to accommodate the non-rigid motion of the TOI.

Applying relaxation labelling to the motion fields produces the smoothed fields shown in Fig. 6.8. The candidate vector sets required for the relaxation labelling were generated


(a) 20:00 UTC

(b) 20:50 UTC

(c) 21:40 UTC

(d) 22:30 UTC

Figure 6.9: Motion vectors field after applying a VMF to the vectors of Fig. 6.8. For clarity, the vectors are subsampled before displaying.

by thresholding the CCC at 0.2. This contrasts with previous studies (e.g. [Evans, 2006]) which used higher threshold values, as the smaller templates used here yield fewer high-quality matches. Ten iterations of the relaxation labelling algorithm were applied. The relaxed results are clearly an improvement on those produced by the MCC method, particularly in the earlier part of the sequence. This can be seen by, for example, comparing the rightmost group of vectors in Figs. 6.7b and 6.8b, where the latter are more locally consistent. Despite the improved smoothness relative to Fig. 6.7, problems with large changes in direction are still prevalent towards the end of the sequence, where the TOI extends and then splits into two regions of enhanced density (see Fig. 6.8d).

The motion fields from Fig. 6.8 can be further smoothed by the application of a VMF, as described in Section 5.7.5, on page 84. Fig. 6.9 presents results created by applying a VMF using a 3×3 window to the vectors in Fig. 6.8. Comparing these vectors with the observable motion (in the sequence) shows good agreement in the earlier frames. The results from approximately 21:40 UTC onwards prove problematic, especially along the ridge towards the top-right. Nevertheless, they show a substantial improvement in vector quality, especially when compared with Figs. 6.6 and 6.7.

In addition to comparing the motion fields with the observable motion in the the TEC sequences, an indication of the expected, underlying motion fields can be obtained by running the $E \times B$ model provided by the MIDAS software package [Spencer and Mitchell, 2007] for the storm event. Comparing Fig. 6.9 with the modelled vectors shown in Fig. 5.3 shows some common features. For example, the early frames in both sequences capture the TOI moving through the centre of the two-celled convection pattern and the latter frames show a reversal in the motion direction around the edges of the TOI.

Table 6.2: Vector entropy (bits/vector) statistics for various kinds of block based motion estimation.

Stage	median H	mean H	σ
Standard	7.457	7.486	0.233
Masked Standard	4.691	4.781	0.544
Sub-pixel	2.924	2.891	0.164
Sub-pixel with RL	2.838	2.809	0.163
Masked Sub-pixel	2.924	2.891	0.164
Masked Sub-pixel with RL	2.838	2.809	0.163

Finally, field entropy values may be examined to show the improvements offered by each additional stage of processing. As seen in Table 6.2 the entropy values were lowered significantly by masking the output vector fields using block standard deviations, this reflects the additional smoothness gained by this, as well as the fact that a large number of the vectors are set to zero.[1] The addition of relaxation labelling further decreases the entropy values, although the savings are nowhere near as large as those afforded by simply masking the vector fields.

[1] The masked entropy values were calculated by setting vector displacements in disabled blocks to zero, and then calculating the entropy of the entire field. Averages were calculated using the number of vectors in the mask.

6.2.2 Correlation-Relaxation Labelling Conclusions

The results presented here show that standard MCC motion estimation does not produce locally smooth motion fields that are consistent with the observable motion. However, relaxation labelling gave improved, more consistent motion fields, particularly in the early part of the sequence. The application of a vector-median post-filter further improves the quality of the output fields. These methods all produce output fields that provide estimates of TOI motion directly from observed data. As such, the results suggest that the proposed method is a viable alternative to the model-based approach for TOI tracking during storm times.

The general conclusions are which may be made regarding template matching and correlation-relaxation labelling are:

- Vector-field density can be increased by overlapping blocks in template matching
- Using sub-pixel motion estimation results in more accurate motion vectors
- Using a low correlation threshold for relaxation labelling increases output quality
- Masking vector fields by using by variance (or standard deviation) filtering input images, and thresholding the output greatly reduces noise, and the incidence of inaccurate vectors.

6.3 Conclusions

This chapter has investigated the application of optical flow and correlation-relaxation labelling to the TEC images described in Section 5.1. Of these two techniques, correlation-relaxation labelling using overlapping blocks and sub-pixel estimation produces fields with the best agreement with the visible motion, modelled vectors and with lower entropies than fields produced using optical flow. This suggests that the correlation-relaxation labelling method is better suited to estimating the motion of non-rigid features in low-resolution remotely sensed images.

Both methods used here would benefit from testing with more data, and data for which ground truths are available would allow more useful performance evaluations to be carried out.

Towards the end of the sequence the motion fields are still hard to interpret. However, this is not unexpected as it difficult for human observers to track the TOI through these frames when, due to a combination of electron recombination and large changes in morphology, the inter-frame correspondences are very hard to discern. The next chapter will describe a less traditional, two-stage segmentation and boundary tracking approach to solving this problem. Of particular interest is the potential improvement to be gained through the fusion of vectors acquired using multiple synergistic techniques.

Chapter 7

Motion Estimation using Boundary Matching

Chapter 5 introduced the problem of the estimating motion of storm enhancements in low-resolution total electron content (TEC) imagery and presented various methods which are typically used to do this. Chapter 6 then examined template matching and correlation-relaxation labelling based methods, and also looked at optical flow techniques. This chapter continues along the same vein, by examining how motion can be estimated by matching shape boundaries. This somewhat less traditional approach has various useful features, such as the ability to resample boundaries, giving more vectors, thus increasing field resolution. One downside is the fact that vectors are estimated along shape boundaries, a problem which can be mitigated by estimating warping transforms by using interpolation or regularisation, both of which can increase vector field density, by estimating displacements within the segmented regions.

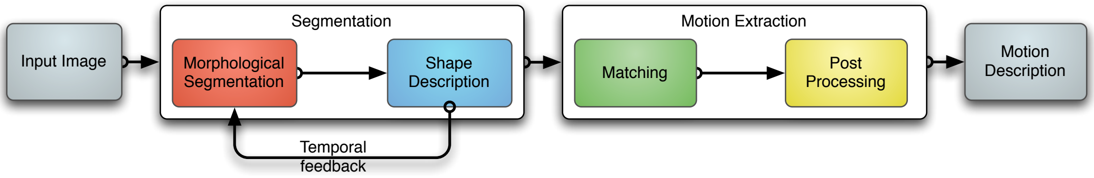


Figure 7.1: The processing pipeline for the curve matching process.

The motion estimation technique described here is based on a two stage approach, illustrated in Fig. 7.1. These two stages consist of *segmentation* and *motion extraction*. The segmentation stage aims to consistently and unambiguously extract first the objects of interest, and then their boundaries. The motion extraction or boundary matching stage aims to find a correspondence between the boundaries of shapes in adjacent frames

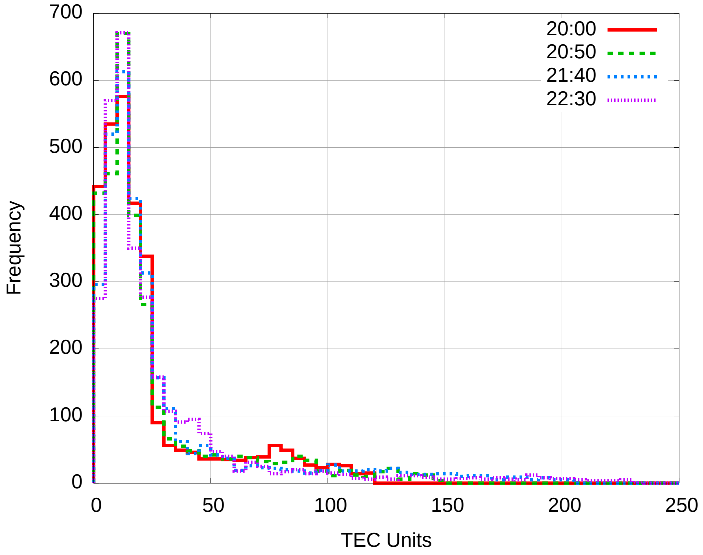


Figure 7.2: Histograms showing the distribution of TEC values in the data. The peaks on the left represent background data (low values), and the area on the right represents both noise and the TOI.

that is in some sense optimal. These two stages are introduced below.

7.1 TOI Segmentation

The TEC images can be considered to be composed of three main components: regions of storm electron density (SED) corresponding to significant regional maxima [assumed to be the tongue of ionisation (TOI)]; noise and artefacts corresponding to other maxima (and minima), and the remainder of the image which can be considered background. The aim of the segmentation stage is to correctly discriminate between the TOI and the other image components. Once the images have been segmented, it is possible to extract information about the enhancements, including their boundaries. These can then be used for estimating the motion of the TOI, and hence studying its evolution.

A very large number of segmentation techniques exist, but are not suitable for the data being considered here. For example, *optimal thresholding*, a simple technique for separating objects from background or clutter [Nixon and Aguado, 2008]. This approach is appropriate when the object of interest has a different range of values than the background, as is the case with the TOI. It is also designed for use on data with a bi-modal intensity distribution. However, histograms of the TEC data for the four images from Fig. 5.2 (on page 71), shown in Fig. 7.2, show that the TEC data is characterised by a distribution with only a single peak, which corresponds to the background value. The TEC values which correspond to the TOI occur in the tails (on the right hand side) of the illustrated distributions. This, coupled with the presence of noise, and the fact that the maximum TEC value changes over time means that simple thresholding is doomed to failure. Other segmentation methods, such as those based on image texture are also inappropriate due to the low resolution and smoothness of the data.

7.1.1 Morphological Segmentation

This section describes the contents of the ‘segmentation’ box in Fig. 7.1. The overall aim of the segmentation stage is to provide accurate descriptions of regions of interest with the frames in order to allow the second motion extraction stage to proceed.

Segmentation can be carried out using various different methodologies, from simple thresholding to more complex methods such as the watershed transform, or graph connectivity [e.g., Soille, 2008], and new methods are continually being developed. The main segmentation framework considered here is mathematical morphology and more specifically, attribute morphology, which is described below.

Mathematical morphology provides a robust toolkit for segmentation and feature extraction based tasks. The extension of traditional morphology for processing greyscale images and later into more general *attribute* morphology has paved the way for the development of a great many different types of segmentation system. Greyscale morphological operators, such as opening and closing by reconstruction, can be used with marker image and thresholding to create simple, shape-preserving systems as-is, and attribute morphology can be used for segmentation in situations where specific knowledge about features being extracted (such as contrast, area or statistical moments) are available. Morphological filters also have a number of advantages over traditional filters, most importantly shape preservation and idempotence – that is, they do not filter what has already been filtered.

The field of mathematical morphology (MM) provides a number of tools for feature extraction and segmentation and many of these have been applied to remotely sensed data. For example, Soille and Pesaresi [2002] provide a topical review of MM applications in remote sensing, ranging from segmentation to skeletonisation. Pesaresi and Benediktsson [2001] propose a morphological segmentation method using the derivative of the morphological profile with residuals from opening and closing by reconstruction. This method is extended to multiple scales by using increasingly large structuring elements. More recently Akçay and Aksoy [2007, 2008] designed a hierarchical MM segmentation technique for remotely sensed imagery. This method creates morphological profiles by applying structuring elements of increasing size to each spectral band. Image patches extracted at different scales are then merged to create meaningful objects using spectral homogeneity and connectivity to optimise the groupings. Greyscale and attribute MM are described in more detail below.

The field of MM in the mid nineteen-sixties, in France, where it was initially used to analyse the geometry of porous media. Initial work on was completed by G. Matheron and J. Serra, and the seminal work entitled: *Eléments pour une théorie des milieux poreux* was published by Matheron in 1967. In this paper, he formalised various concepts, such as granulometries, or morphological profiles which are still heavily used today.

MM has its roots in geoscience and interestingly, the Matheron mentioned here is the very same person who was later credited with pioneering the field of Geostatistics, mentioned in Section 2.5, on page 23.

Greyscale connected component morphology is a branch of mathematical morphology [Soille, 2004, Hlavac et al., 1999, Acton, 2001] which operates on greyscale images, and works by considering the connectivity of level-sets.

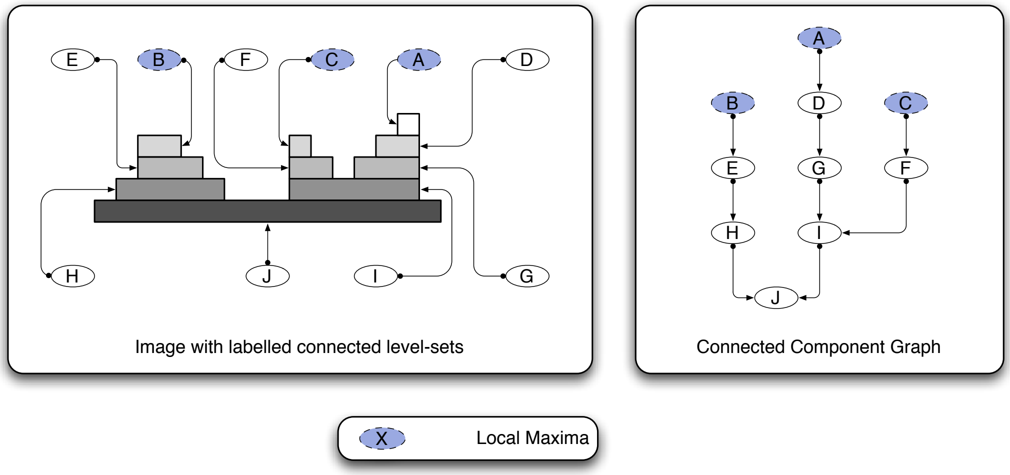

E
B
F
C
A
D
H
J
I
G
Image with labelled connected level-sets
A
B
D
C
E
G
F
H
I
J
Connected Component Graph
X
Local Maxima

Figure 7.3: An example greyscale image cross-section with labelled connected level-sets, and connected component graph showing the relationship between the various level-sets.

For example, imagine the grey-level of a greyscale image to be a vertical displacement. A level-set is a horizontal slice through the image, at a specific value. This means that a given pixel value can be a member of several level-sets (up to 256 for a typical, 8-bit greyscale image). Fig. 7.3 shows an example image cross-section, illustrating the concepts of level-sets and local maxima. For example, in Fig. 7.3, the set A is a subset ($\subset$) of D, which, in turn, is a subset of G, which is a subset of I, which, finally, is a subset of J. The hierarchy of sets can conveniently be represented using a directed graph structure, where the root set (J in this case) is the base node of the tree. As the tree represents connectivity between sets, it is known as a *connected component graph*. Formally, a set (C_1) in a connected component graph is linked by an edge to another set (C_2), only if C_2 is a superset of C_1 (written as $C_1 \subset C_2$). The direction of connectivity is always towards the larger set. In this example, the edge ($\mathbf{E}(\ldots)$) between sets is written as $\mathbf{E}(C_1 \to C_2)$ [Meijster and Wilkinson, 2001, Salembier et al., Apr 1998]. The right hand panel of Fig. 7.3 shows the connected component graph for the example image on the left.

This structure is useful in many situations, as the graph nodes can be used to represent many different image components and properties. This could include lists of pixels in the sets or measures such as area or contrast (see below), perimeter or moment of inertia. One situation in which this structure is particularly useful is area morphology, where the graph nodes (also known as vertices) relate to the areas of the level-sets. The two main operators available in area morphology are *closing* and *opening*. These modify an image by removing features based entirely on their area. This means that (unlike traditional morphological operators) there is no structuring element, and so none of the associated artefacts. As the parameter of interest is component area, area morphology lends itself well to scale-space applications, Acton and Mukherjee [see e.g. 2000], Mukherjee and Acton [see e.g. 2002].

Area opening removes bright or 'ascending' objects which do not meet a specified minimum area. Area closing removes 'descending' objects which do not meet the specified criterion. Area opening is equivalent to moving down from the top of the connected component graph until the area is exceeded, and area closing is the equivalent to moving up from the bottom until the criterion is exceeded. Fig. 7.4 shows how an area opening using an area of three would alter an example image, and its connected component graph. in the image, A, B, C, D, and F would be removed, as they all correspond to components with areas of less than three.

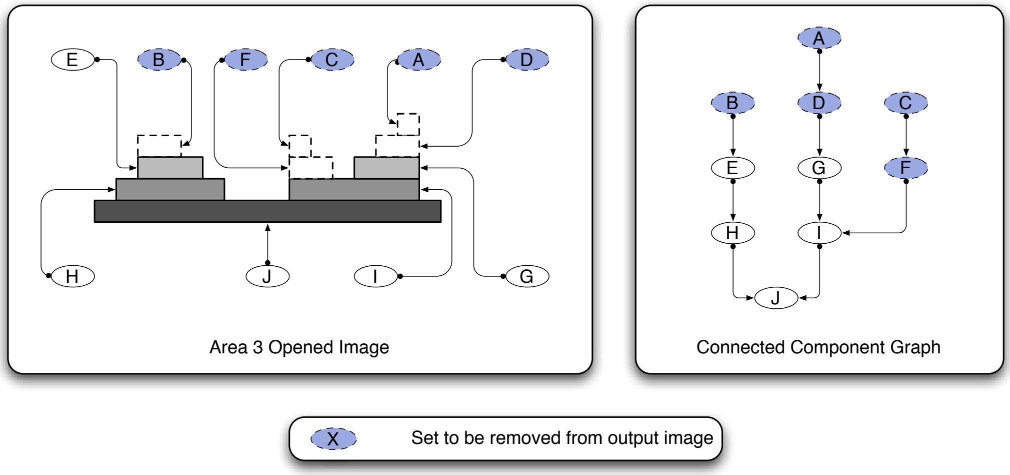

E
B
F
C
A
D
H
J
I
G
Area 3 Opened Image
A
B
D
C
E
G
F
H
I
J
Connected Component Graph
X
Set to be removed from output image

Figure 7.4: The example cross-sectional image from 7.3 showing the effect of an area closing with an area of three (assuming the image depth is only one pixel).

The connected component graph is also useful for describing and implementing *contrast* *morphology*. This is similar to area morphology, except that instead of using the area to decide what to remove (which is purely a property of a single level-set), it makes uses of the set's contrast. In the case of opening, the contrast is the difference between the highest connected local maxima of which the set is a subset, and the set's value. Sets with contrasts below a certain threshold will be removed, whilst others will be left intact. Fig. 7.5 shows how a contrast opening would alter an example image, and its connected component graph. In this case, A, B, C, D, E and F would be removed, as all of these are components whose contrasts are less than three.

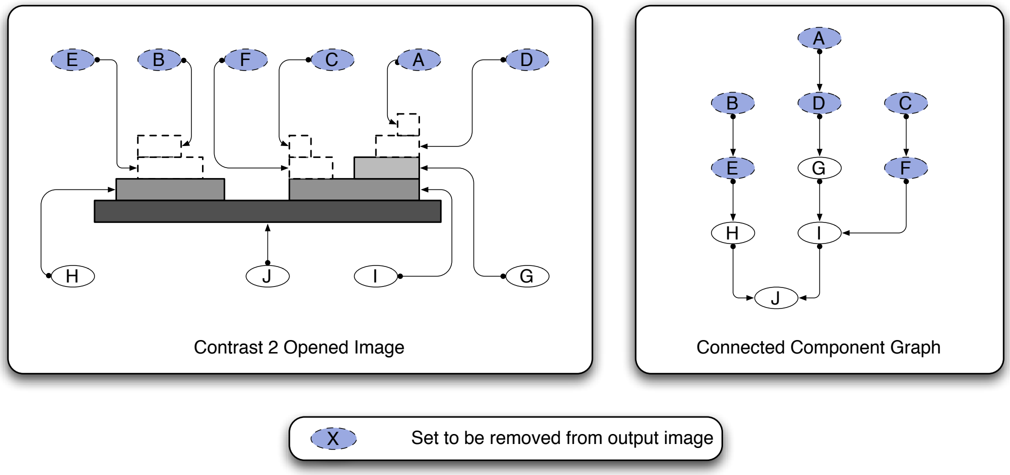

E
B
F
C
A
D
H
J
I
G
Contrast 2 Opened Image
A
B
D
C
E
G
F
H
I
J
Connected Component Graph
X
Set to be removed from output image

Figure 7.5: The example cross-sectional image from 7.3 showing the effect of an contrast closing with a contrast of two.

