

Glacier Surface Analysis

Airborne Laser Scanning for monitoring glaciers and crevasses

M.P. Kodde
Msc. Geomatics

 **TU Delft**

Delft University of Technology



Glacier surface analysis

Airborne Laser Scanning for monitoring glaciers and crevasses

MSc. Geomatics Thesis



Delft University of Technology



Martin Kodde
October 2006

Optical and Laser Remote Sensing
Department of Earth Observation and Space Systems (DEOS)
Faculty of Aerospace Engineering, Delft University of Technology

In cooperation with:
Department for Natural Hazards and Risk Research
Institute of Geography
Faculty of Geo- and Atmospheric Sciences, University of Innsbruck

Cover: Hintereisferner, Tyrol. (Front) Image courtesy of Thomas Geist
Fox Glacier, New Zealand. (Back) Image courtesy of Michael Hambrey

Printed by Fugro, Leidschendam

Graduation Professor:
Delft Supervisor:
Innsbruck Supervisor:
Co-reader:

prof. dr. D.G. Simons
dr. ir. B.G.H. Gorte
Univ.Prof. Dipl.-Ing. Dr.techn. N. Pfeifer
dr. R.C. Lindenbergh

Abstract

Glaciers are interesting phenomena to scientists, mountaineers and tourists. Often, glaciers are important for the local economy, power generation and water supply. The fluctuations of glaciers appear to be good indicators for climate change. Apart from that, glaciers are identified as one of the potential sources for natural hazards in the Alps. For these reasons a great interest is shown in monitoring glaciers. Especially detecting crevasses, these are cracks in the ice surface, is important because they are dangerous features when travelling over a glacier. With Airborne Laser Scanning (ALS) it is possible to acquire high resolution elevation data of glaciers. In this thesis a method is presented for the automatic delineation of the glacier surface and crevasse reconstruction.

In an elevation model the glacier surface is often characterised by its smoothness compared to the surrounding bedrock. Additionally, glaciers are connected areas and their size is restricted by hydrological constraints. This information can be used in a classification of the elevation model in glacier and non-glacier surface. The smoothness criterion is the most important one and can be quantified as the local variance of the terrain. The classification works by applying a threshold to the terrain variance or alternatively performing a segmentation based on the terrain gradient. In the latter case the classification works by setting a minimum segment size. Other criteria such as connectivity and hydrological catchment boundaries are applied afterwards. A raster to vector conversion gives the final map with the delineation of the glacier.

Crevasses appear in the glacier as deep cracks formed due to the dynamics within the glacier. They can measure up to tens of metres deep and are dangerous for those travelling over a glacier. A detection of the location of crevasses is therefore of great interest. Crevasses can be detected by assuming that these are deviations from a regular glacier surface without any crevasses. Such a surface can be calculated with Mathematical Morphology using a closing filter. The difference between this surface and the original measurements show the crevasse locations.

Given the assumption that crevasses have a V-like shape, they can be reconstructed from the data. This is done using the point data, in which the bottom of the crevasse and two edges are reconstructed. The bottom is determined by searching for the lowest points, the edges are found using a special implementation of the Douglas-Peucker algorithm.

Both the delineation of the glacier and the detection of crevasses give good results in the presented approach. Some problems occur in the delineation at crevasse locations. Crevasses are sometimes not detected due to snow bridges, but apart from that it gives good results which were verified with manual measurements. The quality of the reconstruction of crevasses is hard to assess due to the lack of reference data. Many assumptions that are difficult to verify were made during the reconstruction.

New missions with a higher point density and the acquisition of reference data for crevasses with Terrestrial Laser Scanning are recommended to improve the result.

Samenvatting

Gletsjers zijn een interessant fenomeen voor wetenschappers, bergbeklimmers en toeristen. Vaak zijn gletsjers belangrijk voor de plaatselijke economie, energievoorziening middels waterkracht en de zoetwatervoorraad. De fluctuaties van gletsjers blijken goede indicatoren te zijn voor klimaatverandering. Behoudens dat worden gletsjers ook als een van de potentiële bronnen voor natuurgevaren in de Alpen gezien. Om deze redenen bestaat er een grote interesse in het monitoren van gletsjers. Met name het detecteren van gletsjerspleten, dit zijn barsten in het ijsoppervlak, is belangrijk, omdat het gevaarlijke objecten zijn voor bezoekers van gletsjers. Met vliegtuig laseraltimetrie, ofwel Airborne Laser Scanning (ALS), is het mogelijk om hoogtegegevens van gletsjers met een hoge resolutie in te winnen. In deze scriptie wordt een methode voor het automatisch omranden van het gletsjeroppervlak en voor de gletsjerspleetreconstructie voorgesteld.

In een hoogtemodel wordt het gletsjeroppervlak gekarakteriseerd door zijn gladheid vergeleken met het omringende gesteente. Daarnaast zijn gletsjers aaneengesloten oppervlaktes en wordt hun grootte bepaald door hydrologische beperkingen. Deze informatie kan gebruikt worden bij het classificeren van het hoogtemodel in de klassen gletsjer en geen-gletsjer. Het gladheidscriterium is de belangrijkste en kan gekwantificeerd worden als de lokale variantie van het terrein. De classificatie past dan een drempelwaarde toe op deze variantie. Als alternatief kan er ook een segmentatie toegepast worden op basis van de terreingradiënt. In laatstgenoemde methode wordt de classificatie uitgevoerd door een minimum segmentgrootte in te stellen. Andere criteria, zoals aanééngeslotenheid en hydrologische drainagegrenzen worden daarna toegepast. Een raster-naar-vector conversie geeft het uiteindelijke bestand met de omranding van de gletsjer.

Gletsjerspleten zijn in een gletsjer te herkennen als diepe barsten die gevormd worden door de bewegingen in het ijs. Ze kunnen tot tien meter diep zijn en zijn gevaarlijk voor mensen die over de gletsjer trekken. Het detecteren van de gletsjerspleetlocaties is daarom zeer interessant. Gletsjerspleten kunnen gedetecteerd worden door aan te nemen dat het afwijkingen zijn van een gletsjeroppervlak zonder spleten. Een dergelijk oppervlak kan met behulp van Mathematische Morfologie berekend worden door een sluitingsfilter te gebruiken. Het verschil tussen dit oppervlak en de oorspronkelijke metingen toont de gletsjerspleetlocaties.

Als verondersteld wordt dat gletsjerspleten een soort V-vorm hebben, kunnen ze uit de metingen gereconstrueerd worden. Hiervoor wordt de puntdata gebruikt, waarin de bodem van de spleet en twee randen te reconstrueren zijn. De bodem wordt bepaald door te zoeken naar de laagste punten. De randen kunnen gevonden worden via een speciale implementatie van het Douglas-Peucker algoritme.

Zowel de omranding van de gletsjer als de detectie van gletsjerspleten geven goede resultaten via de voorgestelde aanpak. Enkele problemen treden op bij het omranden op de gletsjerspleetlocaties. Als gevolg van sneeuwbruggen kunnen gletsjerspleten soms niet gedetecteerd worden. Behoudens dat zijn goede resultaten te behalen die met handmatige metingen overeenkomen. De kwaliteit van de reconstructie van gletsjerspleten is moeilijk vast te stellen vanwege het gebrek aan referentiedata. Vele aannamen die moeilijk zijn te bewijzen moesten gemaakt worden tijdens de reconstructie.

Nieuwe missies met een hogere punt dichtheid en de inwinning van referentiedata voor gletsjerspleten met Terrestrische Laser Scanning worden aangeraden om het resultaat te verbeteren.

Preface

Glaciers are and have always been objects of great interest to both researchers and tourists. In the past, glaciers were major factors in the formation of our geography. Nowadays, they supply interesting information as indicators for climate change. Getting to know more information of glaciers and being able to closely monitor these is therefore of general importance. A contribution in the form of processing Airborne Laser Scanning data is given in this thesis. This thesis was written as the final part of the MSc. Geomatics at Delft University of Technology.

The work on the graduation project was performed from February till September 2006, which included a three month visit to Innsbruck from March till May. In Austria I did most of the research and programming. The thesis was written at the section for Optical and Laser Remote Sensing within the Department of Earth Observation and Space Systems (DEOS) in the faculty of Aerospace Engineering. During my stay in Austria I worked within the group for Natural Hazards at the Department of Geography in the University of Innsbruck. Financial support was kindly provided by AlpS, the Centre of Natural Hazard Management in Innsbruck.

For some people in Austria it was hard to believe that a Dutchman came to do a thesis about glaciers and crevasses. How can someone who lives in a perfectly flat delta understand the particulars of the Alps? Still, the importance of glaciers crosses borders. For instance, the rivers forming the Dutch delta are partly formed by glacier melting water. Borders were also crossed due to the interdisciplinary approach in the graduation project. During the work I received a lot of support from geographers, glaciologists and GIS experts. This made the graduation project a very educational experience. It also proved the strength of the interdisciplinary approach of the Geomatics master.

For the thesis presented here I have to acknowledge some people who contributed to its development. In particular I want to mention Norbert Pfeifer, who made the project possible, arranged my stay in Austria and also was a great advisor during the project. The hospitality and cooking skills of Norbert and Anja are very much appreciated. Professor Dick Simons from TU Delft read the thesis conscientiously and gave valuable comments to my work. My supervisor Ben Gorte gave many critical but constructive remarks and continuously challenged me to make further improvements, which greatly enhanced the quality of the thesis. In Innsbruck I have worked in close cooperation with the members of the LiSA team: Thomas Geist, Bernhard Höfle and Martin Rutzinger. Their experience with the topic and their dedication to Open Source software was inspiring. Back in Delft, the long discussions with Alexander Bucksch have led to many new insights in frequencies, morphology and the coffee machines.

What I will remember most of the last couple of months is the amount of fun I had with my room mates. Martin and Reinhard introduced me to some interesting aspects of Tyrolean and Austrian culture. They were

also remarkably patient when I was struggling with my German again. Thanks to Fatemeh and Adamantios I had a great time in Delft too. The stay in Innsbruck would have been impossible without the help of my parents, who gave me continuous support in every possible way during my six years of study in the field of Geodesy. However, most of all, thank you Sioeke for your continuous support and understanding.

Martin Kodde
Delft, October 2006

Table of Contents

Abstract	iii
Samenvatting	iv
Preface	v
Table of Contents	vii
List of figures	ix
List of tables	xi
Notation and symbols	xiii
Acronyms	xvii
1 Introduction	1
1.1 Motivation	1
1.2 Background	1
1.3 Problem formulation and research objectives	2
1.4 Methodology	3
1.5 Outline	4
2 Introduction to glaciology	5
2.1 Objectives and history of glaciology	5
2.2 Birth of a glacier	6
2.3 Glacier movement and crevasses	7
2.4 Measurements on glaciers	8
3 Data acquisition	11
3.1 Principles of Airborne Laser Scanning	11
3.2 ALS mission design	13
3.3 The Hintereisferner datasets	14
3.4 Other datasets available	17
3.4.1 Intensity images	17
3.4.2 Terrestrial Laser Scanning and Ground Truth	18
4 Glacier delineation	21
4.1 Methodology for delineation	21
4.2 Smoothness based classification	23
4.2.1 Variance of plane fitting	24
4.2.2 Gradient segmentation	26
4.3 Connectivity	29
4.4 Hydrological constraints	29
4.4.1 Hydrologic flow determination	29
4.4.2 Network and catchment extraction	30
4.5 Laser Intensity map	32
4.5.1 Intensity value correction	32
4.5.2 Surface scattering	35
4.5.3 Implementation in GRASS	36
4.6 Results	38
4.6.1 Implementation	38
4.6.2 Quality assessment	41

5	Reconstructing crevasses	45
5.1	Mathematical Morphology	45
5.1.1	Principles of morphology	45
5.1.2	The Hit-or-Miss transform	48
5.1.3	Application for crevasse extraction	49
5.2	Setting the parameters	51
5.2.1	3D structuring element values	52
5.2.2	Structuring element size and variogram	52
5.2.3	Structuring element shape and anisotropy	56
5.2.4	Threshold value	56
5.2.5	Effects of glacier slope	57
5.3	Limitations of ALS based crevasse detection	57
5.4	Results	58
5.5	Reconstruction of crevasses	61
5.5.1	Introduction	61
5.5.2	Adjusting spline interpolation	62
5.5.3	Boundary representation	64
5.5.4	Reconstruction implementation	66
6	Quality Analysis	69
6.1	Sampling interval	69
6.1.1	Effects of a low sampling interval	69
6.1.2	The optimal sampling interval	71
6.2	Effects of scan angles	72
6.3	Laser intensity	73
7	Conclusions and future work	77
7.1	Conclusions	77
7.2	Future work	78
7.3	Discussion	79
	Bibliography	81
	Appendix A: Alpenvereinskarte	85
	Appendix B: Research Area	87
	Appendix C: Open Source GIS	89
	Appendix D: Module documentation	91

List of figures

1.1	Mountaineers traversing a crevasse in the French Alps.	2
1.2	Workflow of the developed glacier surface analysis toolbox.	3
2.1	Distribution of glaciers around the world.	5
2.2	Surface velocity of a glacier.	7
2.3	Vertical glacier velocity.	7
2.4	Crevasse as an obstacle to glacier travel.	8
2.5	Location and formation of crevasses.	8
3.1	Typical ALS campaign setup.	11
3.2	Multiple returns at crevasses.	13
3.3	Multiple returns when debris lies on the ice.	13
3.4	The Optech ALTM system components.	13
3.5	Laser scanning trajectory as it was flown in 2004.	14
3.6	Valley in the Ötztal with the tongue of the Hintereisferner.	15
3.7	Dates of the twelve laser scanning flights.	15
3.8	Shaded relief view of the laser scanning data.	16
3.9	Difference between geoid and ellipsoid.	16
3.10	Laser intensity image of the glacier snout.	17
3.11	Expedition to the Hintereisferner.	18
3.12	Terrestrial Laser Scanning measurement near the snout.	19
3.13	TLS measurements on the glacier.	19
4.1	A shaded relief view of the ALS data from October 2004.	21
4.2	The concept of a catchment area.	22
4.3	Two approaches for classification.	24
4.4	Smooth surfaces can locally be represented with a plane.	24
4.5	The sobel operators for calculating slopes.	27
4.6	Gradient images.	27
4.7	Result after segmentation based on gradient.	28
4.8	Flow direction map (left) and flow accumulation map (right)	30
4.9	Streams ordered with the Strahler Order	31
4.10	The catchment areas in the Hintereisferner valley.	31
4.11	Laser intensity values interpolated to a raster.	32
4.12	Effects influencing laser signal intensity.	34
4.13	Laser intensity image after correction.	37
4.14	Intensity map of a sloped ice field.	37
4.15	Flow chart of the delineation process.	38
4.16	The variance of the terrain within an 11 x 11 m window.	39
4.17	The window size projected on the glacier surface.	39
4.18	Pixels that are classified as glacier after thresholding at $0.06 m^2$.	40
4.19	After applying the connectivity constraint and closing.	41
4.20	The computed delineation overlaid on the shaded relief view.	41
4.21	Dead ice, covered by dust and rocks, at the snout of the glacier.	42

5.1	The Fox Glacier in New Zealand, clearly showing many crevasses.	45
5.2	Example of a disk shaped structuring element.	46
5.3	Examples of erosion and dilation operations.	46
5.4	Examples of opening and closing operations.	47
5.5	Profile intersecting approximately 1.5 kilometres of the glacier.	50
5.6	Location where the profile was taken.	50
5.7	Profile of the Hintereisferner DEM and the result after closing.	51
5.8	The result of the Top Hat operation.	51
5.9	Effect of a small structuring element on the detected crevasses.	53
5.10	Experimental variogram for a part of the bedrock.	53
5.11	Theoretical variogram for the glacier surface.	55
5.12	Surface of the Hintereisferner	55
5.13	Selecting crevasses.	56
5.14	Crevasse closing under inclination.	57
5.15	A snow bridged crevasse at Antarctica.	58
5.16	Flow chart of crevasse detection method.	59
5.17	Result after top hat filtering.	59
5.18	Detected crevasse locations.	60
5.19	Verification with orthophoto.	60
5.20	Figure showing maximum depth of crevasses.	61
5.21	Side view of a crevasse	61
5.22	Laser points interpolated to a 1m grid.	62
5.23	Wooden splines on the hull of a Viking ship.	63
5.24	Model of a crevasse, interpolated with splines.	64
5.25	Fitted line representing the bottom of the crevasse.	64
5.26	Laser points in the crevasse.	65
5.27	Points in profile perpendicular to bottom line.	66
5.28	Flow chart of the crevasse reconstruction process.	67
6.1	The effect of a low sampling interval.	70
6.2	Shape of the standard crevasse	70
6.3	The measured depth of a crevasse in the worst case.	71
6.4	An error of ΔH as a result of the scan angle θ_s .	72
6.5	Occurrence of crevasses in 2004 dataset.	73
6.6	Spectral reflectance curves for Ice and Snow.	74
6.7	Spectral reflectance curve for Gray Slate.	74
6.8	Blue colours in a crevasse.	75
6.9	Intensity for different surface classes on the Svartisheibreen	76

List of tables

3.1	Optech ALTM 1225 System Parameters	14
4.1	Attenuation of laser energy with $\lambda = 1.06\mu m$	35
4.2	Confusion matrix of the glacier classification	43
5.1	Common used variogram models.	54

Notation and symbols

Raster data storage

\mathbf{X}	Matrix containing a raster map
$\mathbf{X}(r,c)$	Raster map at row/column position (r,c)
\mathbf{C}	Raster map with integer valued classes
\mathbf{H}	Raster map with elevations
\mathbf{I}	Raster map with intensity values
\mathbf{S}	Map with distance travelled by pulse (range map)
Σ	Raster map with variance values
D_x	Domain of map \mathbf{X}
r	Row coordinate
c	Column coordinate
M	Number of rows in the raster map
N	Number of columns in the raster map
p	One point or raster cell
$N_4(p)$	The 4-adjacency of point p
$N_8(p)$	The 8-adjacency of point p
∇_z	Terrain gradient, i.e. first derivative of elevation

