Digital Elevation Model Generation and Fusion

by

Colleen E. Fuss

A Thesis

presented to

The University of Guelph

In partial fulfilment of the requirements

for the degree of

Master of Science

in

Geography

Guelph, Ontario, Canada

© Colleen Fuss, September, 2013
ABSTRACT

DIGITAL ELEVATION MODEL GENERATION AND FUSION

Colleen E. Fuss

University of Guelph, 2013

Advisors:

Dr. Aaron Berg,
Dr. John Lindsay

Digital elevation models (DEMs) are a necessary dataset in modelling the Earth’s surface and the many physical processes that interact with it. There are several ways to acquire elevation data and generate DEMs, and while each method has advantages and disadvantages all DEMs contain error. DEM fusion techniques with the aim of reducing DEM error have been proposed and tested in published literature with several successful results. These techniques have not however, utilized a clustering algorithm on multiple DEMs to exploit consistency in the estimates as an indication of accuracy and precision. This research developed and tested a new DEM fusion algorithm on multiple, overlapping DEMs generated from RADARSAT-2 imagery using stereo-radargrammetric methods. The main steps of the algorithm include slope and elevation thresholding followed by $k$-means clustering of the elevation estimates, as well as filtering and smoothing of the fusion product. Corroboration of the input DEMs, as well as products of each main step of the fusion algorithm, with a higher accuracy reference DEM by landuse class within the study area enabled a detailed analysis of the effectiveness of the DEM generation and the fusion algorithm. The generated DEMs contained systematic errors, large blunders, and regional offsets that varied according to landuse type, as well as the differences in scene acquisition date and sensor parameters. The main findings of the research were: the $k$-means clustering of the elevations improved the global accuracy of the estimates but reduced the
precision; the number of final cluster members and the standard deviation of elevations before clustering both had a strong relationship to the error in the $k$-means estimates. It is therefore recommended that further research be conducted to investigate the relationship between elevation clustering error and the distribution of elevations before clustering, especially for specific landuse classes such as agricultural fields.
ACKNOWLEDGEMENTS

I would like to thank my advisors, Aaron Berg and John Lindsay, for all of their guidance and support throughout my MSc. program. Thank you for believing in me, giving me the opportunity to broaden my knowledge in this field and allowing me to further my experience in academia. It has been wonderful to work with you.

I would like to acknowledge AAFC, CSA, GEOIDE, ORF, and OMAF for providing funding for my research. I would also like to acknowledge Agriculture and Agri-Food Canada for providing the RADARSAT-2 scenes and Stewart Sweeney from OMAF for providing the AgRI polygon dataset used in my research. Thank you especially to Ron Mak of Van Harten Surveyors and Engineers for collecting the GPS points used in this study.

Thank you to all of the faculty, staff, and students of the Geography Department at the University of Guelph who not only supported me on my journey but also made me feel at home. Thanks for the laughs, smiles, and help over these years, especially Nance, Mario, Gihan, and Adam. Neville – thank you for your guidance and advice that are immensely appreciated and have greatly improved some key aspects of my research. Parinaz, Laura, Sarah, Mary Jane, and Kim – your friendship, wisdom and support have been invaluable – thank you so much.

Mom and Dad, your love, support and wisdom are the reason I am where I am today. You are both wonderful people and I feel very blessed that you are my parents. Thank you for encouraging me to follow my interests, and for instilling in me an appreciation for discovery and learning. Thank you, most of all, for teaching me to have a good sense of humour and to laugh when possible at the inevitable ups and downs of life.
# TABLE OF CONTENTS

Abstract  Digital Elevation Model Generation and Fusion ........................................... ii
Acknowledgements ........................................................................................................ iv
Table of Contents .......................................................................................................... v
List of Figures ................................................................................................................ ix
List of Tables ................................................................................................................ x

1 General Introduction ................................................................................................. 1
   1.1 Study Context and Objectives ............................................................................. 1
   1.2 Thesis Outline ...................................................................................................... 4

2 Literature Review ...................................................................................................... 6
   2.1 Introduction to DEMs ......................................................................................... 6
   2.2 Elevation Data Acquisition ............................................................................... 7
      2.2.1 Stereoscopy .................................................................................................. 7
      2.2.2 Interferometry ............................................................................................. 10
      2.2.3 Ranging / Altimetry .................................................................................... 13
   2.3 DEM Error ......................................................................................................... 15
      2.3.1 Error due to landcover and terrain characteristics ....................................... 15
      2.3.2 Interpolation ................................................................................................ 18
      2.3.3 Error estimation ........................................................................................... 21
   2.4 Global DEMs ...................................................................................................... 23
      2.4.1 Multi-source global DEMs ......................................................................... 23
      2.4.2 Single-source global DEMs ...................................................................... 25
   2.5 DEM Fusion ....................................................................................................... 26
      2.5.1 Fusion with weights .................................................................................... 27
2.5.2 Sparse representations ................................................................. 28
2.5.3 Frequency domain filtering .......................................................... 29
2.5.4 Self-consistency ........................................................................... 30
2.5.5 Multi-scale stochastic smoothing .................................................... 32
2.6 Research Gaps and Opportunities .................................................... 32

3 Methods .......................................................................................... 35
3.1 DEM Fusion Algorithm Overview ...................................................... 35
3.2 Study Area ..................................................................................... 36
3.3 RADARSAT-2 Imagery, GCP coordinates, and DEM Extraction .......... 38
  3.3.1 RADARSAT-2 scene selection and processing ............................... 38
  3.3.2 GCP locations and elevations ......................................................... 41
  3.3.3 DEM extraction ........................................................................... 42
3.4 DEM Fusion Algorithm Rationale and Implementation ....................... 43
  3.4.1 Data preparation .......................................................................... 43
  3.4.2 Slope and elevation thresholding ................................................... 44
  3.4.3 k-means clustering ...................................................................... 45
  3.4.4 Filtering and smoothing ................................................................. 48
3.5 Corroboration of DEMs and Fusion Algorithm Products ...................... 49
  3.5.1 Reference DEM and DEMs of difference ....................................... 50
  3.5.2 Classification by landuse ............................................................... 52
4 Results ............................................................................................. 56
4.1 DEM Extraction .............................................................................. 56
  4.1.1 Visual assessment of generated DEMs .......................................... 56
  4.1.2 Correlation between DEMs .......................................................... 57
4.2 Fusion Algorithm Steps and Products ................................................. 58
Appendix B  Correlation Matrices for DEMs by Landuse Class.................................................. 120
Appendix C  Fusion Algorithm Product Figures ........................................................................... 124
Appendix D  DEM of Difference Figures ....................................................................................... 129
Appendix E  Additional Corroboration Figures ............................................................................ 146
Appendix F  $k$-means Clustering Algorithm Flowchart ................................................................. 149
Appendix G  $k$-means Clustering Algorithm Code ........................................................................ 150
LIST OF FIGURES

Figure 3-1  Flowchart of the main steps in the DEMs in the fusion algorithm ..................... 36
Figure 3-2  Location of the study area and larger data collection area................................. 38
Figure 3-3  Map of RADARSAT-2 scene extents ............................................................. 39
Figure 3-4  The reference DEM ....................................................................................... 51
Figure 3-5  Map of the landuse masks created for the study area ....................................... 54
Figure 4-1  Profile of DEM elevations before and after slope and elevation thresholding. ... 62
Figure 4-2  Profile of DEM elevations before k-means clustering and the members of the final cluster after the k-means clustering ................................................................. 65
Figure 4-3  The final product of the DEM fusion algorithm ................................................. 67
Figure 4-4  DEM of difference for the final DEM fusion algorithm product ..................... 69
Figure 4-5  Frequency distribution of the relative accuracies for fusion algorithm products. 74
Figure 4-6  Frequency distribution of the difference in absolute accuracy between Product 01 and the other fusion algorithm products ........................................................................ 75
Figure 4-7  Cumulative frequency distribution of the absolute accuracy of k-means final clusters classified by the number of final cluster members ................................................. 77
Figure 4-8  Map of the number of final cluster members at each cell location .................... 78
Figure 4-9  Cumulative frequency distribution of the absolute accuracy of k-means final clusters classified by the standard deviation of elevations before clustering .................. 79
Figure 4-10 Map of the standard deviation of elevations before k-means clustering, at each cell location .............................................................................................................. 80
LIST OF TABLES

Table 3-1  RADARSAT-2 scenes used for DEM extraction with attributes used for scene selection and pairing ................................................................. 40

Table 3-2  Total area for each landuse class in the study area ........................................ 55

Table 4-1 Summary of fusion algorithm product names and descriptions. ....................... 59

Table 4-2 DEM retention after each thresholding step and k-means clustering ................. 60

Table 4-3 Global means and standard deviations of DEMs or fusion algorithm products.... 71
1 GENERAL INTRODUCTION

1.1 Study Context and Objectives

A digital elevation model (DEM) is a regularly spaced grid of surface elevations. DEMs are a necessary dataset used in many studies of the Earth’s surface and the physical processes that interact with it. For example, hydrological models require DEMs due to the need for the derivation of terrain features, the delineation of stream networks and sub-catchments, and the identification of variable source areas for runoff (Hopkinson et al., 2009; Quinn et al., 1991; Tarboton et al., 1991; Weschler, 2007; Xiao et al., 2010). Fine-resolution DEMs are also useful for catchment geomorphology characterization (Camargo et al., 2009; Martinez et al., 2010; Smith, 2002) and geomorphic interpretation (Chaplot et al., 2006; Smith and Pain, 2009; Wheaton et al., 2010).

All DEMs contain a certain amount of error as a result of the collection and processing of the data used to generate the DEMs (Weschler, 2007). Advancements in the collection of remote sensing data and DEM generation have improved DEM accuracy and precision, however errors still remain. These errors are troublesome since they can be propagated throughout the data processing workflow that they are used in (Hopkinson et al., 2009).

With an increase in the availability of elevation data, efforts have been made to utilize this data redundancy to reduce error. Several methods of DEM fusion have been proposed and examined in the literature. Many of these involve simple techniques such as data gap filling (Karkee et al., 2008), and the weighted averaging of input elevations based on: global measures of error (Papasaika et al., 2009); height error maps from the DEM generation process (Reinartz
et al., 2005; Roth et al., 2002); terrain derivatives (Papasaika et al., 2009); or combinations thereof.

More sophisticated techniques of DEM fusion involve the use of sparse representations (Papasaika et al., 2011), frequency domain filtering (Honikel, 1998; Crosetto and Aragues, 2000; Karkee et al., 2008), self-consistency in the generation process (Schultz et al., 1999, 2002; Stolle et al., 2005); or multi-scale stochastic smoothing (Slatton et al., 2002).

All of the DEM fusion techniques cited improve the elevation estimates for a given area, to varying degrees, and the success of these methods are not to be discounted. Most of the techniques do, however, require other elevation data (e.g. Crosetto and Aragues, 2000; Choussiafis et al., 2012), or height error estimates from pixel correlation (stereogrammetry) or coherence (interferometry) to control the fusion of the input DEMs. The issue with these methods is that ancillary elevation or height error data is not always available, or reliable, for areas under investigation. As well, most of the DEM fusion methods cited involve the fusion of only two or three DEMs, and often these DEMs are from different sources with a more accurate DEM supplementing a less accurate one.

A new method of fusion for multiple, overlapping DEMs is presented here. The algorithm utilizes $k$-means clustering to detect elevations at each grid cell location that are in close agreement. The main assumption of this technique is that elevations in close agreement will be more accurate than those that are not. Based solely on the distribution of elevation and slope values at each cell location, elevations are filtered by slope and elevation thresholding and are then clustered. In this way, the proposed DEM fusion algorithm is mainly empirical with only a
few data distribution parameters controlled by the user. Most importantly, no \textit{a priori} knowledge of the input DEM error is used in this fusion technique.

While the techniques utilized in the main steps of the proposed DEM fusion algorithm (i.e. clustering and slope thresholding) have not been implemented before in a published DEM fusion method, it is important to recognize that some of the concepts underlying these new techniques are the same as the concepts underlying previously published DEM fusion techniques. The proposed DEM fusion algorithm builds on the concept of self-consistency in elevation methods that was introduced by Schultz et al. (1999). While Schultz et al. favoured elevation estimates at the same location that were consistent when target and reference images were reversed in roles of stereo-photogrammetric DEM generation, the proposed algorithm in this study favours estimates that are clustered at each location, as more accurate. As well, the proposed algorithm in this study utilizes consistency in terrain derivatives (i.e. slope) to estimate more accurate elevations. Recognition of the importance of considering consistency in terrain derivatives at each location from overlapping DEMs was a key component of the sparse representations method in the study by Papasaika et al. (2011).

Several, overlapping RADARSAT-2, Synthetic Aperture Radar (SAR) images were available for the same area in Southern Ontario, Canada, and were suitable for use in same-side, stereo-radargrammetric DEM generation. Stereo-radargrammetry is a technique that matches the amplitude information in pixels of overlapping RADAR images to estimate the image parallax (Fayard et al., 2007). The parallax is used in conjunction with the known geometries of the sensor and images to derive an elevation at each location (Toutin and Gray, 2000).
RADARSAT-2 imagery provides four bands of data, based on different polarizations, which are captured simultaneously for each image pixel. The polarization refers to the way the signal is transmitted and received by the sensor. For RADARSAT-2 the signal can be transmitted horizontally (H) or vertically (V) and also received horizontally or vertically, giving rise to the four possible combinations thereof: HH, VV, HV, VH (Fox et al., 2004).

Toutin et al. (2010) have shown that the four polarimetric bands of RADARSAT-2 imagery can be combined within each scene to create a total power (i.e. SPAN) image that is then used for stereo-radargrammetric DEM extraction. Generating the DEM in this way was shown to increase the elevation accuracy compared to a DEM generated from only the HH channels of the images. The general methodology of Toutin et al. (2010) was used to generate DEMs for use in this study.

With suitable imagery, methodology, and software available to generate multiple, overlapping DEMs for the purpose of fusion, the objectives of this research were to:

a) generate multiple, overlapping DEMs of the same area using stereo-radargrammetric techniques and RADARSAT-2 imagery;

b) develop a DEM fusion algorithm and fuse the generated DEMs;

c) corroborate the input DEMs and fusion algorithm products with a higher accuracy reference DEM, to assess the effectiveness of the DEM fusion algorithm.

1.2 Thesis Outline

This thesis contains six chapters. Chapter 1 is an introduction to the thesis, providing an overview of the research context, aim, and objectives. Chapter 2 contains a literature review on DEM generation, error assessment, causes, and mitigation, globally available DEMs, and a
summary of previously published DEM fusion strategies and results. Chapter 3 contains a
description of the methods used to generate and fuse the multiple, overlapping DEMs for the
study. Chapter 4 provides the results of the DEM generation and fusion as well as the
corroborations of the input and fusion DEM products with a higher accuracy DEM. Chapter 5
contains a discussion of the results presented. The conclusions of the thesis study are in Chapter
6. Additional tables and figures for the thesis are provided in Appendices A through E.
2 LITERATURE REVIEW

2.1 Introduction to DEMs

A digital elevation model (DEM) is a regularly spaced grid which contains the elevation of a point on a surface that is coincident with the location of the grid cell. Often DEMs are also referred to as a DTM (digital terrain model), or a DSM (digital surface model) (Poon et al. 2005). The data used to create an elevation surface can be acquired using various technologies and at different scales. Traditionally elevation data was acquired through ground-based surveying methods (Gao, 2007). The development of remote sensing technologies has enabled elevation data to be derived more quickly and at a greater scale than before. Remote sensing techniques have also provided elevation data for areas that are difficult to access and survey (d’Ozouville et al., 2008).

Remote sensing instruments can be passive or active and work with many different wavelengths and polarizations of energy in the transmitted or received signal. The sensor platforms can include satellite, airborne, or ground-based types. Each platform and sensor type allows for a different scale of data collection at a different resolution, and can account for various conditions in terrain and land cover. The main methods used to derive elevation estimates from remote sensing data are stereoscopy, interferometry, and ranging (also known as altimetry).

DEM errors can be propagated throughout the data processing workflow they are used in, for example in hydrological model simulations (Hopkinson et al., 2009; Wescbler, 2007). It is therefore important to understand the causes of DEM error, how error is evaluated, and strategies
to reduce error. With an increase in DEM coverage globally several DEM fusion techniques have emerged with the purpose of reducing error in elevation estimates.

2.2 Elevation Data Acquisition

2.2.1 Stereoscopy

By viewing two images that are acquired from different angles the disparity in the location of features can be seen as displacement and therefore a 3rd dimension (i.e. elevation) can be observed. This method of extracting elevation is stereoscopy, and is built on principles that relate to the depth perception capabilities of a pair of human eyes (Toutin and Gray, 2000).

2.2.1.1 Stereo-photogrammetry

The use of images from film, digital cameras, or digital scanners to characterize features is called photogrammetry. Techniques of photogrammetry include clinometry and stereoscopy; the latter being more commonly used for elevation extraction (Toutin and Gray, 2000). Initially stereoscopy involved viewing stereo pairs of aerial photographs through a stereoviewer. For this technique the accuracy of elevations derived depends on the altitude at which the photographs are taken and the characteristics of the features observed (Lillesand et al., 2008).

Originally aerial photographs were taken with film, but the development of digital cameras allowed the process to be taken into a computer environment. Computer-based stereo workstations were developed, with which users could view the images and see the features in 3-D. Eventually, this included satellite imagery when digital scanners were employed on satellites (Toutin and Gray, 2000).
The most common stereo-photogrammetry procedure currently used involves image matching computer programs which have replaced the stereo workstations. Images are matched and adjusted either in pairs or blocks of several images with the use of tie points – this is referred to as bundle or block adjustment. By knowing the internal geometry of the camera (i.e. focal length, lens distortion) and the external geometry of the image acquisition (altitude of the platform, angle of nadir relative to the ground surface) the image parallax can be calculated for each matched pixel (Lillesand et al., 2008). Topography can be determined from the parallax in the two images since targets at different heights are displaced by an amount related to their elevation (Leberl, 1990, in Rosen, et al., 2000).

Residual error within the elevation model can be estimated with independent check points (ICPs). If the estimated error is too high then the GCPs can be modified (Gao, 2007). Computation of the elevation model from the image parallax allows relative elevations to be calculated. To achieve absolute elevations a number of ground control points (GCPs) with known horizontal and vertical coordinates are required (Gao, 2007).

Aerial photography can be used to produce DEMs with a vertical accuracy of less than a metre, whereas those derived from satellite imagery are in the range of 3 to10 m in the case of IKONOS (Poon et al., 2005) and QuickBird (Toutin, 2004). DEMs generated from automatic stereo image matching (e.g. optical or RADAR) often, however, contain large erroneous blunders due to incorrectly identified match pixel pairs (Milledge et al., 2009a). As well, DEMs derived from optical stereo images specifically can be inhomogeneous since they depend on image feature contrast, and are also compromised by cloud cover (a major issue in the tropics) and lack of sunlight in some cases (Rabus et al., 2003).
2.2.1.2 Stereo-radargrammetry

Radargrammetry involves images acquired from active, RADAR (Radio Detection and Ranging) sensors, instead of cameras in the case of photogrammetry. There are several advantages to working with RADAR systems rather than optical, passive systems, such as digital cameras. Radar operates with the use of microwave energy allowing electrical and geometrical properties of surfaces to be represented. Operating at this wavelength also allows for all weather operation due to the ability of microwaves to penetrate clouds (Bamler and Hartl, 1998; Toutin and Gray, 2000). Because the RADAR system provides its own source of illumination (an active system) it can operate both day and night. (Rosen et al., 2000; Toutin and Gray, 2000).

Stereo-radargrammetry is similar to stereo-photogrammetry, except that Synthetic Aperture RADAR (SAR) sensors are used instead of cameras. With traditional RADAR the antenna length is a limiting factor in the azimuthal resolution that can be achieved with increasing range. SAR is a technology which solves the issue of a limited antenna length by transmitting pulses ahead of the sensor and receiving the pulses further along in the course of the aircraft or satellite (Bamler and Hartl, 1998). A SAR image pixel can contain the amplitude (energy intensity) as well as the phase (time delay) of the signal (Smith, 2002). Only the amplitude portion of the signal is utilized in stereo-radargrammetry; the phase is utilized in InSAR methods (Bamler and Hartl, 1998; Smith, 2002). SAR data is useful not only for deriving elevations, but also for other areas of research including that of polar ice, vegetation, biomass estimation, and soil moisture mapping (Elachi, 1988, in Rosen et al., 2000). SAR technology is implemented on both aerial and satellite platforms.

Advances in stereo-radargrammetry have recently been achieved with higher resolution modes on satellites such as RASARSAT-2 (Ultra-fine mode is 3x3 m pixel imagery), as well as
improved 3-D radargrammetric models. These models incorporate precise satellite orbiting geometry, and reduce the need for GCPs (Toutin and Chénier, 2009). Toutin and Chénier (2009) tested a new version of Toutin’s 3-D radargrammetric model on RADARSAT-2 Ultrafine Mode imagery and were able to produce a DEM with an accuracy of 1 m horizontally and 2m vertically when compared to DEMs created from orthophotos. Though aerial SAR imagery is capable of producing higher resolution DEMs, the limitation of this method is the extent that can be acquired and the cost of the survey. The recent advances in satellite SAR stereo-radargrammetry can allow for a much greater extent of DEM creation than aerial surveys, with accuracies that are not much lower.

Similar to stereo-photogrammetry, the main cause of large blunders (i.e. spikes or pits) in elevation estimates in stereo-radargrammetric DEMs is pixel matching error (Fayard et al., 2007). Poor correlation between image-pair pixels can result from changes in the backscatter amplitude due to target change between image acquisitions (Toutin, 1998), from speckle that is inherent in most RADAR images (Ostrowski and Cheng, 2000), or is due to a lack of texture in the imagery (Paillou and Gelautz, 1999).

2.2.2 Interferometry

Interferometric SAR is usually called InSAR, and sometimes referred to as IFSAR or ISAR (Rosen et al., 2000). InSAR technology uses SAR phase information, rather amplitude data which is used in stereo-radargrammetry (Rosen et al., 2000; Smith, 2002; Toutin and Gray, 2000). The InSAR viewing geometry for a certain point on the ground involves two SAR antenna positions separated by a baseline (i.e. short distance) and the ground location. Each SAR
antenna measures the phase, which is related to the number of wavelengths of the signal needed to cover the distance from the antenna to the ground and back to the sensor (Smith, 2002).

There are three possible configurations for the two InSAR antennas: across-track, along-track, and repeat-pass. In each technique the phase of one antenna is subtracted from the other for each pixel in the image pair resulting in an interferogram. The difference in phase is related to the baseline and surface relief (Smith, 2002).

Across-track interferometry uses 2 antennas on the same platform (Madsen et al., 1993), whereas along-track involves two satellites following each other with a short separation distance. For these techniques the difference in phase is related to the parallax caused by the different acquisition angles (Smith, 2002). The sensitivity of this technique to terrain topography increases with the baseline distance to the point where there is an optimal baseline for DEM generation (Toutin and Gray, 2000). With the repeat-pass technique the sensor must pass over the same area with almost the exact same viewing geometry for two passes. When this condition is met the baseline is nearly zero and the difference in phase is related to a change in elevation at a particular point (Smith, 2002; Toutin and Gray, 2000).