Using contrast as the morphological attribute has several advantages over areas based approaches in images where high gradients are present. Where the gradient is high, a small range of areas can correspond to a large vertical extent, and in these cases using contrast instead of area measures could be beneficial. This is illustrated in Fig. 7.6, which shows how a large contrast can be found across an object which only has small change in area across it.

7.1.2 Morphological TOI Segmentation

Whilst the segmentation schemes proposed in Soille and Pesaresi [2002], Pesaresi and Benediktsson [2001], Akçay and Aksoy [2007, 2008] were developed for application to very-high resolution (VHR) images, the images in this study are characterised by extremely low resolution. In these images, the lack of texture and small size mean that multi-scale segmentation methods are unsuitable. To help overcome these problems a segmentation scheme based on attribute morphology is proposed. Typical attributes include area (the most commonly used feature), moments, orientation, contrast and many other measurable properties of image structure. In this case, *attribute* morphology is

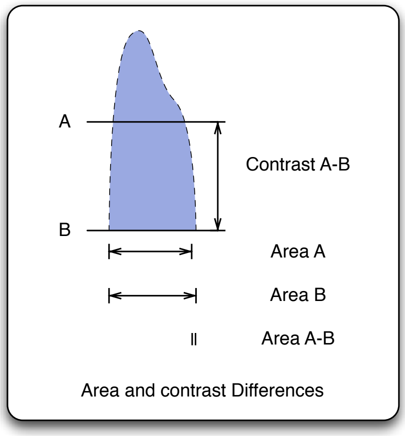


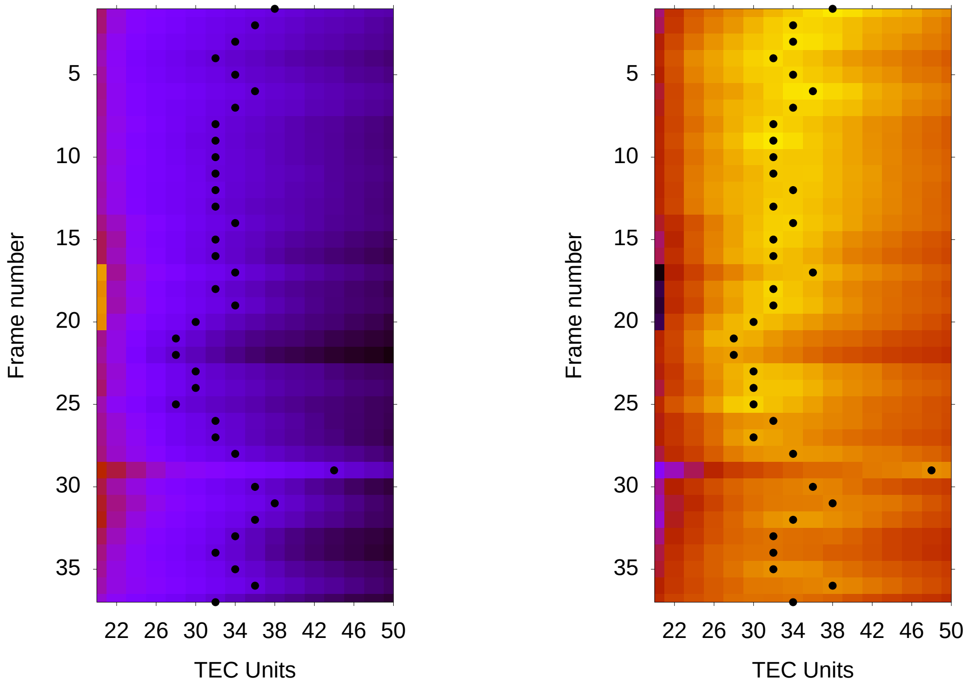
Figure 7.6: The difference between contrast and area parameters. The area can be very insensitive to height, leading to a small change in area consuming a large vertical section of image.

advantageous when compared with *standard* morphology as it does not require the use of structuring elements, all of which would be large when compared to the dimensions of the images, and so introduce significant regions of noise. It also allows the filtering action to be finely controlled by attributes other than shape and size, producing segmentation results that can be tuned according to the properties of the features under consideration. Another useful property is that of shape preservation, a property shared with reconstruction based methods [e.g. Soille and Pesaresi, 2002]. Such methods are interesting as they are able to completely remove specific features, whilst leaving other entirely intact.

Contrast was chosen as the most suitable attribute for segmenting the TEC data due to its low sensitivity to gradient relative to other attributes, such as area. Whilst the peak SED value within the TOI shows considerable variation over the duration of the storm, the background TEC value is much more stable. Performing an attribute closing with a suitable contrast limit essentially flood-fills any basins with a contrast value less than the chosen attribute limit.

In order to choose a suitable contrast value with which to operate the morphological filter, ground-truths were generated by segmenting the images by hand. These were necessary due to the noisy nature of the frames which meant that inter-frame differences were not necessarily consistent and to sudden appearance and disappearance of small features was common. This process gave rise to a set of binary images, one for each frame, in which the salient regions were marked.

To find the contrast value for each frame which gave rise to the results most closely matching the corresponding ground-truth images, the data were filtered using a range of contrasts, and the outputs compared to the ground truths. The optimal value found in



(a) Area Comparison

(b) Correlation Comparison

Figure 7.7: Comparisons of area difference (a) and correlation (b) between frames segmented at various contrasts, and hand segmented frames. In both cases the contrast with the closest matches to the hand segmented frames was 33 TEC Units.

this fashion was then used as the starting point for the segmentations. Fig. 7.7 shows two image maps illustrating the best contrast parameter for each frame, using two different criteria – area (darker means a lower difference between the test and hand-segmented frames), and correlation (brighter means a higher correlation coefficient between the test and hand-segmented frames). In both cases, similar behaviour can be seen; on average, the contrast value which gave rise to results closest to the hand-segmentations was 33 TEC units. Fig. 7.7 shows that the performance of the attribute-based segmentation is relatively insensitive to the choice of contrast value. Therefore, although the contrast value here was selected using hand-segmented TEC images, the availability of more data would enable the starting contrast value to be estimated using training data with little overall loss in performance.

To improve the temporal consistency of the segmentation results over the image sequence, a temporal feedback system was used, along with the ground-truth derived contrast value. This system operated by performing an initial segmentation using a contrast closure, followed by a small-scale area-opening to remove small peaks which were considered clutter. This was then thresholded to give a binary mask identifying the TOI. The feedback system operated by calculating the change in foreground area between each frame and the frame preceding it, and then increasing or decreasing (as appropriate) the contrast level in small increments if the area differed by more than 10%.

If the contrast was altered, the image was re-segmented. This process was repeated in an iterative fashion, until the segmented area was stable or until 10 iterations had occurred. This helped ensure the correspondence of objects between frames, leading to a smoother motion description. A flow diagram illustrating this process is shown in Fig. 7.8.

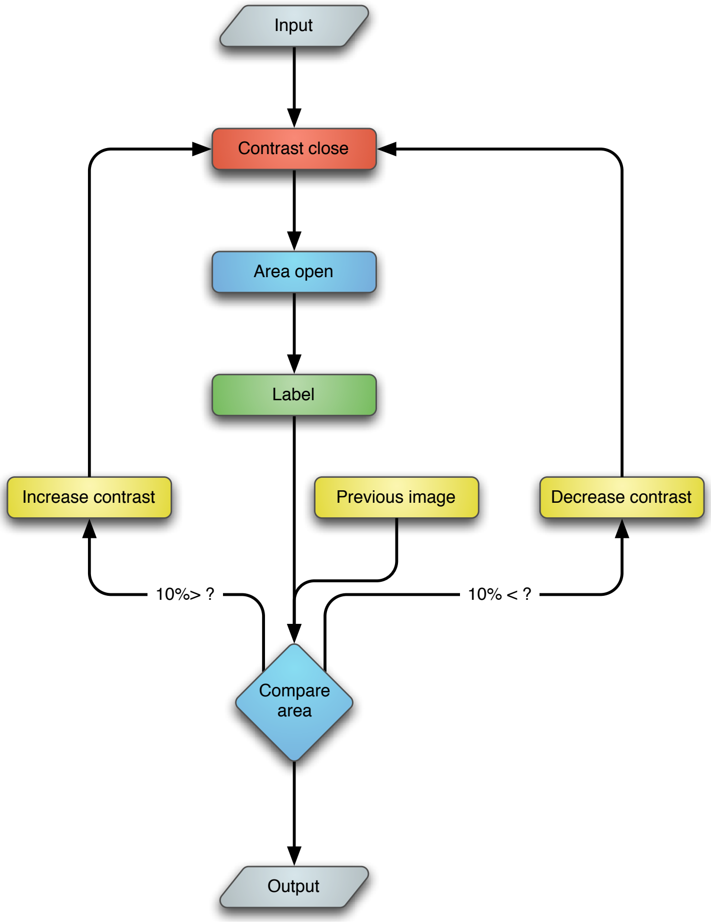
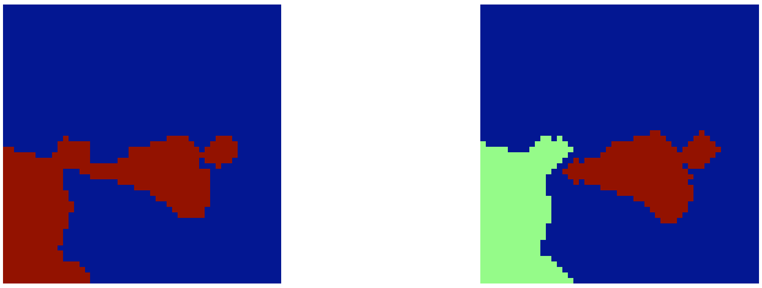


Figure 7.8: Segmentation process showing area based feedback.

The above process yielded results which are similar to the ground truth images whilst maintaining area consistency between frames. However, in a few cases the segmentation was too generous, meaning that some areas which were separated in the hand-segmented images were left joined by the feedback process. This occurred when saddle-shaped structures joined features which would otherwise be separate. To mitigate this problem, a further stage was added to the process. By inverting the input images, masked by the segmented images, and then applying the watershed transform, thin saddle points joining areas were severed, as demonstrated in Fig. 7.9. The segmentation results for the images from Fig. 5.2 are shown in Fig 7.10. To obtain final TOI representations in a form which is smoothed and can be re-sampled, the image boundaries were traced and then converted to a smoothing-spline representation.



(a) (b)

Figure 7.9: Reducing under-segmentation of SED islands using the watershed transform. Example TEC image from 22:35 UTC on 30th August 2003 (a) before and (b) after watershed transform post-processing.

7.2 Boundary Matching

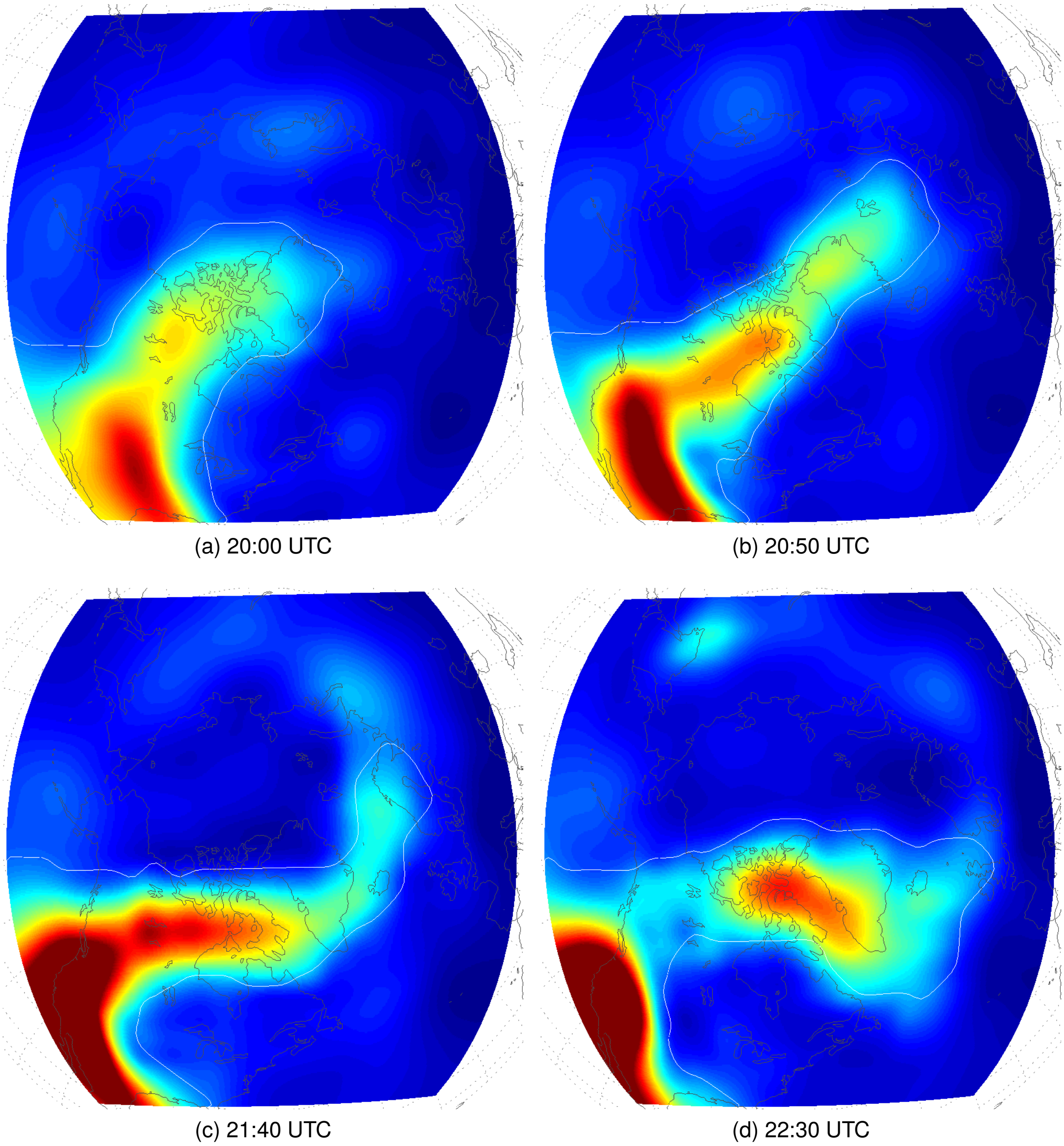
After boundaries have been extracted and described using a chain code or coordinates, it is necessary to compute the correspondences between boundary points. It may also be appropriate to resample them so that the number of points in each boundary is equal, a process which can be made easier by converting the boundaries to a spine-based representation. Splines are piecewise parametric polynomials which are an extremely powerful tool for compactly and smoothly representing complex curves. Using splines allows smoothing of shape boundaries as well as arbitrary resampling along the parameter of the boundary. These factors make splines ideal for describing object boundaries in both low and high-resolution images.

Following appropriate resampling, methods such as shape context matching [Belongie et al., Apr 2002] or the alignment method of Sebastian et al. [2003][1] can be used to determine how boundary points in each frame relate to one another. The coordinate of the corresponding boundary points is then subtracted to give a set of motion vectors, describing how one boundary can be warped into the other. Fig. 7.11 illustrates how boundary correspondences can be used to extract motion estimates. Finally, vector density and smoothness can be increased using interpolation or appropriate models. The shape context method is discussed in more detail below.

Shape Context Matching

The method known as shape context (SC) matching was developed by Belongie et al. [Apr 2002] for measuring similarity between shape boundaries. This is done in three steps, the first two of which can be used to estimate the relative motion of the two

[1] The latter is similar to the edit-distance for strings (a description of how many insertions, changes and deletions are needed to convert from one string to another) [e.g. Klein et al., 2001].



(a) 20:00 UTC

(b) 20:50 UTC

(c) 21:40 UTC

(d) 22:30 UTC

Figure 7.10: Images from Fig. 5.2 segmented using morphological segmentation process described in the text. The TOI boundary is indicated by the white line.

shapes.

Given boundary points for two shapes, the three steps required to measure their similarity are:

1. Solve correspondences between points on their boundaries;
2. Use these to estimate a boundary alignment;
3. Sum matching errors to calculate similarity.

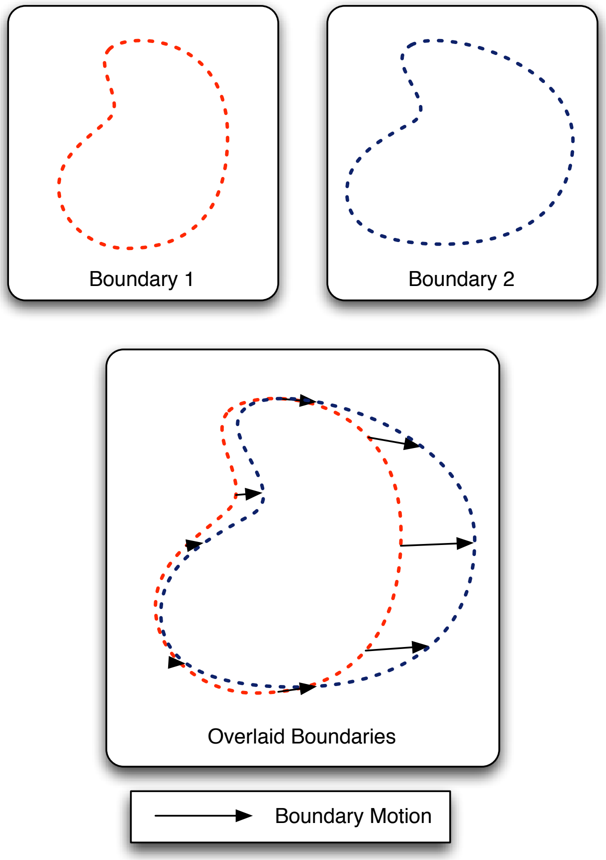


Figure 7.11: A schematic illustrating motion estimation by boundary correspondences

The output of step 2 is a permutation $\pi(i)$ which maps points on the first shape to corresponding points on second. Differencing the coordinates of these corresponding points gives the vectors which are required to warp the boundary of the first shape into that of the second. If both boundaries represent the same object, and the fundamental assumption is that they *do*, then these vectors will represent the relative motion between frames.

SC matching uses properties of shape boundaries known as *contexts* to calculate correspondences between shapes' boundaries. Further stages can then be used to calculate similarity metrics if desired.

A shape context is histogram which describes the distribution of boundary points relative to an origin point (also part of the boundary). A shape context is created in the following manner:

- choose an origin on the shape boundary;
- subtract the origin coordinates from the other boundary coordinates;

- convert the new coordinates into polar form to get r and θ;
- create a 2-D histogram by binning $log(r)$ and θ (Belongie et al. [Apr 2002] uses five bins for $log(r)$ and 12 bins for θ)

In order to establish point correspondences between boundaries, shape contexts must be created for every possible origin point for each shape. This allows a matching cost, $C_{ij} = C(p_i, p_j)$ to be computed for every possible pair of points (p_i, p_j), where p_i is a point on the boundary of the first shape, and p_j is a point on the second. The cost of matching can be calculated using a variety of methods, such as the χ^2 cost, which is given by:

$$C_ij \equiv \chi^2(p_i, p_j) = \frac{1}{2} \sum_{k=1}^{K} \frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)}, \tag{7.1}$$

As this only compares histogram bins with corresponding bins in the second histogram, other methods, such as as the Earth mover's distance (EMD) [Ling and Okada, May 2007] or diffusion distance (DD) [Ling and Okada, 2006] (a dissimilarity measure found to be more robust and accurate than the EMD under certain conditions) can be used. These metrics include cross-bin comparisons and so are more robust to problems such as boundary occlusion and noise. The DD is particularly attractive due to its simplicity (it makes use of a Gaussian pyramid and simple filtering operations).