Vector maps, segments and classifications

\mathbf{R}	The entire region of the data set
\mathbf{R}_x	Sub region x , i.e. one segment or polygon
$P(\mathbf{R}_x)$	A logical operation over the points in set \mathbf{R}_x
\hat{k}	Statistical measure that shows how much a classification is better than a fully random classification
O	Classification confusion matrix
o_{ij}	Elements of the confusion matrix
o_{i+}, o_{+i}	Row and column sum of the elements of the confusion matrix

Signals

c	Velocity of light in vacuum, $c = 299792458 \text{ m/s}$
f	Frequency
λ	Wavelength

Adjustment and variance calculation

σ^2	Variance
m	Mean
\underline{h}	Random variable containing points from map \mathbf{H} that lie in a local neighbourhood
\underline{y}	x and y coordinates of the point data
h_{rms}	The Root Mean Square (RMS) of the values in \underline{h}
$E\{\cdot\}$	Expectation value
\mathbf{A}	Design matrix for the Least Squares Adjustment
\mathbf{W}	Weight matrix
x	Vector with unknown parameters
\hat{x}	Vector with the Best Linear Unbiased Estimation for the unknown parameters
\hat{e}	Residuals after adjustment
$\gamma(\mathbf{d})$	Variogram over distance \mathbf{d} .
$z(r, c)$	Plane elevation at location r, c

Laser Intensity Correction

P_r	Received signal intensity at the sensor
P_t	Signal intensity transmitted from the sensor
G	Antenna gain
η	Efficiency of the antenna (η_{sys}) and efficiency of the atmosphere (η_{atm})
R	Range from the system to the scattering surface
σ^0	Surface backscattering coefficient
A	Area of the scattering surface
t_p	Pulse length of the laser scanner
θ_t	Opening angle of the laser bundle
θ_s	Scan angle, i.e. the angle of the beam with the vertical
θ_i	Angle of incidence of the laser beam

θ_r	Angle of reflection of the laser beam
ϕ_i	Azimuth direction of the incident laser beam
ϕ_r	Azimuth direction of the reflected laser beam
α	Attenuation coefficient
F	Constant multiplication factor for system parameters
$\mathbf{I}(r, c)_{spread}$	Intensity map with correction for spreading
$\mathbf{I}(r, c)_{corr}$	Intensity map with full correction

Mathematical Morphology

\mathbf{B}	Structuring element
$[\varepsilon_{\mathbf{B}}(\mathbf{X})](r, c)$	Erosion of raster map \mathbf{X} with structuring element \mathbf{B}
$[\delta_{\mathbf{B}}(\mathbf{X})](r, c)$	Dilation of raster map \mathbf{X} with structuring element \mathbf{B}
$\gamma_{\mathbf{B}}(\mathbf{X})$	Opening of raster map \mathbf{X} with structuring element \mathbf{B}
$\phi_{\mathbf{B}}(\mathbf{X})$	Closing of raster map \mathbf{X} with structuring element \mathbf{B}
$HMT_{\mathbf{B}}(\mathbf{X})$	Hit-or-Miss Transform of map \mathbf{X}
$BTH_{\mathbf{B}}(\mathbf{X})$	Top Hat filtering of map \mathbf{X}
\mathbf{X}^c	Complement of a raster map
t	Threshold

Crevasse shape

s	Sampling interval, point spacing
w	Crevasse width
d	Crevasse depth
β	Crevasse wall steepness in degrees
ΔH	Error in reconstructed crevasse depth

Acronyms

ALS	Airborne Laser Scanning
ALTM	Airborne Laser Terrain Mapper
ASTER	Advanced Spaceborne Thermal Emission and Reflection Radiometer
BRDF	Bidirectional Reflectance Distribution Function
BTH	Black Top Hat
DEM	Digital Elevation Model
ETRS89	European Terrestrial Reference System 89
GIS	Geographic Information System
GNU	GNU is Not Unix
GPL	General Public License
GPR	Ground Penetrating Radar
GPS	Global Positioning System
GRASS	Geographic Resources Analysis Support System
HMT	Hit-or-Miss Transform
ICESat	Ice, Cloud, and land Elevation Satellite
ICP	Iterative Closest Points
ILRIS	Intelligent Laser Ranging and Imaging System
ILWIS	Integrated Land and Water Information System
INS	Inertial Navigation System
Laser	Light Amplification by the Stimulated Emission of Radiation
LiSA	Lidar Surface Analysis
MSc.	Master of Science
OGC	Open GIS Consortium
OMEGA	Operational Monitoring System for European Glacial Areas
Pixel	Picture Element
Radar	Radio Detection And Ranging
RGB	Red, Green, Blue
RMS	Root Mean Square
SE	Structuring Element
SNR	Signal to Noise Ratio
SQL	Structured Query Language
TIN	Triangulated Irregular Network
TLS	Terrestrial Laser Scanning
UTM	Universal Transverse Mercator
Voxel	Volume Element
WGMS	World Glacier Monitoring Service
WGS84	World Geodetic System 84

1 Introduction

1.1 Motivation

This thesis is written as a part of the graduation project in the newly formed MSc. Geomatics programme at Delft University of Technology. Geomatics is the field that integrates the complete chain of geo-information processing, i.e. acquisition, processing, GIS analysis and finally presentation. The programme emphasizes on the application of geo-information in other professional fields.

In this thesis an attempt is made to fully incorporate the complete processing chain of Geomatics. Most emphasis lies on the processing of data, but database technology and visualization to the end-user are just as important. Also the laser data acquisition and specifically the influence on the quality are analysed.

The work is characterised by an interdisciplinary approach involving contributions from geodesists, GIS experts, geographers and glaciologists.

1.2 Background

Airborne Laser Scanning (ALS) is a relatively new method for the aerial acquisition of digital elevation models. It uses a scanner from a flying platform to acquire a high density collection of 3D points. While the technology is growing to maturity, the range of applications grows rapidly too. One application that has been identified is the monitoring of glaciers, which was up to now only possible with aerial photogrammetry or field visits.

The ALS process allows for a high degree of automation. It is the challenge to develop algorithms to acquire relevant properties of terrains automatically, so that these can be used by specialists such as glaciologists for further analysis. For the purpose of glacier monitoring several elements are of great interest. One of them is the delineation of the glacier, i.e. the border between glacier and bedrock. This is useful for building glacier inventories or for monitoring the changes of a glacier over time. Until now, the delineation used to be performed manually from orthophotos or from shaded relief views derived from elevation models.

A second feature worth monitoring is the crevasses in glaciers. Crevasses are cracks in the ice surface that form due to the dynamics within the ice. They can be deep and thus dangerous for travellers over glaciers. Apart from that there is not much known about crevasses because it is difficult to reach them over a glacier. Using the ALS data more insight about crevasses might be acquired by the glaciologists.



Figure 1.1 Mountaineers traversing a crevasse in the French Alps. (© Corbis)

1.3 Problem formulation and research objectives

Up to now the glacier delineation could only be acquired from aerial photographs, which were not regularly updated. Access to ALS data of glaciers opens new perspectives for frequently updating the glacier delineation. Especially for monitoring the melting of glaciers this data is extremely useful.

An accurate delineation is not only in the interest of researchers. Appendix A shows a cut-out of the *Alpenvereinskarte* covering the research area at the Hintereisferner. These maps are often used by mountaineers and hikers. The *Alpenverein*, an authority if it comes to mapping the Alps, updated this map for the last time in 2003. However, the last update for the glacier data was in 1997. Many other maps for this region even have glacier data that has not been updated for several decades. Considering the fast changes of glaciers this is a very long time.

It is anticipated that new ALS data covering the Alps will become available in the coming years. Governments are working on getting a full coverage with high resolution. Examples of finished and on going projects can be found in the Swiss and Italian Alps (Luethya and Stengeleb 2005; Wack and Stelzl 2005).

The second product that is of interest to glaciologist is a map of the crevasse locations. These maps appear to be useful for numerous purposes. One interested group are the cartographers from the Alpenverein. They might use this map to update the crevasse locations on a more regular interval. Even more interest for the detected crevasses has been shown by glaciologists. The shape, size and depth of the crevasses give much information about the depth of the glacier and the shape of the bedrock. The difference in crevasse area or volume between two epochs can be used to verify hypotheses concerning the speed of glacier movement. The reconstruction of individual crevasses can be an interesting product for some crevasses at specific locations.

These different objectives can be generalised to one broad objective:

To increase safety on and understanding of glaciers by setting up an automated system for glacier monitoring.

However, until now it is not known whether ALS data is a suitable method for this objective. Several attempts are made by different research groups (Baltsavias et al. 2001; Geist et al. 2003). However, little

was done for extracting geometry from the laser data. For other applications, such as the extraction of buildings (Haala et al. 1998; Maas 1999), the methods for processing ALS data is well known. This is also the case for the filtering of unwanted features and detecting break lines (Brügelmann 2000; Kraus and Pfeifer 1998; Vosselman 2000). Also, methods exist for the segmentation of laser data (Vosselman et al. 2004). Most of these methods work well, although not all of them can be implemented efficiently in a production environment. As glaciers and crevasses seem to be less complex than human made objects, the challenge is to detect and reconstruct glaciers and crevasses in a fully automated way for long-term monitoring. Whether this is possible or not is summarised in the research question of this thesis:

Is Airborne Laser Scanning data suitable for reliable and accurate monitoring of glaciers and crevasses?

1.4 Methodology

This graduation project is characterised by an interdisciplinary approach. Developing a method for the reconstruction of crevasses requires close cooperation with those who work in the fields of geography and glaciology. Being the end-users of the ultimately developed program, they are also one of the most important stakeholders within the project.

It is possible to subdivide the entire work in 4 different steps of well defined sub-projects. These steps are:

- Automatic delineation of the glacier
- Automatic detection of crevasse location.
- Reconstruction of crevasses
- Quality description

The data used in this project covers the Hintereisferner in Southern Tyrol and was acquired during the Omega project (Geist and Stötter 2004). Currently, this data is organised within a PostgreSQL database connected to the open source GIS package GRASS (GRASS Development Team 2006). All algorithms developed in this thesis are programmed in either Python or C and implemented as modules within GRASS. As is required by the license used in GRASS, these modules are released under the GNU GPL Open Source license (Free Software Foundation Inc. 1991). Together, these modules form now a complete and fully operational toolbox for glacier surface analysis. Figure 1.2 shows the workflow and the different products that can be produced with this newly developed toolbox.

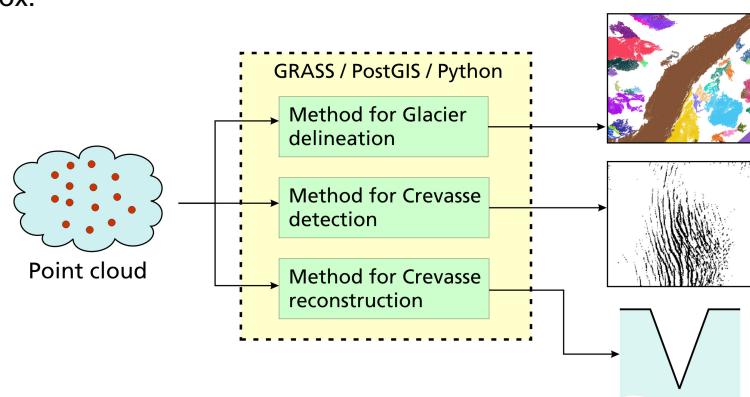


Figure 1.2 Workflow of the developed glacier surface analysis toolbox.

1.5 Outline

Because of the interdisciplinary character of this thesis, some aspects of the disciplines involved will be explained thoroughly in the following chapters. First, a short but comprehensive introduction to the field of glaciology is given in chapter 2. It contains a short overview of the history of this discipline and some essential explanations concerning glaciers in general and crevasses specifically. The geodetic background of the work is described in chapter 3 which is entirely devoted to the data acquisition. It describes some basic principles of Airborne Laser Scanning, the research area in Austria and other datasets that were available for this project.

Chapter 4 focuses on the delineation of the glacier. It gives a short overview of the criteria that are involved in this classification problem and treats these one by one in order to find the best possible implementation. The delineation procedure was tested on ALS data of the Hintereisferner, which is also presented in this chapter. The 5th chapter deals with the problem concerning the detection and reconstruction of crevasses. A method for crevasse detection is outlined and tested on the data. Using data from a number of individual crevasses, several ways to reconstruct these are presented. Chapter 6 follows by analysing several quality aspects that play a role in the methods presented in this work.

The last chapter completes the thesis by giving conclusions and recommendations for future work. More information concerning this thesis is incorporated in the appendices, which are referred to in the text. The thesis comes with a cd-rom containing the source code of the developed GRASS modules.

2 Introduction to glaciology

This chapter provides some background information on glaciology, as far as it is related to this thesis. Glaciology is the study of the properties and occurrences of snow and ice on the earth's surface, usually focussed on active glaciers. Only those glaciological aspects that are most relevant to this project will be treated, for a complete overview about glaciology, the textbooks written by (Paterson 1994) and (Hambrey and Alean 2004) are recommended.

2.1 Objectives and history of glaciology

Over 10% of the land surface of the earth is covered by glaciers: huge layers of slowly moving ice, measuring tens of metres thick. From this enormous glaciated area, approximately 97% lays on Antarctica and Greenland. In fact, these regions are almost completely covered by glaciers. Outside the Polar Regions most glaciers can be found in high mountain areas. In Europe, we find glaciers in Iceland, Norway, the Pyrenees and the Alps. These glaciers are commonly called temperate climate glaciers or mountain glaciers.



Figure 2.1 Distribution of glaciers around the world. White areas show ice sheets, white dots are glaciers. Note that area-scale is distorted (Source: U.S. Geological Survey 2005).

The fascinating sights and environment of glaciers makes them very valuable for all sorts of tourism, such as summer skiing and hiking. For countries such as Austria, glacier related tourism has become essential for the local economy in both summer and winter. As glaciers react rapidly to varying climate conditions, they can also help us understand other phenomena, such as global climate change and its effects on our environment. By monitoring glaciers, the local impact of changes in the climate can be investigated and placed in a historic context using long series of acquired data. Comparing glacier fluctuations allows for

deriving independent estimates of global warming over the last 100 years (Oerlemans 1994).

The large white icecaps of glaciers have attracted the interest of researchers already for two centuries. The first publications describing some physical properties of glaciers were written by adventurous scientists, travelling over glaciers all over the world. James David Forbes (1845) and John Tyndall (1860), who visited some Alpine glaciers in the 19th century, are most famous for their early publications in glaciology. Additionally, a lot has been written in the diaries from expeditions to glaciated areas, most noticeably Antarctica. Nowadays, glaciology still requires a lot of fieldwork, still accompanied by the same life-threatening risks as many years ago. Especially the deep cracks in glaciers are something to be very careful about. Fortunately, new techniques in the field of Earth Observation have made it possible to acquire large datasets with highly accurate information from airborne or space borne systems, leaving the actual investigation to be done safely in the office.

2.2 Birth of a glacier

In the climatically temperate regions on earth, glaciers will only form on high altitudes, where there is enough snowfall and the temperatures are low. The main condition for forming a glacier is that some of the snow that has fallen in the winter lasts throughout the following summer. If this is the case, the snow will accumulate over successive years. Under the high pressure of the snow that has accumulated, the oldest snow will slowly transform to ice. It can take up to 5 or 10 years before the loose snow has been fully transformed to ice.

During the transformation from snow to ice, the material progresses through several intermediate stages. First, the snowflake crystals break due to the weight and gradually change to grains. It also becomes harder and denser. In this stage the snow is called *firn*. Firn can be identified on a glacier between the locations with glacier ice and fresh snow. After a while, the grain-like particles in the firn start to crystallize again, but this time larger crystals form, resulting in solid ice. In this stage, almost no air is left in the ice, except for some small bubbles where air is trapped. The shape of ice crystals will change continuously, caused by the slow movement of the glacier.

Due to the movement of a glacier, this transformation over time is also visible spatially. Starting at the snout of the glacier, the oldest ice will appear. When travelling upwards, the ice becomes younger. At a certain point, the surface ice consists of firn, while at even higher altitudes the snow can be found. For Earth Observation applications this is a relevant aspect as each type of ice has its own absorption and reflection characteristics. A simple example of differing absorption characteristics is the blue colour of very old ice, caused by the high absorption in all but the blue parts of the spectrum. Lutz et. al. (2003) describe an investigation of laser scanning signal intensity on a glacier surface. It was found that differences between snow, ice and firn could clearly be identified from the backscattered intensity. Comparison with orthophotos revealed that the backscatter characteristics of infrared light, as used in Airborne Laser Scanning, are different from visible light.

On the top of the glacier, there is a net growth of the glacier ice. This area is called the accumulation area. Downhill temperatures are higher,

causing more melting and less snowfall. In this area, known as the ablation area, the glacier loses ice. Each glacier has an equilibrium line where the gain and loss of ice are equal. This line forms the border between the accumulation and ablation areas. Knowing the position of the equilibrium line between accumulation and ablation gives information on the current state of a glacier, for instance whether it is growing or retracting.

2.3 Glacier movement and crevasses

In the Alps, the ice layer of a glacier can be up to tens of metres thick; in the arctic regions even up to hundreds of metres. This layer doesn't remain fixed at its position, but is continuously on the move. The movement is a result of gravity forces acting on the ice shelf; hence glaciers are often called 'rivers of ice'.