Since the difference in phase is measured in wavelengths, interferometry is highly accurate in acquiring elevation data in ideal conditions. Smith (2002) gives an example of SAR data from the European Remote Sensing (ERS) satellites being used to create a DEM that is accurate to 2.33 cm in the line-of-site direction. This value is half of the signal wavelength for the sensors. The biggest challenges for InSAR technology are phase unwrapping and decorrelation of pixels (Rabus et al, 2003). The difference in phases of the two SAR images in only known to within one phase cycle, so the appropriate number of phases need to be added to get the true slant range.
of the RADAR signal when it interacted with the target. This is called phase unwrapping, and several methods are reviewed by Bamler and Hartl (1998). Decorrelation is a measure of the reduction of coherence, which is the correlation coefficient of the two SAR images involved in InSAR (Zebker and Villasenor, 1992). Decorrelation most commonly occurs when the orientation of a target changes between RADAR image acquisitions as can occur, for example, in forests on windy days (Reinhartz et al., 2005).

Both InSAR and stereo-radargrammetry are based on SAR technology. The advantage of using InSAR technology instead of stereo-radargrammetry though, is that the accuracy of the elevation values can be in the order of millimetres to centimetres (depending on the platform and sensor), whereas that obtained with stereo-radargrammetry is in the order of metres (Rosen et al., 2000). Also, the InSAR method benefits from automated processing compared to stereoscopy (for RADAR and optical systems) which requires more user interaction in the processing (Madsen et al., 1993; Rosen et al., 2000; Toutin and Gray, 2000). Airborne InSAR can be horizontally accurate to less than a metre (Bamler and Hartl, 1998) which would be appropriate for creating DEMs for hydrological applications. The horizontal accuracy of many satellite InSAR sensors is in the order of 25 m² (in the case of ERS; Bamler and Hartl, 1998) and is more suitable for other applications such as sea ice monitoring and tectonic activity (Gao, 2007).

A ground-based InSAR unit has also been developed, for which the RADAR aperture is made synthetically longer by sliding the transmitting and receiving antennas along a 3 m long track (Nico et al., 2004). A DEM was created using this technique for a test site that was approximately 3 km by 1 km, and had an RMSE of 5 m compared to an existing DEM (Nico et al., 2005). This technology is potentially useful for more localized, large-scale modelling of terrain.
2.2.3 Ranging / Altimetry

One way of acquiring a high-quality DEM is by employing Light Detection and Ranging (LiDAR) technology. LiDAR is a type of ranging technology that is sometimes referred to as laser altimetry. Similar to other active sensors such as RADAR, LiDAR involves the transmission of pulse of energy from a source, the pulse reflecting off a feature, travelling back toward the platform and being received by a sensor (Wehr and Lohr, 1999). The laser wavelength is in the near-infrared range of the electromagnetic spectrum giving LiDAR an advantage that the signal is quite reflective off of natural surfaces and it is more eye-safe than other visible wavelengths (Hopkinson, 2006). The way that the backscatter is recorded is either as a waveform (when the signal is sent as a continuous wave) or as discrete returns (when the signal is sent as a series of pulses) (Bortolota and Wynne, 2005; Hopkinson, 2006). First and last returns of discrete returns can be separated, or the full waveform of a continuous wave can be analysed, to help differentiate the ground location from that of off-terrain objects (Coveny and Fotheringham, 2011).

Often the LiDAR system is mounted on an airborne platform, though ground-based units and satellite sensors are also utilized. A relative coordinate and range of each point is determined using the speed of light, the location and orientation of the source at the time of transmission, and the time between laser pulse transmission and reception (Wehr and Lohr, 1999). In the case of airborne LiDAR, relating the relative coordinates and ranges of the pulse returns to the aircraft trajectory enables the survey points to be translated to ground coordinates with elevations (Hopkinson and Demuth, 2006). This is achieved with an Inertial Motion Unit (IMU), coupled to a high-precision GPS unit, which has enabled the high-precision of LiDAR data acquisition.
(Hopkinson, 2006). The exact precision depends on the survey conditions, but is generally in the range of tens of centimetres (Gao, 2007).

Another advantage of airborne LiDAR is its ability to penetrate forest canopy and other heavily vegetated areas since the pulse (especially small-footprint LiDAR) can pass through relatively small gaps in the ground cover (Wulder et al., 2008). Since aerial LiDAR datasets can produce fine-resolution, high accuracy DEMs they are also useful for assessing the accuracy of other methods of elevation extraction (Toutin et al. 2010). The main disadvantage to aerial LiDAR though, is that it is a relatively expensive method of acquiring elevation data and surveys are often limited to small study areas (Gao, 2007).

LiDAR technology is also available aboard some satellites, including GEOSAT, SEASAT, and ENVISAT, enabling elevation data to be collected at the global scale (Gao, 2007). One such example, the Shuttle Laser Altimeter (SLA) has a sampling interval of 0.75 m vertically and 0.7 km horizontally, with vertical accuracies of 1 m in gentle terrain, and 11 – 46 m in rugged terrain (Garvin et al., 1998). Another example is the Geoscience Laser Altimeter System (GLAS) sensor aboard the Ice, Cloud and land Elevation Satellite (ICESat) that collects LiDAR data with 70 m footprints, spaced 170 m apart. The accuracy of GLAS data is reported to be better than 0.3m (Reuter et al., 2009).

Ground-based LiDAR units are sometimes referred to as Terrestrial Laser Scanners (TLS). They also utilize high accuracy GPS to relate the scan points to a real-world reference system, but do not incorporate an IMU unless they are employed on a moving vehicle (usually TLS systems are stationary). The angle of acquisition is more oblique to the ground surface compared to that of airborne or satellite-based systems, and for this reason multiple scans of terrain from
different directions or azimuth angles are required to minimize the effect of occlusion (see Section 2.3.1). Ground-level vegetation can be a particularly challenging source of occlusion for TLS systems (Coveney and Fotheringham, 2011). TLS systems are more limited in the size of area that can be surveyed compared to the aerial and satellite-based systems, and so they are more suitable to support field-scale elevation data requirements.

2.3 DEM Error

All DEMs contain error that is a result of limited measurement precision, the presence of off-terrain objects in the acquisition area, and interpolation (Burrough and McDonnell, 1998, in Lindsay, 2006), as well as error that occurs during data processing (Weschler, 2007). Also, errors in acquisition can be caused by the characteristics of the terrain or landcover, for example: the moisture content of soil or vegetation, the slope or aspect of topography, and the roughness of surfaces (Bater and Coops, 2009; Reuter et al., 2007; Toutin, 2002). Error can also be propagated if an unsuitable interpolation method is chosen to process the DEM.

2.3.1 Error due to landcover and terrain characteristics

Some forms of error in data acquisition and elevation extraction are unique to the sensor being employed, such as that caused by limited measurement precision, or during the processing of data. There are other errors in data acquisition which are more common amongst sensors though: signal scattering, occlusion, attenuation and multipath. These errors can be propagated into the DEM created from the acquired data (e.g. imagery).

Slope and the orientation of slope (i.e. aspect) can affect how much of the instrument signal is reflected back to the sensor. Toutin (2002) conducted a study on how accuracy relates to the
slope and aspect of terrain in a DEM derived from RADARSAT image stereo-pairs. He found that stereo-radargrammetric DEM error was linearly related to slope, with steeper terrain causing more error. This error is due to radiometric disparity in the images (differences in signal amplitudes for the same pixel location in two images) causing image matching errors. Li et al. (2006) conducted a study using ERS data and also found that DEM error increased linearly with the slope of the terrain. As well, Toutin (2002) also found that topographic aspect (orientation of slope with respect to the position of the sensor) played a minor role in error, with fore-slopes being more accurate, and back-slopes being less accurate.

Specular reflection can cause the transmitted signal to be entirely reflected away from the instrument resulting in none of the energy returning to the sensor. A good example of a surface that causes this is still water, since it can cover a large portion of the terrain at times and can be very smooth. Specular reflection can occur in passive optical systems when the sun is at a low angle to the terrain compared to the sensor (Lillesand et al., 2008), or in active systems such as LiDAR (Hopkinson, 2006) and SAR (Reuter et al., 2007), when the transmitted signal is reflected completely away from the sensor. Denker (2005) reported that water bodies were not well defined in the initial version of SRTM3 data since they cause a low amount of RADAR backscatter. Reflection away from the sensor results is a loss of data for the surface that was the cause of the reflection. In the case of surface water, this is not necessarily a major issue for DEM creation since the elevation of the water surface is usually the same throughout.

Occlusion is another cause of missing elevation data, except in this case the data can be missing for non-uniform areas of elevation. Occlusion is caused when the signal is reflected off of a feature before it is able to reach an area of interest. In a sense, an object blocks the line-of-sight for the instrument. It can occur in terrain of steep relief, in heavily vegetated areas, or urban
centres amongst other cases (Lillesand et al., 2008). The area in the ‘shadow’ of the occluding object is simply missing from the acquired data. As an example, Coveney and Fotheringham (2011) discuss the issues of occlusion due to dense ground vegetation when using a TLS system. Also, Denker (2005) notes the absence of SRTM3 data in the Alps where the mountains are high and there are narrow gorges.

Attenuation occurs when the medium that the signal is travelling through absorbs the energy of the signal. Moisture in soil or vegetation is a common cause of signal attenuation (depending on the signal wavelength). Attenuation can occur in LiDAR (Hopkinson, 2006), RADAR (Dobson and Ulaby, 1986; Reuter et al., 2007), and optical systems (Lillesand et al., 2008). It should be noted that high soil moisture, measured as bulk water differentiated from bound water, can increase the backscatter for RADAR, rather than attenuate the signal (Dobson and Ulaby, 1986). For elevation derived from image pairs, attenuation can cause image matching errors if there is a change in conditions between images. In the case of LiDAR, severe attenuation will cause data drop-outs (Hopkinson, 2006).

Multipath errors cause a much different effect than the aforementioned errors in data acquisition. Multipath occurs when the signal of an active instrument is reflected off of more than one surface before returning to the sensor (Hopkinson, 2006). The assumption of the instrument is that the signal will travel in a straight line to and from a target. When this does not happen, the increased time in transmission and reception of the signal translates into an increased range. The information recorded for that location then has either an erroneous elevation or intensity (Lillesand et al., 2008). Situations that cause multipath include corner reflection from angular, highly reflective features such as buildings (Stilla et al., 2003), and where trees or other heavy vegetation over-hang surface water (Townsend, 2002).
2.3.2 Interpolation

After the acquisition of elevation data, interpolation or aggregation techniques are used to generate DEMs (Weschler, 2007). Interpolation algorithms estimate a variable of interest (elevation in the case of DEMs) at unmeasured locations - usually the centre or corner of a grid cell - using the locations and values of sample points (Chaplot et al., 2006). Often not much is known about the error which occurs as a result of the interpolation process (Desmet, 1997). For this reason, Chaplot et al. (2006) suggest that topographic modelers should be careful when selecting a method that will interpolate values between points of elevation. Many researchers have focused on the uncertainty associated with interpolation methods, but in general there is no single method that is the most accurate when used on terrain data (Fisher and Tate, 2006; Welchler, 2007). It is worth discussing some of the more commonly used techniques in DEM creation, and so an overview of inverse distance weighted (IDW), TIN-based (Triangulated Irregular Network), spline, and kriging methods is provided here.

One of the most widely used interpolation techniques for modeling surfaces is IDW (Aguilar et al., 2005). In the IDW method the value of a point at a certain location is related to the known values of neighbouring points, weighted by the distance from the new point. The weight is inversely proportionate to the power function of the distance (Chaplot et al., 2006). In this way, known points do not have an effect on each other’s weights. IDW seems to be preferred for interpolating DEMs because it creates a very smooth surface that is visually pleasing, especially for bare-earth DEMs, but this does not mean that it is the most accurate. One effect that this method can have on the resultant DEM is that a bulls-eye pattern can occur, where noise in the data is amplified, if the power function is set too high. Aguilar et al. (2005) found that from the various methods they tested, IDW was less appropriate for modelling elevation. Similarly, Bater
and Coops (2009) also found that of the seven interpolation methods they tested on airborne LiDAR data, IDW was the least accurate, and added that it resulted in a stepped pattern in the data which would have a significant impact on any terrain analysis.

In the Bater and Coops (2009) study it was concluded that TIN-based interpolation produced the most realistic surface, compared to the seven other methods including IDW, spline, and linear interpolation, at three different resolutions. The TIN-based method that they chose was Natural Neighbour, which is based on Voronoi polygons derived from the sample points and the TIN surface. The known points (i.e. LiDAR points) are connected to their closest neighbours with lines, creating a Delaunay TIN surface. Polygons are then created with sides that are equidistant from two neighbouring points and perpendicular to the TIN line which would connect those points. This forms the original set of Voronoi polygons where one polygon surrounds each of the known points. When a point of unknown characteristics is inserted into the point network a new Voronoi polygon network is created. The proportion of overlap of the polygon of the unknown point and the original polygons determines the weighting of the values associated with the original polygons (Boissonnat and Cazels, 2000).

IDW and Natural Neighbour are examples of deterministic interpolation methods which estimate the value of unknown points based on the influence of immediate neighbours of known points. Deterministic methods are more computationally efficient, but do not take into consideration the patterns, correlation of values, or errors across the entire area being interpolated. To solve this issue probabilistic geostatistical methods have been created. These methods incorporate not only distance but also direction in determining influential surrounding values, and many also consider spatial autocorrelation of value and errors in the surface (Maune

Kriging is similar to IDW and Natural Neighbour since it estimates values based on the local average, but this concept is taken a step further by considering the spatial variation of the data and the configuration of the data to minimize the variance in interpolated values (Desmet, 1997). Deutsch and Journel (1992, in Grohmann and Steiner, 2008), state that kriging may be most useful for re-sampling elevation data since it honours points at their original locations while interpolating data for the areas between points. For this reason kriging is often chosen for sparse datasets to gain a best estimate of the values for the area between points. Oliver and Webster (1990, in Fisher and Tate, 2006), add that this method is the best linear unbiased estimator and that the error introduced in the estimation can be directly determined, making it a desirable method to use from a statistical stand point. Fisher and Tate (2006) warn that the variance of the kriged surface is directly related to the distance of the estimated value from the known value. Therefore there may be cases where the dataset is too sparse to allow for a realistic surface to be produced. Also, in the case of LiDAR data which is very dense, kriging will produce extra values between points unnecessarily.

Splines are a general class of interpolation techniques that create a surface with minimal curvature while still passing through the sample points (Aguilar et al., 2005). The surface created can be conceptualized as a thin metal plate that has been forced to bend through or very close to the known sample points (Desmet, 1997). This interpolation method is suitable for terrain with smooth slope transitions that does not greatly vary in elevation, and is not suitable for cases where sharp changes occur in short distances (e.g. man-made features or cliffs) (Mitasova and Litas, 1993, in Aguilar et al., 2005). If used on a very large or dense dataset, spline methods
become computationally expensive and there can be an increase in the numerical instability of
the solution found (Lazzaro and Montefusco, 2002, in Aguilar et al., 2005).

Previous research, as mentioned in other studies (Bater and Coops, 2009; Chaplot et al.,
2006; Fisher and Tate, 2006) has found that there is no interpolation method that is universally
superior for the creation of DEMs. Success of any one interpolation method is based on the
nature of the terrain and the distribution of the source data (Fisher and Tate, 2006), as well as the
cell resolution of the desired DEM and the assumptions of the mathematical design of the
method (Bater and Coops, 2009).

2.3.3 Error estimation

DEM error is usually estimated using a more accurate reference dataset such as GPS points
(Coveney and Fotheringham, 2011; Gao, 2007) or other DEMs (Fisher and Tate, 2006). The
most widely used measure of DEM error is the Root Mean Squared Error (RMSE) (Aguilar et
al., 2005; Desmet, 1997). RMSE is the square root of the average of squared differences taken
between the DEM being assessed and reference points that are believed to be of higher accuracy
(Weschler, 2007). The equation for RMSE is given below:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(Z_{i}^{est} - Z_{i}^{ref})^2}{n}}$$

where $Z_{i}^{est}$ is the estimated elevation and $Z_{i}^{ref}$ is the reference elevation at location i, and n is
the number of residuals calculated (modified from Aguilar et al., 2005). A number of studies
have shown that the mean error is not equal to zero, and so the RMSE alone is not a good way to
describe the statistical distribution of the error (Fisher and Tate, 2006). Fisher and Tate (2006),
and Desmet (1997) suggest that at least the Mean Error (ME) and the standard deviation of the error (S) should be reported along with the RMSE.

Even though the RMSE is the most commonly used estimate of DEM accuracy Weschler (2007) as well as Fisher and Tate (2006) stated that it is not necessarily the most appropriate. The RMSE assumes that DEM errors are random, and that they are normally distributed in their values, which is not true of most DEMs. The RMSE also does not reflect how well each cell of the DEM reflects the true elevation (Weschler, 2007). Weschler further argues that error would be better represented in a probability map, and that the contribution of error sources should be quantified to allow for a better understanding of the nature of the error.

Fisher and Tate (2006) argue that the RMSE, ME, and S all fail to represent the spatial pattern of error, which is an important consideration in DEMs since error tends to be spatially correlated. As a solution they recommend that either unconditioned or conditioned error models be used. Unconditioned error simulation models use stochastic simulations of random function realizations that can be applied to the DEM, often through the use of a Monte Carlo simulation method. They are based on properties of the error distribution but actual estimates of error are not honoured. The assumption of unconditioned error simulation models is that the error pattern is uniform over the entire DEM, which is often not the case. Conditional error simulation models differ in that they honour the estimates of error at particular locations and therefore do not assume that the error pattern is uniform (Fisher and Tate, 2006).

It is important to note that estimates of error that are based on reference data sets are not actually estimating absolute error from the true ground surface, but rather are discrepancies from the reference data values, since even the most accurate ground truth data contains a certain level
of error as well (Gao, 2007). Papasaika et al. (2009) suggest that in the absence of a higher quality reference data set, alternative techniques such as the evaluation of slope, aspect, and roughness, can be used to assess the accuracy of a DEM. Reuter et al. (2009) suggest that even with a reference dataset terrain parameters such as slope and curvature should be evaluated to assess the accuracy of a DEM.

2.4 Global DEMs

The first attempts to create a DEM of the globe involved merging elevation from multiple sources into a single product with the greatest coverage possible. More recent advances in space-borne remote sensing instruments have allowed for near global coverage from single sensors. Examples of both types of global DEMs are briefly reviewed in this section.

2.4.1 Multi-source global DEMs

Initial efforts to create a DEM with global coverage involved the merging of elevation data from multiple sources. Two of the main products that resulted from these efforts were GTOPO30 and GLOBE.

GTOPO30 is a global DEM with a grid spacing of 30 arc seconds (approximately 1 km). It was compiled from eight different elevation data sources and was developed for regional and continental scale topographic data use (Harding et al., 1999). The main data sets and the percentage contributing to GTOPO30 for land areas were: Digital Terrain Elevation Data (DTED, 50%), a 1 degree elevation model for the USA (6.7%), and Digital Chart of the World vector data (DCW, 29.9%) (Miliareis and Argialas, 2002). The DCW vector data used were
contours, spot heights, stream lines, lake shorelines and ocean coastlines; all of which were converted to a raster with drainage enforcement (Harding et al., 1999).

Because several raster and vector sources of topographic information were used the accuracy of GTOPO30 varies by location according to the source data (Denker, 2005). For example, Harding et al. (1999) note that the New Zealand DEM RMSE is 9 m whilst the Peru map RMSE is 304 m. Miliaresis and Argialas (2002) also compare the plus or minus 30 m accuracy of DTED to the plus or minus 160 m accuracy of DCW data in the GTOPO30 DEM. Despite the large variation in elevation accuracy, GTOPO30 data was a major contributor to GLOBE (Hastings and Dunbar, 1998) which was the next global DEM initiative.

The Global Land One-kilometer Base Elevation (GLOBE) DEM was initially an empty 2-dimensional, 30 arc-second array that was opened and the best available data used to fill it. It was developed before the scheduled launch of SRTM (Shuttle RADAR Topography Mission; described in the next section of this literature review) in 1999. The U.S. Defense Mapping Agency contributed DTED Level 0 data, and national elevation datasets were also contributed from Australia, Japan, and New Zealand. As well, GLOBE contained data from the GTOPO30 DEM (Hastings and Dunbar, 1998).

Similar to GTOPO30, GLOBE can be locally quite unreliable since the merged datasets were acquired with a variety of sensors and many different techniques were employed during the elevation generation process (Rabus et al., 2003). Hastings and Dunbar (1998) state that perhaps half of GLOBE exceeds 20m RMSE and that the area of Antarctica could be off as much as 300m vertically. They also note that some errors may be from differences in projection and datum since many datasets do not come with proper documentation.
2.4.2 Single-source global DEMs

Advances in space-borne remote sensing instruments have allowed for more DEM coverage globally from single sensors. Examples of these DEMs produced from single sensors are SRTM, GDEM, and the soon to be released WorldDEM.

The Shuttle RADAR Topography Mission (SRTM) acquired RADAR data in C and X bands, over an 11 day period in February 2000, for the entire landmass of the Earth between 60° North and 57° South latitude (Rabus et al., 2003). The objective of the data collection was to obtain the most complete, high-resolution topographic database of the Earth at that time (Sun et al., 2003). The SRTM system was the first single-pass interferometer in space, and was an across-track system on the same sensor so that image pairs were acquired under virtually the same atmospheric and ground conditions which reduced decorrelation (Rabus et al., 2003).

Sun et al. (2003) stated that the absolute vertical accuracy of the SRTM DEM was 16 m (90% LE – linear error). Denker (2005) reported that the first public release of SRTM elevation data was to the research community, to be tested in 2004, and was a 3 arc second (approximately 90 m) resolution DEM called SRTM3. They also reported the SRTM3 product had numerous voids and spurious points, and that most of the water bodies were not well-defined because they likely produced low backscatter. Some of these errors in the SRTM DEM were used as the research context for developing some DEM fusion techniques (which are discussed in the next section of this literature review).

The next attempt to produce a global DEM from a single sensor was the Global DEM (GDEM) produced from optical imagery collected by the Advanced Space borne Thermal Emission and Reflection Radiometer (ASTER) system on the Terrasat satellite (Reuter et al.,
The observation period was from 2000 to 2007 resulting in data coverage between 83° South and 83° North latitude, with many areas imaged several times and the results merged (Hirt et al., 2010). GDEM was released on June 29th 2009 as a 1 arc second (approximately 30 m) resolution product with a vertical accuracy of 20 m (at a 95% confidence interval) (Reuter et al., 2009). This DEM was of a finer resolution and greater coverage than the SRTM DEM. Reuter et al. (2009) also reported that the GDEM was 100 times more detailed than the GTOPO30 and GLOBE DEMs. The GDEM dataset is not without errors though; Hirt et al. (2010) found systematic errors from a strip effect over much of Australia, as well as some un-removed cloud elevations in the DEM.

The next global DEM from a single sensor is anticipated to be WorldDEM generated from the Tandem-X (i.e. X-band RADAR) InSAR system. The 12 m resolution DEM with global coverage from pole to pole is scheduled to be released in 2014. The main source of information available about the WorldDEM and its reported vertical accuracy of 2 m (relative) and 10 m (absolute) is the providers’ website (Astrium, 2013).