The DD is defined by:

$$K(h_i, h_j) = \sum_{l=0}^{L} |d_l(\mathbf{x})|, \tag{7.2}$$

where,

$$d_0(\mathbf{x}) = h_1(\mathbf{x}) - h_2(\mathbf{x}) \tag{7.3}$$

$$d_1(\mathbf{x}) = [d_{l-1} * \phi(\mathbf{x}, \sigma)] \downarrow_2 \quad l = 1, \ldots, L \tag{7.4}$$

Where l represents the current layer of the Gaussian pyramid with L layers, $\downarrow_2$ denotes downsampling to half-size. σ is the standard deviation of the Gaussian filter $\phi(\ldots)$.

Once matching costs have been calculated — the result of which will be a cost matrix, C_{ij} — the task is to minimise the total cost of matching across the two boundaries, given by:

$$H(\pi) = \sum_i C(p_i, p_j), \tag{7.5}$$

subject to π being a permutation. This means that a one-to-one match is required. This problem is known as a *weighted bipartite matching* problem, which can be solved in a variety of ways, such as the Hungarian/Munkres method, which attempts to find an optimal path through the matrix [Kuhn, 1955].

The result of minimising (7.5) is a permutation $\pi(i)$, describing the optimal mapping between shape boundary points. Constraints and extras costs can be added to this technique by modifying the cost matrix to include them as necessary. For example, a distance weighting could be added to help ensure that the minimisation returns points close in the boundary order. Metrics relating to image properties could also be added, although finding suitable metrics is problematic, as many, such as curvature, can be very noisy and introduce 'pits', very low values, into the cost matrix. These serve to pull the match towards them, which can give poor vectors as it can leads to bad boundary correspondences. Fig. 7.12 shows an example cost matrix, created using weighted diffusion distance. The weighting was done using the Euclidean distance between boundary points.

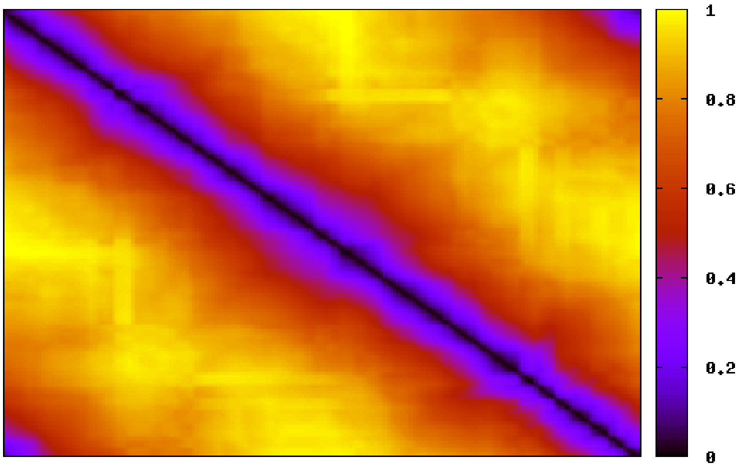


Figure 7.12: An example cost matrix created using the diffusion distance histogram metric, weighted by the distance between matching points. The axes correspond to boundary parameter values, and the scale is normalised cost.

The permutation can then be used to calculate the distances between corresponding boundary points on the two shapes, by subtracting the coordinates of the first shape's boundary points from the corresponding points on the second shape. The result of this will be the set of vectors required to warp the first shape into the second, that is their relative motion.

7.2.1 Boundary Vector Transformations

An additional step which can be performed on estimated boundary vectors is estimating the plane transformation required to warp one shape into the other. This can be done in a variety of ways, and results in a 2-D warping transformation T which allows arbitrary points from one shape to be mapped into the other. This processes allows denser vectors fields to be estimated. For more detailed information on image warping, see Wolberg [1994].

The two most commonly used methods of choosing T are least-squares fitting a standard affine model, and the estimation of warping planes using radial basis function (RBF) interpolation [Belongie et al., Apr 2002]. This interpolation method is described in more detail in Section 2.4. Regularisation can also be applied to smooth the warping planes as needed, reducing the noise in the vectors.

Affine Model

The affine method attempts to fit a standard affine model of the form

$$T(\mathbf{x}) = A\mathbf{x} + o, \tag{7.6}$$

where A is a matrix describing rotation and scaling, and o is a translation vector. The least squares solution is denoted $\hat{T} = (\hat{A}, \hat{o})$. $\hat{o}$ can be found by taking the mean distance between corresponding boundary points, and $\hat{A}$ can be found using:

$$\hat{A} = (Q^{+}P)^{t}, \tag{7.7}$$

P and Q contain the homogeneous coordinates of the boundaries described by p and q, in the same form as equation 2.6. Q^{+} is the pseudo-inverse of Q. The outputs, $\hat{A}$ and $\hat{o}$ can then be used in place of A and o in equation 7.6.

Typical fields produced by affine models and are characterised by an area where the vectors are very small (an origin). Affine modelled fields tend to be have insufficient degrees of freedom to accurately describe complex motion, such as exhibited by the TOI in the TEC images.

RBF Fitting and Regularisation

Belongie et al. [Apr 2002] uses two separate thin plate spline (TPS) surfaces (one for the

x-axis mapping and on for the y-axis), fitted using RBF, giving a model of the form:

$$T(x, y) = (f_x(x, y), f_y(x, y))\,. \tag{7.8}$$

The fitting is carried out by considering the boundary positions of the first shape $p_i = (x_i, y_i)$ as the input coordinates, and then taking the x-component second shapes' coordinate as z_i. RBF interpolation is then performed to give f_x. The same process is then performed using the y-component of the displacement to get f_y.

If the coordinate mapping is considered noisy, it is possible to relax the interpolation condition during the RBF fitting. This process is called regularisation, and allows a smooth surface to be fitted to the data.

This is done by modifying A in equation 2.5, by replacing it with:

$$A + \lambda I \tag{7.9}$$

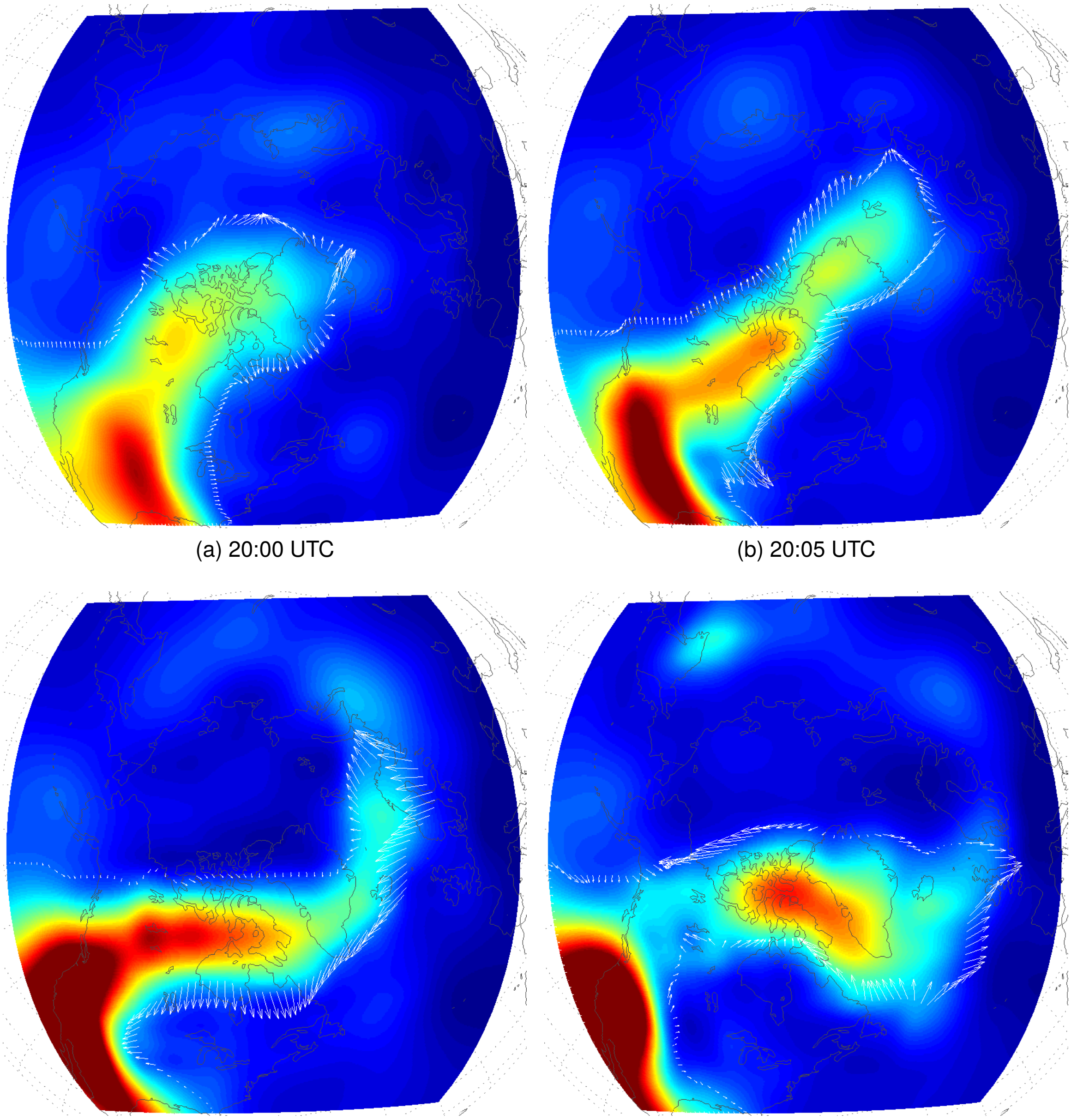
Where λ is a scale dependent *regularisation parameter* which controls the amount of smoothing, and I is an identity matrix. Setting $\lambda = 0$ corresponds to interpolation, and setting λ to large values creates outputs which are similar to those from a fitted affine model.

The scale dependence can be removed by replacing λ with $\alpha^2\lambda_0$, where α is the scale of the inputs points, and λ_0 is normalised λ. This can be estimated by taking the mean edge length between input points. λ_0 can now be varied between zero and approximately one. This makes choosing a value for λ much easier.

7.2.2 Tracking TOI Boundaries

Applying SC based motion estimation to the segmented boundaries provides estimated vectors at the shape boundaries in each frame. In general, this proved to be successful and as shown in Fig. 7.13, it captured the motion in the beginning of the sequence particularly well. In this section of the data-set, the motion is highly constrained, and the TOI is easily distinguished from the background. However, at other points in the sequence this is not always the case.

As described in section 5.1, on page 68, the image sequence corresponds to a patch over the polar cap. One consequence of this is that one side of the image is always in sunlight, and one is in darkness. Because of the nature of the ionosphere, the side in sunlight is continually injected with electrons and ions, and the side in darkness undergoes depletion due to recombination effects. This results in the TOI being drained at its tip,



(a) 20:00 UTC

(b) 20:05 UTC

(c) 20:10 UTC

(d) 20:15 UTC

Figure 7.13: Example false-colour frames from early in the sequence. Vectors are the raw outputs from the segmentation and shape context matching method.

in a fashion similar to the snout of a glacier, where ice melts, and drains away. If the rate of melting increases, the snout will appear to retreat whilst water will always flow away. If only the position of the snout was being measured, the glacier would appear to be flowing backwards, which is clearly never the case. This effect is illustrated in Fig. 7.14.

This problem occurs towards the end of the image sequence under examination. Fig. 7.15 shows an example of this problem occurring due to a slight drop in value at the very tip of the TOI. Unfortunately, these retreating boundaries cannot be detected by simply

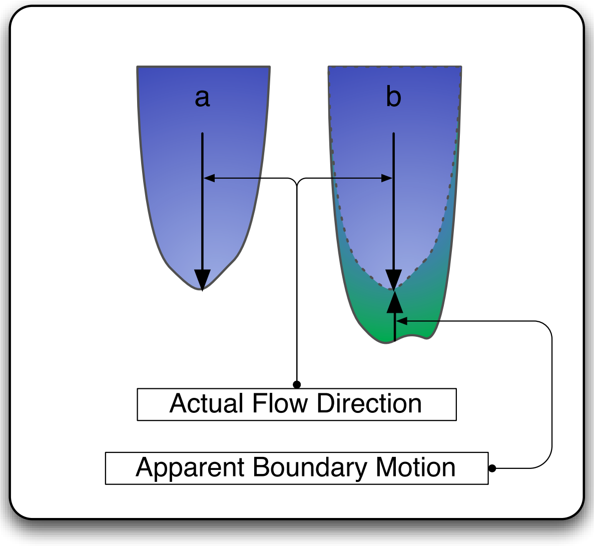


Figure 7.14: A diagram showing how a change in depletion rate can alter the apparent direction of flow.

examining the vector directions relative to other vectors in the same frame, as frames where this problem manifest itself tend to have very noisy vectors in general. For this reason, several techniques to detect and replace these retreating vectors were evaluated for suitability, and effectiveness.

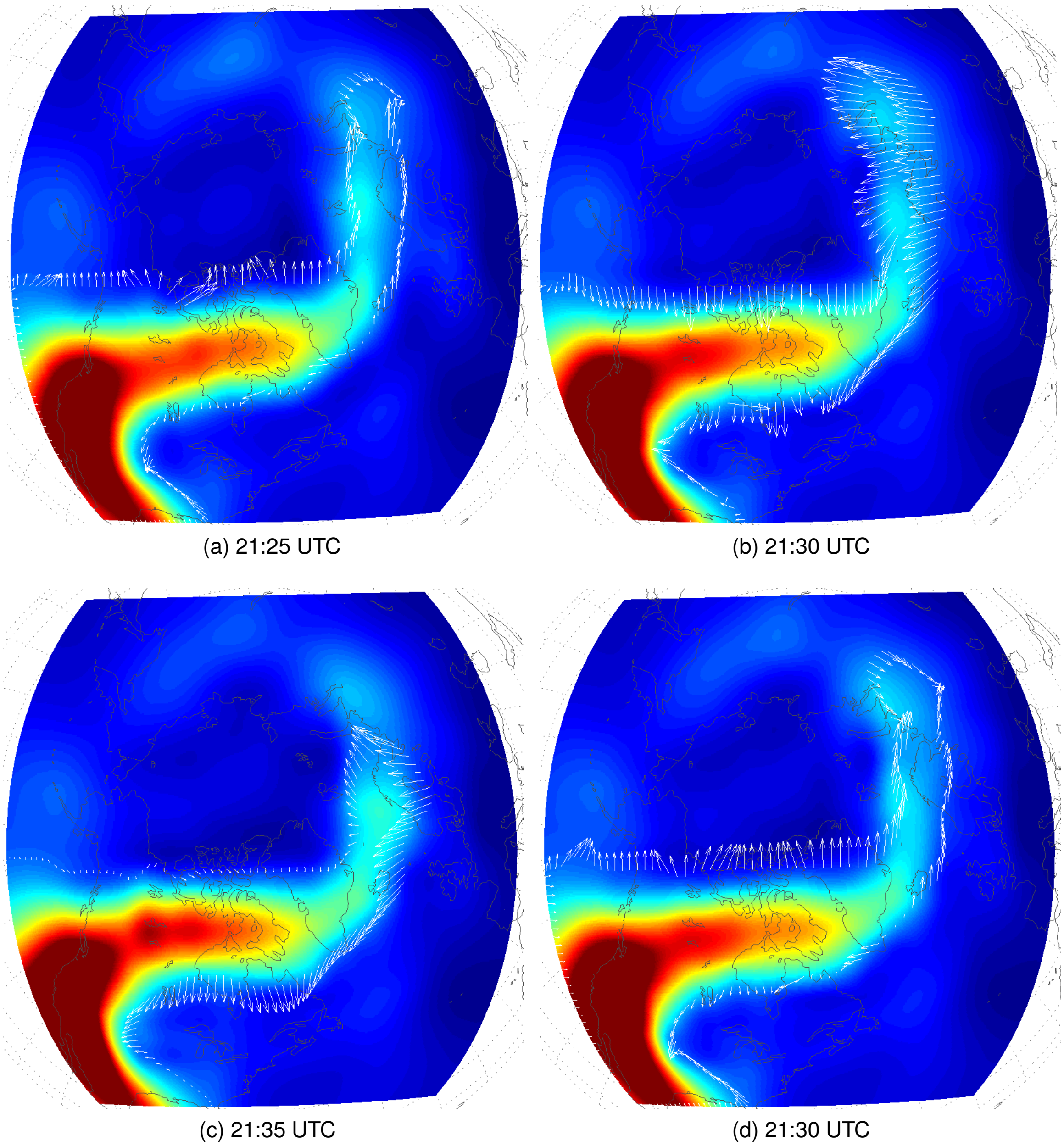
Tested detection methods included:

- Marking vectors by thresholding the magnitude of the current frame's vectors was examined as a simple method of detecting anomalies. However, because no other frames are examined it is unable to detect retreating vectors. For this reason it was deemed unsuitable.

- Marking retreating vectors by first subtracting each given frame's vectors from the previous frame's vectors. These are interpolated to the points at which the current vectors lie. These new difference vectors are then filtered to remove those outside of a 95% confidence limit (two standard deviations either side of the mean) for magnitude, and finally thresholded. This method detects vectors which are significantly different from those in the previous frame, but is very sensitive to the threshold parameter chosen as the histograms of displacements vectors tend to be very flat.

- Marking vectors in a manor similar to the above method, but making use of angle instead of magnitude. This method was found to be unreliable because of the wide variety of vector directions between frames.

- Marking entire frames based on the mean and standard deviation of the vectors



(a) 21:25 UTC

(b) 21:30 UTC

(c) 21:35 UTC

(d) 21:30 UTC

Figure 7.15: Example frames illustrating the retreating snout problem. (a)–(c) Frames from before, during and after the problem occurs, respectively, and (d) the result of detecting the depletion and replacing all the vectors with vectors interpolated from those from the previous frame.

differenced with those from the previous frame. A combination of a mean value of 5, and standard deviation of 3 was found to be effective.

Following detection of the retreating vectors, the selected vectors must be replaced. Vector replacement methods which were tested included:

- Replacing the marked vectors with the vector-median [Astola et al., 1990] of the remaining vectors. As the frames containing the retreating boundary problem tend to contain noisy vectors, this was found to be unpredictable.

- Replacing the detected vectors with new vectors generated using an affine model of the previous frame's vectors. Whilst this tends to give more suitable magnitudes than the previous method, the affine fit often ends up with vectors which seem to point *wrong* direction.

- Replacing the detected vectors with new vectors generated by interpolating the previous frame's vectors onto the marked vector positions. This was carried out using both standard interpolation and regularisation. This gives good results provided the previous frame's vectors are reasonable. For this reason, processing should be done on a frame by frame basis.

- Replacing all of the vectors in the frame with the previous frame's interpolated vectors. This is an effective way of getting smooth vectors over the entire boundary, provided the previous frame's vectors are smooth.

Of the detection and replacement methods tested, replacing all vectors based on measuring vector magnitude statistics was found to be most effective due to the overall smoothness of the output. From an aesthetic point of view, it is advantageous because replacing the entire frame does not result in the discontinuities which are present in the other partial replacement methods at the boundaries of regions where vectors are being replaced.