The speed is not uniform within the layer, but differs with depth and width. The speed is lowest near the bedrock at the sides of the glacier, resulting in a speed pattern as shown in Figure 2.2. For the longitudinal profile, similar aspects hold: the speed is highest at the top layer, while lower near the bedrock (Figure 2.3). The lowest layer of ice literally slides over the bedrock surface, creating the typical smooth surface that can be seen when a glacier has melted.

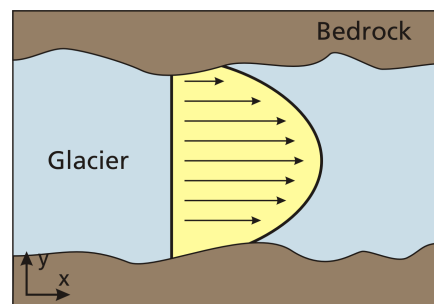


Figure 2.2 Surface velocity of a glacier

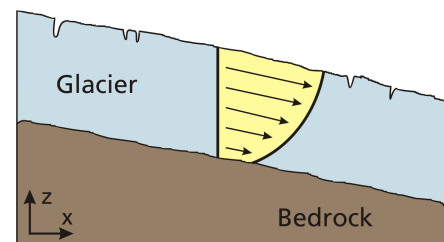


Figure 2.3 Vertical glacier velocity

Crevasses are cracks in the ice surface of a glacier, usually with a V-like shape. Their depth can range up to tens of meters, while their width is a couple of meters. Crevasses can be very nice to look at, for instance because of their bright blue colour. However, crevasses can also be dangerous, especially for mountaineers. Their enormous depths, combined with the fact that crevasses can be hidden by a large pack of snow, makes them very dangerous for anyone on a glacier (Figure 2.4).

Crevasses occur on all glaciers due to the continuous dynamics within the glacier. The glacier slowly moves downhill under gravity forces, but depending on the steepness of the terrain, the speed of the ice bed can change. Also, if the width of the valley becomes wider or narrower, the ice will extend or compress spatially. Due to the changes in speeds of the ice within the glacier, tension will occur within the ice. As ice is brittle, it will crack if the tension becomes too high. These cracks are the typical crevasses that can be found on glaciers.

In old stories, crevasses were mentioned as being bottomless. This is not the case, only the upper layer of approximately 30 meters can brake. Lower laying ice has become plastic by the very high pressure due to the weight of the overlaying snow and ice. Fractures cannot propagate into the plastic ice layer.



Figure 2.4 Crevasse as an obstacle to glacier travel on the Griesgletscher, Canton Valais, Switzerland (Source: Hambrey and Alean 2006).

Crevasses usually form in sets of parallel cracks with their orientation depending on the location in the glacier (Klostermann 1999). Figure 2.5 shows schematic example of how these crevasses form. However, not all cracks in a glacier are really crevasses formed in the way as described above. At the side of the glacier, along a rock wall, one can often find a so called *randkluft*. Because the rock absorbs radiation, the rock warms up thereby melting the ice along the rock. This forms cracks along the outline of a glacier.

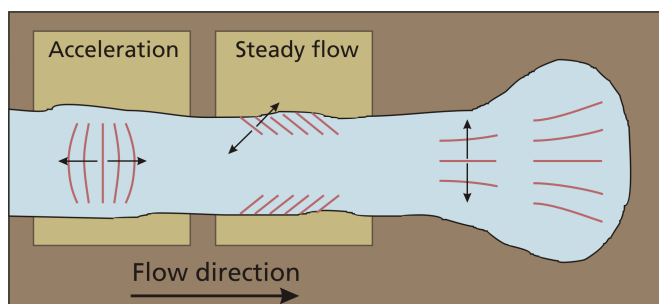


Figure 2.5 Location and formation of crevasses. The arrows indicate the directions in which the ice is pulled apart.

2.4 Measurements on glaciers

Glaciers have already been monitored for a long time. For some glaciers, the World Glacier Monitoring Service (WGMS) has stored data since half a century (WGMS 2006). One interesting parameter for glaciologists is the Mass balance of a glacier, representing the growth or decay of a glacier. Traditionally, determining the mass balance is a labour-intensive

task, done by drilling stakes into the ice over a longitudinal profile. These stakes are then being surveyed over the year. The positions of the stakes tell something about the movement of the glacier. The height of the stake above the ice surface, gives information about the accumulation or melting of ice. Some disadvantages of this method include the huge amount of work required for placing the stakes, potential dangers for those travelling over the glacier and the fact that the absorption of heat by the material used causes additional melting of the glacier, thereby influencing the measurements. Using measurement methods from airborne platforms, such as digital photogrammetry or laser scanning, these measurements can be made much more efficient.

On some glaciers, GPS receivers are placed on fixed positions in the ice. The receivers move together with the glacier, providing valuable information about the glacier movements.

A European Union funded project, Operational Monitoring System for European Glacial Areas (OMEGA), works on permanently monitoring the European glaciers. One of the methods they investigate is Airborne Laser Scanning (Geist and Stötter 2004). Two glaciers, the Sartisen Ice Cap in Norway and the Hintereisferner in Austria, have been measured with Airborne Laser Scanning over several years. The data is placed in a PostgreSQL database, connected to the open source GRASS GIS package. After interpolating this data to a grid, lots of interesting information can be withdrawn from the data. Crevasses for instance are very well visible. Also debris and ice boulders can be seen (Geist and Stötter 2003).

Other research projects use satellite imagery for monitoring glaciers. Whillans and Tseng (1995) have developed a method for automatic tracking of crevasses in satellite images. From 10 meter resolution SPOT images covering a part of Antarctica, the movement of the glacier has been determined by cross-correlating the crevasses in the image. In fact, the crevasses act like moving markers in this case.

Ground Penetrating Radar (GPR) is another measurement method. GPR works by sending Radio waves into the ground. The signal reflects due to heterogeneities and can then be received again. By interpreting the backscattered signal, information on different ice layers can be derived, as well as information on the total thickness of the ice shelf. In fact, the first GPR survey ever in 1929 was performed on a glacier (Olhoeft 1996; Stern 1930). On a regular basis, booklets are published with recent GPR measurements on some of the larger European glaciers (WGMS 2006). A GPR device can also be placed on a rod before a vehicle as a tool for detecting snow-covered crevasses. The aim of these measurements is merely to guarantee safety to those travelling on the ice, rather than mapping the actual location (Delaney et al. 2004).

3 Data acquisition

In the previous chapter several aspects of glaciology were introduced and explained. Monitoring of glaciers is an ongoing activity that can benefit greatly from Airborne Laser Scanning (ALS) data. In Austria several ALS datasets covering a glacier are available. This chapter introduces ALS and how it was used to measure the glacier. Also other means of data acquisition on the glacier are addressed shortly.

3.1 Principles of Airborne Laser Scanning

ALS has become a very efficient method for digital terrain modelling and is commercially available now for more than a decade. Placed in an airplane, helicopter or any other platform, the system can accurately measure the height of the terrain. Figure 3.1 shows a typical setup for an ALS measurement campaign. The ALS system is mounted on an airplane that flies over the area of interest. The scanning system sends out pulses in a wide swath underneath the platform. These pulses, infrared or visible light, scatter on the surface and can be detected by the system again. The travel time gives the distance between platform and terrain. The

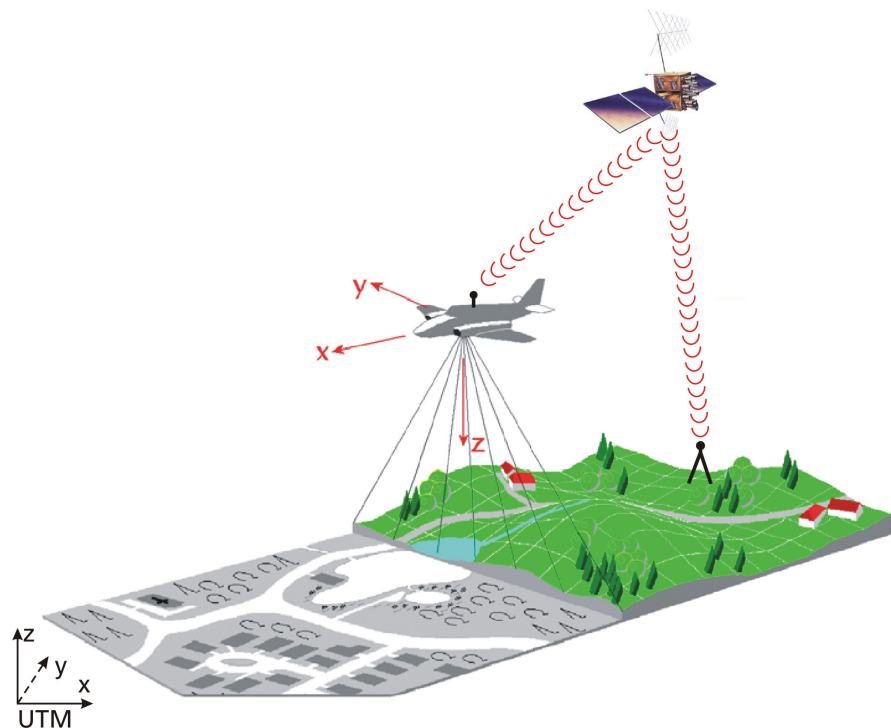


Figure 3.1 Typical ALS campaign setup. The airplane position is determined with GPS/INS using differential GPS. Note that at least 4 satellites are required for GPS processing.

The measurements can only be related to terrain coordinates like a UTM projection if the position and attitude of the platform are accurately known. For this, a GPS/INS system is placed on the platform. With INS (Inertial Navigation System) accelerations and angular change rates are continuously measured. After integrating these measurements over

time, the platform position and attitude can be found. Because INS works at a very high frequency of approximately 50 Hz, the trajectory of the platform can accurately be reconstructed. However, as INS is a relative measuring system based on integration over all previous observations, the observation error increases rapidly with time. Differential GPS (Global Position System) is therefore used to update the INS measurements with accurate absolute values for the platform position. The use of differential GPS requires the availability of one, but preferably more, stationary GPS receivers at known locations, like indicated in Figure 3.1.

The result of an ALS campaign is typically a point cloud: a data set with individual points representing the surface at a certain location. These points can be used for further processing, for instance by interpolating them to a regular grid, a Digital Elevation Model (DEM), noted as $\mathbf{H}(r,c)$ with $r=0\dots M-1$ and $c=0\dots N-1$ as row and column coordinates. When operated from an airplane, the absolute height accuracy for the individual points is about 10 to 15 cm. The precision or relative accuracy, this is the error in the height difference between two nearby points, will be considerably better.

The laser beam that is send out by the system is very narrow: the opening angle lies in the order of milliradians. When flying on an average height of 1000 m, the footprint of the beam on the glacier surface is approximately half a meter. On relatively flat surfaces one can assume that the measured elevation is the mean height for the surface under this footprint. However, in the case of a surface with distinct features or objects on it, there will be multiple returns from this single beam. Figure 3.2 and Figure 3.3 illustrate this for two different situations that can occur on a glacier. When crevasses appear in the glacier, the first pulse reflects on the ice surface. As the footprint of the beam covers the crevasse only partly, the remaining part of the signal will descent into the crevasse. Here it reflects on the wall of the crevasse, causing a stretched return signal.

A similar situation occurs with rocks on the glacier surface, a situation that occurs quite often. In this case there will be two distinct reflections: one on the rock and a second one on the glacier. This is received by the scanner as a first pulse and second or last pulse, both of which are stored by the system (Wagner et al. 2003). It is important to note that the separation between multiple echoes is only possible if the returns are at least one pulse length apart from each other. Assume a pulse length of 1 ns, then taking the speed of light into account, gives that the returns should be 0.3 meters apart from each other. Rocks that are smaller than 0.3 meter can therefore not be recognised from the multiple returns.

Most laser scanning systems supply the X-, Y- and Z-value of both the first and last pulse. Current research involves storing the complete shape of the returned signal: full waveform capturing (Wagner et al. 2003; Wagner et al. 2004). For our application of detecting glaciers and crevasses, we are particularly interested in points that lie on the ice surface or in crevasses. Therefore, the interpolation to a DEM was performed with only the lowest points, i.e. the last pulses.

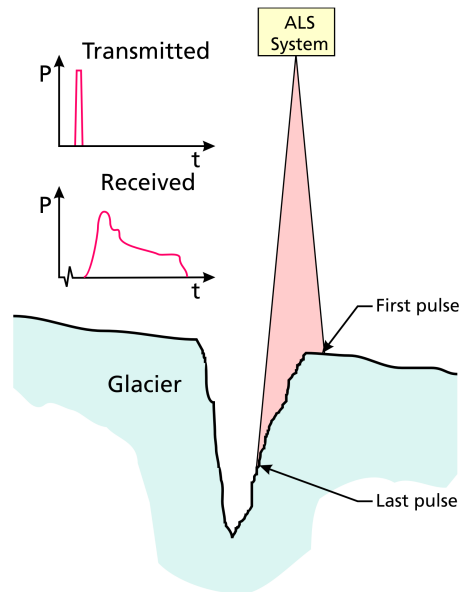


Figure 3.2 Multiple returns at crevasses. The graphs show the transmitted and received power.

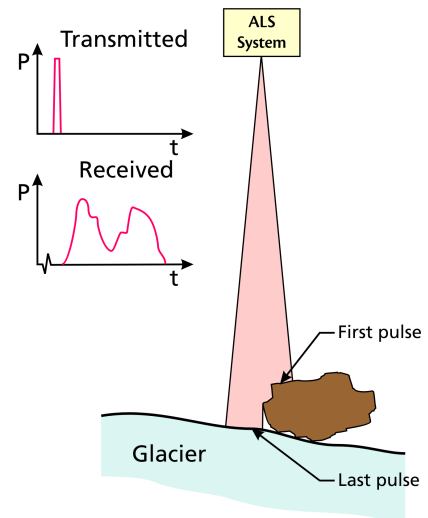


Figure 3.3 Multiple returns when debris lies on the ice.

3.2 ALS mission design

Designing an ALS mission on alpine glaciers is not always trivial, as was nicely described by Baltasvias et al. (2001). The high mountain features constrain the area of operation for the airplane. Additionally, rapidly changing weather and large elevation differences pose a real challenge to the pilot.

The acquisition of the datasets on the Hintereisferner was performed with the Optech ALTM 1225 Laser Scanner, although in some flights a newer edition of the scanner was used (Figure 3.4). The key parameters of the ALTM 1225 scanning system are given in Table 3.1.



Figure 3.4 The Optech ALTM system components. From left to right: control rack, pilot navigation aid, processing software and laser sensor (Source: Optech Inc.).

Table 3.1 Optech ALTM 1225 System Parameters

Operating altitude	410 - 2000 m above ground level
Z-accuracy	15 cm at 1000 m, 25 cm at 2000 m
Laser pulse rate	25 kHz
Swath width	$\pm 20^\circ$ from nadir
Scanning frequency	28 Hz at the 20° scan angle
Beam divergence	0.2 milliradian (half angle)
Footprint	40 cm at 1000 m, 80 cm at 2000 m
Laser wavelength	1024 nm
Laser classification	Class IV laser product

The flights were performed at a mean height of 900 meters above surface with an average point density of 1 point per meter squared. Except for the overlap between scanning strips, all areas were scanned only once. Figure 3.5 shows a plot with the strips as they were flown for one of the datasets. The strips were connected to each other using a strip adjustment based on the points in the overlapping areas. The laser scanning results were checked using a control area in Zwieselstein, the nearest place that is flat enough for having a football field, at about 20 kilometres from the glacier.

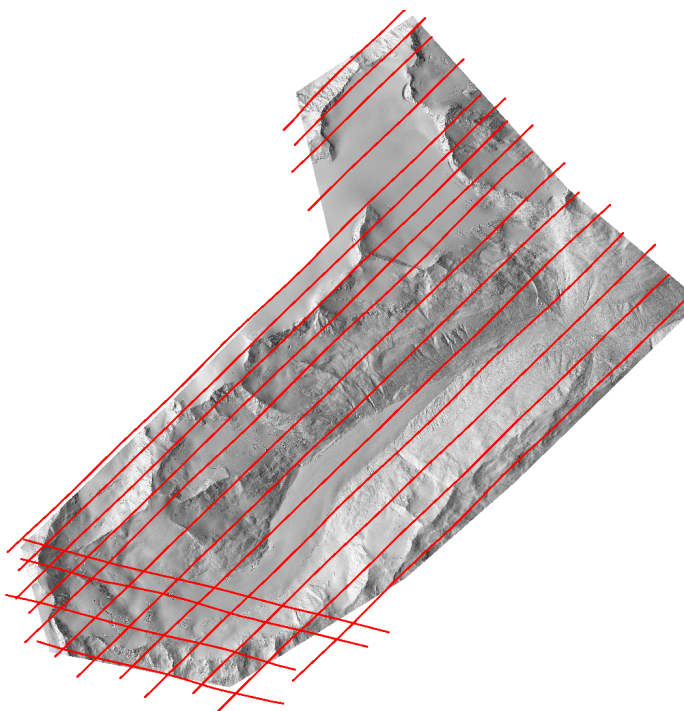


Figure 3.5 Laser scanning trajectory as it was flown in 2004.

3.3 The Hintereisferner datasets

In section 2.4 the OMEGA project was introduced. Within this project two European glaciers were selected as study sites for developing and testing operational monitoring systems. One area was the Engabreen glacier within the Svartisen ice cap located in Northern Norway. The second glacier was the Hintereisferner in Tyrol, Austria. The Hintereisferner is a typical valley glacier located in the Ötztal, just on the border with South-Tyrol in Italy (Figure 3.6). The Hintereisferner was selected for the OMEGA project because monitoring data in the form of maps and photographs for this glacier dates back to the late 19th century. There are few other glaciers with such a long record of

monitoring activity. The laser scanning data acquired above the Hintereisferner and some smaller neighbouring glaciers, the Vernagtferner and Kesselwandferner, was used for this graduation project. A map of the area is included in appendix B.