2.5 DEM Fusion

There are several DEM fusion techniques that have been proposed and tested in the literature over the past three decades. Many of these involve simple techniques such as data gap filling, and the weighted averaging of input elevations based on: global measures of error; height error maps from the DEM generation process; terrain derivatives; or combinations thereof. More sophisticated techniques of DEM fusion involve the use of sparse representations, frequency domain filtering, self-consistency in the generation process, or multi-scale stochastic smoothing.
2.5.1 Fusion with weights

The simplest form of DEM fusion is taking the average of all available overlapping DEMs at each cell location. This type of fusion is not satisfactory though, since large errors can skew the average; the magnitude of the error will be reduced but the resultant DEM will still have blunders (Schultz et al., 2002). A more intelligent approach is to apply weights to the elevation estimates based on error probability to control their influence on the fused elevation estimates. Roth et al. (2002) used weighted averaging to fuse the overlapping portions of DEMs derived from MOMS-2P, SRTM-X, and ERS-Tandem imagery. The weights were derived from the image geometry (an estimated overall accuracy), and height error maps from the DEM generation process (an estimated accuracy at each cell location). Also, a statistical outlier test was performed to identify and correct errors in the fused DEM. The results of the fusion were provided as DEMs and maps of estimated height error for a subset of the study area. Roth et al. (2002) reported that the overall quality of the fused DEM was better than the input DEMs.

Combinations of SRTM-C, SRTM-X (both InSAR) and optical along-track stereo SPOT-5 DEMs were fused in a study by Reinartz et al. (2005). Height error maps created during the InSAR DEM generation process, derived from the coherence and density of residuals, were applied as weights to the SRTM DEM elevations. Weights for the SPOT-5 DEM estimates were derived from the mean standard deviation and the density of matched points in the stereophotogrammetric DEM generation process. The input DEMs and the resulting fusion DEMs (all of which were DSMs) were corroborated with a higher accuracy DTM. The mean, standard deviation, minimum and maximum of offsets were provided for landuse areas defined as fields, suburbs, or forests. For the fields and forests landuse classes the means of the fusion DEMs were not lower than the mean offsets for the input DEMs. The standard deviations of the fusion DEMs
for these classes were however lower than that of the input DEMs except for the SRTM- and SRTM-C DEM fusion. As well, maps of the probability of height error also showed the most improvement when all input DEMs were fused.

A more sophisticated technique for fusing DEMs using weights was presented by Papasaika et al. (2009). Weights were calculated from both *a priori* information about the DEM error (i.e. generation technology and one global estimate of error) and from terrain derivatives of the input DEMs (i.e. slope, aspect, roughness). Land use classes derived from classification of the IKONOS imagery, including trees, buildings, streets, shadows, fields, bare ground, and water, were also used as weights in the fusion process. Two DEMs, generated from LiDAR data and stereo IKONOS imagery, were fused using an active surface model that attracted the less accurate surface to the more accurate one. The shape of the resultant DEM was driven by nominal accuracy and the generation technique of the input DEMs, as well as the land cover; all of which were categorized as internal forces. The fusion was also driven by external forces to coincide with the geomorphological features of the more accurate DEM. Unfortunately this study only presented preliminary results of the fusion in the form of visual assessments for two subset areas that show an improvement in hedge and building representation in the fused DEM.

### 2.5.2 Sparse representations

Building on previous research by the authors, Papasaika et al. (2011) proposed and tested the use of sparse representation theory in DEM fusion. The use of sparse representations would allow DEM fusion to be represented as a mathematical optimization problem that could be solved to global optimality. In this technique the area of interest is segmented into overlying patches of grid cells. Dictionaries of patches (i.e. unique combinations of terrain shape) are
created from higher accuracy DEMs in training areas. Error weights calculated from the slope and roughness of the input DEMs are also used in the fusion. The model is optimized to globally minimize the difference in expected elevations and the input DEMs. This approach was tested on pairs of DEMs, though the authors claimed that it could be extended to more than two DEMs. The three input DEMs were generated from ALOS/PALSAR-2, ERS C-band, and SPOT imagery. A LiDAR DEM was used to corroborate the input and fusion DEMs, and mean offsets as well as RMSE values were provided as results. The ALOS-SPOT DEM fusion had a lower mean and RMSE compared to the input DEMs, whereas the other fusion pairings had either a lower mean, or a lower RMSE than the input DEMs.

2.5.3 Frequency domain filtering

Frequency domain filtering as a method of DEM fusion was first introduced by Honikel in 1998, and has since been tested and published by a few others (Crosetto and Aragues, 2000; Karkee et al., 2008). The basis of this technique is that the lower frequency portion of one DEM (i.e. coarser terrain features) can be isolated and merged with the higher frequency portion of another DEM (i.e. finer terrain features) of the same area. This has been applied to DEMs generated with InSAR and stereo-photogrammetry techniques since the InSAR DEMs tend to be more accurate in the high frequency, and the stereo-photogrammetric DEMs tend to be more accurate in the low frequencies. Frequency domain filtering involves four main steps: converting the DEMs into the frequency domain; applying a low or high pass filter to the appropriate DEM; adding the two desired DEM portions; converting the resultant data back to the spatial domain. Honikel (1998) tested this method on an InSAR DEM generated from ERS imagery, and on a stereo-photogrammetric DEM generated from SPOT imagery. Different cut-off frequencies were
applied and in all cases the mean offset was the same as the higher accuracy SPOT DEM, but the RMSE was lower than that of both input DEMs.

In another study by Crosetto and Aragues (2000) a stereo-radargrammetric DEM generated from RADARSAT-1 imagery and an InSAR DEM generated from ERS-1 imagery were fused using frequency domain filtering. Unfortunately, the DEM fusion was not the focus of the article and so the results of the fusion were not well reported however, the authors did state that the fusion removed systematic errors that were present in the InSAR DEM, and that the DEM precision was also improved.

A more recent study, by Karkee et al. (2008), included a DEM gap filling step before the frequency domain filtering step in the fusion of a stereo-photogrammetric DEM generated from ASTER imagery and an SRTM-C (InSAR) data. The gap filling was necessary since the SRTM DEM contained many holes due to shadow in the RADAR imagery and areas of poor coherence between images. An erosion technique using the slope and aspect of the SRTM cells surrounding the gap was used to fill the gaps. After the gap filling and frequency domain filtering, the input DEM and fusion DEM were corroborated with a 1:25000 scale contour map of the study area. Karkee et al. (2008) reported that the fused DEM had an RMSE that was 42% lower than that of the ASTER DEM, and 10% lower than that of the SRTM DEM. The mean offsets for all DEMs were the same value, due to the co-registration of the input DEMs before fusion.

### 2.5.4 Self-consistency

Another method of DEM fusion, specific to DEMs generated from stereo-photogrammetry techniques, employs the theory of self-consistency (Schultz et al. 1999, 2002). In this method two DEMs are generated from the same pair of images by switching the reference and target
roles for elevation extraction. If the elevation estimates at the same cell location differ by greater than a threshold distance the estimates are not accepted. The threshold is determined by fitting all disparities between the elevations of the DEM pairs to a Gaussian distribution, and threshold is a user specified number of standard deviations from the mean of the distribution.

Schultz et al. (1999) generated 12 DEMs from six pairs of aerial photography images of a barren desert area and applied the self-consistency constraint to filter for reliable estimates. The reliable estimates at each cell location in the study area were simply averaged to create the final fused DEM. The average of each DEM pair (created from the reversal of roles of the imagery) was also created for comparison with the fused DEM. Schultz et al. (1999) reported that the fused DEM was slightly less accurate (measured by the standard deviation of offsets with the ground truth) than the average DEMs generated from three out of five DEM pairs. It was not explained why the results of the sixth DEM pair were not provided. The study by Schultz et al. in 2002 only provided results for five of the original DEM pairs, and not for the fusion product. The fusion strategy was however, applied to DEMs derived from IKONOS imagery over an air force base as an example an urban area, which provided new results. Only visual assessments of the air force base area however were provided for one of the DEM pairs and the fused DEM.

The article by Stolle et al. (2005) was authored by many of the same people that authored the Schultz et al. 2002 paper. In that more recent study (2005) the same methodology for self-consistency fusion was applied to a different area of desert, and to 18 DEMs of a different urban setting that contained high-rise buildings. Only visual assessments and graphed distributions of elevations were provided as results for one of the input DEMs and the fused DEM.
2.5.5 Multi-scale stochastic smoothing

A multi-scale Kalman smoothing filter was used by Slatton et al. (2002) to fuse InSAR DEMs of different resolutions. The multi-scale Kalman filter was employed because it considers the stochastic variability in parameters and is optimal with respect to the minimal mean squared error involved in the DEM fusion model. In the Slatton et al. (2002) study, one low resolution InSAR DEM derived from ERS1/2 imagery was fused with three higher resolution InSAR DEMs generated from TOPSAR imagery. The resultant fused DEM of the ERS DEM fusion with the first TOPSAR DEM was then fused with a second TOPSAR DEM, and that resultant DEM was fused with a third TOPSAR DEM. The results showed that the mean height uncertainty decreased with each additional DEM that was added to the fusion process.

2.6 Research Gaps and Opportunities

There are several methods of acquiring data with remote sensing for the purpose of creating a DEM. Each method has benefits and limitations of use and accuracy depending on the characteristics of the terrain and land cover, processing requirements, cost of acquisition and desired area of coverage. Some data types are useful strictly for elevation acquisition, or are have multiple other uses potentially making them a more desirable data set. Satellite RADAR (specifically SAR) technology has emerged as one of the leading forms of elevation acquisition, in the form of stereo-radargrammetry and interferometry methods, for the coverage of large areas.

A major consideration in the creation of a DEM is the potential sources of error and how the error is reported. Aside from the accuracy of measurement of the instrument employed, interactions of the transmitted energy with features before reception by the sensor are important
to understand and account for when assessing the acquired data. Each elevation dataset requires interpolation to create a gridded DEM, and interpolation can also add to the error of the data. There is not a single method of interpolation which is universally the best, but rather some methods perform better than others depending on terrain and land cover attributes, acquisition technique, and desired outcome of DEM characteristics.

Advancements in DEM generation globally have increased the accuracy and coverage of DEMs. Depending on the method of data acquisition, the characteristics of the land cover and DEM generation, these DEMs still contain inherent errors. This redundancy of DEM data and the need to further reduce errors has provided the opportunity and rationale for the development of DEM fusion techniques.

Many of the DEM fusion techniques published to date are indeed successful in increasing the accuracy or the precision of elevation estimates for a region of interest. Most of these fusion methods however, fuse only two or three DEMs at a time and rely on more accurate estimates of elevation or error to improve a less accurate DEM. Only one of the methods reviewed in this chapter (self-consistency) was entirely data-driven by the elevation estimates themselves, but even in this technique the multiple resultant estimates of elevation at each cell location were simply averaged as the final fusion step.

The question then arises: can a data-driven DEM fusion technique be developed to fuse multiple overlapping DEMs of the same area in an intelligent way that goes beyond simply averaging the estimates at each cell location? To date, such a technique has not been presented in published literature. This presents an exciting opportunity for research in the field of DEM fusion. While keeping the aspects of DEM generation, as well as possible error sources and
estimation, in mind it may be possible to increase the accuracy and precision of a DEM product using multiple, overlapping DEMs, with a newly developed DEM fusion algorithm that exploits the redundancy of data.
3 METHODS

3.1 DEM Fusion Algorithm Overview

The DEM fusion algorithm developed in this study involves the fusion of multiple, overlapping DEMs of the same geographic study area. The goal was to fuse the DEMs without any \textit{a priori} knowledge of the error in the DEMs. This fusion method is therefore based on the distribution of elevations at each cell location, and the user input is limited to data distribution parameters.

At each cell location the elevations available from the multiple DEMs are first removed or retained by slope and elevation thresholding. The threshold values are defined by the user as a multiple of the standard deviation of the slope values or elevations at each location. Retained elevations are then clustered using a \textit{k}-means clustering algorithm. The cluster merging distance is also defined by the user, and is a percentage of the total range of retained elevations at each cell location. Figure F-1 in Appendix F contains a flowchart of the steps involved in the \textit{k}-means clustering portion of the fusion algorithm, and Appendix G contains the programming code for the \textit{k}-means clustering. The results of the clustering step are then filtered with an adaptive mean filter, and smoothed with a Gaussian filter.

The data requirements for the algorithm are multiple, overlapping DEMs generated at same resolution. The assumptions of this algorithm are that the DEMs are referenced to the same horizontal projection and datum as well as the same vertical datum. Figure 3-1 contains a flowchart that outlines the main steps of the fusion algorithm.
3.2 Study Area

The DEM fusion algorithm was tested on an area of overlap for 12 DEMs that was buffered inward by 200 m to avoid edge effects of the DEMs. The 316 km² study area is in Southern Ontario, Canada, Northwest of the city of Guelph (see Figure 3-2). The study area has a post-glacial landscape that is mainly gently sloping with some hummocky topography present. The elevation in the area ranges from 310 to 443 m above mean sea level (referenced to the 2008...
Earth Gravitational Model – EGM 2008). The Grand River flows in the general direction of southwest through the study area from the town of Fergus to the city of Kitchener-Waterloo. The main river has cut a narrow gorge (up to 22 m deep) into the limestone bedrock in the Northern half of the study area. The gorge and a few limestone and aggregate quarries are the only occurrence of steep terrain in the study area.

The land-cover type in the study area is mostly cultivated agricultural fields (51%), with crops consisting mainly of corn, soybeans, wheat, or forages. Pastures or areas with small shrubs are considered rough land in this study and cover 17% of the total study area. Forests are a mix of temperate deciduous and coniferous tree species, and cover 13% of the study area. The urban areas cover only 6% of the study area, and consist mainly of small houses and yards with some larger buildings such as two and three story factories or warehouses. There are no large city centres or high-rise buildings in the study area.

The extent of all RADARSAT-2 scenes used in this study occurs over a 1350 km² area that defines the extent of the data collection and processing for this project (i.e. RADARSAT-2 imagery, GPS data, etc.). The data collection area is shown in Figure 3-2, and also in a map of the RADARSAT-2 scene extents provided Figure 3-3.
3.3 RADARSAT-2 Imagery, GCP coordinates, and DEM Extraction

The general methods of Toutin et al. (2010) were followed to extract DEMs using stereo-radargrammetric methods and RADARSAT-2 (SAR) imagery. This section provides the general steps involved in: processing the RADARSAT-2 imagery; ground control point (GCP) and tie point (TP) collection; DEM generation.

3.3.1 RADARSAT-2 scene selection and processing

Several scenes of RADARSAT-2 Fine-Quad mode, SLC (single look complex) imagery, at various incidence angles (22–49°) were acquired over the study area during the spring and autumn of 2010. The scenes were provided by Agriculture and Agri-Food Canada. The imagery had a resolution of approximately 4.5 by 7.5 m in the slant range, and they were projected in
WGS 1984 UTM Zone 17N. Of this dataset, 16 scenes were used for DEM extraction (see Figure 3-3 for a map of RADARSAT-2 scene extents). Selection and pairing of the scenes for same-side stereo-radargrammetric DEM extraction was based on the date of acquisition, the orbit pass direction, incidence angle, and amount of overlap between scenes (see Table 3-1).

Figure 3-3 Map of RADARSAT-2 scene extents for the imagery used to generate the DEMs.
Table 3-1 RADARSAT-2 scenes used for DEM extraction with attributes used for scene selection and pairing. Scene pairs are numbered according to the ID of the DEM they were used to generate. General pass directions are labeled: A – ascending; D – descending.

<table>
<thead>
<tr>
<th>Scene No.</th>
<th>Acquisition date: dd/mm (in 2010)</th>
<th>Beam Mode</th>
<th>Incidence Angle</th>
<th>Look direction</th>
<th>Pass Direction: azimuth (General)</th>
<th>Scene Pairs (number corresponds to the DEM created)</th>
</tr>
</thead>
<tbody>
<tr>
<td>S01</td>
<td>11/04</td>
<td>FQ2</td>
<td>77°</td>
<td>347°</td>
<td>(A)</td>
<td>1</td>
</tr>
<tr>
<td>S02</td>
<td>14/04</td>
<td>FQ16</td>
<td>79°</td>
<td>349°</td>
<td>(A)</td>
<td>1 2</td>
</tr>
<tr>
<td>S03</td>
<td>05/05</td>
<td>FQ2</td>
<td>77°</td>
<td>347°</td>
<td>(A)</td>
<td>2 3</td>
</tr>
<tr>
<td>S04</td>
<td>08/05</td>
<td>FQ16</td>
<td>79°</td>
<td>349°</td>
<td>(A)</td>
<td>3</td>
</tr>
<tr>
<td>S05</td>
<td>22/09</td>
<td>FQ21</td>
<td>80°</td>
<td>350°</td>
<td>(A)</td>
<td>4</td>
</tr>
<tr>
<td>S06</td>
<td>02/10</td>
<td>FQ31</td>
<td>81°</td>
<td>351°</td>
<td>(A)</td>
<td>4</td>
</tr>
<tr>
<td>S07</td>
<td>26/10</td>
<td>FQ31</td>
<td>81°</td>
<td>351°</td>
<td>(A)</td>
<td>5</td>
</tr>
<tr>
<td>S08</td>
<td>09/11</td>
<td>FQ21</td>
<td>80°</td>
<td>350°</td>
<td>(A)</td>
<td>5 6</td>
</tr>
<tr>
<td>S09</td>
<td>19/11</td>
<td>FQ31</td>
<td>81°</td>
<td>351°</td>
<td>(A)</td>
<td>6 7</td>
</tr>
<tr>
<td>S10</td>
<td>03/12</td>
<td>FQ21</td>
<td>80°</td>
<td>350°</td>
<td>(A)</td>
<td>7</td>
</tr>
<tr>
<td>S11</td>
<td>15/04</td>
<td>FQ20</td>
<td>280°</td>
<td>190°</td>
<td>(D)</td>
<td>8</td>
</tr>
<tr>
<td>S12</td>
<td>18/04</td>
<td>FQ5</td>
<td>282°</td>
<td>192°</td>
<td>(D)</td>
<td>8 9</td>
</tr>
<tr>
<td>S13</td>
<td>09/05</td>
<td>FQ20</td>
<td>280°</td>
<td>190°</td>
<td>(D)</td>
<td>9 10</td>
</tr>
<tr>
<td>S14</td>
<td>12/05</td>
<td>FQ5</td>
<td>282°</td>
<td>192°</td>
<td>(D)</td>
<td>10 11</td>
</tr>
<tr>
<td>S15</td>
<td>02/06</td>
<td>FQ20</td>
<td>280°</td>
<td>190°</td>
<td>(D)</td>
<td>11 12</td>
</tr>
<tr>
<td>S16</td>
<td>05/06</td>
<td>FQ5</td>
<td>282°</td>
<td>192°</td>
<td>(D)</td>
<td>12</td>
</tr>
</tbody>
</table>

The polarimetric channels (HH, HV, VH, and VV) of each RADARSAT-2 scene were initially in the form of un-calibrated GeoTIFF files with an associated metadata file. Each image was imported into PCI Geomatica using no radiometric calibration type and also using a sigma nought calibration type. The DEMs needed to be made from images without calibration, while sigma nought calibration allowed features such as roads to be better distinguished and therefore GCPs and tie points to be more easily placed.

Since Toutin et al. (2010) found that DEMs generated from the total power of the imagery (i.e. the SPAN method, as described by Lee and Pottier, 2009) were more accurate than those
generated from the HH channel alone, the SPAN channels were created and used for DEM extraction in this study. The SPAN corresponds to the sum of the intensities for all four polarimetric channels at each cell location (Toutin et al., 2010). The Total Power tool in Geomatica was used on the un-calibrated polarimetric channels to create the SPAN channel for each scene.

3.3.2 GCP locations and elevations

Road intersection centres were used for GCPs since they could consistently be identified in the RADARSAT-2 images. The GCP locations (x, y) were derived from 2010 SWOOP ortho-images that had a resolution of 20 cm and a horizontal accuracy of 50 cm (Ontario Ministry of Natural Resources, 2010). The elevation of each GCP was interpolated from nearby GPS points collected in a real time kinematic (RTK) survey of the road surface with points collected at least every 50 m.

If the intersection centre was within 10 metres of a GPS point, the nearest GPS point elevation was used directly for the GCP elevation. The distance limit for nearest point interpolation was chosen because the road intersections in the study area are relatively flat and the elevation is therefore not expected to greatly change within 10 metres from the centre of the intersection. All GCP points met this condition, except for two – these had GPS points along a straight section of road to either side, so the elevation was interpolated from the nearest two GPS points using linear interpolation.

Horizontal and vertical coordinates for a total of 49 GCPs were calculated within the data collection area (see Figure 3-3 for a map of GCP locations.). The distribution of GCP locations was moderately patchy due to the location of paved intersections that were consistently visible in
the RADARSAT-2 imagery. The highest density of GCPs in the study area was in the Central to Southeast region.

### 3.3.3 DEM extraction

DEM extraction was performed with PCI OrthoEngine (v. 10.3) software. The general processing steps involved in using this software to generate DEMs is well described by Ostrowski and Cheng (2000).

Toutin’s 3-D Radargrammetric Model (Toutin and Chénier, 2009) was used for DEM generation, based on an EGM08 geoid file. GCPs were placed manually on displayed HV, HH, VV colour composites of the RADARSAT-2 scenes, with residuals of less than 1 m. Tie points with residuals of less than 2 m were collected between image pairs, at road intersections that were not used for GCPs.

Epipolar images were created from the SPAN channels of the image pair. Epipolar images are created by resampling the imagery to a common grid so that the images are aligned, and the Y-parallax is reduced to a value close to the error of the GCPs and tie points. The epipolar images were used as the inputs for the Automatic DEM Extraction tool. The software of the DEM extraction tool segments the images into horizontal strips and then calculates the correlation between pixels of the images within each strip, and an optimal solution is determined within a variably sized moving window. The sensor geometry calculated from the radargrammetric model is used along with the pixel correlation to compute elevations from the parallax for each grid cell location. The ‘fill holes and filter’ option was chosen, which computes the parallax of failed pixels based on a suitable number of successful neighbouring parallax
values, and filters the DEM for noise. The extracted epipolar DEM was then smoothed by the software and geocoded to a 10 m resolution, regularly spaced grid.

### 3.4 DEM Fusion Algorithm Rationale and Implementation

The DEM fusion algorithm was implemented on the 12 generated, overlapping DEMs. An overview of the algorithm is provided in Section 3.1 and a flowchart outlining the major steps in the DEM fusion algorithm is provided in Figure 3-1. This section of the methods chapter provides the rationale for each algorithm step and the details of how the algorithm steps were implemented for the study data.

#### 3.4.1 Data preparation

The first step in the DEM fusion algorithm is to resample all input DEMs to the same grid spacing and cell locations so that the data is aligned. This is referred to as a DEM stack, and it simplifies data processing. As a result, all subsequent DEMs and rasters that are created throughout the processing workflow have the same grid cell resolution and alignment which also facilitates their analysis and comparison at individual cell locations.