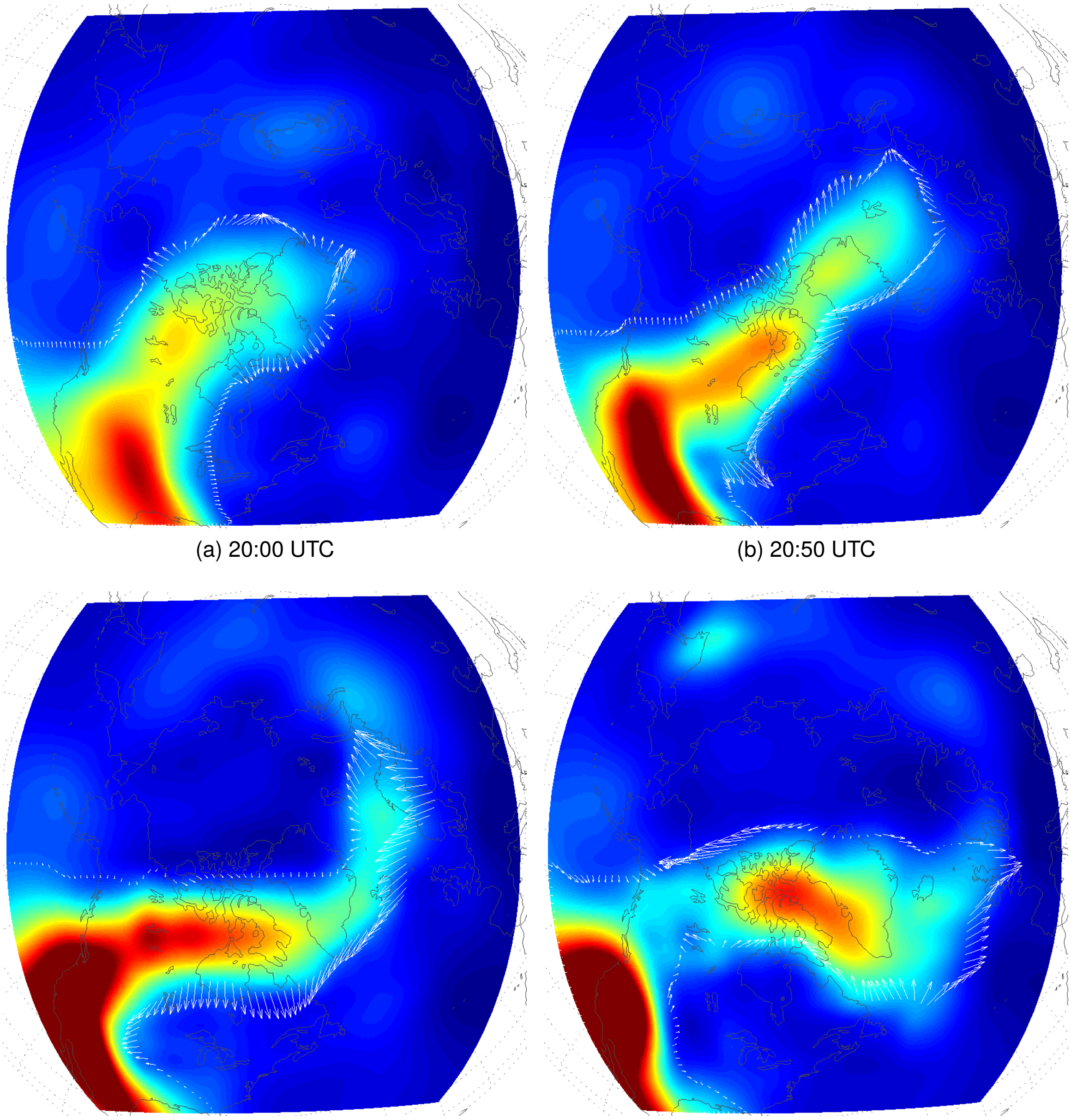
Fig. 7.16 shows example frames with boundary motion vectors plotted at the boundary points (where they were estimated). The next section discusses methods of increasing the density and smoothness of the vector fields produced using this technique.

7.2.3 Increasing Vector Field Density

As an additional processing step, the output vectors can be interpolated to cover the area within the boundaries (and extrapolated to cover the background, if necessary). This could be very useful for interpretation, as large scale trends are easier to observe than those occurring only on boundaries. This process is described in Section 7.2.1.

Fig. 7.17 shows some example frames from the TEC sequence which have been interpolated using RBF TPS interpolation. Other interpolation methods cold also been used, to similar effect, although using linear methods results in far less smooth outputs.

Regularisation can also be used to smooth these vector fields, thus removing noisy and inconsistent vectors. This is similar to the use of relaxation-labelling (see Section 5.7.4,


(a) 20:00 UTC

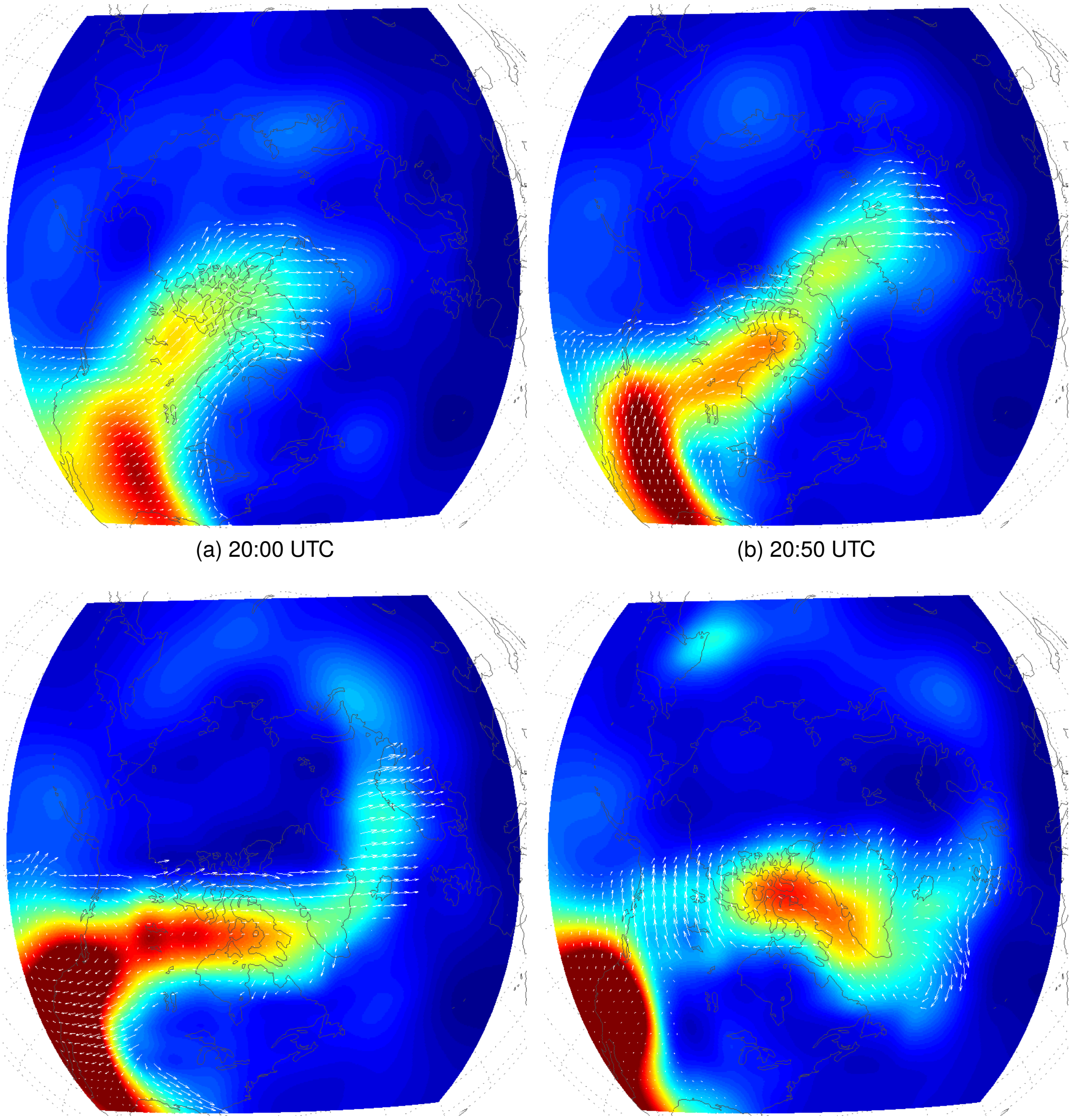
(b) 20:50 UTC

(c) 21:40 UTC

(d) 22:30 UTC

Figure 7.16: Example false-colour frames from Fig. 5.2. Vectors are shown after post-processing by replacing all vectors in frames with detected retreating areas.

on page 82), although it does not pick from a discrete set of available vectors, but uses fitted smooth surfaces. The regularisation is also part of the 'interpolation' method, rather than a add-on step. Fig. 7.18 shows the how changing the scale-dependent regularisation parameter, λ changes the smoothness of a surface being reconstructed. In this example, a sparsely sampled field has had an RBF surface fitted with various different λ values, starting at zero and ending and one, in steps of a quarter. This shows the effect that modifying the regularisation parameter has on the smoothness of the reconstructed field. Fig. 7.19 then shows the same frames as Fig. 7.17, but regularised using a λ of 0.5. This produces smoothed vectors, with a tendency for the field to be



(a) 20:00 UTC

(b) 20:50 UTC

(c) 21:40 UTC

(d) 22:30 UTC

Figure 7.17: Frames from the TEC sequence showing vectors which have been interpolated throughout segmented objects.

less curved than in the non-regularised case.

Comparing Figs. 7.17 and 7.19 with Fig. 5.3 show similar results to the other tested motion estimation techniques. In the beginning of the sequence the behaviours are fairly similar, with a tendency for motion towards the top-right of the images. As the sequence progresses, however, for the right-hand side of Fig. 5.3 feature to appears to show clockwise rotation – a feature which can also be seen in both Figs. 7.17 and 7.19. This suggests that this technique is well suited for capturing large-scale movement, including rotation, something for which the other tested techniques (see Sections 6.2

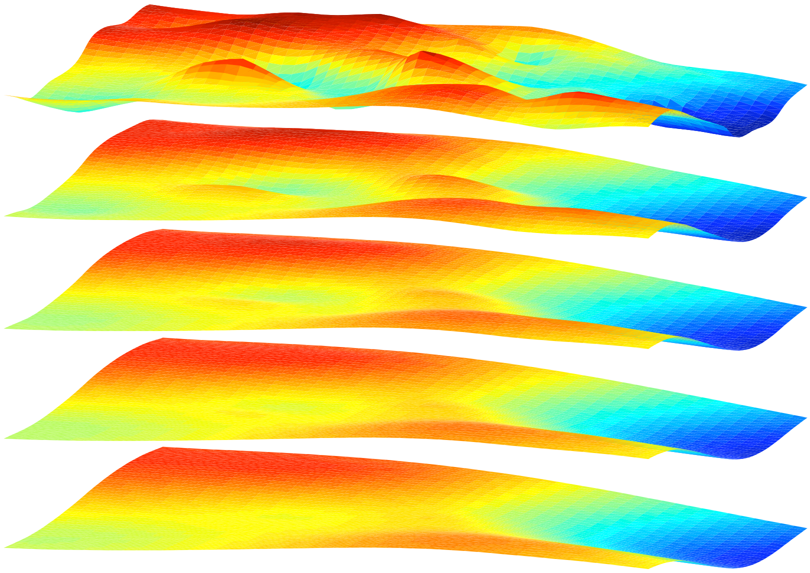


Figure 7.18: The effect of adjusting the regularisation parameter (λ) in steps of 0.25, starting at zero (top), and ending at one (bottom).

and 5.6) have proven ill-equipped.

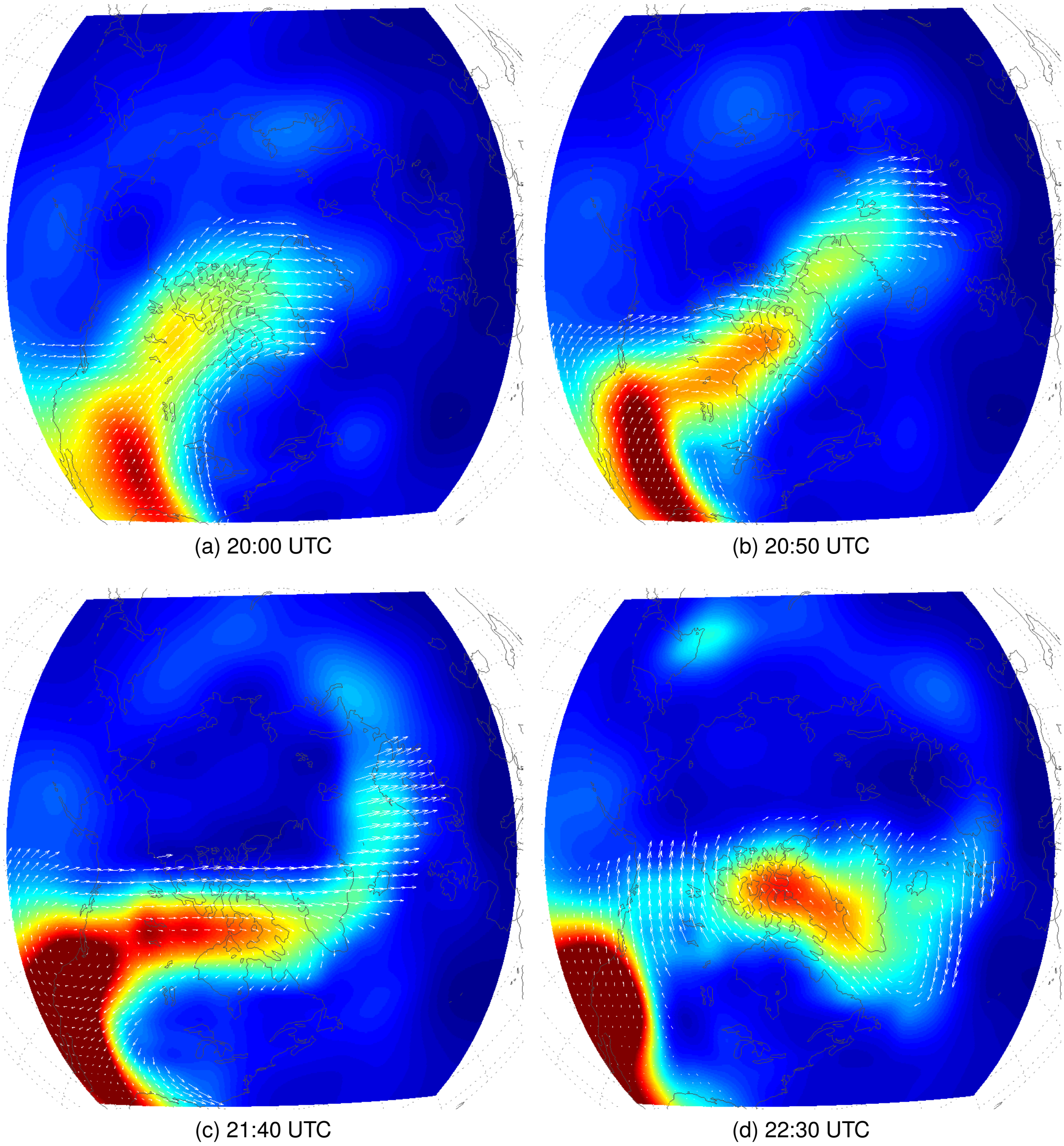
7.3 Conclusions

This chapter has presented a new method for estimating the motion of features in TEC maps of the polar ionosphere. This method makes use of two stages – boundary segmentation and boundary tracking – in order to derive vectors which are accurately describe the motion of the TOI, whilst ignoring uninteresting regions. The segmentation stage uses temporal feedback and morphological filtering based on image contrast to enhance the stability of the motion vector. Finally, a regularisation method is used to estimate smooth vectors throughout the TOI.

Various conclusions can be drawn regarding the techniques introduced in each stage of the work in this chapter.

First, regarding segmentation, it was demonstrated that filters based on attribute morphology, and especially contrast, can effectively segment low-resolution, noisy images. It was also shown that the watershed transform may be used to effectively split regions which are joined by thin saddle shaped areas.

With regards to the conversion of segmented shapes, to boundary descriptions, the most useful observation is that splines can be used to smooth and arbitrarily resample shape


(a) 20:00 UTC

(b) 20:50 UTC

(c) 21:40 UTC

(d) 22:30 UTC

Figure 7.19: Frames from the TEC sequence showing vectors which have been in-filled throughout segmented objects using regularisation, with $\lambda = 0.5$.

boundaries, making them suitable for matching with shape contexts.

Finally, with regards to correspondence matching, this chapter has shown that shape contexts can be applied to boundary matching, in order to extract motion estimates. For this to work, correspondences between entire shapes must be known *a priori*. Diffusion distance was found to be an effective histogram comparison metric for use with shape context matching, and lastly, vector field density can be increased using regularised radial-basis functions, in order to get smooth dense motion fields.

Appendix B compares the different motion estimation techniques against one-another, in order to illustrate how they differ.

Chapter 8

Conclusions and Further Work

This chapter briefly outlines the contents and contributions of this thesis, with particular emphasis on the conclusions from each chapter. Ideas for further work that could be carried out are also listed.

8.1 Summary of Conclusions

Chapter 1 introduced the background, upon which the remainder of this work is based: the ionosphere. Ionospheric mapping methods and storms were introduces, providing context for the work in later chapters.

Chapter 2 reviewed interpolation methods, with a particular emphasis on adaptive normalised convolution (ANC). A technique which was developed specifically for interpolation of total electron content (TEC) data. This and other methods were examined in the following chapter.

Chapter 3 discussed the evaluation of methods of interpolating sparsely sampled ionospheric TEC data. As this is an exercise in interpolation, the results and methodologies should be generally applicable. This chapter represents the first application of normalised convolution (NC) and ANC to geosciences, and is (to the author's knowledge) the most broadly scoped evaluation of interpolation methodologies since Franke [1982]. In this study, natural-neighbour interpolation was found to perform particularly well, ANC performed well for the TEC data, and kriging was found to be somewhat unstable.

Overall, the it was found that natural-neighbour and linear radial basis function (RBF) interpolation work well in situations where stability and well-behaved outputs are important, and cubic triangulation based interpolation as a good staple technique. When interpolating anisotropic data, with high-frequency content, ANC makes is a good choice.

In Chapter 3, on page 42 it was suggested that the stability of an interpolation method can be assessed by examining error distributions, and through knowledge of artefacts and when the occur. To this end, Chapter 4 (page 55) examined various methods for artefacts, and presented some pathological cases which causes techniques to have skewed error distributions. In summary, the main artefacts to consider are:

- Overshoot and undershoot in *all* cubic and higher order based techniques
- Faceting in low order triangulation based techniques
- Spikes in natural-neighbour interpolated surfaces
- Fitting failures in kriging.

It is also important to remember that interpolation outside of the convex hull of a data set is *not* interpolation, but extrapolation. Using edge padding of some for may therefore be necessary to ensure outputs with correct sizes.

Linear RBF interpolation may be the best choice in cases where smooth, and largely artefact free outputs are required.

Chapter 5 and onwards move to the problem of estimating the motion of specific ionospheric features in low-resolution, low-texture images. The features, known as tongue of ionisation (TOI), appear during ionospheric storms, and move across the polar cap. After an initial survey of motion estimation methods in Chapter 5, the performance of optical flow and correlation-relaxation labelling are examined in Chapter 6. In this study, optical flow was found to perform adequately, and correlation-relaxation labelling was found to perform well, when some changes were made to allow for the low resolution and texture. These changed included overlapping blocks, using sub-pixel motion estimation, and the application of a vector-median-filter to the output fields. In addition, a low correlation threshold was set in order to allow more candidate vectors to be entered into the relaxation labelling stage. In both evaluations, thresholding based on variance filtering was used to suppress noisy vectors in 'background' regions. Both optical flow and correlation-relaxation labelling performed poorly during the end of the sequence, which is characterised by confused and irregular motion.