Figure 3.6 Valley in the Ötztal with the tongue of the Hintereisferner. (Image courtesy of Thomas Geist)

In total, twelve epochs of laser scanning data are available for the Hintereisferner. These datasets were acquired between October 2001 and October 2005 in different seasons of the glaciological year. The flight dates are represented in Figure 3.7. The red line in this figure shows the amount of snow that can be found on the glacier. For crevasse detection especially the flights taken in the late summer are useful as the snow cover will be least in this time.

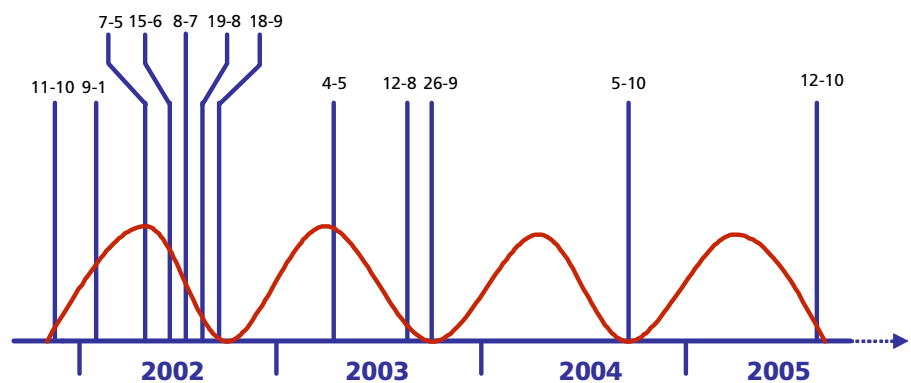


Figure 3.7 Dates of the twelve laser scanning flights over the Hintereisferner. The red line schematically represents the snow height on the glacier. (Image adapted from Thomas Geist)

Most of the flights were performed with the Optech ALTM 1225 Laser Scanner. The average point distance was planned to be either 1.5 meter or 1.0 meter. For three of the 12 missions, the full information of the points, i.e. values for first pulse, last pulse and intensity, is stored in a PostgreSQL database that can be connected to the GRASS GIS (Höfle et al. 2006). The other data sets were not yet added to the database at the time of writing.

Additionally, the data was transformed to a one meter resolution raster using a nearest neighbour interpolation method. Possible gaps in the

raster due to missing laser points within the cell area can be resolved using neighbourhood filters like a mean or median filter. Most of the processing described in this report will be done based on these raster files. For visualization of the elevation model the calculation of shaded relief views is very suitable. In these views the elevation information is used to calculate grey scale values that give a sense of the appearance of shadows due to relief. Figure 3.8 shows an example of a shaded relief view displaying the lower part of the glacier.

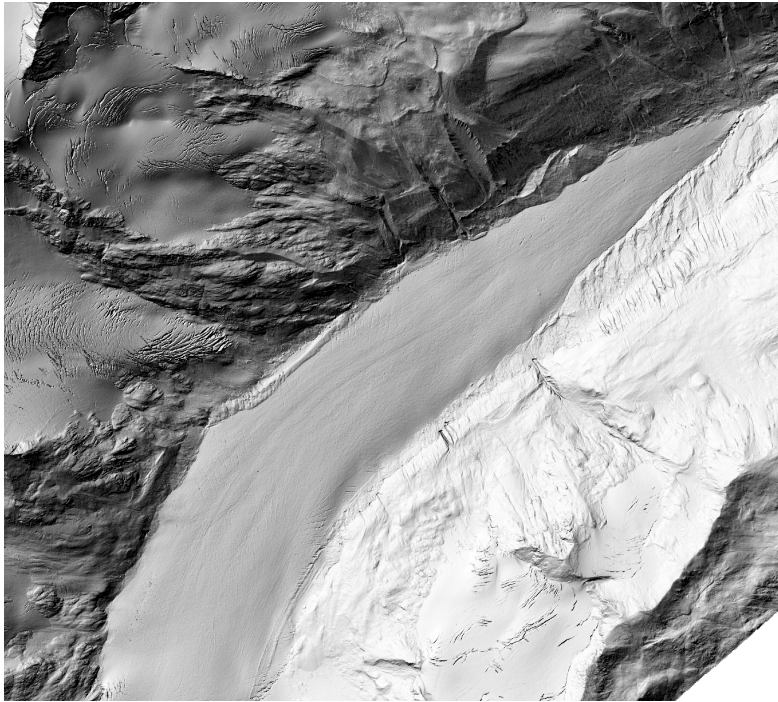


Figure 3.8 Shaded relief view of the laser scanning data. The image shows the tongue of the glacier lying in a steep valley. On the left side the Langtaufferer-Joch-Ferner is visible with some very clear crevasses.

The data was originally acquired in ETRS89 and afterwards projected into the UTM system in zone 32. It is important to note that all elevations are represented as ellipsoidal heights with respect to WGS84 instead of the local geoid. For visualisation purposes and local data analysis this will be fine, but large scale data analysis, such as hydrologic modelling, should be performed with great care. Figure 3.9 gives an impression of the difference between the geoid and an ellipsoid. When the data is not referred to a geoid, one can not be sure that lower points in the data set also represent the direction where water flows in reality. Preferably, the data should be corrected with elevations from the local geoid model. However, for Alpine countries the geoid model is hard to acquire and thus not always available with high accuracy.

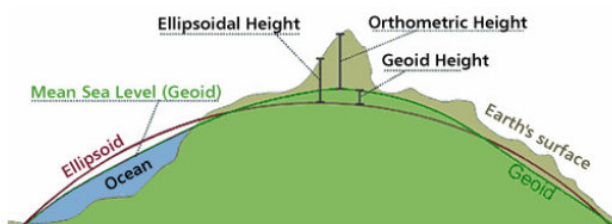


Figure 3.9 Difference between geoid and ellipsoid.

3.4 Other datasets available

The objective of the graduation project was to process ALS data for glacier monitoring purposes. However, besides the DEMs derived from the ALS data, other datasets of the specific area are also available. This includes the intensity data from ALS, orthophotos and terrestrial laser scanning data (TLS).

3.4.1 Intensity images

When the energy of the laser beam emitted by the laser scanner hits the surface, a part of the energy will be absorbed or transmitted. How much energy is reflected depends on surface characteristics such as the colour. The reflected energy is received at the laser scanner again and stored together with the elevation data.

Although often considered as a side-product, the intensity of the reflected laser signal can be a useful complementary data source to be used in further processing or visual analysis. If the received intensity is rasterised and processed, it can be used like a black/white photograph. Figure 3.10 is an example of such an intensity image. As a part of the thesis project a process was designed and implemented in GRASS for generating visually appealing intensity images from laser scanning data. These images may be used as substitutes for traditional aerial photographs. The resulting images will appear in greyscale, but for ice surfaces this is probably not much of a problem.

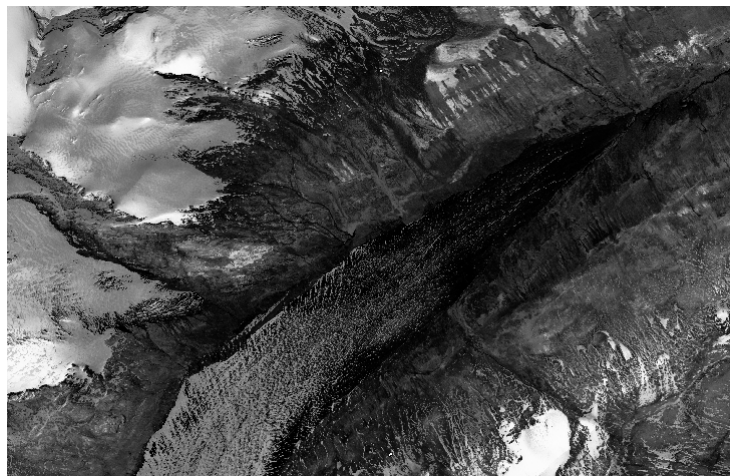


Figure 3.10 Laser intensity image of the glacier snout.

The regular output from an ALS-system after some first post-processing comprises the following elements for each transmitted laser pulse:

- time;
- coordinates first pulse;
- received intensity first pulse;
- coordinates last pulse;
- received intensity last pulse;
- airplane position.

The received intensity I is scaled in the range [0,255] and therefore doesn't show a real physical quantity:

$$I = a_0 \cdot P_r + a_1 \quad (1)$$

In equation (1) a_i are scaling parameters and P_r is the received power in Watt. After interpolating the intensity values to a raster, a greyscale picture appears. The only difference with a real photograph is that the light source for the image was situated in the airplane, while the normal energy source is the sun.

It appears that, especially in mountainous regions, a correction for atmospheric effects has to be applied. Section 4.5 describes a method for applying this correction.

3.4.2 Terrestrial Laser Scanning and Ground Truth

In August 2006 an expedition to the Hintereisferner was organised. The team comprised members from the Department of Geography at the University of Innsbruck and members from the Department of Earth Observation and Space Systems at Delft University. The long range Optech ILRIS 3D Terrestrial Laser Scanner was brought to the glacier to obtain high accuracy measurements of the surface (Figure 3.11).



Figure 3.11 Expedition to the Hintereisferner. Expedition members and ILRIS 3D Terrestrial Laser Scanner.

Apart from a visual assessment of the state and appearance of the glacier, the expedition made it possible to get high accuracy measurements of specific features on and near the glacier with the laser scanner. Figure 3.12 shows the point cloud that was measured near the snout of the Hintereisferner. It features a part of the 'dead ice' that lies right before the Hintereisferner. This is ice that was broken apart from the glacier and is no longer influenced by the glacier dynamics. The melt water of the glacier has cleared a way through the ice and continues as a wild stream. The point cloud shows no reflected points on this water. The colours in the plot represent the received signal intensity. Optionally, RGB values can be attached to the scan from the picture that was taken with the built-in 6 MegaPixel camera.

Another measurement was taken on the glacier surface to get some ground truth for the variance and reflectivity of the surface. The scan covered a part of the glacier and a part of the neighbouring bedrock. However, due to the very grazing angle of the laser beams on the surface, and the general bad reflectance properties of the glacier ice, only a few points were actually returned from the glacier surface.

Additionally, only points on the higher parts of the glacier surface were measured as the lower points were occluded. Figure 3.13 shows the shaded relief view of the glacier and the locations of the terrestrially scanned points. Because the TLS measurements had no absolute orientation, they were transformed to the UTM system by fitting them to the DEM using the Iterative Closest Point method (ICP) (Besl and McKay 1992).

It was impossible to get reliable information about the variance of the terrain because the scanner only scanned the higher points of the glacier. However, the data can be used in future research for gaining more knowledge about the interaction of laser beams with ice.

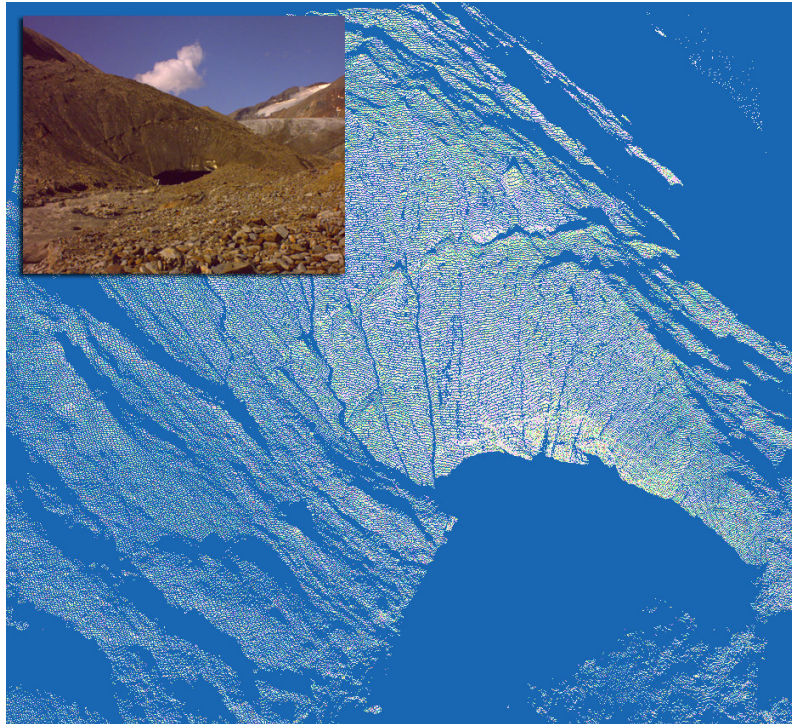


Figure 3.12 Terrestrial Laser Scanning measurement near the snout of the Hintereisferner. The colours represent returned intensity.

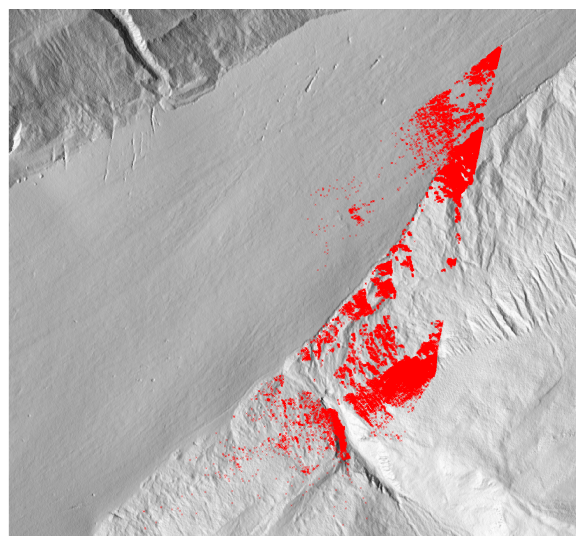


Figure 3.13 TLS measurements on the glacier

4 Glacier delineation

In chapter 2 glaciers and glacial features such as crevasses were introduced. The method of data acquisition and the available data sets were described in chapter 3. This chapter, as well as chapter 5 treats the actual use of Airborne Laser Scanning data for glacier surface analysis.

In the introduction the need for a good delineation of a glacier was formulated. The delineation, basically a map showing the glacier and non-glacier surface, can be used for glacier monitoring and alpine cartography. In this chapter a method for the automatic derivation of the delineation from Airborne Laser Scanning data is presented.

4.1 Methodology for delineation

The ALS data can be organised in two ways: as one huge point cloud or interpolated to a raster based Digital Elevation Model (DEM). For the derivation of the delineation, all processing is performed on the raster based data. This gives a considerable improvement in speed and simplicity, while the loss in accuracy is acceptable because of the intrinsic inaccuracy in the glacier topography due to its continuous dynamics.

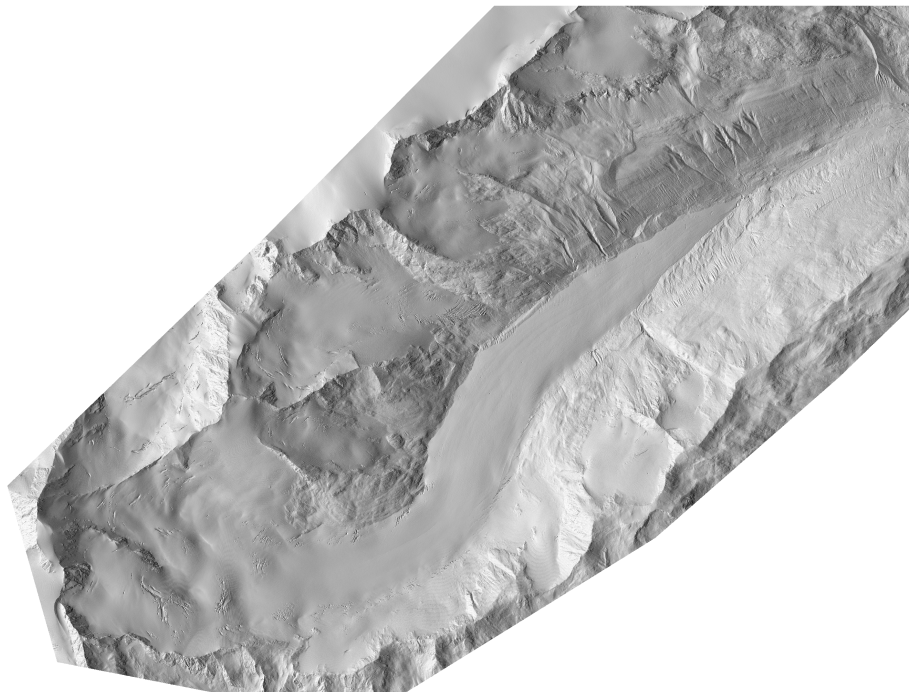


Figure 4.1 A shaded relief view of the ALS data from October 2004.

The exercise of determining the delineation is a classification of the pixels between the glacier and non-glacier classes. The process of classification is well known from Remote Sensing where it normally involves the analysis of multispectral image data and the application of statistically based decision rules. This spectral data is now absent, but other criteria can be developed for the decision rules. For this we revert to the formal definition of a glacier. By definition, glaciers are large

masses of ice set in motion by the Earth's gravity. Additionally, the definition states that the ice mass should have a minimum size of 0.1 km² and the ice should be perennial, i.e. lasting three seasons or more. From this definition a couple of criteria for classifying the data can be derived.

Criterion 1: Smoothness

The most important criterion that can be postulated which applies to ALS data concerns the surface characteristics that can be derived from the elevation data. The ice surface that makes up the glacier is much smoother than the surface of the surrounding bedrock. The smoothness criterion is also apparent when looking at the shaded relief view of the DEM. The shaded relief view (Figure 4.1), technically an image showing the gradient in diagonal direction, clearly gives the appearance of a smooth surface on the glacier.