All DEMs in this study were generated in reference to the same horizontal datum and projection, and the same vertical datum, but were not initially aligned by grid cell location. The DEMs were therefore resampled to the cell alignment of the reference DEM (described in Section 3.5.1) to allow for easier processing of the DEM fusion steps and also easier corroboration of the DEMs and fusion algorithm products with the reference DEM.
3.4.2 Slope and elevation thresholding

The next steps in the DEM fusion algorithm are to filter the data at each cell location for a certain level of consistency with each other. Slope and elevation thresholding are used to achieve this by retaining associated elevations that are within the set thresholds. The threshold values for slope and elevation are pre-determined by the user as a multiple of the standard deviation from the mean slope or elevation value. In this way the threshold value changes with the distribution of slope values or elevations at each location.

To implement slope thresholding, slope rasters are created for each DEM in the stack resulting in a raster stack of slope values. At each cell location the mean and standard deviation of the slope values are calculated, and any slope values that are outside the threshold number of standard deviations from the mean are flagged. The elevations in the DEM stack that correspond to the flagged values in the slope raster stack are then removed.

In this study, a slope raster for each DEM was created using Horn’s (1981) third-order finite difference method, and the grid cell location numbering scheme suggested by Gallant and Wilson (Gallant and Wilson, 1996). Compared to other methods, the third-order finite difference method has the advantage that local errors in elevation contribute less to errors in slope estimation (Horn, 1981).

Slope thresholding was applied at each cell location on the stack of slope rasters and the corresponding DEMs they were derived from. Slope values were compared at each cell location and any values outside two standard deviations of the mean were flagged. The thresholding value of two standard deviations was chosen after a sensitivity test to ensure that the slope thresholding was not too aggressive (i.e. too many initial elevation estimates were removed). The DEM values
at each cell location associated with flagged slope rasters were then removed. As a result, some DEMs had ‘NoData’ values for some cell locations.

After slope thresholding, the next step in the DEM fusion algorithm is to threshold the elevations at each cell location. When elevation thresholding is performed the mean and standard deviation of the elevations retained after slope thresholding are calculated at each cell location in the DEM stack. The elevations outside a threshold number of standard deviations of the mean are removed from the DEM stack.

In this study, elevation thresholding was applied to the DEM stack resultant from the slope thresholding step. At each cell location, any elevations that were outside of two standard deviations of the mean were removed. After testing multiple values this threshold value was chosen since it achieved a level of filtering that was not too aggressive (i.e. it did not remove too many of the elevation estimates). The elevations that were retained were used in the $k$-means clustering step.

### 3.4.3 $k$-means clustering

In the DEM fusion algorithm a modified $k$-means clustering (similar to that described by Mather and Koch, 2011) is performed on the values at each cell location in the DEM stack remaining after the thresholding steps. The clustering step is included in the algorithm to isolate groups or clusters of elevation that are similar to each other. Given the large number of DEMs in the stack, the assumption is made that the more clustered the elevations are, the more accurate they are.

The general $k$-means clustering algorithm is described well by MacQueen (1967). Though many modifications can be made, the basis of this form of clustering is that initially there are $k$
groups with known cluster centres. Points in the data set are added to these groups based on proximity to group centres. In this way, groups become clusters with points as members. Cluster centres are re-calculated as the average value of the members, and members are then re-assigned to clusters based on their proximity to the new centres (Jain, et al. 1999). The clustering is performed until there is no change in the location of the cluster centres (Mather and Koch, 2011). The ISODATA method was introduced to allow clusters to be merged based on proximity of the centres, or for a cluster to split if the variance of a cluster was above a threshold. Also, clusters with few or no members can be removed. This allows the number of clusters to change with the pattern in the data and made the clustering results less dependent on the initial number of clusters.

The modified k-means algorithm proposed by Mather initializes clustering with a relatively large number of cluster centres compared to the anticipated number of clusters and only allows for cluster merging and removal in the case of low membership, but does not allow cluster splitting. This further reduces the dependency of the clustering results on the initial number of cluster centres as well as their placement. Also, removal of the cluster splitting from the algorithm allows the clustering solution to eventually reach a state of equilibrium (i.e. no change in cluster membership) (Mather and Koch, 2011).

In the proposed DEM fusion algorithm, clustering only occurs if there are more than a user-defined minimum number of elevations at each cell location after the thresholding steps, otherwise the average of the elevations is used instead of a clustering result. The cluster merging distance is set by the user as a percent of the range of elevations to be clustered. The stopping condition of the clustering is met when no change occurs in cluster membership, or the user-defined maximum number of iterations has been performed. The final cluster is selected as the
cluster with the highest number of members; if there are two or more clusters that meet this criterion the cluster with the smallest range in elevation is chosen. The average of the final cluster at each cell location is output into a product DEM.

In this study, \( k \)-means clustering was performed at each cell location, using the DEM stack resulting from the slope and elevation thresholding steps. Clustering was performed if there were more than three elevation estimates at a given cell location, otherwise the average of the elevations was calculated and used in the algorithm product. The initial number of clusters was calculated as five times the number of elevations at each location. The centres of these clusters were determined by placing them evenly within the range of elevations at the location. The relatively large number of clusters and their equal placement throughout the DEM range was chosen to provide a dense yet unbiased initial cluster set so that the first round of clustering would yield cluster centres similar to the distribution of the initial elevations.

At each cell location, elevations were then assigned to the nearest cluster centre. Cluster centres were then re-calculated as the average value of the members within each cluster, while clusters with no members were removed. The distance between cluster centres was calculated and the centres nearest to each other identified. If these nearest centres were within the merge distance (calculated as 10 percent of the total range of elevations at that cell location) the clusters were merged and the new cluster centre was calculated as the average of the merged members. Several merge distances were tested and setting the distance equal to 10 percent of the elevation range at each cell location achieved the desired degree of clustering.

The remaining cluster centres were then used in the next iteration and elevations were re-assigned as members of the nearest cluster. The stopping condition for this loop in the algorithm
was a zero percent change in cluster membership or if 50 iterations had been performed. The maximum number of iterations for any cell location in the study area was 11, so the zero percent change condition was always reached. Once the \( k \)-means clustering was complete, the cluster with the largest number of members was identified. If more than one cluster met this criterion, the cluster with the narrowest range in elevations was chosen as the final cluster. The average of the elevations belonging to the final cluster was calculated and assigned as the elevation at that cell location in the resultant fusion product.

3.4.4 Filtering and smoothing

Previous steps in the proposed DEM fusion algorithm are performed on a cell-by-cell basis and, except for the slope thresholding step, the values of neighbouring cells at each cell location are not taken into consideration. The result is an increase in the spatial variance of the fusion product, compared to the input DEMs, after the \( k \)-means clustering step. To create a product with less short-scale variance adaptive mean and Gaussian filters were used to smooth the DEM.

Adaptive mean filters employ a moving window approach that calculates the mean and standard deviation of neighbouring cell values and compares that to the center value. The user is able to specify the window size and the threshold value as a multiple of the standard deviation of values in the window. In this algorithm an adaptive mean filter is applied to the product DEM to remove local outliers (minima or maxima outside the user-specified standard deviation from the mean) in the data. These local spikes or pits in the data are best removed by an adaptive mean filter first so that they do not influence the results of the Gaussian filter.

In this study, several combinations of window sizes and threshold values for the adaptive mean filter were applied to the data and the results were visually inspected. A nine-by-nine
window with an absolute difference threshold of one standard deviation was found to be the most effective for removing local spikes and pits in the data.

The final step in the proposed algorithm is a Gaussian filter applied to the fusion product to smooth the DEM and reduce much of the short-scale variance that can result from the \( k \)-means clustering. A Gaussian filter is a standard technique used for reducing high-frequency noise in elevation data (Milledge, et al. 2009b; Walker and Willgoose 1999). A Gaussian filter can smooth a DEM by a moving window that weights elevations in the neighbourhood based on a Gaussian distribution, with values closer to the centre weighted more strongly. For this algorithm the user defines the standard deviation distance of the Gaussian function as a number of DEM cells. Increasing the standard deviation distance will increase the degree of smoothing in the final product. It should be noted that the Gaussian filter is best suited for gently sloping terrain, and is therefore not recommended for areas with steep cliffs.

In this study, trial and visual inspection were also used to test the settings for the Gaussian filter, which was applied to the fusion product after the adaptive mean filter. A standard deviation distance of 8 grid cells was determined as an appropriate setting to reduce short-scale variance in the data, and yet not result in an overly-smoothed final fusion product.

### 3.5 Corroboration of DEMs and Fusion Algorithm Products

The accuracy of the RADARSAT-2 DEMs and the fusion algorithm products (created at various stages in the fusion procedure) were assessed by comparing these data with a more accurate reference DEM. Statistics for comparisons with the reference DEM were computed for the whole DEM as well as areas classified by landuse.
3.5.1 Reference DEM and DEMs of difference

To corroborate the RADARSAT-2 DEMs and fusion algorithm products another DEM interpolated from higher accuracy data was used as a reference DEM (rDEM). These data were mass points derived from SWOOP 2010 ortho-imagery. The mass points were created at a regularly spaced 10 m interval with an accuracy of 50 cm both horizontally and vertically. Points that were classified as trees or buildings were removed from the dataset and the resultant data gaps were filled with interpolated points (Ontario Ministry of Natural Resources, 2010).

The mass points were interpolated into a DEM using a spline with tension algorithm. The interpolation was performed in overlapping tiles that were later mosaicked. The resultant rDEM possessed a 10 m resolution. Since the rDEM vertical datum was originally CGG2000 (Canadian Gravimetric Geoid Model of 2000) it was transformed to EGM 2008 using a series of grid shifts. The transformation was performed to compare the rDEM with the RADARSAT-2 DEMs and the fusion algorithm products. Figure 3-4 contains the rDEM, clipped to the study area. The rDEM is void of trees and buildings, though it should be noted that several aggregate and limestone quarries are the cause of prominent, rectangular depressions in the study area.

Since all DEMs (and therefore algorithm products) were resampled to the rDEM, the values of the rDEM were simply subtracted from all DEMs and products of the fusion algorithm, at each cell location. The resultant rasters are referred to as DEMs of difference. Global statistics were then performed on each DEM of difference (i.e. the mean, minimum, maximum, and standard deviation as well as the 10th, 25th, 50th (median), 75th, and 90th percentiles) for the overall study area, and also by areas classified as certain types of landuse. The global mean and median offsets were used to assess DEM and fusion accuracy, whilst the standard deviation and
the spread of the percentile ranges were used to assess DEM and fusion precision. The DEMs of difference were also used for further statistical analyses of the fusion algorithm products.

Figure 3-4 The reference DEM (rDEM) created from SWOOP 2010 ortho-imagery at 10 m grid spacing, and clipped to the study area extent.
3.5.2 Classification by landuse

To gain a better sense of how accurate the extracted DEMs were and how well the fusion algorithm performed on different land cover types (i.e. trees, crops, buildings, etc.) the DEMs of difference were segmented by landuse class (i.e. forest, fields, urban, etc.) within the study area. The Ontario Agricultural Resource Inventory (AgRI) dataset is a polygon coverage representing landuse classes and was utilized in this study. The polygons were digitized from 2006 and 2010 SWOOP imagery, with complete coverage of the study area. This data (version 15) was provided by Dr. Stewart J. Sweeney, of the Ontario Ministry of Agriculture and Food.

For this study, the AgRI dataset was checked with SWOOP 2010 imagery, and polygon editing took place to account for any landuse changes between the time of original digitization (potentially on SWOOP 2006 imagery) and the time of the RADARSAT-2 imagery acquisition in 2010. The main changes made to the AgRI polygons were modifying field or forest polygons where new buildings had been constructed between 2006 and 2010. Another significant change was the removal of two golf courses and several single, isolated rural homes from the urban landuse class. These features were assigned to other landuse classes that were not used in this study.

The edited AgRI polygon shapefile was processed to create raster masks of the landuse classes chosen for this study: Fields, Roads, Rough Land, Urban, Water, and Forest. Other classes available but not chosen were: Farmstead, Fencerow, Railway, Riparian and Quarry. The polygon coverage was clipped to the study area, and polygons were buffered inward by 20 m for all chosen landuse classes, except for water and roads. The water and road features were so narrow that buffering inward would result in almost no remaining area representing those features. Buffering was performed at a distance of 2 grid cells (20 m) to avoid mixed pixel
effects from the RADARSAT-2 imagery, and to avoid possible error in the AgRI polygon locations. A separate binary raster mask for each landuse class was created from the associated buffered polygons with a cell size of 10 m and aligned to the rDEM (see Figure 3-5).

Each landuse mask was used as a binary grid to select only regions identified to be of that class from each DEM of difference. The grid cell count and percent of the study area that each landuse mask covered is provided in Table 3-2. Global statistics for the landuse specific DEMs of difference were then calculated (i.e. the mean, minimum, maximum, and standard deviation, as well as the 10th, 25th, 50th (median), 75th, and 90th percentiles). The global mean and median offsets were used to assess DEM and fusion accuracy, whilst the standard deviation and the spread of the percentile ranges were used to assess DEM and fusion precision, for each landuse class.
Figure 3-5  Map of the landuse masks created for the study area.
Table 3-2 Total area for each landuse class in the study area by cell count, and as a percentage of the study area. Note: the landuse class percentages are less in this table than reported for the study area since many of the landuse polygons were buffered inward and there can also be a change of area when polygons are converted to rasters. As well, the total of the landuse classes do not sum to the overall area because of the inward buffering, and the fact that there were other landuse classes in the study area (i.e. the overall area) that were not analysed individually.

<table>
<thead>
<tr>
<th>Landuse Class</th>
<th>Cell Count</th>
<th>Percent of Study Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fields</td>
<td>1310668</td>
<td>41.48</td>
</tr>
<tr>
<td>Forest</td>
<td>277922</td>
<td>8.80</td>
</tr>
<tr>
<td>Roads</td>
<td>29408</td>
<td>0.93</td>
</tr>
<tr>
<td>Rough Land</td>
<td>113017</td>
<td>3.58</td>
</tr>
<tr>
<td>Urban</td>
<td>119308</td>
<td>3.78</td>
</tr>
<tr>
<td>Water</td>
<td>26367</td>
<td>0.83</td>
</tr>
<tr>
<td>Overall</td>
<td>3159814</td>
<td>100.00</td>
</tr>
</tbody>
</table>
4 RESULTS

4.1 DEM Extraction

4.1.1 Visual assessment of generated DEMs

The twelve RADARSAT-2 DEMs that were extracted using stereo-radargrammetric methods are shown in Appendix A (see Figures A-1 to A-12). A visual assessment of the DEMs was performed and several general observations were made. Mainly, there existed varying degrees of obvious errors, with some DEMs containing more systematic errors, large blunders, or regional offsets than others.

The first noticeable difference between DEMs is their range in elevation. DEM 02 has the largest range of 248.7 m, and DEM 12 has the smallest range of 152.8 m. The second difference noticed is the repeating patterns of elevation, with a high degree of short-scale variation, often causing a diagonal striping (or diamond shape) effect approximately every 300 m to 1000 m in the DEMs. In DEMs 01-03 these patterns were the most obvious, with those of DEM 02 being the most pronounced.

There were also large blunders in the DEMs in the form of erroneous hills and depressions (e.g. an elevation change of 50 or 100 m in a distance of 100 m). When all DEMs were compared, the DEMs with the largest number of blunders, and to the worst degree, were DEMs 04 and 02. When the areas affected by these large blunders were examined in 2010 SWOOP imagery or the landuse polygons provided by the AgRI dataset, the blunders were found to occur mainly in agricultural fields and in some cases where fields bordered forest patches or roads.
Valley delineation for the Grand River and its subsidiaries also varied between DEMs. DEM 10 appeared to have the most complete channel inclusion and continuity, whereas DEMs 04 and 06 were the most incomplete.

4.1.2 Correlation between DEMs

Since the DEMs appeared to be quite different from each other, the Pearson’s product-moment correlation coefficient ($r$) was calculated for all possible pairs of DEMs, overall and by landuse class. The resultant correlation matrices are provided in Appendix B. For the overall $r$ values (see Table B-1) the lowest correlation was between DEM 02 and all other DEMs, with $r$ values between 0.49 and 0.78. Pairings involving DEMs 03 and 04 had $r$ values between 0.85 and 0.91, while all other pairings (except for those involving DEM 02) had $r$ values above 0.91. The highest correlation was between DEMs 08 and 12, with an associated $r$ value of 0.98.

The landuse-specific correlation coefficients were calculated and are provided in the matrices of Tables B-2 to B-7. As well, the difference between each landuse-specific $r$ value and the overall $r$ value for each respective DEM pair were calculated and are also given in the matrices. A positive difference indicates that there was a higher correlation between the DEMs within the given landuse class compared to correlation of the overall study area.

For the rough land and urban landuse classes the $r$ values were higher than the overall $r$ values for every DEM pairing; with differences of up to 0.12 for rough land, and 0.18 for urban. The greatest change for urban and rough land $r$ values were for pairings with DEMs 02 and 04.

Correlation coefficients between DEMs for the water and road landuse classes were slightly higher than the overall correlation coefficients, for most of the DEM pairings. The greatest difference between $r$ values for these landuse classes and the overall $r$ values was for pairings
with DEM 04 (ranging from 0.05 to 0.09 higher than the overall $r$ values). The exceptions were for most pairings with DEMs 01, 02 and 03, where the $r$ values for the water and road classes were slightly lower than the overall $r$ value.

For the field landuse class, all DEM pairings were less correlated than the overall pairings, except for a few pairings with DEM 03. The pairing between DEMs 02 and 04 had the most negative difference between the $r$ value for fields and that of the overall study area, with a difference of -0.06.

For the forest landuse class there were an approximately equal number of cases where the $r$ value was higher or lower than the overall $r$ value. The negative differences were however, of a greater magnitude than the positive differences. For example, the difference in forest and overall $r$ values for pairings between DEM 02 and DEMs 05-12 ranged from -0.08 to -0.12. In contrast, the highest positive difference was 0.03 for the pairing of DEMs 03 and 04.

### 4.2 Fusion Algorithm Steps and Products

In this section the main steps of the fusion algorithm are assessed by a comparison of the DEM retention percentages (for thresholding and clustering steps) as well as a visual assessment of the algorithm products. The algorithm products were created to help understand the contribution of each step in the fusion algorithm to the final fused DEM. A brief description of each product is provided in Table 4-1, and a summary below.
Table 4-1  Summary of fusion algorithm product names and descriptions.

<table>
<thead>
<tr>
<th>Product Name</th>
<th>Brief Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product 01</td>
<td>Average of elevations at each DEM cell location before any fusion algorithm steps</td>
</tr>
<tr>
<td>Product 02</td>
<td>Average of elevations at each DEM cell location after slope and elevation thresholding (the inputs for k-means clustering)</td>
</tr>
<tr>
<td>Product 03</td>
<td>Average of elevations that are members of the final k-means cluster at each DEM cell location</td>
</tr>
<tr>
<td>Product 04</td>
<td>Resultant DEM after adaptive mean filtering is performed on Prod. 03</td>
</tr>
<tr>
<td>Product 05</td>
<td>The final algorithm product. The resultant DEM after Gaussian smoothing is performed on Prod. 04</td>
</tr>
</tbody>
</table>

Product 01 is the average at each cell location of all aligned DEMs before any other algorithm processing. It is used as the simplest form of fusion for a comparison to the contribution of other algorithm steps. Product 02 is the average of all elevations, at each cell location, after slope and elevation thresholding only. Product 03 is the average of the final cluster members, at each cell location, after k-means clustering is performed on the elevations that are retained after slope and elevation thresholding. Product 04 is the result of applying the adaptive mean filter to Product 03. Product 05 is the result of applying the Gaussian filter to Product 04. The fusion algorithm products are shown in Figures C-1 to C-5 in Appendix C.

4.2.1  Slope and elevation thresholding assessment

The purpose of the slope thresholding step in the fusion algorithm was to remove elevations that were associated with slopes in the DEM that were significantly different than other slopes at that location. As a result of slope thresholding, 77% of cell locations had an elevation from one DEM removed, and 2% of cell locations had elevations from two DEMs removed. At the
remaining locations no elevations were removed. The number of elevations removed at each cell location did not have a particular pattern across the study area.

Table 4-2 contains the percentage of cell locations for each DEM where elevations were retained after slope thresholding, for the overall study area and for the field landuse class. The overall and field DEM retention percentages were similar except for DEM 02 (76% overall and 80% in fields) and DEM 04 (69% overall and 64% in fields). All other DEMs retained values at 93% or more of the overall and field cell locations. DEM 12 had the highest retention, at 99.3%, followed by DEMs 08 (98.7%) and 09 (98.4%).

Table 4-2 DEM retention after slope thresholding, elevation thresholding, and k-means clustering, as a percentage of the total available cell locations, for the overall study area and for the field landuse class.

<table>
<thead>
<tr>
<th>DEM</th>
<th>Overall</th>
<th>Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>98.72, 97.82, 35.77</td>
<td>98.99, 98.13, 35.54</td>
</tr>
<tr>
<td>02</td>
<td>76.26, 38.54, 7.75</td>
<td>80.18, 42.40, 8.41</td>
</tr>
<tr>
<td>03</td>
<td>95.01, 91.26, 21.34</td>
<td>95.83, 92.27, 23.04</td>
</tr>
<tr>
<td>04</td>
<td>69.24, 57.99, 12.98</td>
<td>64.34, 52.65, 12.45</td>
</tr>
<tr>
<td>05</td>
<td>96.49, 90.74, 28.82</td>
<td>96.28, 91.92, 29.73</td>
</tr>
<tr>
<td>06</td>
<td>93.86, 92.52, 32.32</td>
<td>92.99, 91.40, 31.35</td>
</tr>
<tr>
<td>07</td>
<td>97.45, 96.01, 34.94</td>
<td>96.97, 95.20, 33.21</td>
</tr>
<tr>
<td>08</td>
<td>98.37, 97.67, 36.25</td>
<td>98.28, 97.36, 35.91</td>
</tr>
<tr>
<td>09</td>
<td>98.69, 98.47, 39.97</td>
<td>98.76, 98.54, 40.16</td>
</tr>
<tr>
<td>10</td>
<td>98.24, 96.09, 31.56</td>
<td>98.37, 95.75, 31.55</td>
</tr>
<tr>
<td>11</td>
<td>98.05, 97.01, 34.38</td>
<td>98.06, 96.87, 34.28</td>
</tr>
<tr>
<td>12</td>
<td>99.30, 98.99, 44.79</td>
<td>99.30, 98.99, 44.79</td>
</tr>
</tbody>
</table>

Elevation thresholding was used as the next step in the fusion algorithm to remove elevations that were outside two standard deviations of the mean elevation at each cell location. As a result of elevation thresholding, 64% of cell locations overall had an elevation from one DEM.
removed, and 1% of cell locations had elevations from two DEMs removed. At the remaining locations no elevations were removed by the elevation thresholding.

Overall retention percentages for each DEM after elevation thresholding (see Table 4-2) were slightly less than the percentages after the previous algorithm step (slope thresholding), except for DEMs 02 and 04. The overall percentage for DEM 02 changed from 76 to 39%, and for DEM 04 from 69 to 58%. All other DEMs had at least 91% retention overall after elevation thresholding. The DEMs with the highest retention were again DEMs 12, 08, and 09. The retention percentages for the field landuse class were also similar to those for the overall study area, except for DEM 02 (42% for fields) and DEM 04 (53% for fields).