In an attempt to improve the quality of the vector fields, Chapter 7, followed the design and development of a two stage motion estimation process based on tracking using shape boundaries. The two stages involved in this process were segmentation, using attribute mathematical morphology, and boundary tracking, using shape contexts. As an addition step, the vector fields were fitted with regularised surface, in order to increase their density, and 'fill-in' the shape boundaries. One particularly notable feature of the segmentation method designed here is the use of temporal feedback, which stabilises the area of the segmented objects by examining the properties of previous frames. This technique was found to perform well, once several problems had been overcome. The main problem were object joins, due to saddle points, and the *retreating boundary problem* which manifests as a reversal in the direction of boundary motion. The problems were solved using the watershed transform, and by selectively replacing boundary vectors respectively.

Whilst these performance evaluations have been specifically applied to TEC data, most of the conclusions are generalisable to other types of data. This chapter also demonstrated that a well known shape recognition paradigm can be adapted to the estimation of shape-motion.

8.2 Further Work

A by product of any long term project must surely be a large list of unanswered questions. This section lists a few such questions, collected over the course of the work that completes into this document.

8.2.1 Interpolation and TEC Mapping

A large number of older interpolation methods exist which have not been tested in recent studies [e.g. Akima, 1978, Shepard, 1968, Ripley, 2004]. Many of these are also implemented in languages such as FORTRAN (or older languages). Porting these methods to C or C++ and then including them in the studies in Chapter 3 would be a useful benchmark. It would show how well these methods can cope with highly-sparse data (something which seems to have only been given attention recently), as well as how well they compare to more modern methods.

Another useful advancement would be improving the speed of the RBF interpolation methods by implementing domain decomposition. This was briefly mentioned in Chapter 3. The result in Chapter 4 show that these methods are largely free from the artefacts which manifest in triangulation based methods, meaning that a very fast implementation would be an excellent tool for day-to-day interpolation.

Work on examining how sampling patterns alter interpolation outputs should also prove very interesting and informative. This should concentrate on the spatial frequency content that can be reconstructed for different sampling methods and sparsities, and might pave the way for a universal description of the limits of reconstruction quality at differing sparsity. A good sampling point for this work would be Ripley [2004] and references therein.

Finally, with regards to TEC mapping: these maps do not *necessarily* need to be created using interpolation. An evaluation of regularisation methods (such as those described in Chapter 7) would be a useful evolution of the work on evaluating interpolation performance. The resilience to noise would also be an excellent property of these methods.

8.2.2 Motion Estimation

Future work in the area of motion estimation of storm enhancements should focus on the extension of these techniques into an unsupervised system which does not require the manual creation of ground truths. This will require more data which have been gathered during storms. A suitable training set could then be created which would allow contrast attributes to be chosen. Extending this to be proportional to the maximum data value would be more general and allow the use of the same parameter on many different storms.

Independent data sources such as vector data from the SuperDARN [Greenwald et al., 1995] or EISCAT radars could also be used both for validation of results and perhaps more interestingly, the production of improved motion fields by combining them estimates of boundary positions.

Implementation of a tracking system, using for example, a Kalman filter [Hlavac et al., 1999] would be a useful extension of the work, which could allow for forecasting of storms motion they progress. These results could also be of scientific interest.

Integration of motion estimation into real-time TEC mapping systems would be relatively simple, and allow near-instant analysis of the motion of ionospheric features.

Exhaustive testing of the different motion estimation methods examined here could be carried out by simulating data with known motion, thereby proving a ground truth.

This data would have to be generated in such a way as to ensure realism, whilst ensuring confidence in the estimated vectors. Examining warping methods [e.g. Wolberg, 1994] which could be modified for image generation would be a good first-step towards implementing this study. Although it should be noted that simple warping models, such as the affine model, can not appropriately capture the complex motion of the TOI.

Extension of the motion estimation methods to 3-D would allow feature tracking in full tomographic image sequences. Boundary tracking based on shape contexts would be a good starting point, although correspondence-based tracking would probably less computationally expensive.

References

S. T. Acton. Fast algorithms for area morphology. *Digital Signal Processing*, 11(3):187–203, 2001. 105

S. T. Acton and D. P. Mukherjee. Scale Space Classification Using Area Morphology. *IEEE Trans. Image Process.*, 9(4):623, 2000. 106

H. G. Akçay and S. Aksoy. Automatic detection of geospatial objects using multiple hierarchical segmentations. *IEEE Trans. Geosci. Remote Sens.*, 46(7):2097–2111, July 2008. ISSN 0196-2892. doi: 10.1109/TGRS.2008.916644. 104, 107

H. G. Akçay and S. Aksoy. Morphological segmentation of urban structures. *Urban Remote Sensing Joint Event, 2007*, pages 1–6, April 2007. doi: 10.1109/URS.2007.371765. 104, 107

H. Akima. A method of bivariate interpolation and smooth surface fitting for irregularly distributed data points. *ACM Transactions on Mathematical Software*, 4(2):148–159, 1978. 59, 129

I. I. Alexeev, E. S. Belenkaya, S. Y. Bobrovnikov, V. V. Kalegaev, J. A. Cumnock, and L. G. Blomberg. Magnetopause mapping to the ionosphere for northward IMF. *Ann. Geophys*, 25:2615–2625, 2007. 70

A. Almansa and T. Lindeberg. Fingerprint enhancement by shape adaptation of scale-space operators with automatic scale selection. *IEEE Trans. Image Process.*, 9(12):2027–2042, 2000. 31

P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2(3):283–310, 1989. 80

F. Arikan, O. Arikan, and C. B. Erol. Regularized estimation of TEC from GPS data for certain midlatitude stations and comparison with the IRI model. *Advances in Space Research*, 39(5):867–874, 2007. 8, 41

J. Astola, P. Haavisto, and Y. Neuvo. Vector median filters. *Proceedings of the IEEE*, 78 (4), 1990. 84, 95, 120

C. Bradford Barber, D. P. Dobkin, and H. Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Trans. Math. Softw.*, 22(4):469–483, 1996. 20

J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994. 77

R. K. Beatson, W. A. Light, and S. Billings. Fast solution of the radial basis function interpolation equations: Domain decomposition methods. *SIAM Journal on Scientific Computing*, 22:1717–1740, 2001. 22

T. J. Bellerby. High-Resolution 2-D Cloud-Top Advection From Geostationary Satellite Imagery. *IEEE Trans. Geosci. Remote Sens.*, 44(12):3639–3648, 2006. 85

S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, Apr 2002. ISSN 0162-8828. doi: 10.1109/34.993558. 111, 114, 116

D. Bhat, N. Nayar, and K. Shree. Ordinal measures for image correspondence. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(4):415–423, April 1998. 79

D. Bilitza. International Reference Ionosphere 2000. *Radio Sci*, 36(2):261–275, 2001. 88

J. Blanch. *Using Kriging to Bound Satellite Ranging Errors Due to the Ionosphere*. PhD thesis, Dept. of Aeronautics and Astronautics, Stanford University, 2003. 42, 52

J. Blanch, T. Walter, and P. Enge. Application of spatial statistics to ionosphere estimation for WAAS. *Proceedings of ION NTM*, 2002. 10, 24, 40

A. Boucher, K. C. Seto, and A. G. Journel. A novel method for mapping land cover changes: Incorporating time and space with geostatistics. *IEEE Trans. Geosci. Remote Sens.*, 44(11):3427– 3435, 2006. ISSN 0196-2892. 23

E. Brockmann and W. Gurtner. Combination of GPS solutions for densification of the European network: Concepts and results derived from 5 European associated analysis centers of the IGS. *EUREF workshop, Ankara, May*, 1996. 9

G. S. Bust and G. Crowley. Tracking of polar cap ionospheric patches using data assimilation. *Journal of Geophysical Research (Space Physics)*, 112:A05307, May 2007. doi: 10.1029/2005JA011597. 71

J. C. Carr, W. R. Fright, and R. K. Beatson. Surface interpolation with radial basis functions for medical imaging. *IEEE Trans. Med. Imag.*, 16:96–107, 1997. ISSN 0278-0062. 18, 21, 46

M. Chen, W. Shi, P. Xie, V. Silva, V. E. Kousky, R. W. Higgins, and J. E. Janowiak. Assessing objective techniques for gauge-based analyses of global daily precipitation. *J. Geophys. Res*, 113, 2008. 42, 45, 59

S. W. H. Cowley. Excitation of Flow in the Earth's Magnetosphere-Ionosphere System: Observations by Incoherent-Scatter Radar. In J. Moen, A. Egeland, and M. Lockwood, editors, *Polar Cap Boundary Phenomena*, 1998. 70

N. Cressie. The origins of kriging. *Mathematical Geology*, 2:239–252, 1990. 15, 18, 23

N. Cressie. *Statistics for Spatial Data*. John Wiley & Sons, Inc., 1991. ISBN 0-471-84336-9. 23, 24

F. de Jong, L. J. van Vliet, and P. P. Jonker. Gradient estimation in uncertain data. 1998. 31

S. Dransfeld, G. Larnicol, and P.-Y. Le Traon. The potential of the maximum cross-correlation technique to estimate surface currents from thermal AVHRR global area coverage data. *IEEE Geosci. Remote Sens. Letters*, 3(4):508– 511, 2006. ISSN 1545-598X. 85, 94

R. S. J. Estepar, M. Martin-Fernandez, C. Alberola-Lopez, J. Ellsmere, R. Kikinis, and C.-F Westin. Freehand ultrasound reconstruction based on roi prior modeling and normalized convolution. *Lecture Notes In Computer Science*, pages 382–390, 2003. 25

A. N. Evans. Cloud tracking using ordinal measures and relaxation labelling. *IEEE International Geoscience and Remote Sensing Symposium*, 2:1259–1261, 1999. 79, 85

A. N. Evans. Cloud motion analysis using multichannel correlation-relaxation labeling. *IEEE Geosci. Remote Sens. Letters*, 3(3):392–396, July 2006. 82, 85, 98

A. N. Evans. Glacier surface motion computation from digital image sequences. *IEEE Trans. Geosci. Remote Sens.*, 38(2), 2000a. 80, 82, 83

A. N. Evans. On the use of ordinal measures for cloud tracking. *International Journal of Remote Sensing*, 21(9):1939–1944, 2000b. 79

Q Fan, A Efrat, V Koltun, S Krishnan, and S Venkatasubramanian. Hardware assisted natural neighbour interpolation. *Proc. 7th Workshop on Algorithm Engineering and Experiments (ALENEX)*, 2005. 21

G. Farneback. *Polynomial Expansion for Orientation and Motion Estimation*. PhD thesis, Linköping University, Sweden, 2002. 25

J. C. Foster, A. J. Coster, P. J. Erickson, J. M. Holt, F. D. Lind, W. Rideout, M. McCready, A. van Eyken, R. J. Barnes, R. A. Greenwald, et al. Multiradar observations of the polar tongue of ionization. *Journal of Geophysical Research*, 110, 2005. 5

R. Franke. Scattered data interpolation: Tests of some methods. *Mathematics of Computation*, 38(157):181–200, 1982. 40, 127

J. Fried and S. Zietz. Curve fitting by spline and akima methods: possibility of interpolation error and its suppression. *Physics in Medicine and Biology*, 18(4):550–558, 1973. 59

M. Gasca and T. Sauer. On the history of multivariate polynomial interpolation. *Journal of Computational and Applied Mathematics*, 122(1-2):23–35, 2000. 14

J. Geusebroek, A. W. M. Smeulders, and J. van de Weijer. Fast anisotropic gauss filtering. *IEEE Trans. Image Process.*, 12(8):938–943, August 2002. 35, 36, 46

M. Gianinetto and P. Villa. Rapid response flood assessment using minimum noise fraction and composed spline interpolation. *IEEE Trans. Geosci. Remote Sens.*, 45(10): 3204–3211, Oct. 2007. ISSN 0196-2892. doi: 10.1109/TGRS.2007.895414. 10

R. C. Gonzalez and R. E. Woods. *Digital Image Processing 2/E*. Prentice Hall, 2001. ISBN 0-13-094650-8. 77

R. A. Greenwald, K. B. Baker, J. R. Dudeney, M. Pinnock, T. B. Jones, E. C. Thomas, J.P. Villain, J.C. Cerisier, C. Senior, C. Hanuise, et al. DARN/SuperDARN. *Space Science Reviews*, 71(1):761–796, 1995. 11, 69, 130

M. Hernandez-Pajares, J. M. J. Zornoza, J. S. Subirana, R. Farnworth, and S. Soley. EGNOS test bed ionospheric corrections under the October and November 2003 storms. *IEEE Trans. Geosci. Remote Sens.*, 43(10):2283–2293, 2005. 49

V. Hlavac, M. Sonka, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. Brooks/Cole Publishing Company, 2 edition, 1999. ISBN 0-534-95393-X. 28, 73, 82, 86, 105, 130

B. Hoffmann-Wellenhof, H. Lichtenegger, and J. Collins. *GPS Theory and Practice 5/e*. Springer Wein New York, 2001. ISBN 3-211-83534-2. 8

J. Karvanen and A. Cichocki. Measuring sparseness of noisy signals. In *4th Inernational Symposium on independent Component Analysis and Blind Signal Separation (ICA2003)*, Nara, Japan, April 2003. 17

M. Kass and A. Witkin. Analyzing oriented patterns. *Computer Vision, Graphics, and Image Processing*, 37(3):362–385, 1987. 31

M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988. 86

P. N. Klein, T. B. Sebastian, and B. B. Kimia. Shape matching using edit-distance: An implementation. In *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA-01)*, pages 781–790, New York, January 2001. ACM Press. 111

H. Knutsson and C.-F Westin. Normalized and differential convolution: Methods for interpolation and filtering of incomplete and uncertain data. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 515–523, New York City, USA, June 1993. 25, 26

R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, 2(12):1137–1143, 1995. 42

H. W. Kuhn. The hungarian method for solving the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97, 1955. 86, 115

A. Leick. *GPS Satellite Surveying 2/e*. John Wiley & Sons, Inc., 1995. ISBN 0-471-30626-6. 7

M. Liao, T. Wang, L. Lu, W. Zhouzhou, and D. Li. Reconstruction of DEMs from ERS-1/2 tandem data in mountainous area facilitated by SRTM data. *IEEE Trans. Geosci. Remote Sens.*, 45(7):2325–2335, July 2007. ISSN 0196-2892. doi: 10.1109/TGRS.2007.896546. 10

W. A. Light. *Approximation Theory, Spline Functions and Applications*, chapter Some aspects of radial basis function approximation, pages 163–190. Kluwer Academic Publishers, Boston, MA, 1992. 22

H. Ling and K. Okada. An efficient earth mover's distance algorithm for robust histogram comparison. *Pattern Analysis and Machine Intelligence, IEEE Trans.*, 29(5):840–853, May 2007. ISSN 0162-8828. doi: 10.1109/TPAMI.2007.1058. 114

H. Ling and K. Okada. Diffusion distance for histogram comparison. In *CVPR*, pages 246–253, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2597-0. doi: http://dx.doi.org/10.1109/CVPR.2006.99. 114

D. G. Lowe. Object recognition from local scale-invariant features. *International Conference on Computer Vision*, 2:1150–1157, 1999. 86

E. Maeland. On the comparison of interpolation methods. *IEEE Trans. Med. Imag.*, 7(3): 213–217, Sep 1988. ISSN 1558-254X. doi: 10.1109/42.7784. 59

M. H. Mahdian, E. Hosseini, and M. Matin. Investigation of spatial interpolation methods to determine the minimum error of estimation: Case study, temperature and evaporation. In *Proceedings of GeoComputation*, 2001. 40

A. Mannucci, B. Iijima, U. Lindqwister, L. Sparks X. Pi, and Wilson B. D. *Review of Radio Science 1996 - 1999*, chapter 9 GPS and Ionosphere. Oxford University Press, 1999. ISBN 0198565712. 9

G. Matheron. The intrinsic random functions and their applications. *Advances in Applied Probability*, 5:439–468, December 1973. ISSN 00018678. 23

G. Matheron. The theory of regionalized variables, and its applications. *Centre de Geostatistique, Fontainebleau, Paris*, 1971. 23

R. W. Meggs, C. N. Mitchell, and P. S. J. Spencer. Simulations of thin shell and 4-d inversion techniques for mapping of total electron content. In *7th General Assembly of the Union of Radio Science International (URSI)*, volume 15, page 20, Maastricht, Netherlands, August 2002. 8, 41, 42

E. Meijering. A chronology of interpolation: from ancient astronomy to modernsignal and image processing. *Proceedings of the IEEE*, 90(3):319–342, 2002. 14

A. Meijster and M. H. F. Wilkinson. Fast computation of morphological area pattern spectra. In *International Conference on Image Processing, 2001*, volume 3, pages 668–671 vol.3, 2001. doi: 10.1109/ICIP.2001.958207. 106

C. N. Mitchell and P. S. J. Spencer. A three-dimensional time-dependent algorithm for ionospheric imaging using GPS. *Annals of Geophysics*, 46(4):687–696, 2003. 8, 9

D. P. Mukherjee and S. T. Acton. Cloud tracking by scale space classification. *IEEE Trans. Geosci. Remote Sens.*, 40(2):405–415, 2002. 106

M. Nitzberg and T. Shiota. Nonlinear image filtering with edge and corner enhancement. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(8):826–833, 1992. 31

M. Nixon and A. Aguado. *Feature extraction & image processing*. Academic Press, 2 edition, 2008. 75, 76, 77, 86, 89, 103

H. Omre. The variogram and its estimation. *Geostatistics for Natural Resources Characterization, Part*, 1:107–125, 1984. 23, 43

J. M. Pallares, G. Ruffini, and L. Ruffini. Ionospheric tomography using GNSS reflections. *IEEE Trans. Geosci. Remote Sens.*, 43(2):321–326, 2005. 9

S. W. Park, L. Linsen, O. Kreylos, J. D. Owens, and B. Hamann. Discrete sibson interpolation. *IEEE Trans. on Visualization and Computer Graphics*, 12(2):243– 253, 2006. ISSN 1077-2626. 21

M. Pesaresi and J. A. Benediktsson. A new approach for the morphological segmentation of high-resolution satellite imagery. *IEEE Trans. Geosci. Remote Sens.*, 39(2):309–320, Feb 2001. ISSN 0196-2892. doi: 10.1109/36.905239. 104, 107

T. Q. Pham. Robust fusion of irregularly sampled data using adaptive normalized convolution. *EURASIP Journal on Applied Signal Processing*, 2006, 2006. 25, 26, 28, 29

T. Q. Pham and L. J. van Vliet. Normalized averaging using adaptive applicability functions with applications in image reconstruction from sparsely and randomly sampled data. *Image Analysis, Proc.*, 2749:485–492, 2003. 15, 25, 26, 31

G. M. Philip and D. F. Watson. Matheronian geostatistics – quo vadis? *Mathematical Geology*, 18(1):93–117, 1986. 10, 25

R. Piroddi and M. Petrou. *Advances in Imaging and Electron Physics*, volume 132, chapter Dealing with Irregular Samples, pages 109–165. Elsevier Inc., 2004. 29

M. Powell. The Theory of Radial Basis Function Approximation in 1990. *Advances in numerical analysis. Vol. 2: Wavelets, subdivision algorithms, and radial basis functions, Proc. 4th Summer Sch., Lancaster/UK*, page 2, 1990. 22

M. Rauth. *Gridding of geophysical potential fields from noisy scattered data*. PhD thesis, University of Vienna, May 1998, 1998. 40

K. Rijkse. H.263: video coding for low-bit-rate communication. *Communications Magazine, IEEE*, 34(12):42–45, Dec 1996. ISSN 0163-6804. doi: 10.1109/35.556485. 88