Criterion 2: Connectivity

Additionally to the smoothness criterion which was derived from the elevation measurements, the classification can be extended with what is known in Remote Sensing as spatial pattern recognition. This involves the classification based on the shape of the objects. For glaciers this might appear to be quite problematic, as glaciers usually have rather irregular and varying shapes. However, from the definition it follows that glaciers are always single ice fields. Two separate bodies of ice cannot constitute one single glacier. Geometrically this means that all pixels that belong to one single glacier should be connected to each other.

Criterion 3: Hydrological constraints

Glaciers are sometimes referred to as "Rivers of Ice". Although this analogy is not entirely true, glaciers indeed do obey the hydrological principles. That glaciers, like rivers, always flow downwards is obvious. Consequently, the notion of a catchment area also applies to glaciers. A catchment area is the area in which all water, ice or snow, flows to the same single outlet. A river can never cross the border of the catchment area, and so can't a glacier. Any pixel classified as glacier should therefore lie within the catchment area of the glacier (Figure 4.2).

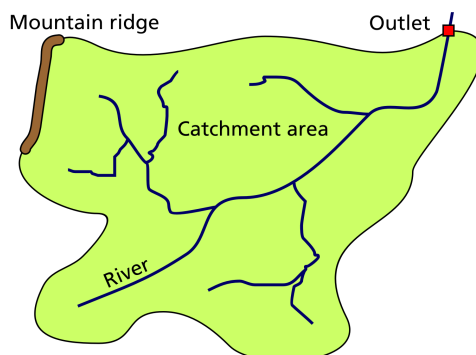


Figure 4.2 The concept of a catchment area

Criterion 4: Laser intensity

Laser Intensity images were already introduced as a side product of ALS in chapter 3. It is a reasonable expectation that the bright ice surface has a higher reflection than the darker surrounding bedrock. A classification can be based on these characteristics, although this requires a correction for atmospheric effects on the intensity. However, reflection also depends on many physical properties of the surface, including debris cover, moisture and roughness. This makes it hard to build an automatic classification on intensity. Alternatively, the data can be used for manual inspection of the result.

Criterion 5: Slope and aspect

Lastly, two other parameters could be relevant for the classification. These are the slope and aspect of the terrain. Slope is defined as the steepness of the terrain in degrees, aspect the direction that the terrain faces with respect to the north. Both parameters can be calculated for each pixel and combined with the smoothness criteria they can make up a three dimensional feature space. However, as it appears, the correlation between these two parameters and the pixels that belong to the glacier is rather weak: there is no direct relation between slope, aspect and pixels belonging to a glacier. Even when both parameters are calculated for patches of 11 by 11 meters, thereby cancelling small scale relief influence, the correlation remains small. For this reason, these parameters are not further considered in this report.

Experiments showed that the classification based on smoothness gives the best result for delineating the glacier. The other criteria can be used to improve the result of the smoothness based classification. In the next section the methodology behind the smoothness based classification will be described. Two different methods are presented

4.2 Smoothness based classification

In this section the use of the first criterion is investigated. Smoothness is a qualitative description for a surface of a terrain. In order to base a classification on smoothness a geometrical quantification of this property should be developed. The dictionary definition of smooth refers to properties like "free of irregularities and roughness". Geometrically we could say that within a small window a smooth surface looks like a plane, which is perfectly smooth by definition. This also means that within a small window, the first derivative of the surface signal should be constant.

For classifying between glacier and non-glacier surface, these properties of a smooth surface can be used. Either we look how well a plane fits in the surface by looking at the variance after fitting, or we search for areas where the first derivative of the surface is constant. Classification can be performed by searching the most appropriate class for each pixel individually, followed by a connected component labelling. Alternatively, the pixels can be segmented first and then classified on the basis of properties of the segments (Figure 4.3).

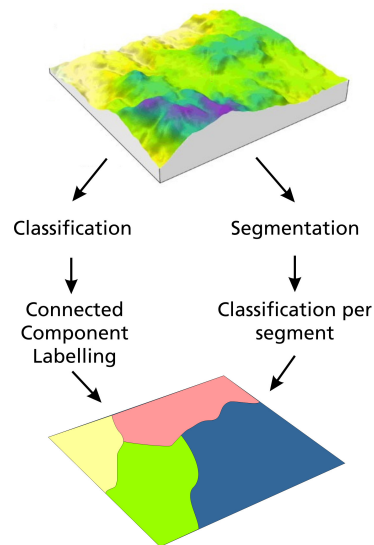


Figure 4.3 Two approaches for classification.

4.2.1 Variance of plane fitting

Figure 4.4 shows how a small part of the surface is represented by a plane. If the surface is very similar to the plane, it can be called smooth and is therefore likely to be part of the glacier. The variance of the difference between a plane and the surface, i.e. the variance of the residuals after fitting a plane, can be used as a measure for smoothness.

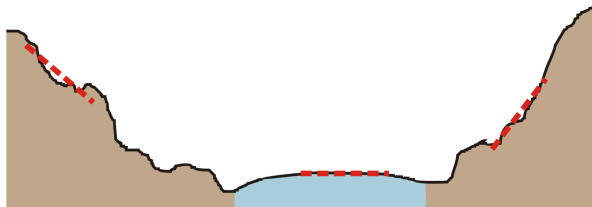


Figure 4.4 Smooth surfaces, such as glaciers, can locally be represented with a plane.

The variance is calculated within a moving window over the elevation model. The glacier surface is then selected by thresholding at a certain variance level. Let a window of arbitrary shape and size covering n cells move over the entire elevation model. Let \underline{h} be a random variable containing the elevations within the window from the DEM \mathbf{H} . The variance of these points around the mean elevation m is then defined as:

$$\sigma^2 = \frac{1}{n} \sum_{i=0}^{n-1} (h_i - m)^2 \quad (2)$$

This measure is also known as the second statistical moment. However, for our application the variance as such is not really useful. It is known that glaciers can occur on very steep slopes, while this measure only gives the variance with respect to horizontal slopes. In fact, equation (2) is not a good measure for smoothness. This is because the signal is not second order stationary, a condition which states that the mean of the signal should be constant. We can achieve this condition by first fitting a

plane through the points and then look at the variance with respect to this plane, i.e. the variance of the residuals.

The plane $z(r, c)$ that will be fitted through the points is defined as:

$$z(x, y) = a_0 + a_1 r + a_2 c \quad (3)$$

The fitting was performed with the least squares adjustment. The points within the window have an elevation \underline{h} and are expected to lie on the plane. When the plane parameters are put in x and the model parameters in matrix \mathbf{A} , the plane equation can be written as:

$$E\{\underline{h}\} = \mathbf{A}x \quad (4)$$

with

$$\mathbf{A} = \begin{pmatrix} 1 & r_1 & c_1 \\ \vdots & \vdots & \vdots \\ 1 & r_n & c_n \end{pmatrix}; \quad x = \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix}$$

where n is the number of points in the window. The problem can be solved with the well known least squares solution, assuming an identity matrix for the variance-covariance matrix, i.e. giving all points equal weight. The best fitting plane though all these points is described by the parameters for which values are estimated with the Least Squares Adjustment.

$$\hat{x} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \underline{h} \quad (5)$$

The residuals after fitting the plane follow from

$$\hat{e} = \underline{h} - \mathbf{A} \hat{x} \quad (6)$$

Now, the data is second order stationary, because by definition the mean of \hat{e} is zero, provided that the fitted plane only represents a small area of the surface. From \hat{e} the variance with respect to the plane can now be calculated as:

$$\sigma^2 = \frac{\hat{e}^T \hat{e}}{n} \quad (7)$$

which is equivalent to equation (2), but with an inclined mean.

If we define a window and let it move over the elevation model, the variance can be calculated for every position of the window. Using this method, smooth but steep surfaces can well be represented using the variance. Because of the nature of the Least Squares method and the plane model used, the residuals will always be calculated in z-direction, rather than perpendicular to the plane. Generally, this will cause a slight increase of the variance, compared to the variance calculated normal to

the plane. However, for the purposes described in this report, this effect can easily be disregarded.

The result of the described calculation is a new map Σ which contains the variance of n surrounding points in each pixel $\Sigma(r,c)$. The classification is now simply defined as applying a threshold to this map:

$$\begin{cases} \mathbf{C}(r,c) = \text{TRUE} & \text{for } \Sigma(r,c) < t \\ \mathbf{C}(r,c) = \text{FALSE} & \text{for } \Sigma(r,c) \geq t \end{cases} \quad (8)$$

The resulting classification \mathbf{C} is a binary map with TRUE for pixels that are assumed to be part of the glacier and FALSE for pixels that are not.

The whole method runs on two parameters which are the window size n and the threshold value t . The size of the window strongly depends on the characteristics of the terrain. The window should be large enough in order to get significant differences in the variance. However, by using windows that are too large, local phenomena will get lost and the delineation between low and high variance will become less accurate. Related to this is the selection of a threshold level, which is also heavily dependent on the type of terrain.

One problem of this method is the computational efficiency. The design matrix \mathbf{A} remains constant, so the inversion and some other steps can be pre-processed, but many evaluations remain to be done for every window position. On the other hand, the variance only has to be calculated once for each map, after which any threshold can be applied individually. Additionally, the slope and aspect are two other properties that are automatically available after fitting the plain.

4.2.2 Gradient segmentation

When the surface behaves locally like a plane, the first derivative of the surface is locally constant. Areas with constant first derivatives can be grouped in segments. If these segments are large enough, the surface that belongs to them can be considered smooth. In image processing, the first derivative of the data is usually called the gradient ∇_z .

$$\nabla_z = \left(\frac{\partial z}{\partial x} \quad \frac{\partial z}{\partial y} \right)^T \quad (9)$$

Numerically the gradient can be computed with a filter. The Sobel operators are most suited for that. Figure 4.5 shows the Sobel operators for the gradients in y and x direction. The Sobel filter calculates the gradient by taking the difference between the two pixels left and right of the centre pixel. At the same time, it smoothens the result a bit by taking the diagonally pixels into account with a lower weight (Lillesand et al. 2004).

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Figure 4.5 The sobel operators for calculating slopes. Left: calculation in y-direction; right: calculation in x-direction.

Applying both filters results in two gradient images like in Figure 4.6. These images can now be processed in a segmentation algorithm that makes segments of pixels with constant gradient. After that a classification can be based on these segments. During the segmentation algorithm, each individual point is checked to see if it belongs to a certain object or not. After segmentation, the complete area is partitioned into individual regions. The borders of these regions are formed by discontinuities (Jähne 2005). Segmentation is more formally defined by Gonzalez and Woods (2002) as follows:

Let \mathbf{R} represent the entire region. We may view segmentation as a process that partitions \mathbf{R} in n sub regions, $\mathbf{R}_1, \mathbf{R}_2, \dots, \mathbf{R}_n$, such that

1. $\bigcup_{i=1}^n \mathbf{R}_i = \mathbf{R}$,
2. \mathbf{R}_i is a connected region, $i = 1, 2, \dots, n$,
3. $\mathbf{R}_i \cap \mathbf{R}_j = \emptyset$ for all i and j , $i \neq j$,
4. $P(\mathbf{R}_i) = \text{TRUE}$ for $i = 1, 2, \dots, n$, and
5. $P(\mathbf{R}_i \cup \mathbf{R}_j) = \text{FALSE}$ for $i \neq j$.

In the above, $P(\mathbf{R}_i)$ is a logical operation over the points in set \mathbf{R}_i that returns TRUE if a certain segment criterion applies to the region.

From the above, it is clear that segmentation requires a full partitioning and that overlapping regions or mixed pixels are not allowed. Nearly all methods for segmentation work by looking to the similarities between points. Two very similar points will probably belong to the same region.

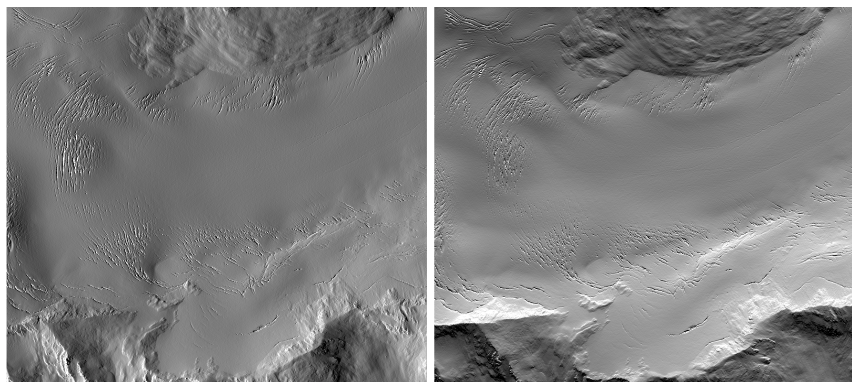


Figure 4.6 Gradient images. Left: gradient in x-direction; right: gradient in y-direction.

Vosselman et. al. (2004) and Hoover et. al. (1996) treat different methods for segmentation in order to recognise structure in elevation models. One of the segmentation algorithms treated is the split-and-

merge algorithm. For this work such an segmentation algorithm based on quad trees is used. The algorithm was designed by Gorte (1996) and has the advantage that it allows to segment on multiple bands simultaneously. In this case the x- and y-gradient images are the two bands on which the segmentation algorithm operates.

A quad tree is a hierarchical data model that provides a compact raster representation by using a variable-sized grid cell. Finer subdivisions are used in areas requiring finer detail, providing a higher level of resolution. The segmentation algorithm takes the quad tree as input and merges the leaves of the quad tree into segments, controlled by a homogeneity measure. The homogeneity measure consists of two parts. First there is a threshold for the difference between the mean segment value and the value of a pixel that is candidate for merging. Secondly the variance after merging is restricted by a threshold. The result when applied to the glacier is shown in Figure 4.7.

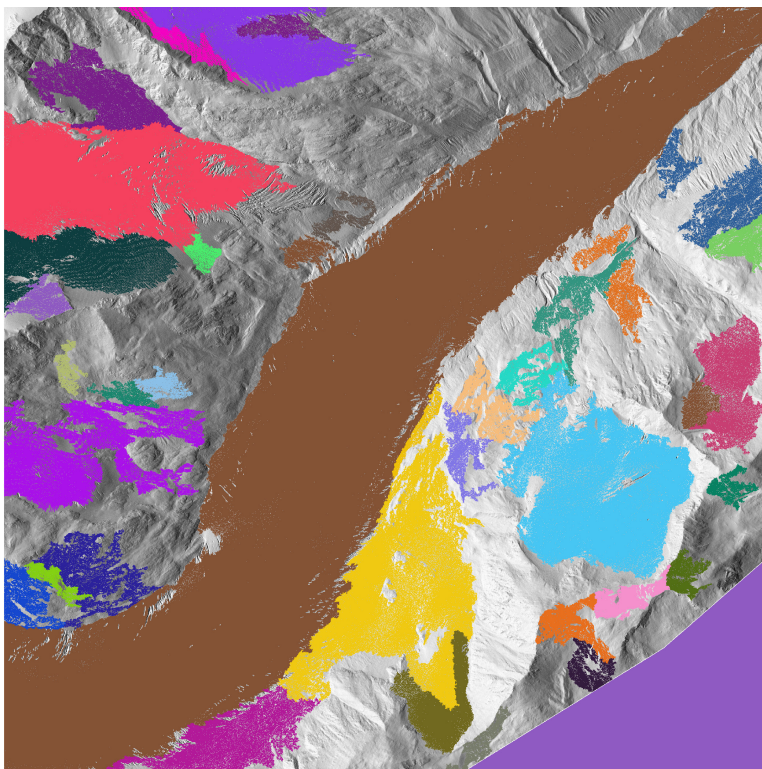


Figure 4.7 Result after segmentation based on gradient.

There are a high number of different segments after the segmentation, which should now be classified in one of the classes 'glacier' and 'non-glacier'. Reverting back to the original classification measures, we see that we have to look for smoothness. Only when a segment is relatively large, the surface can be called smooth. The problem of classifying glacier pixels can therefore now be translated to the problem of selecting segments that are greater than a certain threshold. By applying this threshold, the parts of the terrain that can be considered smooth are selected.

In comparison to the variance based segmentation this method is computationally much more efficient because calculating the gradients requires less computational power than fitting the planes.

4.3 Connectivity

The second classification criterion was the connectivity of all pixels that belong to the glacier. Applying this constraint introduces the important design decision that only one glacier can be delineated at a time.

As all processing is performed within the raster domain, connectivity is quite well defined. The set of four pixels neighbouring a pixel p is denoted as $N_4(p)$ and is called *4-adjacency*. Finding all connected pixels of a given seed pixel is called connected component labelling. All pixels that have the same value and are connected to each other are labelled with the same label in this process. For the delineation work, such a connected component labelling is applied to the data. After running the labelling program, the component with the largest area is selected, as this is assumed to be the glacier in the dataset. All other pixels not belonging to this component are discarded.

4.4 Hydrological constraints

From a glaciological point of view, one large connected ice field doesn't have to be exactly one glacier. It can be that two glaciers are connected somewhere, but behave completely independent. Often, these glaciers are also given different names. A classification method based on geometrical aspects like smoothness and connectivity doesn't notice this; including constraints concerning the catchment area makes therefore sense.

4.4.1 Hydrologic flow determination

The boundaries of a catchment area can easily be calculated in any GIS given a DEM which covers the complete catchment area (Tarboton et al. 1991). The principle behind the algorithm is the calculation of the run-off direction of water for all pixels. This direction follows from the lowest elevation in the $N_8(p)$ pixels of the centre pixel in a moving window. Given this flow direction quantified in one of the eight compass points, a flow accumulation graph can be calculated.

Assuming a unit rainfall over the whole area, the flow accumulation process calculates how much water flows in each pixel. When visualising the accumulation, a hydrologic network appears with little channels or rivers. Local minima in the data might disturb the continuous flow of water in the model. The process is therefore preceded by a sink filling program which alters the data in such a way that for every pixel there is a route towards the outlet.