The combined effects of the two thresholding steps could be seen in a visual assessment and comparison of Products 01 and 02 (Figures C-1 and C-2, respectively). Both Product 01 and Product 02 did not possess the systematic errors or large blunders that were present in the input DEMs. Visually, valley inclusion and continuity was also much improved in Products 01 and 02. Similar to the input DEMs, there were erroneous hills and depressions in Products 01 and 02, however, these were on a much smaller scale and were less extreme than several of the input DEMs. In Product 02 the erroneous hills and depressions are less pronounced than in Product 01, though there is more short-scale variance present in some areas of Product 02.

The effects of the thresholding steps could also be seen in a profile of a subset of the study area (see Figure 4-1). The slopes or elevations that were significantly different from the other values at each position in the profile were removed. The averages of the values before thresholding (Product 01), and after (Product 02), are shown along with the elevation of the
rDEM for comparison. For each position in the profile the elevation of Product 02 was closer than that of Product 01 to the rDEM.

4.2.2 k-means clustering assessment

k-means clustering was performed on the elevations remaining after slope and elevation thresholding. Clustering occurred at each cell location within multiple iterations until either there was a zero percent change in cluster membership, or until 50 iterations were performed. The minimum number of iterations was 3 and the maximum was 11 (only occurring at five cell locations), so the zero percent change stopping condition was always met. The average number of iterations was 5, which occurred at 38% of the cell locations in the overall study area.
The clustering algorithm then selected the cluster with the most members to be the final cluster. If more than one cluster met this criterion, the cluster with the smallest range was chosen. The percentage of cell locations where there was only one possible final cluster was 72%, while 21% of the cell locations had two possible final clusters. Once the final cluster was chosen the average of the members was calculated for each cell location, resulting in Product 03 (see Figure C-3).

The mean number of members in the final clusters was 3, which occurred at 46% of cell locations, while final clusters with 4 members occurred at 34% of the cell locations overall. The three DEMs that were most often in final clusters were DEMs 08, 09, and 12, both for the overall study area and in fields. The DEM retention percentages after k-means clustering, for the overall study area and for fields, are provided in Table 4-2. The overall final cluster membership percentage was highest for DEM 12 (46%) and DEM 09 (41%), and lowest for DEM 04 (13%) and DEM 02 (8%). The final cluster membership percentages for fields were within one percent of the associated overall values, for all DEMs. As well, it is interesting to note that of the available values after the thresholding steps in fields, DEMs 02 and 04 had only 20% and 24% of the elevations (respectively) that were in a k-means final cluster; the same figure for DEM 12 was 45%.

Some of the effects of the k-means clustering could be seen when a visual assessment of Product 03 was performed. Compared to Product 02, there was much more short-scale noise present in Product 03. The noise was most prevalent in the Southwest region of the study area. Though there were not as many erroneous hills or depressions in Product 03 (which may be an improvement over Product 02), the valleys in Product 03 were not as continuous as they were in Product 02.
The effects of the \(k\)-means clustering step could also be seen in the same profile area shown previously, except with elevation data from before and after \(k\)-means clustering (see Figure 4-2). Especially when compared to the available elevations shown in Figure 4-1, it was apparent that at each location the elevations most clustered were flagged as final cluster members. The averages of the elevations before thresholding (Product 01), after thresholding (Product 02), and after clustering (Product 03) are shown in the profile along with the elevation of the rDEM for comparison. For each position in the profile the elevation of Product 03 was closer than that of Products 02 and 01 to the rDEM.
Figure 4-2  Profile of input DEM elevations before $k$-means clustering and the members of the final cluster after the $k$-means clustering. The average elevation at each position before thresholding (Product 01), after thresholding (Product 02), and after clustering (Product 03) are shown along with the elevation of the reference DEM (rDEM). The profile is the same position as the one shown in Figure 4-1.

4.2.3  Adaptive mean, and Gaussian filtering assessment

Adaptive mean filtering was applied to remove local outliers in the data (spikes and pits) after the $k$-means clustering step. The effect of this step on the fusion algorithm product (Product 04) was not easily apparent when viewed at the study area scale (see Figure C-4 in Appendix C), but when viewed at a larger scale removal of the spikes and pits were more obvious. When looking at Product 04 at the study area scale, however, some erroneous high areas and low depressions in the data were noticeably enlarged, in comparison to Product 03. This effect was especially apparent in the Southwest of the study area.
Gaussian filtering was applied as a final step to smooth the fusion algorithm product by reducing short-scale variance in the data (Product 05). The effect of the smoothing was apparent when viewing Product 05 at the study area scale (see Figure 4-3), as well as at a larger scale. Because the short-scale variation was decreased the patterns in elevations were more continuous and smooth than in Products 03 and 04. Product 05 also appeared to be smoother than Product 01, and though it contained several erroneous hills and depressions they are at a much smaller and finer scale than Product 01. It was also noted that in Product 05 the river valleys were not as continuous or well included as in Products 01 and 02.
4.3 Corroboration of Input DEMs and Fusion Algorithm Products

To measure how accurate and precise the input DEMs and fusion algorithm products were with reference to the higher accuracy DEM (rDEM), DEMs of difference from the rDEM were created (see Figures D-1 to D-17 in Appendix D). The global mean and median offsets were used
to assess DEM and fusion accuracy, whilst the standard deviation and the spread of the percentile ranges were used to assess DEM and fusion precision. In each DEM of difference, positive values correspond to locations where the DEM has a higher elevation than the rDEM, and negative values correspond to locations where the DEM has a lower elevation than the rDEM. In this section results of the visual assessments and the global measures of accuracy and precision of the DEMs of difference are presented.

4.3.1 Visual assessments of the DEMs of difference

A visual inspection of the DEMs of difference for the 12 original DEMs (Figures D-1 to D-12) revealed that the assumed to be erroneous elevation features in the DEMs (systematic noise, hill and depression blunders, and a lack of channel inclusion) were indeed errors when compared the rDEM. Most notably, in all DEMs of difference for the original DEMs the river valleys were over estimated to varying degrees, except for DEMs 02 and 04 where they were generally underestimated. The exception was for the Elora Gorge where all DEMs over-estimated the gorge, though this was least pronounced in DEMs 02 and 04. As for the river banks, most were underestimated in the DEMs, especially in DEMs 08 to 12.

When the DEMs of difference for the original DEMs were compared, some regions of poor elevation estimation varied between DEMs. The Northwest region of the study area was most prevalently underestimated in DEMs 01, 02 and 03, while the Southwest region was most prevalently underestimated in DEM 04 and DEMs 08 to 12.

The same general areas of underestimation were manifested differently in the algorithm products. When the DEMs of difference of the algorithm products were visually assessed (Figures D-13 to D-17; Product 05 DEM of difference is also shown in Figure 4-4), the
Northwest region of the study area was generally over estimated in Product 01 (except for the edge of a large channel) and underestimated in Product 05. There was also a noticeable transition for the Southwest region of the study area: it was generally well estimated in Product 01, but gradually became underestimated, for a greater area and to a greater degree, with each step of the algorithm.

Figure 4-4 DEM of difference for Product 05, clipped to the study area.
Another dissimilarity noticed between the DEMs of difference for the products was in the regional elevation estimation of the Central and Southeastern region of the study area. In this area the elevation was generally over estimated in Product 01, but was less so for Product 02, and even less for Product 03. In Product 05 this area had localized portions that were over or underestimated, but they were not as extreme as Product 01 or even Product 02.

The other apparent difference between the DEMs of difference for the algorithm products was the estimation of the river valleys and banks. The Elora Gorge was over estimated in every product, but least so in Product 03. The banks of the main Grand River valley were generally more underestimated in Product 01 than Product 05. The river valleys themselves, however, were more prominently overestimated in Product 05 compared to Product 01.

### 4.3.2 Accuracy and precision of input DEMs and fusion algorithm products

Vertical offsets comprising the DEMs of difference for the input DEMs and fusion algorithm products were summarized by percentiles, minimum and maximum (extremes), mean and standard deviation values. Box plots of the percentiles, extremes, and mean values are provided in Figures E-1 to E-4 in Appendix E. The following section presents results from the DEMs of difference as the mean of offsets (i.e. global accuracy) and standard deviation of offsets (i.e. global precision) with respect to the rDEM (see Table 4-3).

For the overall study area, DEMs 01 to 06, and DEM 08 had the highest mean and standard deviation combinations of all original DEMs. The largest standard deviation was for DEM 02, and secondly for DEM 04. DEMs 10 and 11 had larger standard deviations than many other DEMs, however their mean offsets were quite low in contrast to the other DEMs. It is interesting to note that all of the DEMs had a positive mean offset, except for DEM 10.
Table 4-3  Global means (µ) and standard deviations (σ) of DEMs or fusion algorithm products (Prod.) offsets from the reference DEM value: for the whole study area (overall), and by landuse class. Positive mean values indicate overestimation, and negative mean values indicate underestimation, compared to the reference DEM.

<table>
<thead>
<tr>
<th>µ (σ)</th>
<th>Overall</th>
<th>Fields</th>
<th>Forest</th>
<th>Roads</th>
<th>Rough</th>
<th>Urban</th>
<th>Water</th>
</tr>
</thead>
<tbody>
<tr>
<td>(m)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Land</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEM 01</td>
<td>7.6 (8.8)</td>
<td>7.0 (8.9)</td>
<td>12.0 (8.4)</td>
<td>6.7 (8.7)</td>
<td>6.6 (7.4)</td>
<td>3.0 (6.6)</td>
<td>13.0 (9.7)</td>
</tr>
<tr>
<td>DEM 02</td>
<td>16.2 (32.0)</td>
<td>18.5 (31)</td>
<td>23.5 (36.8)</td>
<td>12.4 (31.9)</td>
<td>8.5 (33.3)</td>
<td>-0.9 (22.7)</td>
<td>-3.9 (27.6)</td>
</tr>
<tr>
<td>DEM 03</td>
<td>6.2 (13.4)</td>
<td>6.5 (13.0)</td>
<td>11.5 (15.0)</td>
<td>4.8 (13.0)</td>
<td>3.6 (13.3)</td>
<td>-1.6 (10.3)</td>
<td>2.1 (13.5)</td>
</tr>
<tr>
<td>DEM 04</td>
<td>3.6 (20.0)</td>
<td>4.1 (21.6)</td>
<td>11.2 (18.1)</td>
<td>1.9 (18.8)</td>
<td>-1.2 (16.9)</td>
<td>-5.1 (8.9)</td>
<td>-10.0 (10.8)</td>
</tr>
<tr>
<td>DEM 05</td>
<td>10.1 (9.0)</td>
<td>9.0 (9.4)</td>
<td>12.7 (8.7)</td>
<td>9.9 (8.4)</td>
<td>11.1 (8.1)</td>
<td>13.4 (6.2)</td>
<td>14.3 (9.0)</td>
</tr>
<tr>
<td>DEM 06</td>
<td>7.8 (9.0)</td>
<td>6.9 (10.0)</td>
<td>11.3 (7.7)</td>
<td>7.3 (8.1)</td>
<td>9.1 (7.5)</td>
<td>7.2 (5.1)</td>
<td>11.0 (6.7)</td>
</tr>
<tr>
<td>DEM 07</td>
<td>1.5 (7.2)</td>
<td>0.6 (7.7)</td>
<td>5.2 (6.1)</td>
<td>0.3 (6.4)</td>
<td>2.9 (7.2)</td>
<td>-1.2 (4.7)</td>
<td>7.5 (6.4)</td>
</tr>
<tr>
<td>DEM 08</td>
<td>7.6 (8.1)</td>
<td>7.3 (8.4)</td>
<td>11.2 (8.0)</td>
<td>6.6 (7.5)</td>
<td>7.5 (6.7)</td>
<td>5.0 (6.3)</td>
<td>6.8 (8.3)</td>
</tr>
<tr>
<td>DEM 09</td>
<td>2.6 (8.8)</td>
<td>2.4 (9.0)</td>
<td>4.5 (8.6)</td>
<td>2.4 (8.6)</td>
<td>3.1 (7.3)</td>
<td>-0.1 (6.6)</td>
<td>0.1 (10.9)</td>
</tr>
<tr>
<td>DEM 10</td>
<td>-2.3 (10.1)</td>
<td>-2.9 (10.5)</td>
<td>0.5 (9.2)</td>
<td>-3.7 (9.9)</td>
<td>-1.0 (8.3)</td>
<td>-0.9 (7.4)</td>
<td>-5.5 (11.3)</td>
</tr>
<tr>
<td>DEM 11</td>
<td>0.5 (9.8)</td>
<td>0.1 (10.3)</td>
<td>2.1 (8.8)</td>
<td>-0.9 (9.5)</td>
<td>2.4 (8.0)</td>
<td>2.4 (6.3)</td>
<td>-0.8 (9.8)</td>
</tr>
<tr>
<td>DEM 12</td>
<td>5.2 (6.9)</td>
<td>4.8 (6.8)</td>
<td>9.9 (5.9)</td>
<td>4.7 (6.2)</td>
<td>5.1 (6.4)</td>
<td>-2.0 (5.0)</td>
<td>7.7 (6.9)</td>
</tr>
<tr>
<td>Prod. 01</td>
<td>5.5 (6.0)</td>
<td>5.4 (5.9)</td>
<td>9.6 (6.0)</td>
<td>4.4 (5.7)</td>
<td>4.8 (5.7)</td>
<td>1.6 (3.5)</td>
<td>3.5 (4.7)</td>
</tr>
<tr>
<td>Prod. 02</td>
<td>4.6 (4.9)</td>
<td>4.2 (4.8)</td>
<td>8.3 (4.4)</td>
<td>3.7 (4.5)</td>
<td>4.4 (4.5)</td>
<td>1.6 (3.1)</td>
<td>4.6 (5.2)</td>
</tr>
<tr>
<td>Prod. 03</td>
<td>4.3 (7.0)</td>
<td>3.8 (7.2)</td>
<td>8.1 (6.5)</td>
<td>3.6 (6.9)</td>
<td>5.0 (6.1)</td>
<td>1.2 (5.1)</td>
<td>6.4 (8.0)</td>
</tr>
<tr>
<td>Prod. 04</td>
<td>4.4 (6.4)</td>
<td>3.8 (6.5)</td>
<td>8.1 (5.8)</td>
<td>3.6 (6.2)</td>
<td>5.0 (5.5)</td>
<td>1.1 (4.7)</td>
<td>6.7 (7.3)</td>
</tr>
<tr>
<td>Prod. 05</td>
<td>4.4 (5.2)</td>
<td>3.8 (5.2)</td>
<td>8.0 (4.5)</td>
<td>3.6 (4.9)</td>
<td>5.2 (4.6)</td>
<td>1.1 (3.7)</td>
<td>7.3 (5.9)</td>
</tr>
</tbody>
</table>

The overall study area contains many landuse types, so it is informative to look at the results of several different landuse types in isolation from others. All of the DEMs had less of a mean offset for the field landuse class when compared to the overall values, except for DEMs 02, 03, 04, and 10. The standard deviation of offsets was greater for fields than the overall study area, for all DEMs except for DEMs 02, 03, and 12. All of the mean values for DEMs in the forest class were higher than the respective values in the field class, and most of the forest class standard deviations were less than those of the field class. In the urban class the mean offsets were lower (more negative) than the respective values in the fields class, except for DEMs 05, 06, and 11. The standard deviation values for all DEMs were the lowest in the urban class. Rough land areas had a higher mean than the respective value for fields, except for DEMs 01.
through 04. The standard deviation values for the rough land class were lower than the forest except for DEMs 06 and 12, and lower than fields except for DEM 02. The roads and water classes did not have a particular pattern of comparison with the other classes with respect to the mean and standard deviations of the offsets for the original DEMs.

The pattern of change in the mean and standard deviation of offsets when the fusion algorithm products were compared was more apparent. When compared to Product 01, the mean and standard deviation of the offsets for Product 02 were less overall and for all landuse classes, except for urban (the mean was the same and the standard deviation was lower) and water (both the mean and standard deviation were higher).

Application of the $k$-means clustering step (Product 03), when compared to Product 02, decreased the mean offset from the rDEM, but increased the standard deviation of the offsets for all classes except for rough land and for water where the means were increased. Filtering for extreme values (Product 04) and smoothing the data (Product 05), when compared to Product 03, resulted in the mean decreasing for all classes except rough land and water (increased slightly), and fields (remained the same). The standard deviation for this same comparison decreased for all classes.

The final fusion algorithm product (Product 05) had a lower standard deviation of offsets than any of the original DEMs, overall and for each landuse class. The Product 05 standard deviations were higher than those of Product 02, overall and for each landuse class. The mean offset of Product 05 was, however, less than those of Product 01 and 02, overall and for all landuse classes, except rough land and water.
4.4 Further analyses of fusion algorithm products

Further analyses were performed to explore possible reasons for the differences in the accuracy and precision results of the fusion algorithm products. Data from the field landuse class was isolated and analysed since fields are areas without prominent off-terrain objects such as trees or buildings. In the case of fields, therefore, the offsets of the fusion products (that are DSMs) with respect to the rDEM (a DTM) are taken to be a better measure of accuracy than in other areas. Fields are also open spaces not as likely to be influenced by adjacent landuse classes, which is more likely an issue with the narrow corridor features such as roads and surface water features.

In this section the results from two analyses of the fusion algorithm products’ accuracy are presented. Firstly, the frequency distributions for the fusion product’s accuracy with respect to the rDEM, and the absolute accuracy of the products, compared to each other at each cell location, are examined. Secondly, the fusion algorithm product accuracy is classified and presented based on: the number of members in each final cluster and the standard deviation of elevations before the clustering step.

4.4.1 Frequency distributions

The frequency distributions (number of cell locations) of offsets from the rDEM, for all of the fusion algorithm products in the field landuse class, are provided in Figure 4-5. The data used to create the plots were binned at an interval of 1 m. All distributions were negatively skewed, with that of Product 01 the most negatively skewed. Product 01 also had the most positive mode at 6 m, while the modes of Product 03 and 04 (both at 4 m) were the least positive. Products 02 and 05 had the same mode at 5 m, though Product 05 had a higher frequency at that offset value.
compared to Product 02. Products 02 and 05 also had a similar amount of spread in their distributions, which was much less than the other products.

![Frequency distribution graph](image)

**Figure 4-5** Frequency distribution of the relative accuracies for fusion algorithm products. The full extent of the positive and negative tails are not shown.

The data was also analysed to see how the absolute accuracy of each product compared to other products, at each cell location. Of particular interest was how much the fusion algorithm steps (Products 02 to 05) made an improvement in the elevation estimate compared to simply taking the average of all DEMs (Product 01). To achieve this, absolute accuracies for each product were calculated at each cell location for the field landuse class. The values for Product 01 were then subtracted from the values of the other products, creating difference rasters. A negative cell value in the difference raster indicates a location where the given product (i.e. Product 01 to 04) was more accurate than Product 01, and a positive value indicates the given product was less accurate than Product 01. Since the stated accuracy of the rDEM was +/- 0.5 m,
values in the difference raster between -0.5 m and 0.5 m indicated the products compared had the same accuracy.

Instead of just summarizing the aforementioned differences, they are presented in more detail in the frequency distributions of Figure 4-6. Most notable of these results was that the primary mode of all distributions was at 0 m, indicating the same accuracy as Product 01. The secondary mode of Product 02 was more negative than the modes of other products, and the distribution had a narrower spread. As well, all of the products had a negatively skewed distribution in this graph.

![Figure 4-6 Frequency distribution of the difference in absolute accuracy, at each field cell location, between Product 01 and the other fusion algorithm products. Positive values indicate that Product 01 is more accurate, and negative values indicate it is less accurate, than the respective other products.](image-url)
4.4.2 Number of final cluster members and standard deviation before clustering

To gain a better understanding of the factors influencing the accuracy of the $k$-means clustering other variables were analysed in relation to the accuracy of Product 03. Data for the field landuse class was again used since there were no off-terrain objects expected to be present in both the rDEM and the 12 input DEMs. The variables analysed were the number of members in the final $k$-means cluster, and the standard deviation of elevations at each cell location before $k$-means clustering. The results of the analysis for these variables are presented in this section.

The $k$-means final cluster accuracy (the offset of Product 03 from the rDEM) was classified by the number of members in the final cluster at each cell location. The absolute accuracy of Product 03 was then plotted as a cumulative frequency distribution (CFD) for each class (see Figure 4-7). The progression of curves in the CFD showed that with an increase in the number of final cluster members the standard deviation and the mean of the absolute offsets for Product 03 decreased. The curves for the 7 and 8-9 cluster member classes were not as smooth as the others since there were fewer observations (cell locations) in those classes.
Figure 4-7 Cumulative frequency distribution of the absolute accuracy of $k$-means final clusters (Product 03), classified by the number of final cluster members.

To see if there was a spatial pattern in the number of members in the final cluster, the number of final cluster members was mapped for each cell location in the field landuse class (see Figure 4-8). There was a slight increase in the occurrence of high cluster membership in the Southeast region of the study area, but generally the spatial patterns of this variable occurred at the field scale.
Figure 4-8 Map of the number of final cluster members at each cell location in the field landuse class.

The $k$-means final cluster accuracy was also classified by the standard deviation of elevations before clustering. The absolute accuracy of Product 03 was then plotted as a cumulative frequency distribution (CFD) for each class (see Figure 4-9). The progression of curves in the CFD showed that with an increase in the standard deviation of elevations before clustering the
standard deviation and the mean of the absolute offsets for Product 03 increased. The curves for the 0-2, 16-18, 18-20, and 20-35 classes of standard deviation before clustering were not as smooth as the others since there were fewer observations (cell locations) in those classes.

![Cumulative frequency distribution of the absolute accuracy of k-means final clusters (Product 03), classified by the standard deviation of elevations before clustering.](image)

The standard deviation of elevations before clustering, at each cell location in the field landuse class, was mapped to see if there was a spatial pattern in the variable (see Figure 4-10). Though variation existed at the field scale, there was a much stronger spatial pattern in the standard deviation at the study area scale. The standard deviation was generally the highest in the Southwest region of the study area, and was also relatively high in the Northwest region. In the Central and Southeast regions, the standard deviation tended to be lower than in the other regions of the study area.
Figure 4-10  Map of the standard deviation of elevations before $k$-means clustering, at each cell location in the field landuse class.
5 DISCUSSION

The discussion of the results for this study is organized into three main sections. The first section discusses the input DEM quality as it relates to the types and scale of errors present, linking visual assessments to result statistics where appropriate. The potential sources of error are also discussed. In the second section, the effectiveness of the DEM fusion algorithm is discussed as a whole, and as a function of the contribution of each major algorithm step. Visual assessments and statistics from the results of the fusion algorithm products are compared to each other and to those of the input DEMs. In the third section, potential future research directions are briefly outlined.

5.1 DEM Quality

When the original DEMs and associated DEMs of difference were visually assessed there were several noticeable patterns in the elevations and offsets from the reference DEM. These patterns and their potential causes are discussed in this section, accordingly: systematic errors; large blunders; regional differences; riverbank underestimation and channel overestimation. Statistical results of the DEM correlation and corroboration are discussed, where appropriate, to support the observed elevation patterns.