B. D. Ripley. *Spatial Statistics*. Wiley-Interscience, 2004. 17, 59, 129, 130

J.M̃ Ruohoniemi and K.B̃ Baker. Large-scale imaging of high-latitude convection with super dual auroral radar network hf radar observations. *jgr*, 03:20797–20811, 1998. 70

P. Sakov. Natural neighbour interpolation software, 2005. URL http://www.marine.csiro.au/~sakov. 46

P. Salembier, A. Oliveras, and L. Garrido. Antiextensive connected operators for image and sequence processing. *IEEE Trans. on Image Processing*, 7(4):555–570, Apr 1998. ISSN 1057-7149. doi: 10.1109/83.663500. 106

T. Samardjiev, P. A. Bradley, L. R. Cander, and M. I. Dick. Ionospheric mapping by computer contouring techniques. *Electronics Letters*, 29(20):1794–1795, 1993. 41

M Sambridge, J Braun, and H McQueen. Geophysical parameterization and interpolation of irregular data using natural neighbours. *Geophys. J. Int*, 122(3):837–857, 1995. 19, 21

D. T. Sandwell. Biharmonic spline interpolation of GEOS-3 and SEASAT altimeter data. *Geophysical Research Letters*, 14(2):139–142, 1987. 23, 46

J. Schmetz, K. Holmlund, J. Hoffman, B. Strauss, B. Mason, V. Gaertner, A. Koch, and L. Van De Berg. Operational Cloud-Motion Winds from Meteosat Infrared Images. *Journal of Applied Meteorology*, 32(7):1206–1225, 1993. 94

T. B. Sebastian, P. N. Klein, and B. B. Kimia. On aligning curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):116–125, 2003. 111

D. Shepard. A two-dimensional interpolation function for irregularly-spaced data. *Proceedings of the 1968 23rd ACM national conference*, pages 517–524, 1968. 59, 129

R. Sibson. A brief description of natural neighbour interpolation. *Interpreting Multivariate Data*, pages 21–36, 1981. 21

P. Soille. Constrained connectivity for hierarchical image decomposition and simplification. 30(7):1132–1145, July 2008. 104

P. Soille. *Morphological Image Analysis*. Springer, 2 edition, 2004. 105

P. Soille and M. Pesaresi. Advances in mathematical morphology applied to geoscience and remote sensing. *IEEE Trans. Geosci. Remote Sens.*, 40(9):2042–2055, Sep 2002. ISSN 0196-2892. doi: 10.1109/TGRS.2002.804618. 104, 107, 108

PSJ Spencer and CN Mitchell. Imaging of fast moving electron-density structures in the polar cap. *Annals of Geophysics*, 50(3):427–434, June 2007. 69, 70, 99

I. Stanislawska, G. Juchnikowski, L. R. Cander, L. Ciraolo, P. A. Bradley, Z. Zbyszynski, and A. Swiatek. The kriging method of TEC instantaneous mapping. *Advances in Space Research*, 29(6):945–948, 2002. 10, 52

C. Stolle, S. Schluter, S. Heise, C. Jacobi, N. Jakowski, S. Friedel, D. Kurschner, and H. Luhr. GPS ionospheric imaging of the north polar ionosphere on 30 october 2003. *Advances in Space Research*, 36(11):2201–2206, 2005. 5

K. Sugihara, A. Okabe, and B. Boots. Spatial tessellations: Concepts and applications of voronoi diagrams. *Probability and Statistics*, 2000. 19

N. A. Thacker, A. F. Clark, J. L. Barron, R. J. Beveridge, P. Courtney, W. R. Crum, V. Ramesh, and C. Clark. Performance characterization in computer vision: A guide to best practices. *Computer Vision and Image Understanding*, 109(3):305–334, 2008. 41

The Mathworks, Inc. griddata, 2007. URL http://www.mathworks.com/access/helpdesk/help/techdoc/ref/griddata.html. 46

M. H. Trauth. *MATLAB Recipes for Earth Sciences*. Springer, 2006. 15, 18, 24, 46

M. van Ginkel, J. van de Weijer, L. J. van Vliet, and P. W. Verbeek. Curvature estimation from orientation fields. In P. Johansen B. K. Ersboll, editor, *Proc. 11th Scandinavian Conference on Image Analysis*, pages 545–551, Kangerlussuaq, Greenland, June 1999. 33

L. J. van Vliet and P. W. Verbeek. Estimators for orientation and anisotropy in digitized images. In *Proceedings of the first annual conference of the Advanced School for Computing and Imaging*, pages 442–450, Heijen, NL, 1995. 31

R. Warnant and E. Pottiaux. The increase of the ionospheric activity as measured by GPS. *Earth, Planets and Space*, 52(11):1055–1060, 2000. 8

D. F. Watson. Natural neighbour sorting. *Aust. Comput. J.*, 17(4):189–193, 1985. 20, 21

D. F. Watson and G. M. Philip. Triangle based interpolation. *Mathematical Geology*, 16 (8):779–795, 1984. 19

DF Watson and GM Philip. Neighborhood-based interpolation. *Geobyte*, 2(2):12–16, 1987. 19

D. R. Weimer. Models of high-latitude electric potentials derived with a least error fit of spherical harmonic coefficients. *J. Geophys. Res*, 100(19,595), 1995. 70, 88

C-F. Westin and H. Knutsson. Tensor field regularization using normalized convolution. *Proceedings of the Ninth International Conference on Computer Aided Systems Theory (EUROCAST)*, 2809, February 2003. 25, 29

P. Wielgosz, D. A. Grejner-Brzezinska, and I. Kashani. Regional ionosphere mapping with kriging and multiquadric methods. *Journal of Global Positioning Systems*, 2(1): 48–55, 2003. 10, 52

G. Wolberg. *Digital Image Warping*. IEEE Computer Society Press Los Alamitos, CA, USA, 1994. 116, 131

Q. X. Wu, S. J. McNeill, and D. Pairman. Correlation and relaxation labelling: an experimental investigation on fast algorithms. *International Journal of Remote Sensing*, 18(3):651–662, 1997. 82, 85

Q.X. Wu. A Correlation-Relaxation-Labeling Framework for Computing Optical Flow Template Matching from a New Perspective. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, page 843853, 1995. 83, 84

I. T. Young and L. J. van Vliet. Recursive implementation of the gaussian filter. *Signal Processing*, 44:139–151, 1995. 28, 35

Chapter A

Publications

The following publications are the result of work contained within this thesis:

- M. P. Foster and A. N. Evans. An evaluation of interpolation techniques for reconstructing ionospheric TEC maps. *IEEE Transactions on Geoscience and Remote Sensing*, 46(7):2153–2164, July 2008.
- M. P. Foster and A. N. Evans. Tongue of Ionisation Motion Estimation from Polar TEC Sequences. *IEEE Geoscience and Remote Sensing Letters, Accepted March 2009*
- M. P. Foster and A. N. Evans. Performance evaluation of multivariate interpolation methods for scattered data in geoscience applications. *Proceedings of the IEEE Geoscience and Remote Sensing Symposium 2008*, (4):565–568, July 2008.
- M. P. Foster and A. N. Evans. Segmentation and tracking of ionospheric storm enhancements. *Proceedings of the XIV Conference on Image and Signal Processing for Remote Sensing*, volume code 71090O. SPIE, 2008b.

Chapter B

Comparisons of TEC Motion Estimation Methods

This chapter briefly compares the motion estimation methods described in Chapters 6 and 7 with the modelled motion described in Section 5.1.

To this end, motion vectors from each method were aggregated, converted to magnitude and subtracted from those of the other methods. Fig. B.1 shows the difference between vectors from the $E \times B$ model and relaxation labelling.

The aggregated values from each method were differenced, and values binned to create histograms. These are shown in Figs. B.4, B.3, B.4, and B.5 respectively, and illustrate the overall difference in vector magnitude between each of the methods. Examination of these figures allows several conclusions to be drawn.

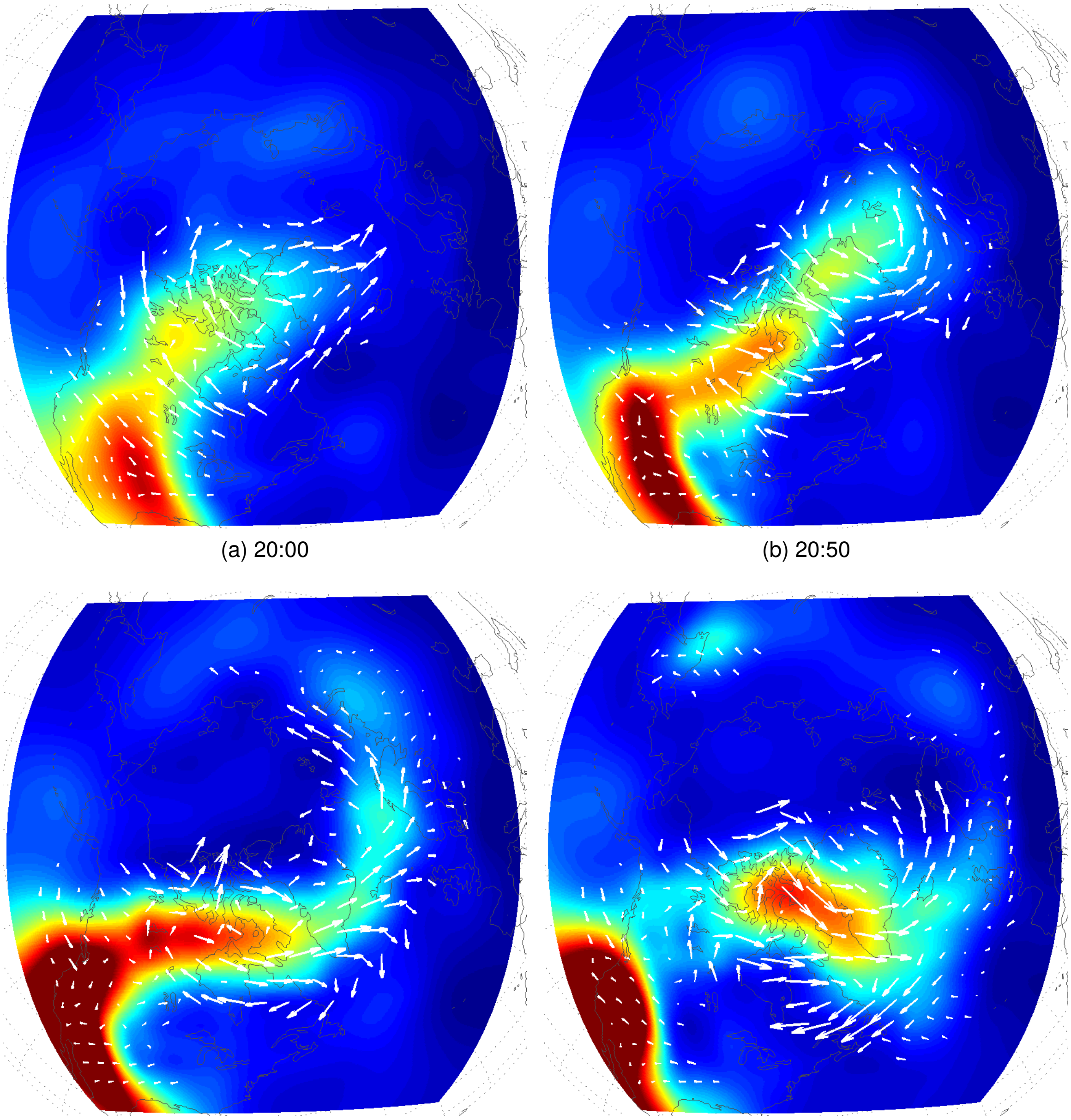
First, the magnitudes of the $E \times B$, optical flow and boundary derived vectors are all quite similar. This can be seen in the symmetrical zero-mode histograms of these methods.

Second, the relaxation labelling vectors have larger magnitudes than the the other methods. This is evidenced by the positive modes of the histograms in Fig. B.4.

Table B.1 shows some statistics confirming these observations. However, the table also shows that the comparisons with the boundary generated vectors resulted in the highest variances. This may be due to the boundary generated vectors being excessively smooth, as they are generated using regularisation.

Table B.1: Motion field comparison statistics

Comparison	Mode	Mean	Variance
ExB - Optical Flow	0	0.7615	3.0678
ExB - Relaxation Labelling	-1.8	0.0597	3.8143
ExB - Boundary Vectors	-0.02	0.4181	10.4442
Optical Flow - Relaxation Labelling	-2	-1.5519	0.8721
Optical Flow - Boundary Vectors	-0.2	-0.8607	7.0120
Relaxation Labelling - Boundary Vectors	1.6	0.5210	8.0967



(a) 20:00

(b) 20:50

(c) 21:40

(d) 22:30

Figure B.1: Frames illustrating the difference between $E \times B$ and relaxation labelling vector fields.

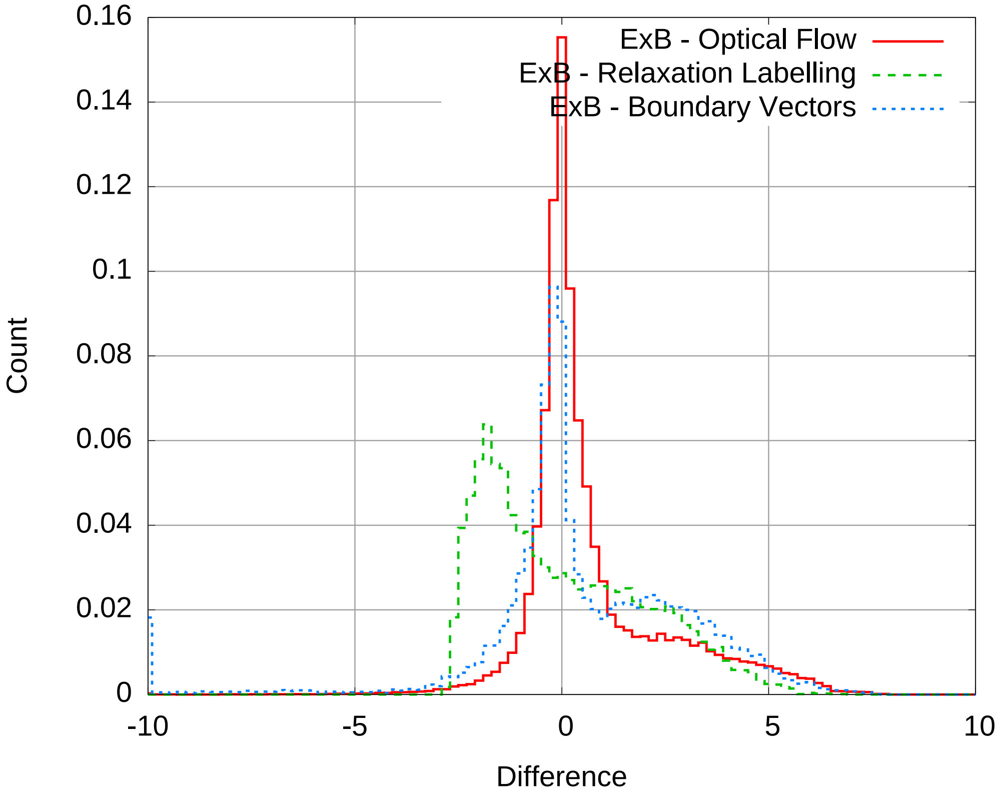


Figure B.2: $E \times B$ vector magnitude comparisons.

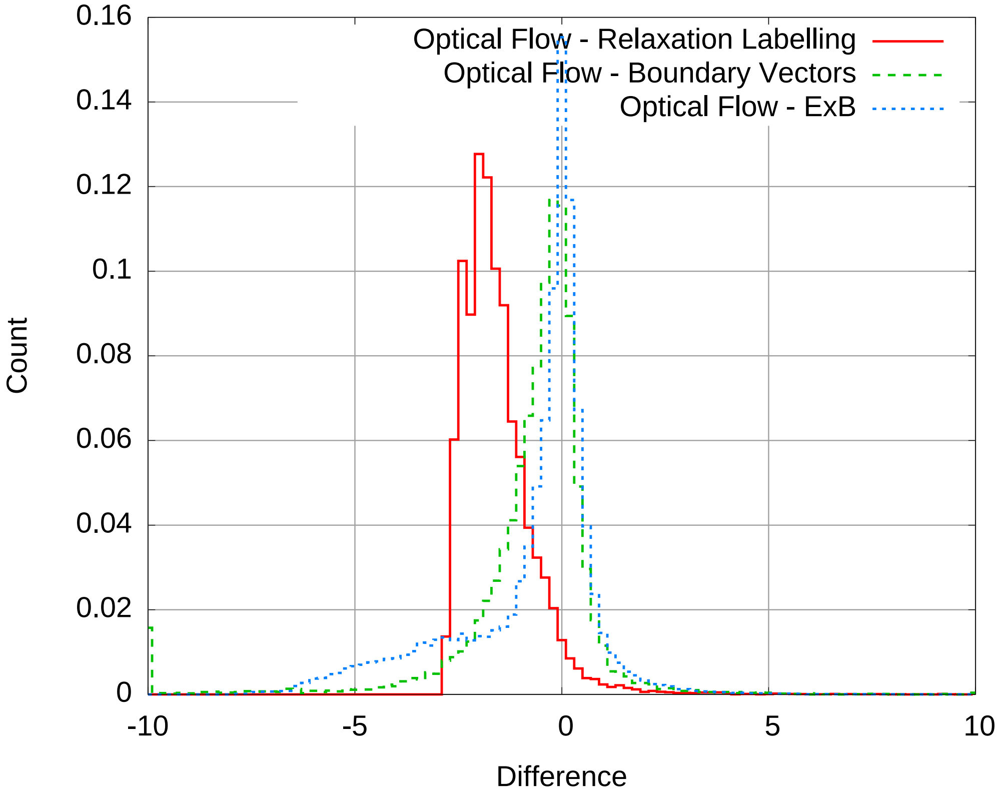


Figure B.3: Optical flow vector magnitude comparisons.

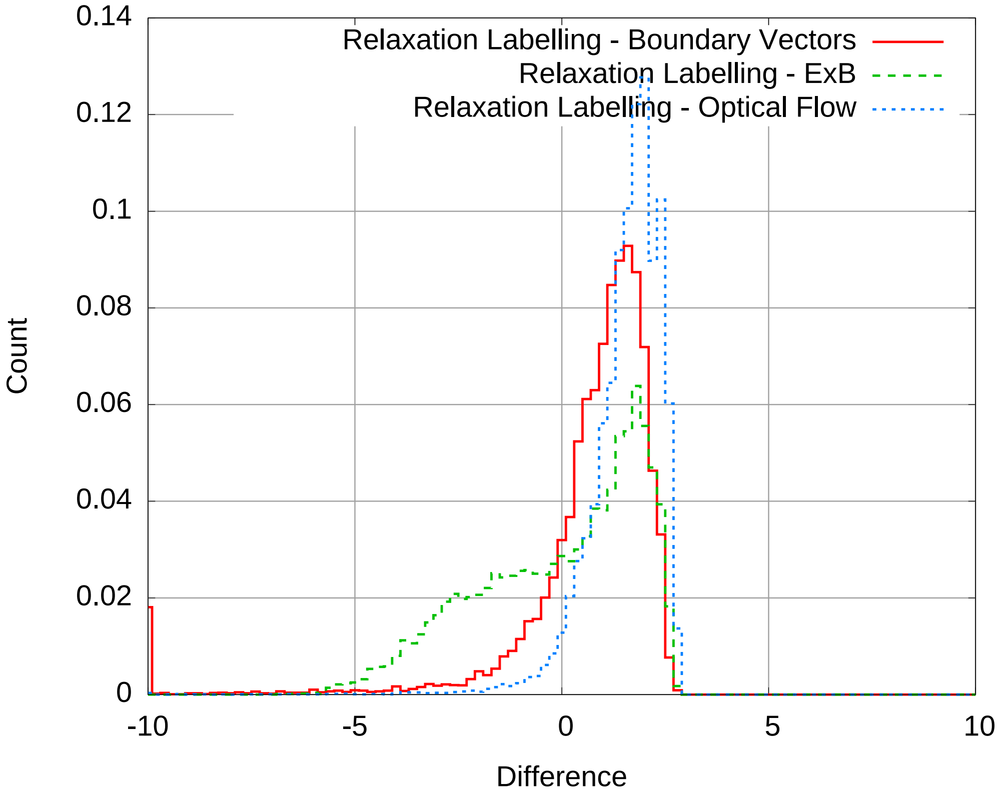


Figure B.4: Relaxation labelling vector magnitude comparisons.