Figure 4.8 shows an example for the flow direction map and flow accumulation map covering a part near the snout of the glacier. The flow accumulation map clearly shows where most water flows in the terrain. It is for instance clear that the drainage of glacial melt water occurs mainly along the border of the glacier.

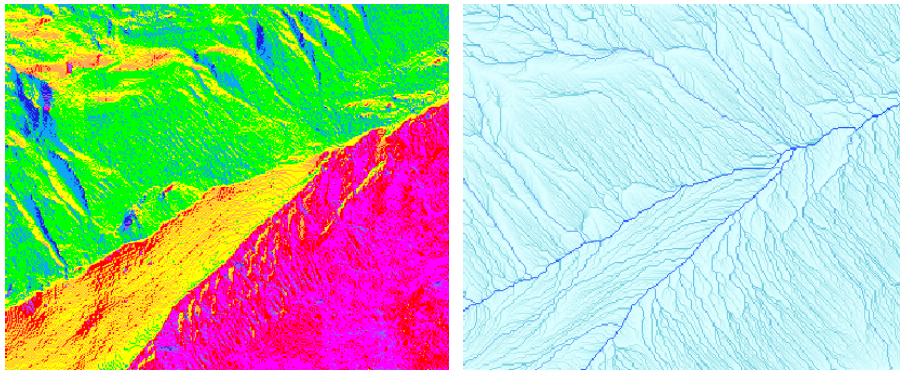


Figure 4.8 Flow direction map (left) and flow accumulation map (right)

Continuing with the accumulation map, a drainage network can be extracted. This is simply done by applying a threshold on the accumulation graph; only pixels with a value above the threshold transport enough water to be selected as part of the hydrologic network. In the terrain one will find streams or rivers along the lines that are identified in the network map.

4.4.2 Network and catchment extraction

It is this drainage network map that can be used to derive the catchment area in a certain terrain. However, the drainage network doesn't differentiate for the importance of streams in the terrain. This can be done with the Strahler Network Ordering, a method from the field of geomorphology (Strahler 1952). The Strahler Order is a hierarchical ordering of the streams in a hydrological network. The more important a stream is, the higher Strahler Order is given to it. Some GIS programs contain algorithms for calculating the Strahler Order given the accumulation map. (ITC 2005) First, the streams are found that form the upper-most starting points of the drainages in the network. These streams obtain ordering number one until a node is found that connects the stream with a following stream. When two or more streams of equal order join each other, the stream order value is increased by 1. When a higher-order stream joins a lower-order stream, the order number for the next stream does not increase; instead, the largest order number of the streams that contribute to it is assigned. The resulting ordering shows the importance of a stream in the hydrologic system and can be used as a mean for generalisation.

Figure 4.9 shows the streams coloured to their Strahler order overlaid over a hill shade. A minimum drainage length of 1000 meters was chosen for this selection. For each stream in the drainage ordering map a catchment can now be calculated. This calculation is basically a search for all pixels that flow towards the stream of interest.

For the application of finding the glacier surface, we are only interested in one large catchment area for the complete glacier. A very long minimum length of 5000 meters was therefore selected in the ordering. This resulted in three separate catchment areas for the area covered by the DEM as showed in Figure 4.10. One large catchment area for the whole area can be obtained by merging the separate catchments. This catchment area gives the maximum extends of the glacier surface.

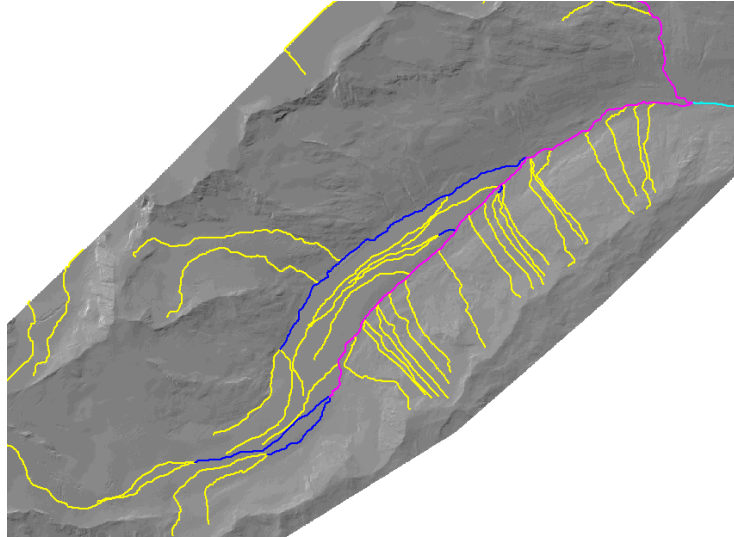


Figure 4.9 Streams ordered with the Strahler Order

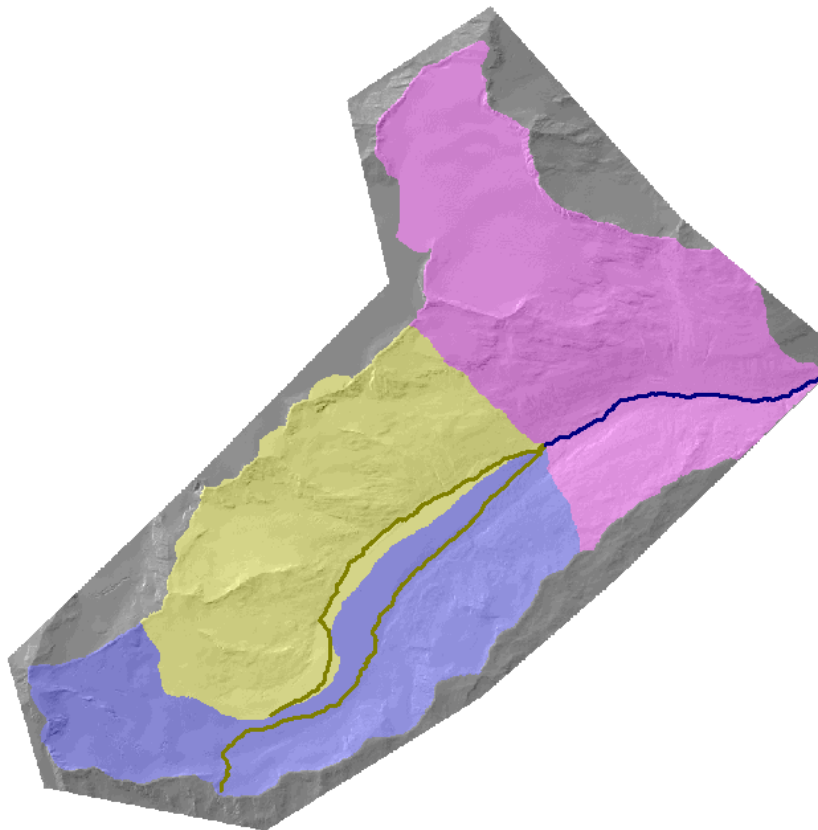


Figure 4.10 The catchment areas in the Hintereisferner valley.

Finding the glacier pixels is now achieved by determining the intersection of the regions that were generated by the smoothness classification and the catchment area boundary.

$$\mathbf{R}_{glacier} = \mathbf{R}_{smooth} \cap \mathbf{R}_{catchment} \quad (10)$$

The calculation of the catchment areas can be done in programs like TarDEM and ILWIS, the latter being used in this case. In contrary to many of the other processes described in this work, the hydrologic

calculations are global instead of local calculations. This makes the processing rather extensive and time consuming. The problem was circumvented by resampling the data from a 1 meter resolution to a 5 meter resolution, gaining a factor 25 in computation time. The calculations could now be performed in a couple of minutes instead of a couple of hours; an inherent reduction in the accuracy of the catchment border is naturally the price to pay.

4.5 Laser Intensity map

In section 3.4 the Laser Intensity values were presented as a side product of ALS. If these intensity values are interpolated to a grid, they might be used for the classification of glacier pixels or for visual supervision by the operator. As mentioned in section 4.1, the intensity values require some further processing before they can be used for surface analysis. This section elaborates on these processing steps.

4.5.1 Intensity value correction

Figure 4.11 shows the result of interpolating the intensity values to a raster. Clearly, the result is visually not really appealing. The contrast within the image is low and the brightness is not constant over the image. This is due to the differences in range from the measurement platform to the surface. At the snout of the glacier, the surface is approximately 200 meters lower than at the top. Because the travel time of the pulse is not constant, the loss of energy is not constant either.

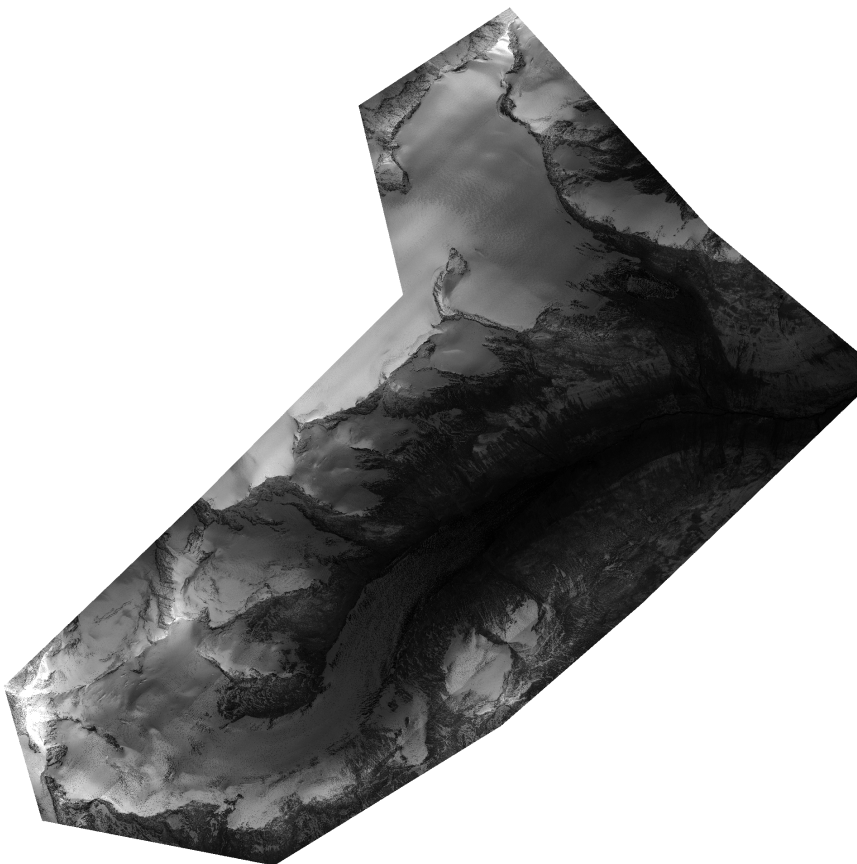


Figure 4.11 Laser intensity values interpolated to a raster without any further processing. Terrain height differences introduce variation in received intensity.

One way to correct the image is by modelling all components that make up the received energy P_r in the sensor. The value for P_r can be described with the Laser Range Equation. Rees (2001: 215-16) gives the equation as:

$$P_r = \frac{\lambda^2 G^2 P_t}{(4\pi)^3 R^4} \sigma^0 A \eta_{sys} \eta_{atm} \quad (11)$$

where:

P_r is the received signal intensity at the sensor;

P_t is the signal intensity transmitted from the sensor;

λ is the wavelength of the signal;

G is the antenna gain;

η is the efficiency of the antenna and atmosphere;

R is the range from the system to the scattering surface;

σ is the backscattering coefficient, which depends on surface material, incidence angle, etc;

A is the area of the scattering surface.

To find the amount of reflected energy it is required to know the area of the scattering surface. This area depends on range and beam aperture. Pulse sensors can operate in either the *beam-limited case* or the *pulse-limited case*. In the latter, also the pulse length has to be taken into account for calculating the scattering area. A sensor is beam-limited if (Rees 2001: 200-02):

$$R\theta_t \ll 2\sqrt{cRt_p} \quad (12)$$

With a mean flying height of $R = 900$ m, pulse length $t_p = 1$ ns and opening angle $\phi_t = 0.4$ mrad, laser scanning is indeed beam-limited. This means that it can be assumed that the whole footprint is illuminated by the laser beam at once. According to Jelalian (1992: 3-10) the area of the scattering surface depends can then be written as:

$$A = \pi R^2 \left(\frac{1}{2}\theta_t\right)^2 \quad (13)$$

In this equation we assume that the target lies normal to the beam direction. Combining equations (11) and (13) gives for the range equation:

$$P_r = \frac{\lambda^2 G^2 P_t \theta_t^2}{2 \cdot 4^3 \pi^2 R^2} \sigma^0 \eta_{sys} \eta_{atm} \quad (14)$$

Within one flight, many of these parameters can assumed to be constant, so that they have no effect on the scaled image and need not to be corrected.

$$F = \frac{\lambda^2 G^2 P_t \theta_t^2 \eta_{sys}}{2 \cdot 4^3 \pi^2} = \text{constant}$$

This gives the simplified form:

$$P_r = \frac{1}{R^2} \sigma^0 \eta_{atm} F \quad (15)$$

From this equation, the three components that need to be corrected can be seen. First, a correction of has to be applied as a result of spherical spreading. This loss can be explained as the effect that the photons in the beam have to be divided over a continuously increasing area. The second correction is for the attenuation due to travelling through the atmosphere and the last one due to scattering. These effects are depicted in Figure 4.12.

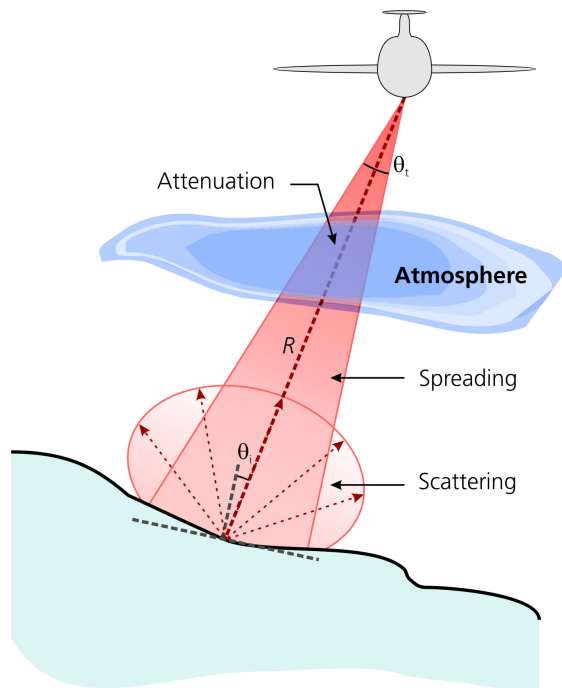


Figure 4.12 Effects influencing laser signal intensity.

The atmospheric efficiency η_{atm} is the attenuation of the signal while travelling to the surface and back. It can be quantified with the attenuation coefficient α in dB/km (Table 4.1). Usually a value of $\alpha = 1 \text{ dB/km}$ suffices. The attenuation α relates to η_{atm} as:

$$\alpha \cdot \frac{2R}{1000} = 10 \cdot \log(\eta_{atm}) \quad (16)$$

with a factor 1000 to convert metres to kilometres. From this equation the value for η_{atm} can easily be derived:

$$\eta_{atm} = 10^{\left(\frac{\alpha \cdot 0.002R}{10}\right)} \quad (17)$$

The backscatter coefficient σ^0 in equation (15) depends primarily on the angle of incidence. In the case of Lambertian scattering this proportionality can be written as (Rees 2001: 48):

$$\sigma^0 \propto \cos^2 \theta_i \quad (18)$$

The angle of incidence θ_i depends on both the scan angle as well as the terrain slope. Using this relation a correction for incidence angle can easily be added to the programme. A more elaborate description on the requirement for Lambertian scattering is given in the next section.

Table 4.1 Attenuation of laser energy with $\lambda = 1.06\mu m$

Weather condition	α in dB / km
Extremely clear	0.2
Standard clear	0.5
Clear	0.8
Light haze	1.5
Medium haze	2.3
Haze	3.9

Source: (Byers et al. 1951)

4.5.2 Surface scattering

Equation (13) for the scattering surface presented in the previous section applies only if the surface is a Lambertian surface. On a Lambertian surface, the energy is equally reflected into all directions. This distribution is usually quantified with the bidirectional reflectance distribution function (BRDF). Given an the angle of incidence (θ_i) and the azimuth direction (ϕ_i), this function gives the proportional reflection for every direction (θ_r, ϕ_r). For a Lambertian scattering, the BRDF is a constant (Rees 2001):

$$BRDF(\theta_i, \phi_i, \theta_r, \phi_r) = \frac{1}{\pi} \quad (19)$$

This constant is implied in equation (18) of the previous section. The full equation describing the backscatter coefficient includes the BRDF from equation (19) and reads:

$$\sigma^0 = 4\pi BRDF(\theta_i, \phi_i, \theta_r, \phi_r) \cos^2 \theta_i = 4 \cos^2 \theta_i \quad (20)$$

Lambertian scattering only occurs if the surface is rough compared to the wavelength of the signal. In the extreme opposite case, where the surface is very smooth compared to the signal wavelength, specular scattering occurs. The surface behaves then like a mirror, reflecting all energy into one direction according to the law of reflection:

$$\theta_i = \theta_r$$

Now, the BRDF is zero for all but one direction. The question whether a surface causes specular or Lambertian scattering can be answered with

the Rayleigh roughness criterion. For a surface to be smooth, it should adhere to the following:

$$h_{rms} < \frac{\lambda}{8 \cos \theta_i} \quad (21)$$

where h_{rms} is the Root Mean Square (RMS) of the terrain elevation h that is within the footprint area A . Given the laser wavelength λ of approximately 1000 nm, this holds that the RMS of the surface should be smaller than 125 nm. There are few natural surfaces that have such a smooth surface and as such it is safe to assume that all surfaces in the dataset are Lambertian to a certain degree. This assumption of a Lambertian surface is fundamental to ALS. On a non-lambertian surface, the laser beam would be reflected in opposite direction to the laser scanning and can therefore not be measured.