5.1.1 Systematic errors

Repeating patterns of elevation, often with a large amount of short-scale variation, were noticed in several DEMs, but especially in DEMs 01, 02, and 03. This diamond effect was likely caused by the methods used to resolve the pixel correlation between the images when the DEMs
were generated. The DEM generation software is generally described by Ostrowski and Cheng, (2000). The epipolar images are segmented into horizontal strips and variably sized windows are used within each strip to search for matching pixels between the images. The disparity between matched pixels is translated into parallax, which is used to generate elevations in the DEMs. Linear interpolation is used on a narrow area of overlap between adjacent strips to blend the values between strips. This relatively simple interpolation method over a small transition area was likely responsible for the systematic short scale variance in the DEMs. DEM 02 was affected the most by this error because of the significant presence of other forms of error (i.e. large blunders).

5.1.2 Large blunders

Large blunders (i.e. grossly erroneous hills and depressions) were present to varying degrees in all DEMs, but the worst cases were in DEMs 04 and 02. These blunders were most likely due to pixel matching error between the images used to create the DEMs. Pixel matching is a critical step in stereoscopy, since the disparity between the pixels is translated to image parallax and elevation at each cell location in the DEMs (Fayard et al., 2007). Pixel matching error is most commonly caused by a change in nature of the RADAR backscatter (and therefore the imagery itself) within the search window (Toutin, 1998), or is due to a lack of texture in the imagery (Paillou and Gelautz, 1999).

A raster of pixel correlation values is output along with each generated DEM, however when these values were compared to the offsets of the DEMs from the reference DEM a strong relationship did not exist. The pixel correlation maps were therefore not found to be a reliable indicator of DEM accuracy. In this discussion, pixel matching error is inferred from the
relationship between large blunders in the DEMs and the likelihood of differences in backscatter response from the temporal change of ground targets in the RADARSAT-2 scenes.

Characteristics of the features that influence the RADAR backscatter are termed target parameters. The blunders in the DEMs were mostly found within the field landuse class, and were least present in the urban class. Of all the landuse classes considered in this study, the fields were expected to have the most variation in target parameters between scene acquisitions. These variable target parameters include, but are not limited to: soil moisture; crop height, density and coverage; the random or orientated roughness of the surface due to either row crops or tilled soil (Dobson and Ulaby, 1986). These target parameters are especially variable in the spring and fall which is when the RADARSAT-2 scenes were acquired.

Fields can often produce a backscatter response that is quite uniform over the area or has a cyclic component to the pattern (due to row crops, or ridges and furrows). In these cases the resultant texture in the imagery can be smooth or repetitive (Dobson and Ulaby, 1986). In either case, pixel matching errors can result from confusion within the imagery texture (Paillou and Gelautz, 1999). Smooth textures for fields coupled with changes in backscatter between scene acquisitions, may explain why most of the blunders in fields occurred where two fields met or where a field boarded another landuse type.

The blunders in the field landuse class were also not consistent between DEMs. This was supported by the Pearson product moment correlation ($r$) values for fields which were lower than the overall $r$ values, for all DEM pairs. As mentioned, the blunders were the worst for DEMs 02 and 04. This fact seems to be related to the timing of the scene acquisitions within the spring and fall seasons, as well as the amount of lag time between scene acquisitions.
DEM 02 was generated from scenes acquired in mid-April and early May when fields were being tilled, or planted, and crops were beginning to emerge in the study area, and therefore target parameters were rapidly changing. DEMs 01 and 03 were generated from imagery with the same system parameters (i.e. incidence angle and look direction) as DEM 02, but were less erroneous, likely because of less lag time (3 days, instead of 21 days for DEM 02) between scene acquisitions. DEM 04 was generated from scenes acquired in late September and early October when fields were being harvested and tilled, and similar to the early spring, the target parameters were expected to be rapidly changing. DEMs 05, 06, and 07 were generated from imagery with the same system parameters as those used to generate DEM 04, however they were acquired later in the fall when the target parameters were likely more stable.

Not surprisingly, the urban areas contained the least number of DEM blunders, and had consistently higher correlation values ($r$) between DEMs than those of the overall study area. The absence of large blunders was also supported by the fact that the lowest standard deviation of offsets within a given DEM was that of the urban landuse class. The urban areas would not be expected to change significantly between scene acquisitions due to a relatively large percentage of the area comprising of buildings and roads; both of which do not vary in height, area, roughness, or position with time. As well, in much of the area between the buildings and roads is mainly covered with manicured grass which does not vary much throughout the spring and fall.

### 5.1.3 Regional offsets

There were also regional patterns of offset from the rDEM that existed in many of the DEMs. These were large areas that were generally underestimated or overestimated in groups of DEMs. Though landuse type certainly played a role, there were other factors in the RADARSAT-2
imagery acquisition and the DEM generation process that might have been responsible for the regional variation.

The DEMs and scenes used to generate them were organized into three groups based on similarities in sensor parameters during the scene acquisition. These groups were: spring ascending (DEM 01-03); fall ascending (DEM 04-07); spring descending (DEM 08-12). By examining the incidence angles and look directions for the RADARSAT-2 scene acquisitions (see Table 3-1), as well as the scene extents (Figure 3-3), the relative image geometries were determined. Relative image geometry is important to consider because there is a balance of radiometric consistency between images taken from similar angles, and enhanced geometric disparity (or parallax) from images taken at dissimilar angles (Toutin, 1999). The look direction and incidence angle are also important because they determine the relative illumination of ground features based on their slope and aspect (Toutin, 2002).

For the spring ascending group, the scenes had the least amount of overlap, the steepest incidence angles (22° and 37°), a relatively large intersection angle (15°), and the look direction of approximately 78° azimuth. The Northwest region of the study area has a general aspect of South to Southeast that is nearly perpendicular to the look direction. It is very likely that this region of the study area was improperly represented in DEMs 01-03 because of a combination of poor surface illumination due to the system look direction, and a relatively large radiometric disparity from the large intersection angle of the scenes.

In contrast, the Northwest region was not poorly represented in the fall ascending, and spring descending DEMs. The fall ascending group of scenes had a similar look angle (80.5°) to the spring ascending group and the incidence angles are shallower (42° and 49°), however the
intersection angle is smaller (7°). The smaller intersection angle between the fall ascending scenes would create less radiometric disparity between the image pairs used to generate the DEMs. The spring descending scenes have a larger intersection angle (16°), and slightly steeper incidence angles (25° and 41°) than the spring ascending scenes, but the look direction is of the other groups (281°) so the Northwest region is likely well illuminated.

The Southwest corner of the study area is underestimated in all of the spring descending DEMs (08 to 12), as well as DEM 04. This region has a many rows of elongated, low hills that have a general aspect of either Northeast or Southwest. The prominent aspect of these terrain features are approximately perpendicular to the look direction of the scenes acquired in the spring descending DEMs, and so they are likely poorly illuminated in those scenes. The region is relatively well represented in the spring and fall ascending groups of DEMs because the look directions favour the illumination of the prominent terrain features. The exception to this is for DEM 04, in which case the underestimation of the region may be due to differences in target parameters causing poor pixel matching, as discussed earlier. This may especially be the case because the Southwest region of the study area is covered by many agricultural fields.

The linear terrain features described for the Southwest region of the study area are also present in much of the Central-east to Southeast region of the study area, but the latter mentioned region is well represented in most DEMs. This is likely due to the distribution of GCPs, which is relatively dense in the Central to Southeast region, and relatively sparse in the other regions of the study area. GCPs help to refine the 3-D radargrammetric model that is used to convert image parallax into elevations, and so a relatively high density of GCPs may have improved elevation estimates in the DEMs.
The regional offset patterns that are noticeable in visual assessments and comparisons of the DEMs and DEMs of difference are also statistically supported. The Southwest and Northwest regions of the study area generally have the highest standard deviation of elevations after most of the large blunders have been removed by slope and elevation thresholding (see Figure 4-10).

5.1.4 Riverbank underestimation and valley over estimation

Many of the riverbanks in the study area are underestimated, and the valleys themselves are overestimated. These effects are both likely a product of DEM interpolation and smoothing. Channel overestimation may also be amplified by the presence of nearby forests.

During the DEM generation process the elevations are interpolated and smoothed. These processes are proprietary to the software and are not well described in the literature (Ostrowski and Cheng, 2000). It is likely that the smoothing algorithm is not optimized to preserve edges, which would be necessary to maintain the steep edges of the main riverbanks. Instead the riverbank edges are smoothed down to gradually match the lower elevations of the valley bottom. If the valley is narrow enough the combined effect of smoothing from both sides would likely result in an overestimation of the valley itself. The Elora Gorge is a good example of this, with overestimation occurring in all DEMs.

A relatively large section of the main Grand River valley is wide in comparison to other portions of the valley. In much of the main valley bottom forest and rough land areas are present. The forest areas have some of the highest mean offsets for a given DEM, compared to the other landuse classes (see Table 4-3). The mean offsets do not represent the true height of the trees since C-band SAR causes an underestimation of tree height, even in ideal conditions, due to the signal penetration being a function of the height, density, and species of the trees (Reinartz, et al.
2005). None of these variables were sampled for the study area, so only approximations can be made, but the presence of trees in the main river valley is likely responsible for much of the overestimation of elevation there.

The water features and roads are similar in that they were both expected to be well estimated in DEMs and to have low offsets from the rDEM. This was due to the consistent backscatter response of these features in all images, and the likelihood of correct pixel matching between image pairs. As demonstrated by the large mean and standard deviation of offsets for the road and water landuse classes in most DEMs (see Table 4-3) these narrow features are more likely to be affected by the elevations of the adjacent landuse classes.

5.2 Effectiveness of the DEM fusion algorithm

Though many of the landuse class results for the fusion products follow the same pattern as the field class, the fields are the only places in the study area that are void of high off-terrain objects such as trees and buildings. The offsets of fusion algorithm products from the rDEM in fields are therefore taken to be true measures of accuracy and precision. This section of the discussion will therefore focus mainly on the results for the field landuse class.

5.2.1 Effectiveness of the fusion algorithm as a whole

The goal of the DEM fusion algorithm was to create a DEM that was more accurate and precise than the input DEMs. The final fusion algorithm product (Product 05) was more precise than each of the input DEMs, but not necessarily more accurate. Since elevations were clustered at each cell location, and an estimate of the accuracy for each input DEM was not considered in
the fusion algorithm, it is understandable that the fusion process created a product that was a compromise of accuracy of the input DEMs, to some degree.

The main benefit of the DEM fusion algorithm was the increased global accuracy compared to most input DEMs, and the precision of the estimates, in the final fusion product. This increased consistency suggested that much of the error and noise had been removed from the data and the accuracy of the product as a whole was therefore less influenced by extreme values, in comparison to the original DEMs. This was also noticeable in the visual assessments of Product 05 and the input DEMs. Product 05 did not contain the systematic errors, large blunders, or same degree of channel discontinuity as the input DEMs.

Since the accuracy and precision of the input DEMs was not factored into the fusion algorithm a more appropriate comparison of the fusion algorithm final product (Product 05) was with a less sophisticated form of DEM fusion: averaging all of the input DEMs at each cell location (Product 01). Product 05 was more accurate and precise than Product 01, as evident when the global mean and standard deviation of offsets from the rDEM were compared for the products.

The statistically increased accuracy of elevation estimates from the DEM fusion, compared to simple averaging was also apparent in the frequency distributions of offsets for Products 01 and 05 (see Figure 4-5). Compared to Product 01, the distribution of Product 05 had a mode that was closer to zero, a higher peak, less spread about the mean, and was less negatively skewed.

The difference in absolute accuracy of Product 05 from that of Product 01, at each cell location, was also graphed as a frequency distribution (Figure 4-6). For this graph, the Product 05 mode was at zero, which indicated that at most of the field cell locations the Product 05
elevation estimates were of similar accuracy as those of Product 01. The skew of the distribution, however, was negative which indicated a greater propensity for Product 05 to be more accurate than Product 01 and to a greater degree.

The proposed DEM fusion algorithm tested in this study was therefore superior statistically to a simple averaging of the elevations. When the products were visually assessed, some patterns arose as to which regions and features in the study area the fusion algorithm product is not successful at representing. The prominent differences between Product 01 and 05 were for regional offsets and channel continuity.

Elevation estimates for the Central and Southeast region of the study area, as well as the Northwest region, were improved in Product 05, compared to Product 01. Conversely, the Southwest region of the study area was more underestimated in Product 05. As well, the river valley bottoms are more over estimated, and the banks are less underestimated in Product 05, compared to Product 01.

These trends are related to the distribution of elevations and errors in each of the regions and features, as is evident from the visual assessment of the input DEMs, their associated DEMs of difference, and the standard deviation of elevations after most blunders are removed (see Figure 4-10). In the affected regions and river valley features, the presence of outliers that skewed the average of the elevations at those locations to be more accurate than the clustered values, benefited the elevation estimates in Product 01, and hindered them in Product 05.

The influence of each step in the algorithm was explored to gain a better understanding as to why the fusion algorithm was successful in some ways, but not in others. Products 02 to 05 were created to represent the effect of each successive step of the fusion algorithm on the input data.
(see Table 4-1 for product descriptions). The contribution of each step in the algorithm is discussed in the following sections.

### 5.2.2 Slope and elevation thresholding assessment

When compared to Product 01, Product 02 had a higher global accuracy and precision. This indicated that the slope and elevation thresholding steps removed many erroneous elevation estimates, which improved the resultant DEM. These statistics were also supported by the characteristics of the frequency distribution of offsets in fields for Product 02 (Figure 4-5). Compared to Product 01, the distribution of Product 02 had a mode that was closer to zero, a higher peak, less spread about the mean, and was less negatively skewed.

The difference in absolute accuracy of Product 02 from that of Product 01, at each cell location in the fields, was also graphed as a frequency distribution (see Figure 4-6). For Product 02 primary mode was at zero, which indicated that at many of the field cell locations the Product 02 elevation estimates had the same accuracy (and were therefore the same elevation) as those of Product 01. The secondary mode is negative, indicating that when the elevation estimate of Product 02 was not the same as Product 01, there was a greater likelihood that Product 02 was more accurate than Product 01. The distribution for Product 02 also had a narrow spread about the mode, compared to other products, signifying that most of the differences in accuracy compared to Product 01 are not large differences. Elevation estimates as a result of slope and elevation thresholding were, therefore, neither improved nor worsened by a large magnitude, compared to those of averaging before thresholding.

The retention of particular DEMs after slope and elevation thresholding was not related as strongly to the accuracy of the DEMs as it was to the precision of the DEMs, though a
combination of accuracy and precision was likely involved. The DEMs with the lowest precision (DEM 02 and 04) were also those with the lowest retention of cell locations after the slope and elevation thresholding. Conversely, the DEM with the highest precision (DEM 12) had the highest retention percentage after slope and elevation thresholding. These results support the idea that similar slopes and elevations in DEMs tend to be more accurate than those that are dissimilar.

When the visual assessments of Product 01 and 02 were compared, the most prominent difference was that Product 02 had fewer blunders than Product 01. As well, the Central to Southwest region, and the Northwest region were better represented as a whole. There was not a noticeable change in the valley bottom inclusion and connectivity of Product 02, compared to Product 01.

5.2.3  k-means clustering assessment

The beneficial effect of adding the k-means clustering step to the processing chain of the fusion algorithm was visible in the profile of Figure 4-2, where clustering was shown to increase the accuracy of the elevation estimates. The fact that the most accurate DEMs were most often in the final cluster, and that the least accurate DEMs were least often in the final cluster were also encouraging results that supported the ideas that accurate elevations were clustered and could be isolated with k-means clustering.

The statistics and resultant graphs conveyed a slightly less optimistic set of results from the k-means clustering step. The global mean and standard deviations of offsets for Product 03 indicated that it was more accurate, but less precise, than Product 02. Supporting these values was the frequency distribution of offsets in fields (see Figure 4-5) for Product 03. The mode of
Product 03 was closer to zero than that of Product 02, however, the spread of values about the mode was much larger. As well, the frequency distribution revealed that the offsets for Product 03 were more negatively skewed than those of Product 02, which would have biased the global mean offset of Product 03.

The frequency distribution of the difference in absolute accuracy of Product 03 from that of Product 01, at each cell location in the fields (see Figure 4-6), also provided information on the effectiveness of the k-means clustering step. When compared to the distribution for Product 02, the distribution of Product 03 also had its primary mode at zero, but the distribution peaked at a lower frequency, meaning fewer cell locations had an accuracy that was the same as that of Product 01. As well, the distribution for Product 03 was more wide-spread about the mode, and was more positively skewed than the distribution for Product 02. These characteristics of the distribution indicated that while there were many locations that Product 03 was more accurate than Product 02, and to a greater degree, there were a greater number of locations where Product 03 was less accurate, and to a greater degree.

These findings did not support the idea that the addition of the clustering step in the algorithm is beneficial. The main issue with the k-means clustering step was that the resultant product was noisy. This noise was very noticeable in the visual assessment of Product 03 as abrupt changes in elevation forming spikes, pits, or small patches of elevation that did not match the surrounding elevations. For this reason, factors influencing the accuracy of the k-means clustering were investigated.
5.2.4 Factors influencing k-means clustering accuracy

Since the clustering was performed without *a priori* information about the potential error in input DEMs, or for given landuse types or features, it seemed probable that the erroneous noise resulting from the clustering was related to the distribution of the elevation estimates before clustering. Two variables were then examined: the number of final cluster members, and the standard deviation of elevations before clustering.

The cumulative frequency distributions (CFDs) in Figure 4-7 showed that the accuracy and precision of Product 03 increased with the number of final cluster members. This result supported the idea that more accurate elevation estimates would be clustered in greater numbers. The chance of error in the elevation estimate resulting from clustering is therefore greater when fewer elevations are isolated.

The number of final cluster members at each cell location in the field landuse class was mapped to see if there was a spatial pattern (Figure 4-8). The pattern was only moderately apparent at a regional scale, since there was a tendency toward greater cluster membership in the Central to Southeastern region of the study area. This pattern was expected, since that region was well represented in all input DEMs.

The pattern of the number of final cluster members appeared to be stronger at the field scale. The pattern within a given field might be indicative of processes at the field scale that are affecting the consistency of elevation estimates. These processes could include agricultural practices or fluctuating soil moisture. This pattern might also indicate terrain features with certain slopes or aspects that are well characterized in some DEMs but not in others. Further research is required to investigate the role of field scale processes in stereo-radargrammetric
DEM accuracy and precision and the related effect on clustering elevations at each cell location within fields.

The other variable investigated was the standard deviation of elevations before clustering. The CFDs of the Product 03 accuracy for various classes of standard deviation (Figure 4-9) showed that an increase in the standard deviation of elevations before clustering resulted in an increase in both the global mean and standard deviation of offsets for Product 03. This result supported the idea that, without \textit{a priori} information on DEM error, the successful fusion of the DEMs is highly dependent on the distribution of the input elevations.

A map of the standard deviation of elevations before clustering was created (Figure 4-10) and visually assessed. Patterns of standard deviation in elevations existed at the field-scale however, the regional pattern in this variable was much more apparent. Larger standard deviations occurred more often in the Southwest and Northwest regions of the study area. Those patterns were not surprising given the mix of poor and good representation of the aforementioned regions in the input DEMs. The visual assessment of the DEM of difference for Product 03 supports the relationship between the standard deviation of elevations and the accuracy of clustering, as well as the regional pattern of the relationship.

5.2.5 \textit{Filtering and smoothing assessment}

Errors due to clustering erroneous elevation estimates were anticipated, and that is why the data was then filtered for noise. It was assumed that many of the errors would be isolated and not reflected in neighbouring cells, so the adaptive mean filter was employed to remove any elevations that were significantly higher or lower than those of neighbouring cells, and replace the elevations with the mean value of the neighbouring cells (Product 04). The filter removed
noise and decreased the standard deviation, while maintaining the same mean and mode of the offsets from the rDEM from Product 03 (see Figure 4-5). These statistics supported the idea that the clustering of significantly erroneous elevations often occurred in isolated locations. After a visual assessment of Product 04 and the associated DEM of difference, the adaptive mean filter was least beneficial for areas where erroneous estimates predominated. This occurred mainly in the Southwest and Northwest regions of the study area and for several portions of the main Grand River valley bottom.

Product 04 still contained a lot of short-scale variation, so a Gaussian filter was included in the algorithm (resulting in Product 05). The Gaussian filter reduced even more noise in the DEM, while maintaining the global accuracy that was established in the k-means clustering step. The distribution of offsets for Product 05 (see Figure 4-5) had the same mode (with a higher frequency) as that of Product 02, though the standard deviation of offsets was slightly larger. The Product 05 distribution was also more negatively skewed than that of Product 02, and therefore the increased accuracy of Product 05 compared to 02 was slightly biased.

The frequency distribution of the difference in absolute accuracy of Product 05 and Product 01 (Figure 4-6) had a wider spread of values about the mode, but was more negatively skewed than the distribution for Product 02. This indicated that, while there were many cell locations in the field landuse class where Product 05 was less accurate than Product 02, there were a greater number that were more accurate, to a greater degree.

The Gaussian filter was very effective in smoothing the data to reduce short-scale variance, which was necessary after the k-means clustering step. As expected, this effect was most beneficial on areas with relatively flat or gently rolling topography, which is a characteristic of a
large portion of the study area. The smoothing affect did, however, cause a greater underestimation of many of the river banks, and a greater overestimation of the river valley bottoms. If landuse masks were used in the fusion algorithm, an edge-preserving method of smoothing would have been a more appropriate choice for the river valley bottoms and banks.

5.3 Future Research Directions

The generated DEMs were erroneous to varying degrees, and landuse type played a major role in the cause of the errors. Better discrimination of landuse classes based on target parameters (e.g. vegetation or building height, coverage, or density) and ancillary data for target parameter change would enable a better understanding of the errors present in stereo-radargrammetric DEMs. This research would be most beneficial if it was focused on agricultural fields since large blunders were mostly found in this landuse class in this study, and very little has been published on this topic.

If the DEM quality and fusion algorithm effectiveness were to be further studied for landuse types other than fields, a higher accuracy DSM (i.e. from LiDAR data) would be useful for quantifying the accuracy of the elevation estimates for areas with off-terrain objects such as trees and buildings. With this better understanding of the nature of elevation estimates for various landuse types, the settings of the DEM fusion algorithm could be altered by landuse, as suggested by Papasaika and Baltsavias (2009).

k-means clustering of elevations proved to be beneficial to the overall accuracy of the fused DEM product when the associated noise was removed by filtering and smoothing. The error in clustering had a strong relationship with the number of cluster members and the standard
deviation of elevations before clustering. It is therefore recommend that further research be conducted to investigate the relationship between elevation clustering error and the distribution of elevations before clustering. We also suggest that a probability of error map based on the distribution of elevations be tested with a spline interpolator to remove error from the elevation estimates produced after $k$-means clustering.
6 CONCLUSIONS

In this study, a new fusion algorithm was developed and tested on twelve overlapping DEMs of a mainly agricultural area. The input DEMs were derived from multiple RADARSAT-2 fine-quad mode images using same-side stereo-radargrammetric methods. The main steps of the algorithm were slope and elevation thresholding followed by $k$-means clustering of elevations at each cell location, and finally, filtering and smoothing the fusion product with moving window filters. Several algorithm products were created to better understand the contribution of each main step in the fusion process. The input DEMs and algorithm products were corroborated with a higher accuracy reference DEM, and the results were presented and discussed.