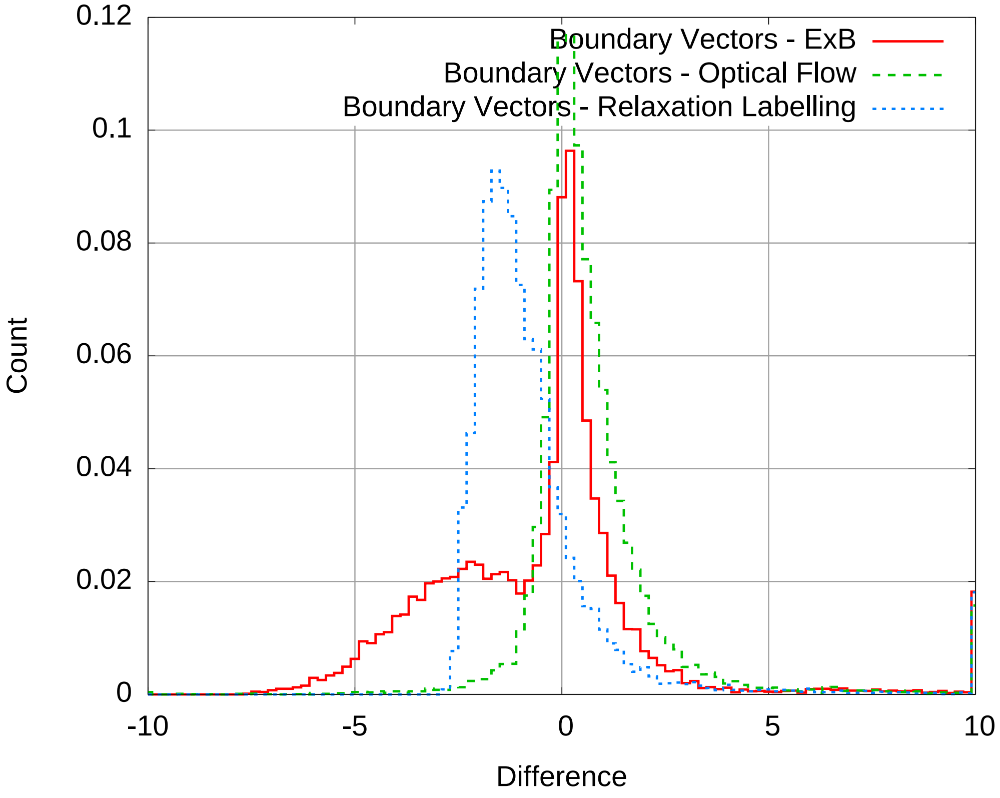
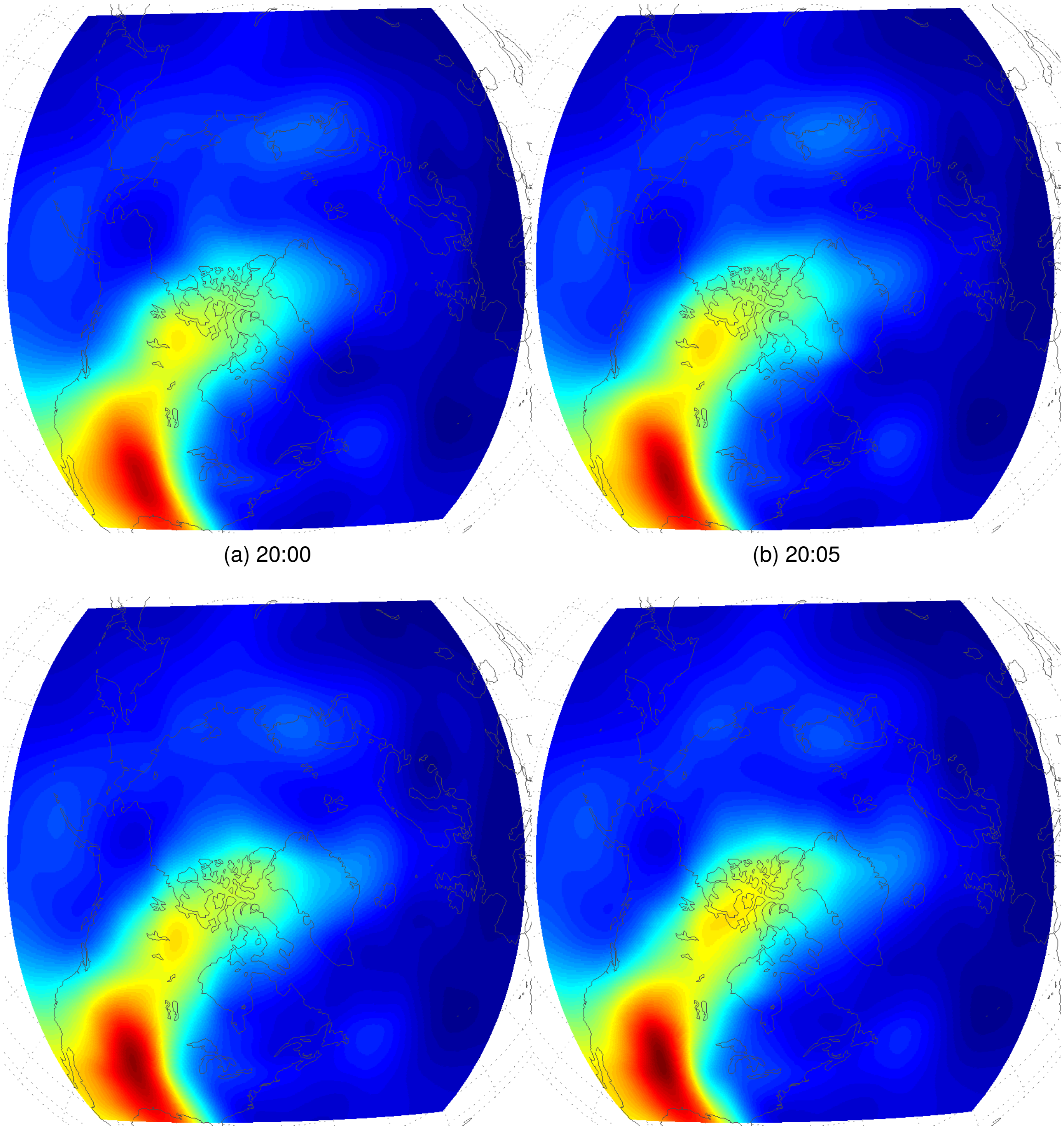


Figure B.5: Boundary vector magnitude comparisons.

Chapter C

Polar TEC Data Sequence

This chapter contains the entirety of the image sequence use in the motion chapters.