However, in reality the surface will be somewhere in-between Lambertian and specular scattering. For a good reconstruction, reflectance properties of snow and ice should be known. Current research at the University of Innsbruck focuses on finding a good reflectance model. Measurements with the terrestrial laser scanner might help to find out these properties.

4.5.3 Implementation in GRASS

A program was written in GRASS, which takes a raster map with the raw intensity values. A second raster map containing the range values can be supplied to the program. This map \mathbf{S} contains for every pixel the distance to the airplane position at the time of acquisition. The range map was calculated from the database organising the point cloud. This database stores for each point the position of the airplane at the time of acquisition, making it possible to do this kind of analysis.

Using these two datasets, it can apply a correction for spreading loss and attenuation. The first step that is performed is the correction for the spreading loss:

$$\mathbf{I}(r, c)_{spread} = \mathbf{I}(r, c) \cdot \mathbf{S}(r, c)^2 \cdot s \quad (22)$$

with s a scale factor of 10^{-4} , required for computational reasons. The next step is the correction for attenuation using the relation found in equation (17):

$$\mathbf{I}(r, c)_{corr} = \mathbf{I}(r, c)_{spread} \cdot 10^{\left(\frac{\alpha \cdot 0.002 \cdot \mathbf{S}(r, c)}{10}\right)} \quad (23)$$

The final step involves a histogram equalisation for scaling the intensities back to the [0,255] range again and increasing the contrast at the same time. In order to reduce noise, it might be useful to apply a median filter afterwards. More information about the program can be found in appendix D.

Note that the program does not correct for the angle of incidence of the signal with the surface although this would be possible using the

relations given in previous sections. It would require the calculation of the beam angle and the terrain slope. After the spreading loss this is the second most important correction term that needs to be applied. It has a systematic character, as can be seen in Figure 4.14. The wave like pattern is due to the varying angle of incidence.

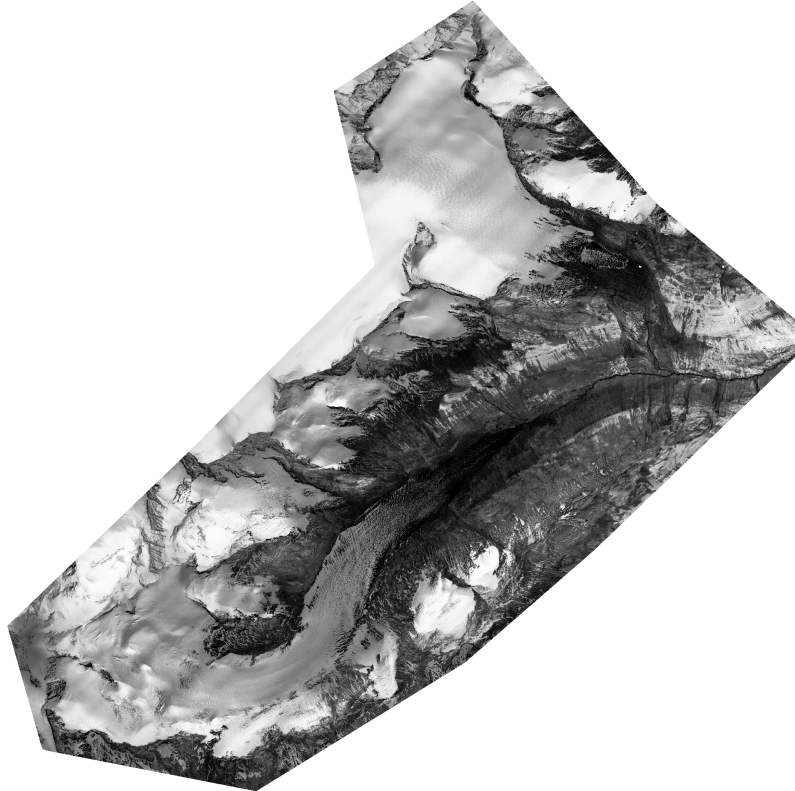


Figure 4.13 Laser intensity image after correction for spreading loss and attenuation. The contrast has been enhanced using histogram equalisation.

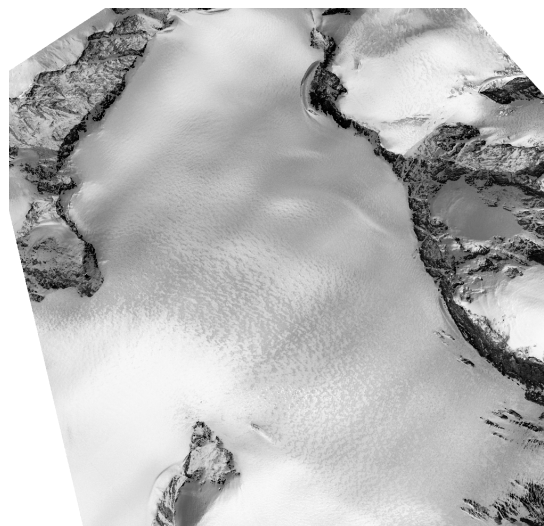


Figure 4.14 Intensity map of a sloped ice field. A large scale wave pattern is visible due to influence of slope on intensity measurement. The wave length is equal to the in-between strip distance.

The correction for angle of incidence was not implemented because it appeared that intensity data from ALS is not suitable for the automatic delineation of a glacier. The reflectance of terrain depends on many

aspects like the soil moisture, amount of snow on the glacier or the appearance of a thin layer of water over the ice. These aspects are hard to predict and can therefore not be handled by a general automated procedure. However, for visual assessment and manual mapping the corrected intensity data will still be very useful.

4.6 Results

A program was written in GRASS to determine the delineation of the glacier by classifying all pixels as either "glacier" or "non-glacier". This is done by using the variance based smoothness classification and the constraints concerning the hydrological properties and connectivity. In the following section the implementation of these methods into a program will be presented. The glacier delineation test was performed on the 11th dataset from the Hintereisferner, acquired in October 2004. This dataset is characterised by a lower number of crevasses compared to acquisitions on other dates.

4.6.1 Implementation

Figure 4.15 features a flow chart with the steps that are performed in the new GRASS program for glacier delineation. It also shows which modules were written for the Glacier surface analysis toolbox to extend the existing GRASS functionality.

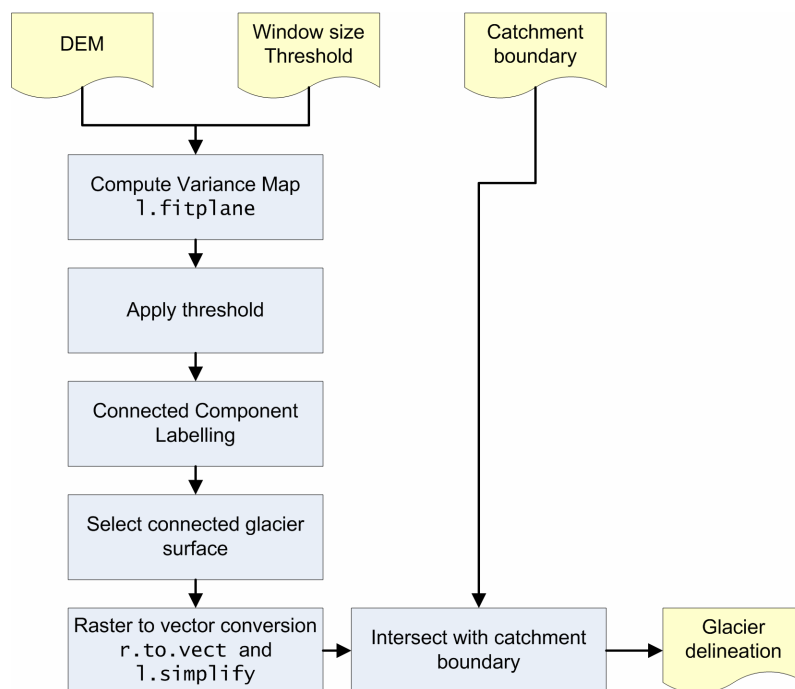


Figure 4.15 Flow chart of the delineation process. The identified l.* modules are part of the glacier surface analysis toolbox developed for GRASS.

As is clear from the flow chart, the first step in the process is to calculate the variance of the terrain. A program was written to calculate for all points the variance, using the data available in a window surrounding it. Additionally it also gives the slope and aspect of these points. For the Hintereisferner dataset a window size of 11 by 11 pixels at 1 meter resolution gave a good result. Over the complete area that was acquired by laser scanning, the variance ranges from 0 to 147 m^2 . The result is

represented in Figure 4.16 using a logarithmic scale. The smooth surface of the glacier can very well be distinguished from this image. High variance values appear at terrain discontinuities, such as mountain ridges and foothills.

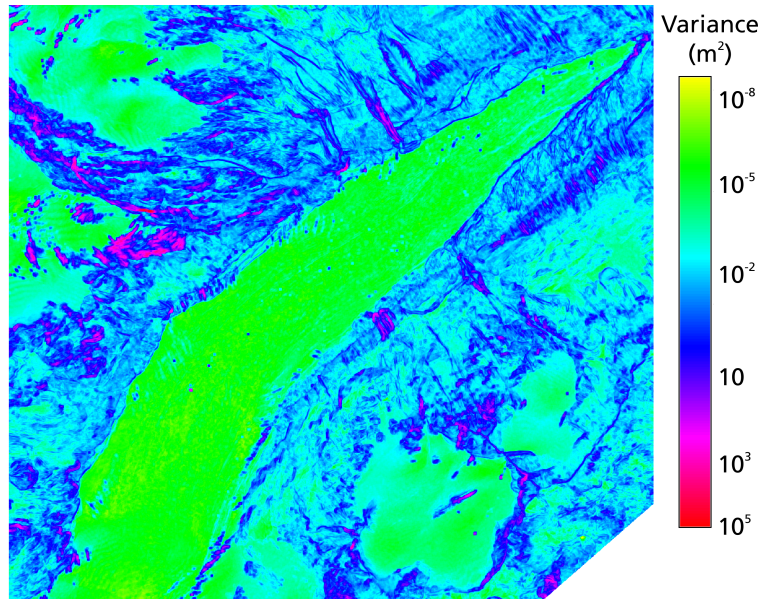


Figure 4.16 The variance of the terrain within an 11 x 11 m window, displayed in a logarithmic scale.

For the window size, a window of 11 by 11 meters appeared to work well, although this choice is rather arbitrarily. Figure 4.17 shows an image taken on the glacier with the window projected on the surface. It gives an impression of the variance that can be expected within such a window.

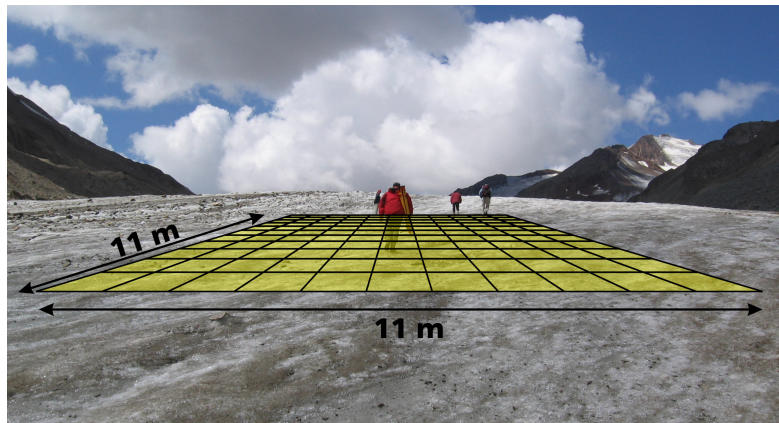


Figure 4.17 The window size projected on the glacier surface

The surface variance that we find in the DEM is a combination of the measurement precision and the variation in the terrain.

$$\sigma_{DEM}^2 = \sigma_{ALS}^2 + \sigma_{SURFACE}^2 \quad (24)$$

The minimum possible variance is expected to be equal to the laser scanning precision, higher variance values can be accounted to the terrain properties. For the glacier surface of the Hintereisferner variance

values up to 0.06 m^2 were found, which is approximately twice the laser scanning precision, i.e. a standard deviation of 0.25 m. This value of 0.06 m^2 was determined empirically and used as threshold value in order to classify between glacier and non-glacier surface: all terrain with a lower variance is supposed to be part of the glacier. The selected area is shown in Figure 4.18.

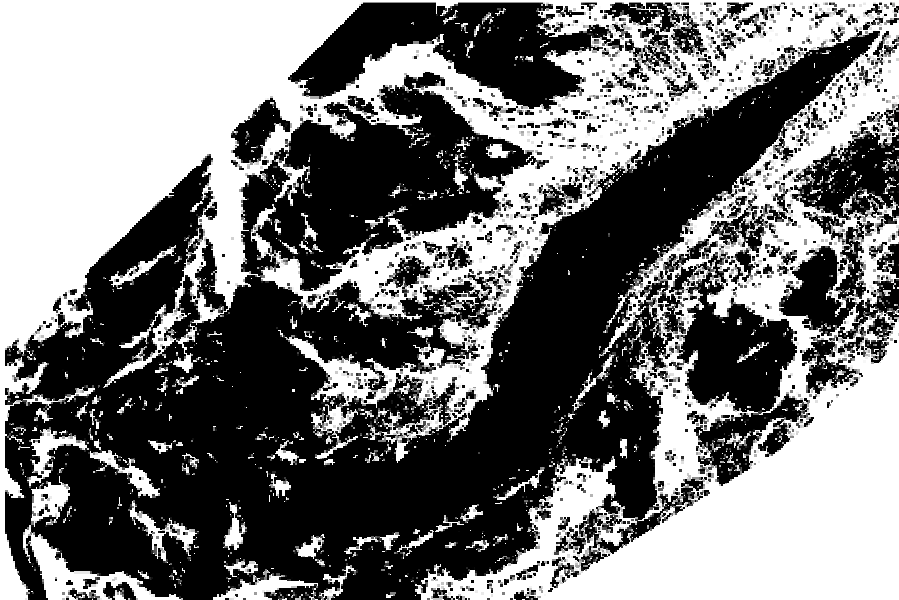


Figure 4.18 Pixels that are classified as glacier after thresholding at 0.06 m^2 . Compare with Figure 4.16.

The classification criterion of connectivity was also added to the program. Applying this property holds the additional advantage that noise and incorrect classification based on variance alone will be disregarded. In the program this property is implemented by selecting the largest connected component after thresholding (Figure 4.19). A morphologic closing filter is run over the data before selecting the largest connected component in order to join small separate parts after selection.

The resulting raster map can be converted to a vector based polygon representing the outline of the glacier. Using the Douglas-Peucker algorithm, which was newly implemented in GRASS, the line can be simplified. It can then be stored as a polygon in GRASS format, or any other industry-standard format.

This above presented raster map is almost the final result of the delineation process. Apart from some small errors, which will be elaborated on in the next section, it resembles the glacier delineation quite well. The only remaining problem is that the delineation includes some other glaciers, i.e. glaciers in another basin. The last step is therefore to limit the delineation at the catchment area border. The catchment border is imported in the system, after calculating it in any suitable software package. Then, a simple GIS intersection of the two polygons results in the final product for the glacier delineation.



Figure 4.19 The glacier surface after applying the connectivity constraint and morphological closing.

4.6.2 Quality assessment

The computed result can be checked for its quality in several ways. First of all, the result was overlaid onto the shaded relief view. Using available maps and expert knowledge the correctness of the classification can be assessed (Figure 4.20). This qualitative check of the delineation result was performed in cooperation with a glaciologist. It appears that the delineation is generally good, with the notable exception of the debris covered area besides the snout of the glacier. This part of the glacier consists of "dead ice" that is covered by rocks and snow. Due to the rough surface caused by the debris, it is impossible to detect this area as glacier with the method as it is described in this work (Figure 4.21).

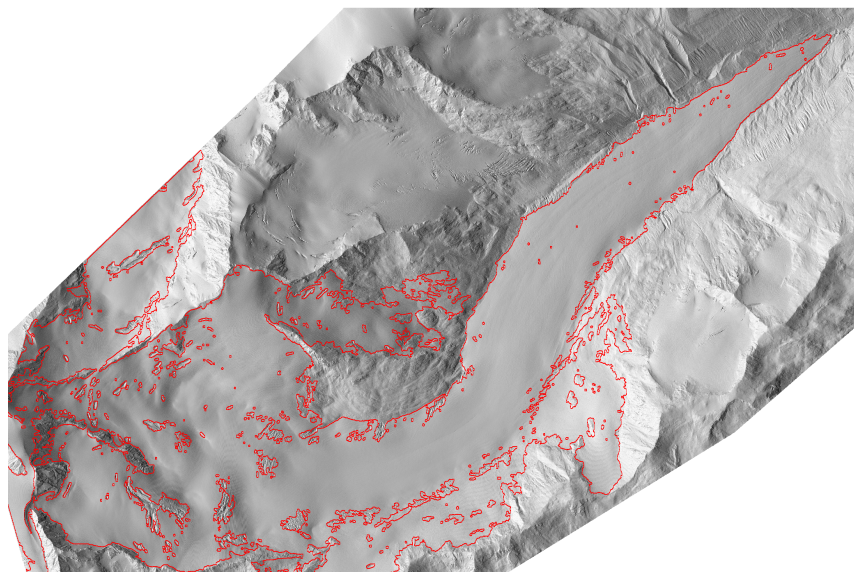


Figure 4.20 The computed delineation overlaid on the shaded relief view.