The main findings of this study were:

- The generated DEMs contained systematic errors, large blunders, and regional offsets that varied according to landuse type, as well as the differences in scene acquisition date and sensor parameters.
- The slope and elevation thresholding steps of the fusion algorithm were effective in removing systematic errors and large blunders from the elevation estimates.
- The $k$-means clustering of the elevations improved the global accuracy of the estimates but reduced the precision. The number of final cluster members and the standard deviation of elevations before clustering both had a strong relationship to the error in the $k$-means estimates.
- The adaptive mean and Gaussian filtering steps were effective in removing many erroneous elevation estimates that resulted from the $k$-means clustering step but the
smoothing effect caused river valley bottoms to be more overestimated, and the bank edges to be more underestimated in the final fusion product.

Further research should be conducted to investigate the relationship between elevation clustering error and the distribution of elevations before clustering. As well, better discrimination of landuse classes, and ancillary data for target parameter change between imagery acquisitions should be incorporated into a more rigorous study of the causes of stereo-radargrammetric DEM error, especially in agricultural fields.


Ontario Ministry of Natural Resources. 2010. SWOOP 2010 Deliverables.


Figure A-1  DEM 01, clipped to the study area. No contrast stretch was applied to symbolize the data.
Figure A-2  DEM 02, clipped to the study area. No contrast stretch was applied to symbolize the data.
Figure A-3  DEM 03, clipped to the study area. No contrast stretch was applied to symbolize the data.
Figure A-4  DEM 04, clipped to the study area. No contrast stretch was applied to symbolize the data.
Figure A-5  DEM 05, clipped to the study area. No contrast stretch was applied to symbolize the data.
Figure A-6  DEM 06, clipped to the study area. No contrast stretch was applied to symbolize the data.
Figure A-7 DEM 07, clipped to the study area. No contrast stretch was applied to symbolize the data.
Figure A-8  DEM 08, clipped to the study area. No contrast stretch was applied to symbolize the data.
Figure A-9 DEM 09, clipped to the study area. No contrast stretch was applied to symbolize the data.
Figure A-10  DEM 10, clipped to the study area. No contrast stretch was applied to symbolize the data.
Figure A-11  DEM 11, clipped to the study area. No contrast stretch was applied to symbolize the data.
Figure A-12 DEM 12, clipped to the study area. No contrast stretch was applied to symbolize the data.
**APPENDIX B**

**CORRELATION MATRICES FOR DEMS BY LANDUSE CLASS**

Table B-1  The overall study area Pearson product moment correlation coefficient ($r$) matrix for DEMs.

<table>
<thead>
<tr>
<th></th>
<th>DEM01</th>
<th>DEM02</th>
<th>DEM03</th>
<th>DEM04</th>
<th>DEM05</th>
<th>DEM06</th>
<th>DEM07</th>
<th>DEM08</th>
<th>DEM09</th>
<th>DEM10</th>
<th>DEM11</th>
<th>DEM12</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEM01</td>
<td></td>
<td>0.6365</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEM02</td>
<td>0.9129</td>
<td></td>
<td>0.7780</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEM03</td>
<td>0.8478</td>
<td>0.6129</td>
<td>0.8600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEM04</td>
<td>0.9367</td>
<td>0.4985</td>
<td>0.8471</td>
<td>0.8525</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEM05</td>
<td>0.9347</td>
<td>0.5752</td>
<td>0.8815</td>
<td>0.8606</td>
<td>0.9486</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEM06</td>
<td>0.9395</td>
<td>0.5446</td>
<td>0.8667</td>
<td>0.8601</td>
<td>0.9623</td>
<td>0.9519</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEM07</td>
<td>0.9213</td>
<td>0.5567</td>
<td>0.8730</td>
<td>0.8855</td>
<td>0.9566</td>
<td>0.9410</td>
<td>0.9537</td>
<td>0.9790</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEM08</td>
<td>0.9123</td>
<td>0.6100</td>
<td>0.8935</td>
<td>0.8851</td>
<td>0.9331</td>
<td>0.9310</td>
<td>0.9366</td>
<td>0.9642</td>
<td>0.9699</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEM09</td>
<td>0.9253</td>
<td>0.5896</td>
<td>0.8839</td>
<td>0.8729</td>
<td>0.9417</td>
<td>0.9355</td>
<td>0.9426</td>
<td>0.9623</td>
<td>0.9654</td>
<td>0.9728</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEM10</td>
<td>0.9351</td>
<td>0.5498</td>
<td>0.8729</td>
<td>0.8859</td>
<td>0.9623</td>
<td>0.9480</td>
<td>0.9611</td>
<td>0.9798</td>
<td>0.9788</td>
<td>0.9588</td>
<td>0.9641</td>
<td></td>
</tr>
</tbody>
</table>

Table B-2  The field landuse class Pearson product moment correlation coefficient ($r$) matrix for DEMs (lower left half of the table). Difference between $r$ values for the overall area and those of the field landuse class are in the upper right half of the table; positive values are those where the $r$ value for the field class is higher than the overall $r$ value, and are only highlighted to help distinguish them from negative values.

<table>
<thead>
<tr>
<th></th>
<th>DEM 01</th>
<th>DEM 02</th>
<th>DEM 03</th>
<th>DEM 04</th>
<th>DEM 05</th>
<th>DEM 06</th>
<th>DEM 07</th>
<th>DEM 08</th>
<th>DEM 09</th>
<th>DEM 10</th>
<th>DEM 11</th>
<th>DEM 12</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEM 01</td>
<td></td>
<td>-0.0201</td>
<td>-0.0079</td>
<td>-0.0236</td>
<td>-0.0049</td>
<td>-0.0153</td>
<td>-0.0082</td>
<td>-0.0059</td>
<td>-0.0053</td>
<td>-0.0124</td>
<td>-0.0090</td>
<td>-0.0037</td>
</tr>
<tr>
<td>DEM 02</td>
<td>0.6164</td>
<td></td>
<td>-0.0387</td>
<td>-0.0622</td>
<td>-0.0076</td>
<td>-0.0129</td>
<td>-0.0058</td>
<td>-0.0110</td>
<td>-0.0082</td>
<td>-0.0110</td>
<td>-0.0086</td>
<td>-0.0109</td>
</tr>
<tr>
<td>DEM 03</td>
<td>0.9050</td>
<td>0.7393</td>
<td></td>
<td>-0.0293</td>
<td>0.0036</td>
<td>-0.0050</td>
<td>-0.0004</td>
<td>0.0010</td>
<td>0.0030</td>
<td>-0.0008</td>
<td>0.0012</td>
<td>0.0019</td>
</tr>
<tr>
<td>DEM 04</td>
<td>0.8242</td>
<td>0.5507</td>
<td>0.8307</td>
<td></td>
<td>-0.0118</td>
<td>-0.0225</td>
<td>-0.0168</td>
<td>-0.0189</td>
<td>-0.0168</td>
<td>-0.0172</td>
<td>-0.0190</td>
<td></td>
</tr>
<tr>
<td>DEM 05</td>
<td>0.9318</td>
<td>0.4909</td>
<td>0.8507</td>
<td>0.8407</td>
<td></td>
<td>-0.0146</td>
<td>-0.0123</td>
<td>-0.0069</td>
<td>-0.0067</td>
<td>-0.0107</td>
<td>-0.0107</td>
<td>-0.0066</td>
</tr>
<tr>
<td>DEM 06</td>
<td>0.9194</td>
<td>0.5623</td>
<td>0.8765</td>
<td>0.8381</td>
<td>0.9340</td>
<td></td>
<td>-0.0148</td>
<td>-0.0137</td>
<td>-0.0131</td>
<td>-0.0176</td>
<td>-0.0166</td>
<td>-0.0115</td>
</tr>
<tr>
<td>DEM 07</td>
<td>0.9313</td>
<td>0.5388</td>
<td>0.8663</td>
<td>0.8433</td>
<td>0.9500</td>
<td>0.9371</td>
<td></td>
<td>-0.0085</td>
<td>-0.0077</td>
<td>-0.0115</td>
<td>-0.0117</td>
<td>-0.0082</td>
</tr>
<tr>
<td>DEM 08</td>
<td>0.9309</td>
<td>0.5465</td>
<td>0.8772</td>
<td>0.8615</td>
<td>0.9533</td>
<td>0.9299</td>
<td>0.9485</td>
<td></td>
<td>-0.0044</td>
<td>-0.0054</td>
<td>-0.0071</td>
<td>-0.0045</td>
</tr>
<tr>
<td>DEM 09</td>
<td>0.9160</td>
<td>0.5485</td>
<td>0.8760</td>
<td>0.8687</td>
<td>0.9499</td>
<td>0.9279</td>
<td>0.9460</td>
<td>0.9746</td>
<td></td>
<td>-0.0047</td>
<td>-0.0062</td>
<td>-0.0040</td>
</tr>
<tr>
<td>DEM 10</td>
<td>0.8999</td>
<td>0.5990</td>
<td>0.8927</td>
<td>0.8662</td>
<td>0.9224</td>
<td>0.9134</td>
<td>0.9251</td>
<td>0.9588</td>
<td>0.9652</td>
<td></td>
<td>-0.0078</td>
<td>-0.0049</td>
</tr>
<tr>
<td>DEM 11</td>
<td>0.9163</td>
<td>0.5810</td>
<td>0.8851</td>
<td>0.8557</td>
<td>0.9310</td>
<td>0.9189</td>
<td>0.9309</td>
<td>0.9552</td>
<td>0.9592</td>
<td>0.9650</td>
<td></td>
<td>-0.0036</td>
</tr>
<tr>
<td>DEM 12</td>
<td>0.9314</td>
<td>0.5389</td>
<td>0.8748</td>
<td>0.8669</td>
<td>0.9557</td>
<td>0.9365</td>
<td>0.9529</td>
<td>0.9753</td>
<td>0.9748</td>
<td>0.9539</td>
<td>0.9605</td>
<td></td>
</tr>
</tbody>
</table>
Table B-3  The forest landuse class Pearson product moment correlation coefficient ($r$) matrix for DEMs (lower left half of the table). Difference between $r$ values for the overall area and those of the forest landuse class are in the upper right half of the table; positive values are those where the $r$ value for the forest class is higher than the overall $r$ value, and are only highlighted to help distinguish them from negative values.

| DEM 01 | 0.0850 | -0.0216 | **0.0163** | -0.0074 | **0.0044** | **0.0016** | 0.0001 | -0.0024 | **0.0012** | -0.0029 | 0.0003 |
| 0.5515 | DEM 02 | -0.0133 | **0.0156** | -0.1200 | -0.1093 | -0.1165 | -0.1093 | -0.1131 | -0.0825 | -0.1005 | -0.1013 |
| 0.8913 | DEM 03 | 0.7647 | **0.0298** | -0.0584 | -0.0352 | -0.0433 | -0.0426 | -0.0457 | -0.0287 | -0.0392 | -0.0375 |
| DEM 04 | 0.8641 | 0.6285 | 0.8898 | DEM 04 | -0.0111 | **0.0054** | -0.0025 | **0.0038** | -0.0009 | **0.0103** | -0.0004 | **0.0037** |
| 0.9293 | DEM 05 | 0.3785 | 0.7887 | 0.8414 | DEM 05 | **0.0064** | **0.0074** | **0.0011** | **0.0020** | -0.0049 | -0.0037 | **0.0044** |
| 0.9391 | DEM 06 | 0.4659 | 0.8463 | 0.8660 | 0.9550 | DEM 06 | **0.0087** | **0.0072** | **0.0070** | **0.0070** | **0.0068** | **0.0070** |
| 0.9411 | DEM 07 | 0.4281 | 0.8234 | 0.8576 | 0.9697 | 0.9606 | DEM 07 | **0.0054** | **0.0054** | -0.0021 | -0.0012 | **0.0076** |
| 0.9369 | DEM 08 | 0.4482 | 0.8336 | 0.8842 | 0.9613 | 0.9508 | 0.9624 | DEM 08 | -0.0001 | -0.0052 | **0.0005** | **0.0017** |
| 0.9189 | DEM 09 | 0.4436 | 0.8273 | 0.8846 | 0.9586 | 0.9480 | 0.9591 | 0.9789 | DEM 09 | -0.0014 | **0.0017** | **0.0007** |
| 0.9135 | DEM 10 | 0.5275 | 0.8648 | 0.8954 | 0.9282 | 0.9380 | 0.9345 | 0.9590 | 0.9685 | DEM 10 | **0.0031** | -0.0030 |
| 0.9224 | DEM 11 | 0.4891 | 0.8447 | 0.8725 | 0.9380 | 0.9423 | 0.9414 | 0.9628 | 0.9671 | 0.9759 | DEM 11 | -0.0051 |
| 0.9354 | DEM 12 | 0.4485 | 0.8354 | 0.8896 | 0.9667 | 0.9550 | 0.9687 | 0.9815 | 0.9795 | 0.9558 | 0.9590 | DEM 12 |

Table B-4  The road landuse class Pearson product moment correlation coefficient ($r$) matrix for DEMs (lower left half of the table). Difference between $r$ values for the overall area and those of the road landuse class are in the upper right half of the table; positive values are those where the $r$ value for the road class is higher than the overall $r$ value, and are only highlighted to help distinguish them from negative values.

| DEM 01 | 0.0023 | 0.0049 | 0.0238 | -0.0032 | **0.0039** | -0.0017 | **0.0009** | **0.0029** | -0.0029 | -0.0030 | -0.0009 |
| 0.6388 | DEM 02 | DE 02 | 0.0017 | -0.0022 | -0.0046 | -0.0056 | -0.0096 | -0.0219 | -0.0209 | -0.0226 | -0.0137 | -0.0160 |
| 0.9178 | DEM 03 | 0.7797 | DEM 03 | 0.0115 | **0.0054** | **0.0036** | **0.0063** | -0.0049 | -0.0034 | -0.0100 | -0.0034 | -0.0023 |
| 0.8716 | DEM 04 | 0.6107 | DEM 04 | 0.8715 | **0.0297** | **0.0319** | **0.0238** | **0.0218** | **0.0194** | **0.0214** | **0.0296** | **0.0239** |
| 0.9335 | DEM 05 | 0.4939 | 0.8525 | 0.8822 | DEM 05 | 0.0094 | 0.0070 | 0.0085 | 0.0073 | 0.0078 | 0.0090 | **0.0063** |
| 0.9366 | DEM 06 | 0.5966 | 0.8511 | 0.8925 | 0.9580 | DEM 06 | 0.0067 | 0.0070 | 0.0095 | 0.0073 | 0.0067 | **0.0083** |
| 0.9378 | DEM 07 | 0.5350 | 0.8730 | 0.8839 | 0.9693 | 0.9606 | DEM 07 | **0.0054** | **0.0053** | **0.0048** | 0.0067 | **0.0061** |
| 0.9377 | DEM 08 | 0.5356 | 0.8713 | 0.9022 | 0.9687 | 0.9506 | 0.9624 | DEM 08 | **0.0027** | **0.0026** | **0.0036** | **0.0030** |
| 0.9242 | DEM 09 | 0.5358 | 0.8696 | 0.9049 | 0.9639 | 0.9505 | 0.9590 | 0.9817 | DEM 09 | **0.0035** | **0.0049** | **0.0032** |
| 0.9094 | DEM 10 | 0.5874 | 0.8835 | 0.9065 | 0.9409 | 0.9383 | 0.9414 | 0.9668 | 0.9734 | DEM 10 | **0.0024** | **0.0047** |
| 0.9223 | DEM 11 | 0.5759 | 0.8805 | 0.9025 | 0.9507 | 0.9422 | 0.9493 | 0.9659 | 0.9703 | 0.9752 | DEM 11 | **0.0052** |
| 0.9342 | DEM 12 | 0.5338 | 0.8706 | 0.9098 | 0.9686 | 0.9563 | 0.9672 | 0.9828 | 0.9820 | 0.9635 | 0.9693 | DEM 12 |
Table B-5  The rough land landuse class Pearson product moment correlation coefficient ($r$) matrix for DEMs (lower left half of the table). Difference between $r$ values for the overall area and those of the rough land class are in the upper right half of the table; positive values are those where the $r$ value for the rough land class is higher than the overall $r$ value, and are only highlighted to help distinguish them from negative values.

<table>
<thead>
<tr>
<th>DEM 01</th>
<th>0.0963</th>
<th>0.0157</th>
<th>0.0476</th>
<th>0.0120</th>
<th>0.0193</th>
<th>0.0105</th>
<th>0.0159</th>
<th>0.0210</th>
<th>0.0359</th>
<th>0.0279</th>
<th>0.0132</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.7328</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9286</td>
<td>0.8614</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.8954</td>
<td>0.7582</td>
<td>0.9063</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9487</td>
<td>0.9540</td>
<td>0.6914</td>
<td>0.9020</td>
<td>0.8999</td>
<td>0.9626</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9500</td>
<td>0.6429</td>
<td>0.8773</td>
<td>0.8834</td>
<td>0.9692</td>
<td>0.9672</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9527</td>
<td>0.6471</td>
<td>0.8819</td>
<td>0.9009</td>
<td>0.9699</td>
<td>0.9560</td>
<td>0.9614</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9423</td>
<td>0.6546</td>
<td>0.8870</td>
<td>0.9092</td>
<td>0.9666</td>
<td>0.9583</td>
<td>0.9570</td>
<td>0.9842</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9482</td>
<td>0.7154</td>
<td>0.9169</td>
<td>0.9268</td>
<td>0.9481</td>
<td>0.9583</td>
<td>0.9523</td>
<td>0.9697</td>
<td>0.9747</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9532</td>
<td>0.6938</td>
<td>0.9026</td>
<td>0.9172</td>
<td>0.9582</td>
<td>0.9608</td>
<td>0.9590</td>
<td>0.9726</td>
<td>0.9743</td>
<td>0.9829</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9483</td>
<td>0.6448</td>
<td>0.8809</td>
<td>0.9069</td>
<td>0.9717</td>
<td>0.9622</td>
<td>0.9673</td>
<td>0.9836</td>
<td>0.9832</td>
<td>0.9671</td>
<td>0.9744</td>
<td>0.9744</td>
</tr>
</tbody>
</table>

Table B-6  The urban landuse class Pearson product moment correlation coefficient ($r$) matrix for DEMs (lower left half of the table). Difference between $r$ values for the overall area and those of the urban class are in the upper right half of the table; positive values are those where the $r$ value for the urban class is higher than the overall $r$ value, and are only highlighted to help distinguish them from negative values.

<table>
<thead>
<tr>
<th>DEM 01</th>
<th>0.1802</th>
<th>0.0434</th>
<th>0.0696</th>
<th>0.0076</th>
<th>0.0389</th>
<th>0.0180</th>
<th>0.0143</th>
<th>0.0112</th>
<th>0.0402</th>
<th>0.0376</th>
<th>0.0062</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.8167</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9563</td>
<td>0.9148</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9174</td>
<td>0.7538</td>
<td>0.9158</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9443</td>
<td>0.6545</td>
<td>0.8765</td>
<td>0.9442</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9736</td>
<td>0.7582</td>
<td>0.9319</td>
<td>0.9490</td>
<td>0.9681</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9575</td>
<td>0.6966</td>
<td>0.9017</td>
<td>0.9492</td>
<td>0.9916</td>
<td>0.9718</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9511</td>
<td>0.7067</td>
<td>0.9070</td>
<td>0.9620</td>
<td>0.9866</td>
<td>0.9738</td>
<td>0.9833</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9325</td>
<td>0.6992</td>
<td>0.8991</td>
<td>0.9660</td>
<td>0.9802</td>
<td>0.9601</td>
<td>0.9774</td>
<td>0.9944</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9525</td>
<td>0.7472</td>
<td>0.9271</td>
<td>0.9700</td>
<td>0.9752</td>
<td>0.9741</td>
<td>0.9794</td>
<td>0.9914</td>
<td>0.9917</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9629</td>
<td>0.7382</td>
<td>0.9259</td>
<td>0.9644</td>
<td>0.9819</td>
<td>0.9802</td>
<td>0.9863</td>
<td>0.9927</td>
<td>0.9891</td>
<td>0.9960</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9413</td>
<td>0.6879</td>
<td>0.8947</td>
<td>0.9643</td>
<td>0.9877</td>
<td>0.9671</td>
<td>0.9844</td>
<td>0.9958</td>
<td>0.9950</td>
<td>0.9896</td>
<td>0.9915</td>
<td>0.9744</td>
</tr>
</tbody>
</table>
Table B-7  The water landuse class Pearson product moment correlation coefficient \((r)\) matrix for DEMs (lower left half of the table). Difference between \(r\) values for the overall area and those of the water landuse class are in the upper right half of the table; positive values are those where the \(r\) value for the water class is higher than the overall \(r\) value, and are only highlighted to help distinguish them from negative values.