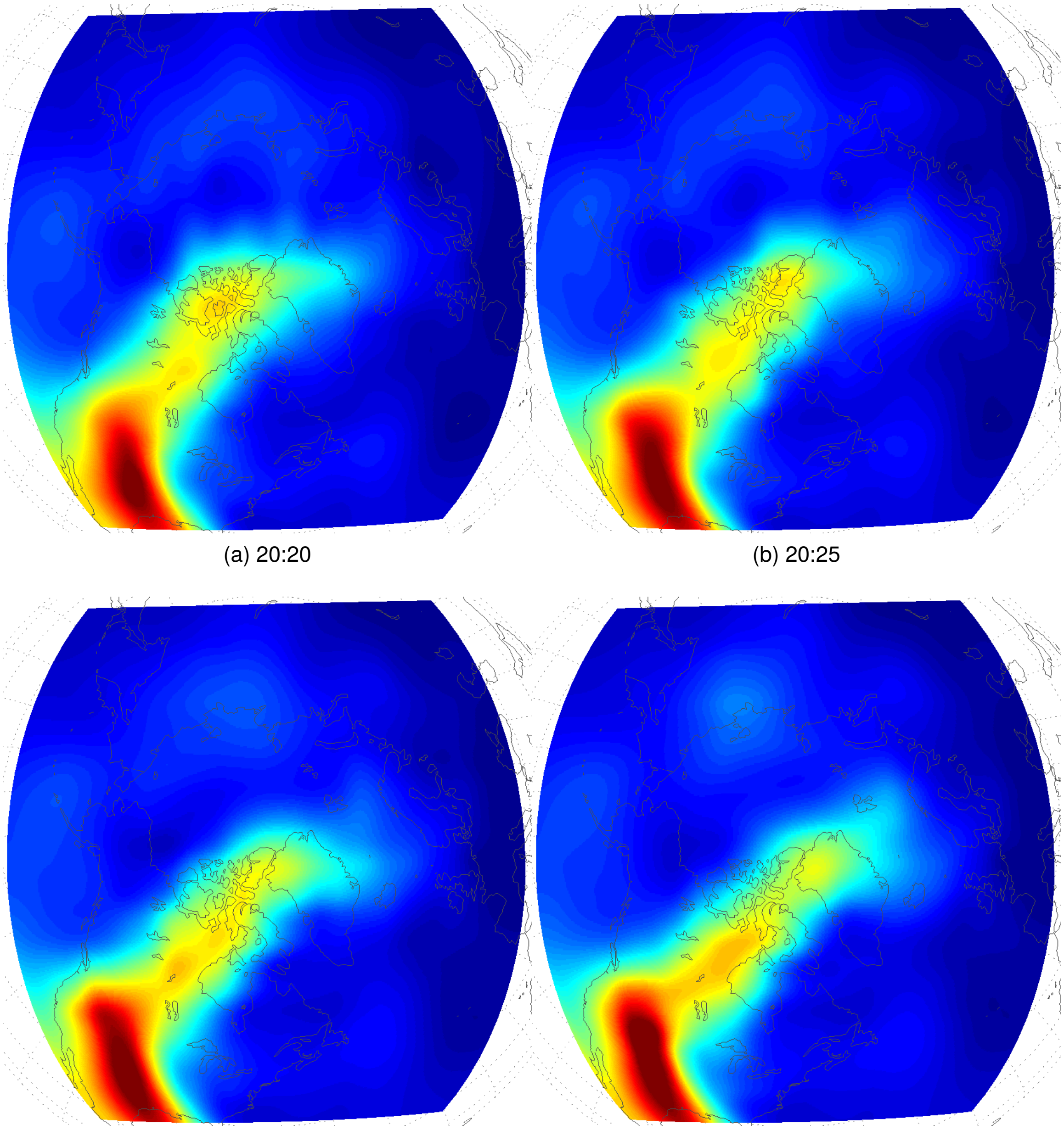
(a) 20:00

(b) 20:05

(c) 20:10

(d) 20:15

Figure C.1: The first four frames from the TEC image sequence



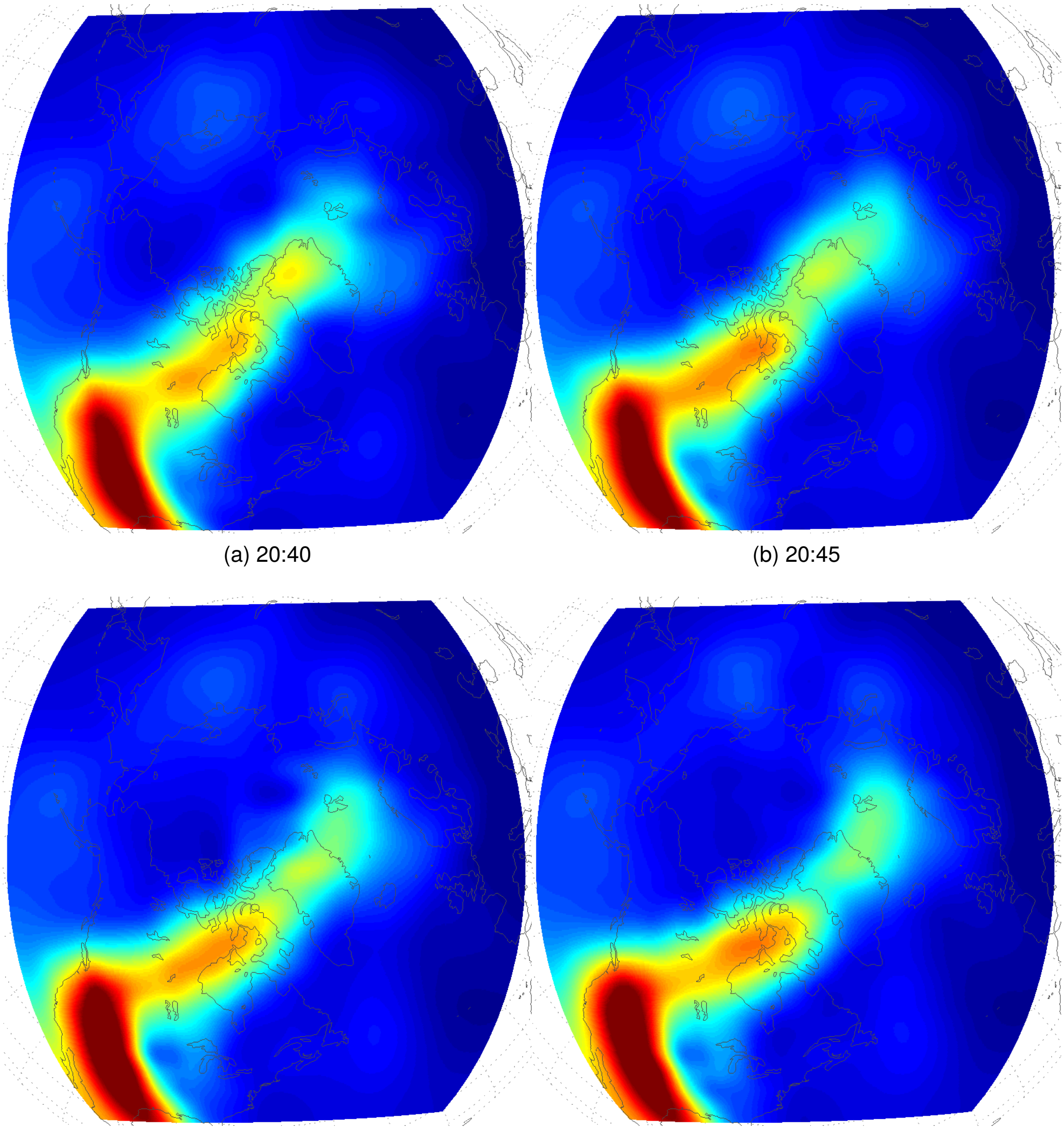
(a) 20:20

(b) 20:25

(c) 20:30

(d) 20:35

Figure C.2: The second four frames from the TEC image sequence



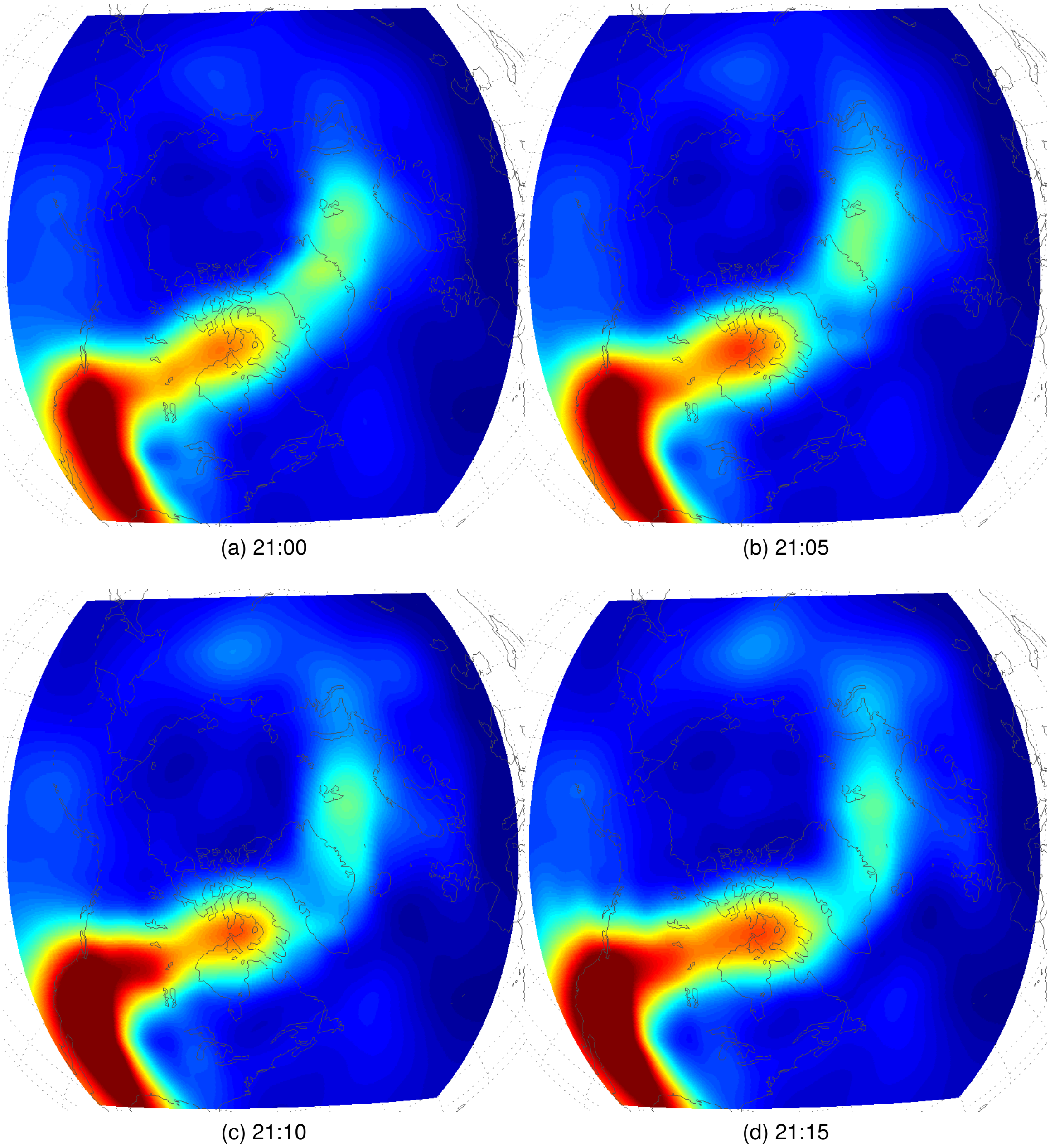
(a) 20:40

(b) 20:45

(c) 20:50

(d) 20:55

Figure C.3: The third four frames from the TEC image sequence

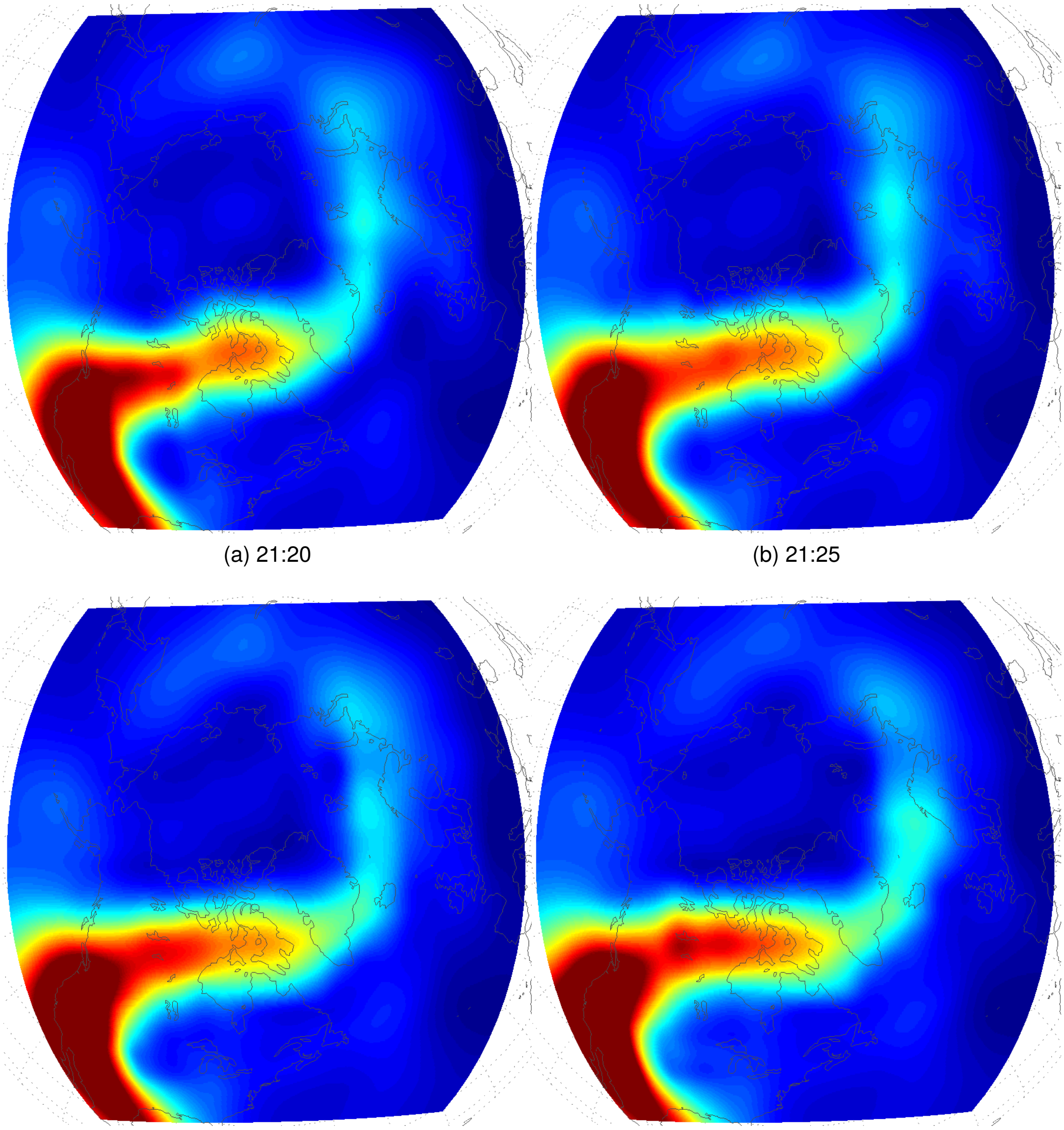

(a) 21:00

(b) 21:05

(c) 21:10

(d) 21:15

Figure C.4: The fourth four frames from the TEC image sequence



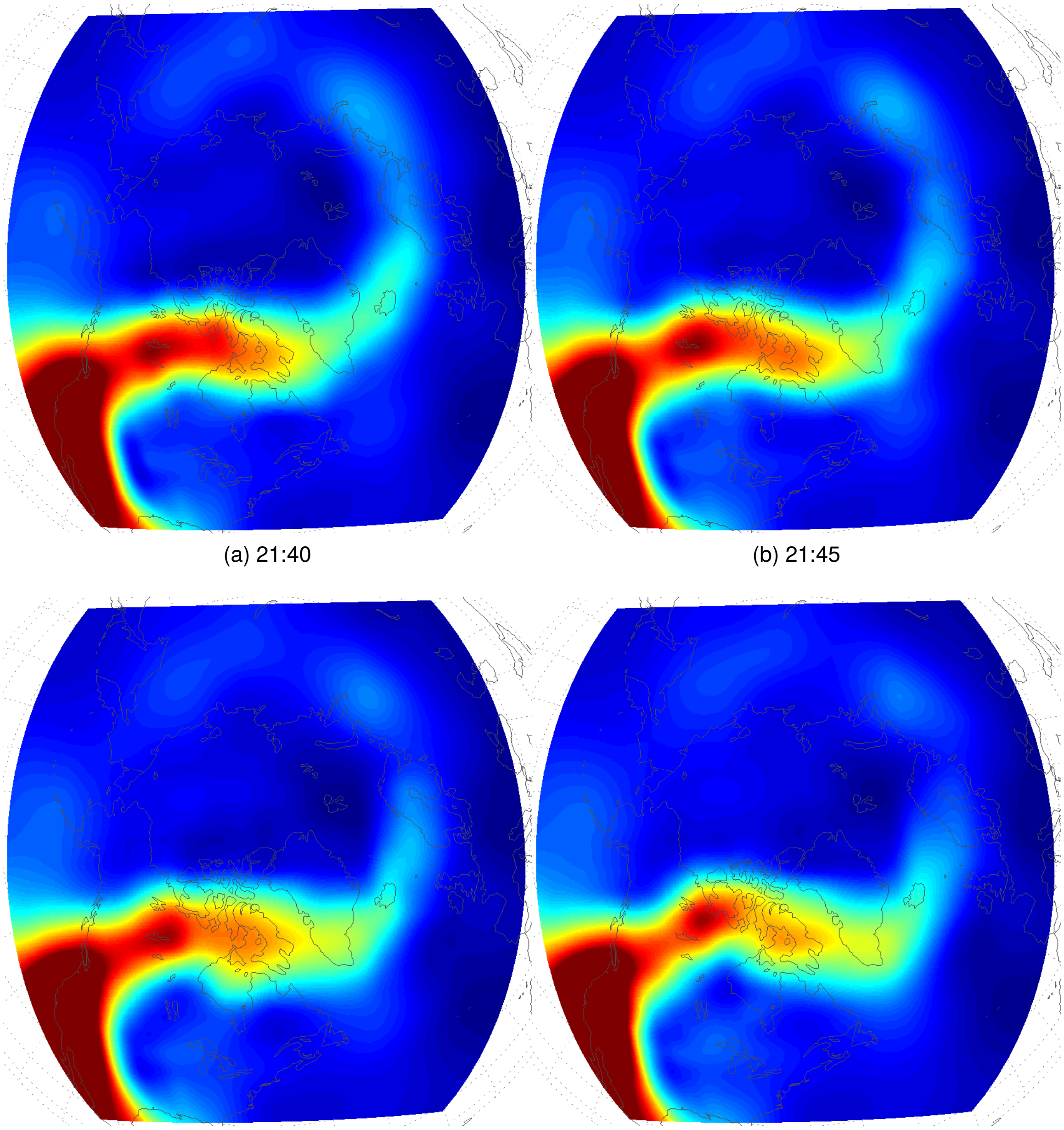
(a) 21:20

(b) 21:25

(c) 21:30

(d) 21:35

Figure C.5: The fifth four frames from the TEC image sequence



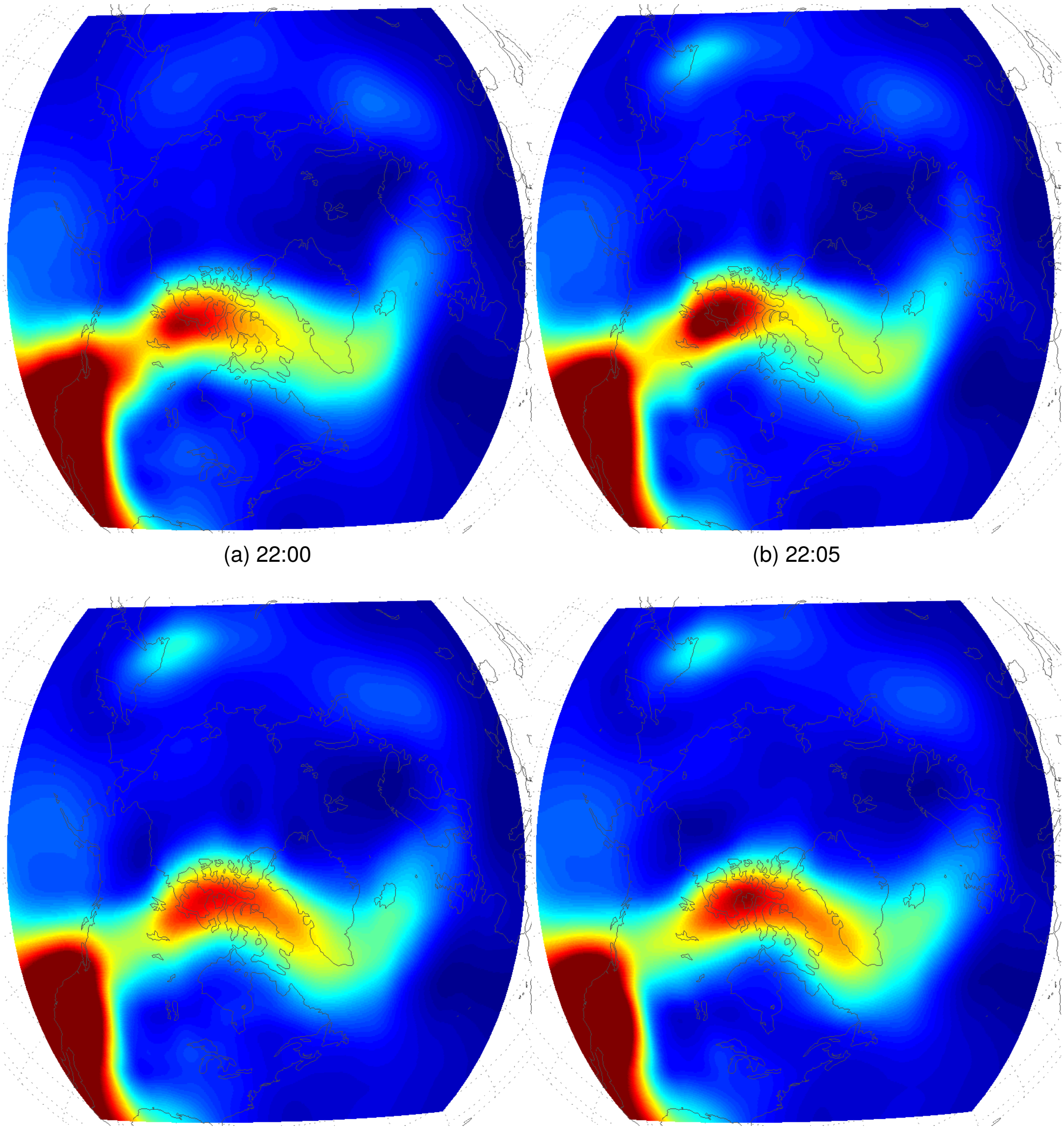
(a) 21:40

(b) 21:45

(c) 21:50

(d) 21:55

Figure C.6: The sixth four frames from the TEC image sequence



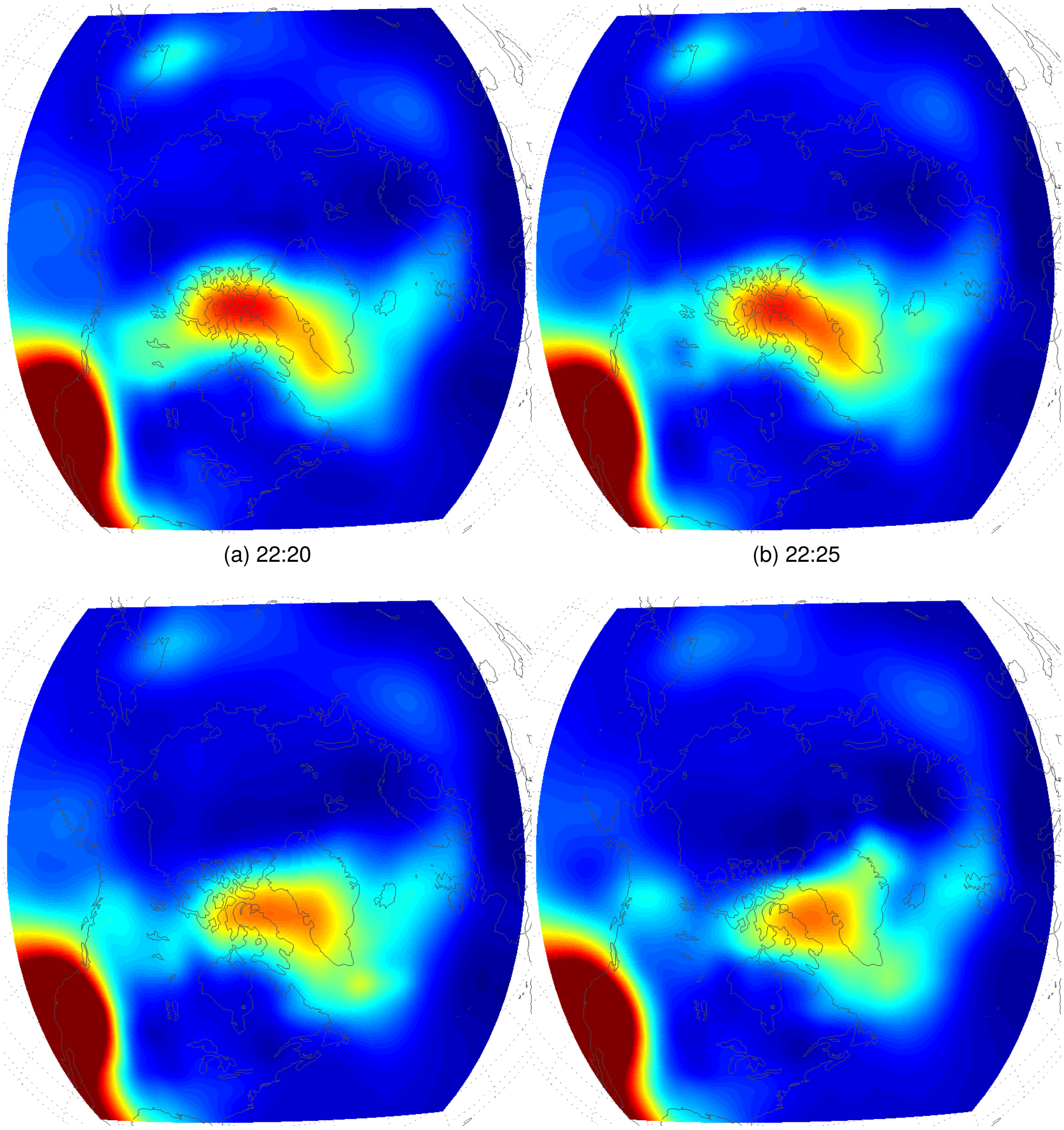
(a) 22:00

(b) 22:05

(c) 22:10

(d) 22:15

Figure C.7: The seventh four frames from the TEC image sequence



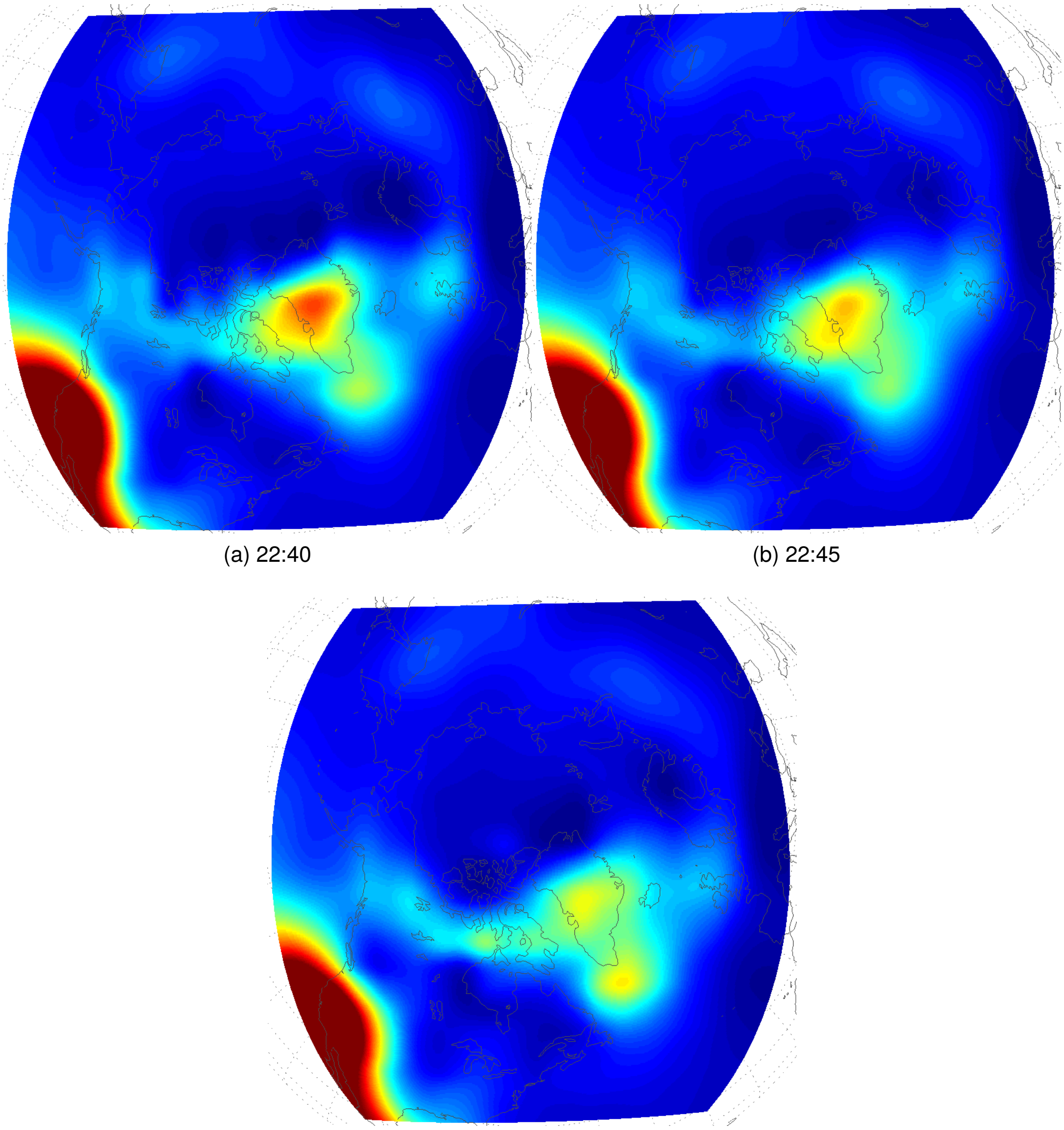
(a) 22:20

(b) 22:25

(c) 22:30

(d) 22:35

Figure C.8: The eighth four frames from the TEC image sequence



(a) 22:40

(b) 22:45

(c) 22:50

Figure C.9: The ninth four frames from the TEC image sequence

Chapter D

Software Used

A variety of both free and commercial software was used in the creation of this thesis. Development was done on the Macintosh and linux platforms, using TextMate and vim, the text editors, and PDFLaTeX (from TexLive 2007/8 [http://texlive.org]).

The fonts used in the this theses are: TexGyre Heros for the section titles, and TexGyre Pageela for the body.

Many graphs were created using gnuplot (http://gnuplot.info), and diagrams drawn using OmniGraffle (http://omnigroup.com).

Data processing was done using Matlab (http://mathworks.com), and octave (http://octave.org), a free clone.

Git (http://git-scm.com) and subversion were used for version control.

Free software created by the author, including octave code for Kriging and RBF interpolation, and various TextMate bundles (e.g. for driving gnuplot) available at: http://github.com/mattfoster.