<table>
<thead>
<tr>
<th>DEM 01</th>
<th>0.0707</th>
<th>0.0224</th>
<th>0.0885</th>
<th>-0.0136</th>
<th>-0.0014</th>
<th>-0.0139</th>
<th>-0.0228</th>
<th>-0.0351</th>
<th>-0.0035</th>
<th>-0.0034</th>
<th>-0.0313</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.7072</td>
<td>DEM 02</td>
<td>0.0577</td>
<td>0.0494</td>
<td>-0.0055</td>
<td>-0.0249</td>
<td>-0.0342</td>
<td>-0.0271</td>
<td>-0.0681</td>
<td>-0.0533</td>
<td>-0.0367</td>
<td>-0.0664</td>
</tr>
<tr>
<td>0.9353</td>
<td>0.8357</td>
<td>DEM 03</td>
<td>0.0536</td>
<td>-0.0078</td>
<td>-0.0106</td>
<td>-0.0193</td>
<td>-0.0218</td>
<td>-0.0452</td>
<td>-0.0292</td>
<td>-0.0182</td>
<td>-0.0410</td>
</tr>
<tr>
<td>0.9363</td>
<td>0.6623</td>
<td>0.9136</td>
<td>DEM 04</td>
<td>0.0922</td>
<td>0.0932</td>
<td>0.0857</td>
<td>0.0760</td>
<td>0.0577</td>
<td>0.0724</td>
<td>0.0871</td>
<td>0.0611</td>
</tr>
<tr>
<td>0.9231</td>
<td>0.4930</td>
<td>0.8393</td>
<td>0.9447</td>
<td>DEM 05</td>
<td>0.0351</td>
<td>0.0263</td>
<td>0.0216</td>
<td>0.0207</td>
<td>0.0391</td>
<td>0.0364</td>
<td>0.0236</td>
</tr>
<tr>
<td>0.9333</td>
<td>0.5503</td>
<td>0.8709</td>
<td>0.9538</td>
<td>0.9837</td>
<td>DEM 06</td>
<td>0.0265</td>
<td>0.0317</td>
<td>0.0287</td>
<td>0.0391</td>
<td>0.0375</td>
<td>0.0294</td>
</tr>
<tr>
<td>0.9256</td>
<td>0.5104</td>
<td>0.8474</td>
<td>0.9458</td>
<td>0.9886</td>
<td>0.9784</td>
<td>DEM 07</td>
<td>0.0173</td>
<td>0.0167</td>
<td>0.0310</td>
<td>0.0317</td>
<td>0.0172</td>
</tr>
<tr>
<td>0.9140</td>
<td>0.5304</td>
<td>0.8544</td>
<td>0.9564</td>
<td>0.9818</td>
<td>0.9753</td>
<td>0.9743</td>
<td>DEM 08</td>
<td>0.0111</td>
<td>0.0230</td>
<td>0.0249</td>
<td>0.0086</td>
</tr>
<tr>
<td>0.8862</td>
<td>0.4886</td>
<td>0.8278</td>
<td>0.9432</td>
<td>0.9773</td>
<td>0.9697</td>
<td>0.9704</td>
<td>0.9901</td>
<td>DEM 09</td>
<td>0.0166</td>
<td>0.0172</td>
<td>0.0063</td>
</tr>
<tr>
<td>0.9088</td>
<td>0.5567</td>
<td>0.8643</td>
<td>0.9575</td>
<td>0.9722</td>
<td>0.9701</td>
<td>0.9676</td>
<td>0.9872</td>
<td>0.9865</td>
<td>DEM 10</td>
<td>0.0186</td>
<td>0.0199</td>
</tr>
<tr>
<td>0.9219</td>
<td>0.5529</td>
<td>0.8657</td>
<td>0.9600</td>
<td>0.9781</td>
<td>0.9730</td>
<td>0.9743</td>
<td>0.9872</td>
<td>0.9826</td>
<td>0.9914</td>
<td>DEM 11</td>
<td>0.0170</td>
</tr>
<tr>
<td>0.9038</td>
<td>0.4834</td>
<td>0.8319</td>
<td>0.9470</td>
<td>0.9859</td>
<td>0.9774</td>
<td>0.9783</td>
<td>0.9884</td>
<td>0.9871</td>
<td>0.9787</td>
<td>0.9811</td>
<td>DEM 12</td>
</tr>
</tbody>
</table>
Figure C-1  Product 01 of the DEM fusion algorithm: average of all input DEMs, before any other algorithm processing, at each cell location. No contrast stretch was applied to symbolize the data.
Figure C-2  Product 02 of the DEM fusion algorithm: average of all input DEMs, after slope and elevation thresholding, at each cell location. No contrast stretch was applied to symbolize the data.
Figure C-3  Product 03 of the DEM fusion algorithm: average of all final cluster members after $k$-means clustering, at each cell location. No contrast stretch was applied to symbolize the data.
Figure C-4  Product 04 of the DEM fusion algorithm: result of filtering Product 03 with an Adaptive Mean filter. No contrast stretch was applied to symbolize the data.
Figure C-5  Product 05 of the DEM fusion algorithm: the final product; result of smoothing Product 04 with Gaussian filter. No contrast stretch was applied to symbolize the data.
Figure D-1 DEM of difference for DEM 01: difference in elevation as a result of subtracting the reference DEM from DEM 01 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-2 DEM of difference for DEM 02: difference in elevation as a result of subtracting the reference DEM from DEM 02 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-3  DEM of difference for DEM 03: difference in elevation as a result of subtracting the reference DEM from DEM 03 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-4  DEM of difference for DEM 04: difference in elevation as a result of subtracting the reference DEM from DEM 04 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-5 DEM of difference for DEM 05: difference in elevation as a result of subtracting the reference DEM from DEM 05 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-6  DEM of difference for DEM 06: difference in elevation as a result of subtracting the reference DEM from DEM 06 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-7 DEM of difference for DEM 07: difference in elevation as a result of subtracting the reference DEM from DEM 07 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-8  DEM of difference for DEM 08: difference in elevation as a result of subtracting the reference DEM from DEM 08 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-9 DEM of difference for DEM 09: difference in elevation as a result of subtracting the reference DEM from DEM 09 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-10  DEM of difference for DEM 10: difference in elevation as a result of subtracting the reference DEM from DEM 10 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-11  DEM of difference for DEM 11: difference in elevation as a result of subtracting the reference DEM from DEM 11 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-12 DEM of difference for DEM 12: difference in elevation as a result of subtracting the reference DEM from DEM 12 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-13  DEM of difference for Product 01: difference in elevation as a result of subtracting the reference DEM from Product 01 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-14  DEM of difference for Product 02: difference in elevation as a result of subtracting the reference DEM from Product 02 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-15 DEM of difference for Product 03: difference in elevation as a result of subtracting the reference DEM from Product 03 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-16 DEM of difference for Product 04: difference in elevation as a result of subtracting the reference DEM from Product 04 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
Figure D-17  DEM of difference for Product 05: difference in elevation as a result of subtracting the reference DEM from Product 05 at each cell location in the study area. No contrast stretch was applied to symbolize the data.
APPENDIX E

ADDITIONAL CORROBORATION FIGURES

Figure E-1  Box plots of the DEM of difference statistics for DEMs 01 to 04, for the overall study area and by landuse class.
Figure E-2  Box plots of the DEM of difference statistics for DEMs 05 to 08, for the overall study area and by landuse class.

Figure E-3  Box plots of the DEM of difference statistics for DEMs 09 to 12, for the overall study area and by landuse class.
Figure E-4 Box plots of the DEM of difference statistics for Products 01 to 05 (left to right), for the overall study area and by landuse class. Note that the scale of the y-axis is different from the other box plots in this section.
Figure F-1  Generalized flowchart of the modified $k$-means clustering algorithm adapted to cluster elevation estimates at each grid cell location, from multiple overlapping input DEMs of the same cell resolution and alignment.
APPENDIX G

K-MEANS CLUSTERING ALGORITHM CODE

/*
 * Code Name: Modified k-means clustering of elevation estimates for multiple, overlapping
DEMS
 * Developed by: Colleen Fuss, MSc. University of Guelph, 2013
 * Description: The modified k-means clustering
 * algorithm (original ref: Paul M. Mather; see Mather and Koch, 2011) was adapted
 * to cluster elevations from overlapping DEMs in 1 dimension (i.e. vertical) at each
 * cell location. Multiple overlapping DEMs with the same grid cell resolution and
 * overall grid size are input into the code.
 * Notes: 1) The input DEMs are required to be in .dep format
 * 2) The user-defined parameters are labelled in the comments above the
 * initialization of each. 3) This code was run in NetBeans IDE v.7.1.2, linking
 * to the .dll file for the software program Whitebox GAT v.2.0.3.
 */
package modkmeans_1;
import java.util.ArrayList;
import whitebox.geospatialfiles.WhiteboxRaster;
import whitebox.geospatialfiles.WhiteboxRasterInfo;
import whitebox.interfaces.WhiteboxPlugin;
import whitebox.interfaces.WhiteboxPluginHost;
/**
 * @author fussc
 */
public class ModKmeans_1 {

public static void main(String[] args) {
   ModKmeans_1 DSS2 = new ModKmeans_1();
   //Input DEM file locations and names
   String inputDemsString = "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v4/i_d2_ElevFilter_2SD/n01_i_d.dep;
+C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v4/i_d2_ElevFilter_2SD/n02_i_d.dep;"
+C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v4/i_d2_ElevFilter_2SD/n03_i_d.dep;"
+C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v4/i_d2_ElevFilter_2SD/n04_i_d.dep;"
+C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v4/i_d2_ElevFilter_2SD/n05_i_d.dep;"
+C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v4/i_d2_ElevFilter_2SD/n06_i_d.dep;"
+C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v4/i_d2_ElevFilter_2SD/n07_i_d.dep;"
+C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v4/i_d2_ElevFilter_2SD/n08_i_d.dep;"
+C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v4/i_d2_ElevFilter_2SD/n09_i_d.dep;"
+C:/a1 CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v4/i_d2_ElevFilter_2SD/n10_i_d.dep;"
+C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v4/i_d2_ElevFilter_2SD/n11_i_d.dep;"
+C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v4/i_d2_ElevFilter_2SD/n12_i_d.dep;";}
// Output raster file directories and names for binary grids that show if final cluster membership for each input DEM has occurred at each cell location: '1' indicates DEM membership.

String outputFinalClustMembString = "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/n01_fcm.dep;" + "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/n02_fcm.dep;" + "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/n03_fcm.dep;" + "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/n04_fcm.dep;" + "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/n05_fcm.dep;" + "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/n06_fcm.dep;" + "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/n07_fcm.dep;" + "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/n08_fcm.dep;" + "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/n09_fcm.dep;" + "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/n10_fcm.dep;" + "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/n11_fcm.dep;" + "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/n12_fcm.dep;"

DSS2.run(inputDemsString, outputFinalClustMembString);
}

public void run(String inputDemsString, String outputFinalClustMembString) {
    int cols, rows;
    int numDems;
    int a, b, i, j, k;
    int numClusters;

    // User-defined parameter: Minimum number of DEM elevations at each cell location for clustering to proceed
    int minNumNeeded = 3;

    // User-defined parameter: Maximum number of iterations in the cluster merging and re-assigning loop
    int maxIterations = 50;

    // Directory where processing is taking place
    String myDirectory = "C:/a1_CFuss/a_DEM_Compare/a_a_New_DEMs/FF_v5/j_b_FinalClusterMemb/";

    String[] inputDems, outputFinalClustMemb;
    inputDems = inputDemsString.split(";");
    outputFinalClustMemb = outputFinalClustMembString.split(";");
    numDems = inputDems.length;

    WhiteboxRasterInfo[] inDem = new WhiteboxRasterInfo[numDems];
    WhiteboxRaster[] outFCM = new WhiteboxRaster[numDems];

    // Output raster names and extensions
    String outMinHeaderName = "FcmMin.dep";
    String outMaxHeaderName = "FcmMax.dep";
    String outAveHeaderName = "FcmAve.dep";
    String outRangeHeaderName = "FcmRange.dep";
    String outNumMembsHeaderName = "FcmNumMembs.dep";
    String outNumLcHeaderName = "NumPosFcm.dep";
    String outNumJHeaderName = "NumIterations.dep";
try {
    for (i = 0; i < numDems; i++) {
        inDem[i] = new WhiteboxRasterInfo(inputDems[i]);
    }

    rows = inDem[0].getNumberRows();
    cols = inDem[0].getNumberColumns();
    double noData = inDem[0].getNoDataValue();

    for (i = 0; i < numDems; i++) {
        outFCM[i] = new WhiteboxRaster(outputFinalClustMemb[i], "rw", inputDems[i],
                WhiteboxRaster.DataType.FLOAT, noData);
    }

    double[][] dataElev;
    double[][] demData = new double[numDems][cols];
    double dataNow = 0;
    int numUsed = 0;
    double Total, Min, Max, Ave, Range, Spacing;
    double TotalSqVar, stdDev;
    double[] clusterCentre;
    ArrayList centres = new ArrayList();
    double[][] clusterStats;
    double[] newClusterStats;
    double[] largestClusterStats;
    double minDistance;
    double dist;
    double minDist;
    int mergeCluster1, mergeCluster2;
    double centreMergeDist;
    int[] pointCluster;
    int[] oldPointCluster;
    int numPointsChanged;
    double percentChanged;

    //User-defined parameter: Percent change to be achieved before cluster merging and
    //re-assigning loop can stop
    double percentChangedThreshold = 0;

    double mostMembers;
    double numMembs;
    double narrowestRange;
    double numLrgClusters;
    double rangeNow;

    double[] demFinalClusterMemb;
    numLrgClusters = 0;
    j = 0;

    //Set-up new output rasters
    outLcMin = new WhiteboxRaster(myDirectory + outMinHeaderName, "rw", inputDems[0],
                WhiteboxRaster.DataType.FLOAT, noData);
    outLcMax = new WhiteboxRaster(myDirectory + outMaxHeaderName, "rw", inputDems[0],
                WhiteboxRaster.DataType.FLOAT, noData);
outLcAve = new WhiteboxRaster(myDirectory + outAveHeaderName, "rw", inputDems[0],
WhiteboxRaster.DataType.FLOAT, noData);
outLcRange = new WhiteboxRaster(myDirectory + outRangeHeaderName, "rw",
inputDems[0],
WhiteboxRaster.DataType.FLOAT, noData);
outLcNumMemb = new WhiteboxRaster(myDirectory + outNumMembHeaderName, "rw",
inputDems[0],
WhiteboxRaster.DataType.FLOAT, noData);
outNumLc = new WhiteboxRaster(myDirectory + outNumLcHeaderName, "rw",
inputDems[0],
WhiteboxRaster.DataType.FLOAT, noData);
outNumJ = new WhiteboxRaster(myDirectory + outNumJHeaderName, "rw", inputDems[0],
WhiteboxRaster.DataType.FLOAT, noData);
for (row = 0; row < rows; row++) {
    numUsed = 0;
    //Get values for elevation for each row, each raster
    for (i = 0; i < numDems; i++) {
        dataElev = inDem[i].getRowValues(row);
        demData[i] = dataElev;
    }
    for (col = 0; col < cols; col++) {
        //Re-set all variables
        Min = Double.POSITIVE_INFINITY;
        Max = Double.NEGATIVE_INFINITY;
        Total = 0;
        numUsed = 0;
        Ave = 0;
        Range = 0;
        Spacing = 0;
        stdDev = 0;
        TotalSqVar = 0;
        //Determine how many values are present for each cell
        for (i = 0; i < numDems; i++) {
            dataNow = demData[i][col];
            if (dataNow != noData) {
                numUsed += 1;
                Total += dataNow;
                if (dataNow < Min) {
                    Min = dataNow;
                }
                if (dataNow > Max) {
                    Max = dataNow;
                }
            }
        }
        //Initiate and Reset array for whether DEMs were used in final cluster
        demFinalClusterMemb = new double[numDems];
        //If there are any elevations present
        if (numUsed != 0) {
            //Calculate the Average and Range of elevations
            Ave = Total / numUsed;
            Range = Max - Min;
            //re-set the number of rasters used
            numUsed = 0;
        }
//Find the squared variance value for all elevations
for (i = 0; i < numDems; i++) {
    dataNow = demData[i][col];
    if (dataNow != noData) {
        TotalSqVar += ((dataNow - Ave) * (dataNow - Ave));
        numUsed += 1;
    }
}

//Calculate the standard deviation
stdDev = Math.sqrt(TotalSqVar / numUsed);

//User Defined Parameter: set the cluster centre merge min distance
to 10% of the range of elevation values
centreMergeDist = (0.10 * Range);

//If more than minimum number of elevations present, proceed with
cluster analysis
if (numUsed > minNumNeeded) {
    //set-up array of elevation values that are present at this
    location
double valuesUsed[] = new double[numUsed];
k = 0;
    for (i = 0; i < numDems; i++) {
        dataNow = demData[i][col];
        if (dataNow != noData) {
            valuesUsed[k] = dataNow;
            k += 1;
        }
    }
    numClusters = numUsed * 5;
    Spacing = Range / (numClusters + 1);
    //initialize Cluster Centres
    clusterCentre = new double[numClusters];
    centres.clear();
    for (a = 0; a < numClusters; a++) {
        clusterCentre[a] = Min + Spacing * (a + 1);
        centres.add(clusterCentre[a]);
    }

    //Re-set cluster data arrays for this location (row,col)
    pointCluster = new int[numUsed];
    clusterStats = new double[numClusters][5];
    newClusterStats = new double[5];
    largestClusterStats = new double[5];

    //Start iterating though clustering until a stopping condition is
    met
    j = 0;
    do {
        if (j > 0) {
//update the new number of clusters
numClusters = centres.size();

//Merge close clusters
minDist = Double.POSITIVE_INFINITY;
mergeCluster1 = 0;
mergeCluster2 = 0;

for (a = 0; a < numClusters; a++) {
   double centreA = (double) centres.get(a);
   for (b = 0; b < numClusters; b++) {
      if (b > a) {
         double centreB = (double) centres.get(b);
         dist = Math.sqrt((centreA - centreB) *
                          (centreA - centreB));
         if (dist < minDist) {
            minDist = dist;
            mergeCluster1 = a;
            mergeCluster2 = b;
         }
      }
   }
}

if (minDist < centreMergeDist) {
   newClusterStats[0] = 0;
   newClusterStats[1] = 0;
   newClusterStats[4] = 0;
   for (i = 0; i < numUsed; i++) {
      if (pointCluster[i] == mergeCluster2 ||
          pointCluster[i] == mergeCluster1) {
         newClusterStats[0] += 1;
         newClusterStats[1] += valuesUsed[i];
         //update cluster Min
         if (valuesUsed[i] < newClusterStats[2]) {
            newClusterStats[2] = valuesUsed[i];
         }
         //update cluster Max
         if (valuesUsed[i] > newClusterStats[3]) {
            newClusterStats[3] = valuesUsed[i];
         }
      }
   }
                        newClusterStats[0]);
   centres.remove(Math.max(mergeCluster1,
                           mergeCluster2));
   centres.remove(Math.min(mergeCluster1,
                           mergeCluster2));
   centres.add(newClusterStats[4]);
   numClusters = centres.size();
}
j++;
oldPointCluster = pointCluster;
pointCluster = new int[numUsed];

// assign each point to a cluster
for (i = 0; i < numUsed; i++) {
dataNow = valuesUsed[i];
minDistance = Double.POSITIVE_INFINITY;
dist = 0;
for (a = 0; a < numClusters; a++) {
    clusterCentre[a] = (double) centres.get(a);
    dist = Math.sqrt((dataNow - clusterCentre[a]) * (dataNow - clusterCentre[a]));
    if (dist < minDistance) {
        minDistance = dist;
        pointCluster[i] = a;
    }
}
}

numPointsChanged = 0;
for (i = 0; i < numUsed; i++) {
    if (oldPointCluster[i] != pointCluster[i]) {
        numPointsChanged += 1;
    }
}

// Re-set cluster stats
for (a = 0; a < numClusters; a++) {
    clusterStats[a][0] = 0;
    clusterStats[a][1] = 0;
    clusterStats[a][2] = Double.POSITIVE_INFINITY;
    clusterStats[a][3] = Double.NEGATIVE_INFINITY;
    clusterStats[a][4] = 0;
}

// Update cluster stats
for (i = 0; i < numUsed; i++) {
    for (a = 0; a < numClusters; a++) {
        if (pointCluster[i] == a) {
            clusterStats[a][0] += 1;
            clusterStats[a][1] += valuesUsed[i];
            // update cluster Min
            if (valuesUsed[i] < clusterStats[a][2]) {
                clusterStats[a][2] = valuesUsed[i];
            }
            // update cluster Max
            if (valuesUsed[i] > clusterStats[a][3]) {
                clusterStats[a][3] = valuesUsed[i];
            }
        }
    }
}
//update cluster centres
centres.clear();
for (a = 0; a < numClusters; a++) {
    if (clusterStats[a][0] > 0) {
        clusterStats[a][4] = clusterStats[a][1] / clusterStats[a][0];
        clusterCentre[a] = clusterStats[a][4];
        centres.add(clusterCentre[a]);
    }
}
numClusters = centres.size();
percentChanged = (double) numPointsChanged / numUsed * 100;
} while ((percentChanged > percentChangedThreshold) && (j < maxIterations));

//Find the cluster with the highest membership
mostMembers = Double.NEGATIVE_INFINITY;
for (a = 0; a < numClusters; a++) {
    if (clusterStats[a][0] > mostMembers) {
        mostMembers = clusterStats[a][0];
        largestClusterStats[0] = clusterStats[a][0];
        largestClusterStats[1] = clusterStats[a][1];
        largestClusterStats[2] = clusterStats[a][2];
        largestClusterStats[3] = clusterStats[a][3];
        largestClusterStats[4] = clusterStats[a][4];
    }
}
Min = largestClusterStats[2];
Max = largestClusterStats[3];
Ave = largestClusterStats[4];
umMembs = largestClusterStats[0];

// Search to find any clusters with same num of members as largest cluster
//Results in final cluster selected for algorithm product
numLrgClusters = 0;
narrowestRange = Double.NEGATIVE_INFINITY;
for (a = 0; a < numClusters; a++) {
    if (clusterStats[a][0] == numMembs) {
        numLrgClusters += 1;
        rangeNow = (clusterStats[a][0] - clusterStats[a][2]);
        if (rangeNow < Range) {
            mostMembers = clusterStats[a][0];
            largestClusterStats[0] = clusterStats[a][0];
            largestClusterStats[1] = clusterStats[a][1];
            largestClusterStats[2] = clusterStats[a][2];
            largestClusterStats[3] = clusterStats[a][3];
            largestClusterStats[4] = clusterStats[a][4];
        }
    }
}

Min = largestClusterStats[2];
Max = largestClusterStats[3];
Ave = largestClusterStats[4];
umMembs = largestClusterStats[0];

// Search to find any clusters with same num of members as largest cluster
//Results in final cluster selected for algorithm product
numLrgClusters = 0;
narrowestRange = Double.NEGATIVE_INFINITY;
for (a = 0; a < numClusters; a++) {
    if (clusterStats[a][0] == numMembs) {
        numLrgClusters += 1;
        rangeNow = (clusterStats[a][3] - clusterStats[a][2]);
        if (rangeNow < Range) {
            mostMembers = clusterStats[a][0];
            largestClusterStats[0] = clusterStats[a][0];
            largestClusterStats[1] = clusterStats[a][1];
            largestClusterStats[2] = clusterStats[a][2];
            largestClusterStats[3] = clusterStats[a][3];
            largestClusterStats[4] = clusterStats[a][4];
        }
    }
}
Min = largestClusterStats[2];
Max = largestClusterStats[3];
Ave = largestClusterStats[4];
umMembs = largestClusterStats[0];

//NEW code for members of final cluster

for (i = 0; i < numDems; i++) {
    dataNow = demData[i][col];
    if (dataNow <= Max && dataNow >= Min) {
        demFinalClusterMemb[i] = 1;
    } else {
        demFinalClusterMemb[i] = 0;
    }
}

} else {

    numMembs = 0;
    Min = noData;
    Max = noData;
    Ave = noData;
    Range = noData;
    numLrgClusters = noData;
    j = 0;
    for (i = 0; i < numDems; i++) {
        demFinalClusterMemb[i] = 0;
    }
}

} else {

    Min = noData;
    Max = noData;
    Ave = noData;
    Range = noData;
    numMembs = noData;
    numLrgClusters = noData;
    j = 0;
    for (i = 0; i < numDems; i++) {
        demFinalClusterMemb[i] = noData;
    }
}

outLcMin.setValue(row, col, Min);
outLcMax.setValue(row, col, Max);
outLcAve.setValue(row, col, Ave);
outLcRange.setValue(row, col, Range);
outLcNumMembs.setValue(row, col, numMembs);
outLcNum.setValue(row, col, numLrgClusters);
for (i = 0; i < numDems; i++) {
    outFCM[i].setValue(row, col, demFinalClusterMemb[i]);
}

} //end of column

} //end of row

for (i = 0; i < numDems; i++) {
    inDem[i].close();
    outFCM[i].close();
}
outLcMin.close();
outLcMax.close();
outLcAve.close();
outLcRange.close();
outLcNumMems.close();
outNumLc.close();
outNumJ.close();

} catch (Exception e) {
    System.out.println(e.getMessage());
}
}