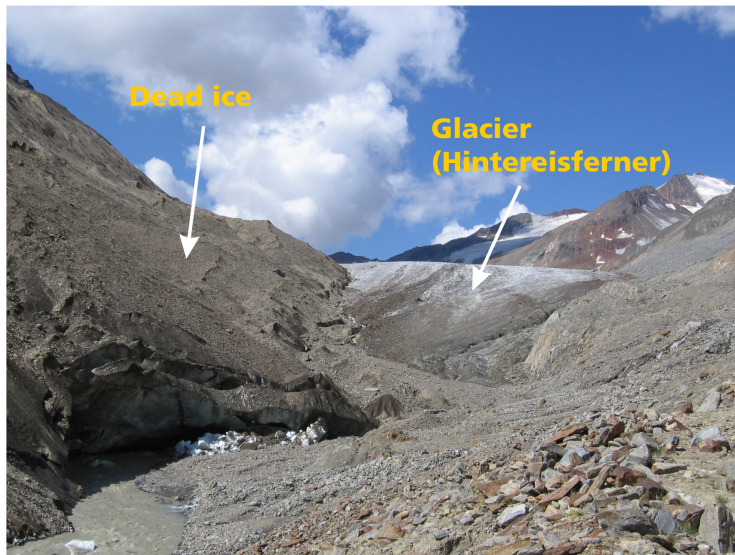


Figure 4.21 Dead ice, covered by dust and rocks, at the snout of the glacier.

Other problems with the classification occur at and near the crevasses, because here the glacier surface no longer adheres to the smoothness constraint that was postulated earlier. Very flat rock in the dataset, formed due to former glacier cover or snow fall, can result in over segmentation when connected to the glacier surface.

Another inspection was made by comparing the result with a manual delineation that was once made by a graduation student at the geography department of the University of Innsbruck. Visually, the manually measured line and the automatically derived line agreed very well.

A quantitative assessment of the classification accuracy can be made by generating a confusion matrix. The confusion matrix shows how many pixels were classified false and is given in Table 4.2. It gives the number of pixels that were classified in one of the classes for both the reference and the classification map. In the second table, percentages for commission and omission are given. The commission represents the number of pixels that were classified to a class where they do not belong, omission on the other hand is the number of missing pixels in a class. In fact, they represent the error types that in statistics are known as Type I and Type II errors.

It is important to note that also a completely random classification will give some correct classified pixels. The \hat{k} value is a statistical value that represents how much better the classification is than a fully random classification. Conceptually it is defined as (Lillesand et al. 2004):

$$\hat{k} = \frac{\text{observed accuracy} - \text{chance agreement}}{1 - \text{chance agreement}}$$

For instance, a value of $\hat{k} = 0$ means that the classification is no better than a fully random one. The value for \hat{k} can be calculated from the confusion matrix with the number of pixels in a certain class. If we name the confusion matrix O , then the individual elements of this matrix are o_{ii} with $i = 1 \dots n$ where n is the number of classes. For simplicity we

call the Row Sum in Table 4.2 o_{i+} and the Column Sum o_{+i} . The calculation is now as follows:

$$\hat{k} = \frac{MN \cdot \sum_{i=1}^n o_{ii} - \sum_{i=1}^n (o_{i+} \cdot o_{+i})}{(MN)^2 - \sum_{i=1}^n (o_{i+} \cdot o_{+i})} \quad (25)$$

M and N are respectively the number of rows and columns in the classification raster C .

The overall kappa value for the glacier delineation was 0.82, which can be interpreted as that the classification was 82% better than a fully random classification. This classification result could be much better if the errors due to crevasses can be removed. Currently, there is no automatic way to do that, except for removing all holes in the glacier polygon. This is not the most optimal method, because it will also remove rocks standing out above the glacier and clearly not forming part of it.

Table 4.2 Confusion matrix of the glacier classification

Classes		Reference Map		
		No Glacier	Glacier	Row Sum
Classifi- cation	No Glacier	19665553	1122511	20788064
	Glacier	611266	5639118	6250384
	Column Sum	20276819	6761629	27038448

Classes	Commission	Omission	Estimated \hat{k}
No Glacier	5.88 %	3.29 %	0.780
Glacier	9.78 %	16.60 %	0.867

5 Reconstructing crevasses

Crevasses are one of the most typical features in glaciers and of interest to both mountaineers and researchers. This chapter describes an outline for the detection and reconstruction of crevasses from ALS data. Figure 5.1 features a heavily crevassed glacier in the Alps of New Zealand. The aim of this section is to develop a method for detecting and visualizing these crevasses from ALS data. The presented approach is based on mathematical morphology, which will be explained first.



Figure 5.1 The Fox Glacier in New Zealand, clearly showing many crevasses. (Source: Hambrey and Alean 2006)

5.1 Mathematical Morphology

5.1.1 Principles of morphology

Mathematical morphology is the theory of the analysis of spatial structures in data sets (Serra 1982). It works like a convolution, but uses decision operators instead of multiplication. A morphological filter is used to detect or modify structural elements in the image, i.e. the morphology of the terrain. It is an alternative to frequency filtering, where terrain information is first converted to the frequency domain using the Fourier Transform. The advantage of Mathematical Morphology is that it operates directly on the morphology of the surface and is therefore more intuitive. Most operations in Mathematical Morphology have been developed in the field of image processing. Therefore, they often refer to operations in the grey scale domain. Although the grey scale domain is usually limited to $[0,255]$, these values can easily be replaced with the elevation values from the DEM. As such, there is no fundamental difference to applying these operations to grey scale images or other raster based data.

Mathematical morphology is based on two principal operations that form the basis of almost any morphological analysis. These are erosion and dilation. In the greyscale domain, erosion is defined as assigning the

minimum pixel value within a window to the centre pixel, while dilation is exactly the opposite operation as it takes the maximum value. In Mathematical Morphology, the window is called the structuring element B . The structuring element can have any shape, but squares and disks appear to be most useful in general cases (Figure 5.2).

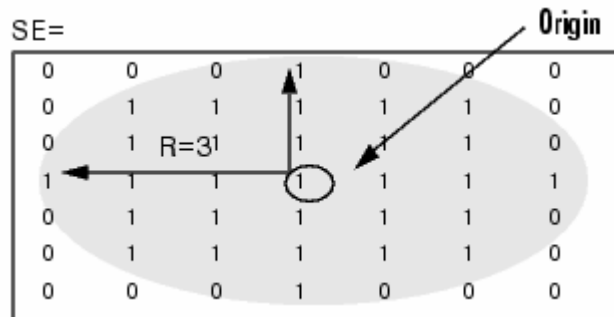


Figure 5.2 Example of a disk shaped structuring element. (© The Mathworks)

If the structuring element B is centred over the pixel with coordinates (s, t) , the eroded image of a raster X is defined as (Gonzalez and Woods 2002; Soille 1999):

$$[\varepsilon_B(\mathbf{X})](s, t) = \min \{ \mathbf{X}(s+r, t+c) \mid (s+r), (t+c) \in D_X; (r, c) \in D_B \}$$

Similarly, the dilated image is defined by taking the maximum:

$$[\delta_B(\mathbf{X})](s, t) = \max \{ \mathbf{X}(s+r, t+c) \mid (s+r), (t+c) \in D_X; (r, c) \in D_B \}$$

Figure 5.3 shows the result of erosion and dilation on a one dimensional signal. Clearly, the dilation process magnifies protruding elements in the signal, but covers pits in the surface. In contrast, the erosion operator shrinks the protruding elements but magnifies pits, holes and thus crevasses. Erosion and dilation have a duality: the erosion is in fact a dilation applied on the background, the complement of the data.

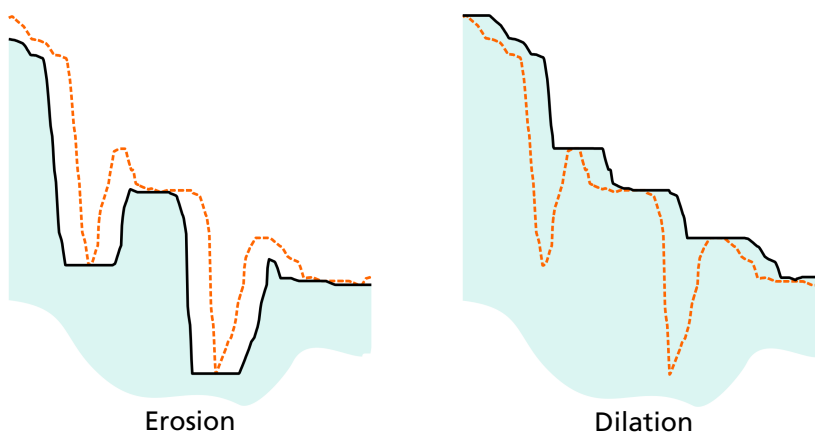


Figure 5.3 Examples of erosion and dilation operations performed on a crevassed glacier surface with a flat structuring element.

One of the properties of dilation and erosion is that these are increasing operators. This means that if a set X is a subset of another set Y , then $\delta_B(X)$ is still a subset of $\delta_B(Y)$ (Serra 1982):

$$X \subseteq Y \Rightarrow \delta_B(X) \subseteq \delta_B(Y) \quad (26)$$

and

$$X \subseteq Y \Rightarrow \varepsilon_B(X) \subseteq \varepsilon_B(Y) \quad (27)$$

This increasing property is important because it was a requirement to transform the dilation and erosion operation from the binary to the greyscale domain.

From the two intrinsic operations of erosion and dilation, the processes of opening and closing are derived. Opening and Closing are the so-called morphological filters. Opening is simply the combined operation of erosion and dilation and Closing the same but in reverse order. Figure 5.4 shows the result of applying opening and closing to a one dimensional signal.

$$\text{Opening: } \gamma_B(\mathbf{X}) = \delta_B[\varepsilon_B(\mathbf{X})] \quad (28)$$

$$\text{Closing: } \phi_B(\mathbf{X}) = \varepsilon_B[\delta_B(\mathbf{X})] \quad (29)$$

Although defined as a combined operation of erosion and dilation, the filters do have a physical meaning. When performing Opening on a surface, it means that one searches for the highest position where the structuring element is completely covered by the surface. Closing is the search for the elevation that the structuring element has when it is laid horizontally on the surface.

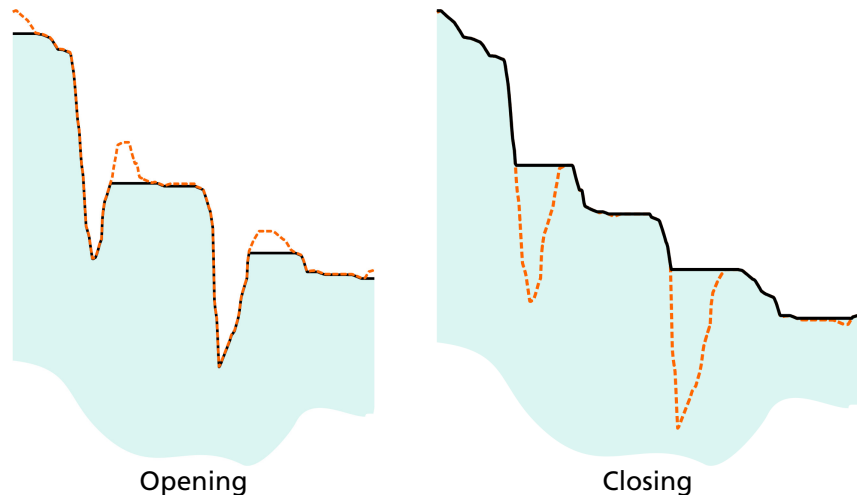


Figure 5.4 Examples of opening and closing operations performed to the same surface as in Figure 5.3.

Opening and Closing have the properties of being increasing and idempotent. Idempotence means that if the filter is applied once, no more changes to the signal will result from repeated applications. For instance, applying a second closing to the result in Figure 5.4, will still

give the same black line as a result. Idempotence is important for the formal definition of mathematical morphology.

$$\gamma_B(\gamma_B(X)) = \gamma_B(X) \quad (30)$$

and

$$\phi_B(\phi_B(X)) = \phi_B(X) \quad (31)$$

5.1.2 The Hit-or-Miss transform

The third well known Morphological operation besides Opening and Closing is the Hit-or-Miss transform (Serra 1982). The basic idea behind the Hit-or-Miss Transform (HMT) is the extraction of image pixels that fulfil a certain neighbourhood configuration requirement. The HMT uses two Structuring Elements: one to define what the object should look like and a second element to define the shape of the background of the image. This makes it the morphological alternative for template matching and therefore a potential method for crevasse detection. Some aspects of the HMT will be given in this section.

Like the other operations as erosion, dilation, opening and closing, the hit-or-miss transform was originally defined for binary images. In this case, the transform of the binary image \mathbf{X} with the elements \mathbf{B}_1 and \mathbf{B}_2 is defined by (Soille 1999):

$$[HMT_B(\mathbf{X})](r,c) = \{\mathbf{X}(r,c) | (\mathbf{B}_1)_{r,c} \subseteq \mathbf{X}, (\mathbf{B}_2)_{r,c} \subseteq \mathbf{X}^C\} \quad (32)$$

with \mathbf{X}^C the complement of the binary image. Both structuring elements have the same origin and by definition $\mathbf{B}_1 \cap \mathbf{B}_2 = \emptyset$. The hit-or-miss transform thus searches for those pixels where \mathbf{B}_1 hits the object pixels and where \mathbf{B}_2 misses the object pixels. This makes the transform very useful for detecting certain shapes. One defines what an object should look like and what the background should look like. If both structuring elements feature a 0 at the same position, these pixels are called *don't care* pixels. Here the object can either be true or false.

In contrast to the definition of erosion and dilation, which were directly given in the grey scale domain, equation (32) cannot be extended to the processing of grey scale images. This is because the presented hit-or-miss transform is not increasing like it was defined in equation (26). However, the definition can be rewritten as an intersection of two erosions:

$$HMT_B(\mathbf{X}) = \varepsilon_{\mathbf{B}_1}(\mathbf{X}) \cap \varepsilon_{\mathbf{B}_2}(\mathbf{X}^C) \quad (33)$$

Because erosion can be extended to the greyscale domain, the HMT should also be extendable. However, there is not one definite way to extend hit-or-miss operations to the greyscale domain. One problem is that the definition of the complement of the grey scale image \mathbf{X}^C is not trivial, for which several authors have proposed different solutions (Khosravi and Schafer 1996; Ronse 1996). Only very recently a unified theory for the grey level hit-or-miss transform has been presented and applied in practice (Naegel et al. 2006a, 2006b).

For the application of detecting crevasses, we would like to define a SE with grey level values that represent a crevasse. The advantage is that prior knowledge concerning a crevasse, e.g. the V-like shape, can be put into the detection. A second SE for the miss-part of the HMT should have values representing the glacier surface. According to Naegel (2006b), exactly this problem of correctly determining the SE for getting correct results is the first issue when using HMT. There have been very few examples of the successful application of grey scale SEs for the HMT. In fact, the use of a non-flat SE is generally dissuaded because setting the right grey level values appears to be cumbersome, especially if one wants the detection to be somewhat general. For instance, information on which exact grey-level values to use should be available on beforehand, while in practice these values are varying. In the case of the glacier dataset this could probably be achieved by detrending the data, so that the glacier surface always has elevation zero. Another point raised by Naegel is that experience shows that most often the same results can be achieved by the smart application of the less complex flat SEs. For this reason, the design of grey level structuring elements will no longer be considered.

However, there are alternatives. One notable one, often used in angiography for detecting blood vessels, is to partition the data into a full 3D space built of voxels (Passat et al. 2005). For the glacier DEM this means that the data has to be transformed to a 3D voxel map. This map will be a binary one, with ones for the voxels that are part of the terrain and zeros for voxels that are part of the air. Note that the result brings a loss in the vertical accuracy because the elevations will have to be quantified in a limited number of voxels. Now, a pair of 3D structuring elements could be designed to detect crevasses in the dataset. Note that when a 3D structuring element is applied on a 3D voxel map, one is actually still using a flat SE. Because of this and because the map has become binary, the basic definition of the HMT from equation (32) can be used in a slightly extended form:

$$\left[HMT_B(\mathbf{Z}) \right](r, c, d) = \left\{ \mathbf{Z}(r, c, d) \mid (\mathbf{B}_1)_{r,c,d} \subseteq \mathbf{Z}, (\mathbf{B}_2)_{r,c,d} \subseteq \mathbf{Z}^C \right\} \quad (34)$$

with $\mathbf{Z}(r, c, d)$ the 3D map with row, column and depth coordinates.

With such a map it is possible to search in the dataset to a crevasse with a specific shape. If a crevasse with a certain length, width and depth is defined in the SE, locations of similar crevasses in the dataset will be detected. However, if we want the method to be very general and able to detect all possible crevasses, an alternative approach probably suits better. In the next section a more general method that is also based on morphological filtering will be presented. In this approach crevasses are regarded as deviations from the glacier surface. It has the disadvantage that prior knowledge concerning the shape of the crevasse is not used in the detection, but the method is therefore more sensitive to atypical crevasses and requires less tuning of the SE.

5.1.3 Application for crevasse extraction

In his book, Serra (1988) writes about object recognition: "Recognition of an object simply means that all the rest has been eliminated from the scene." This is exactly what is done in order to extract the crevasses from the DEM and visualise their locations. It is rather difficult to see the

crevasses in the DEM, and similarly it is difficult to select only that part of the DEM that represents a crevasse. This is obvious when looking to figures Figure 5.5 and Figure 5.6 which represents a profile over a glacier with crevasses. Because of the difference in scale, the crevasses are difficult to see in large-scale relief of the glacier. In fact, detecting crevasses appears to be very similar to applying a contrast enhancement.

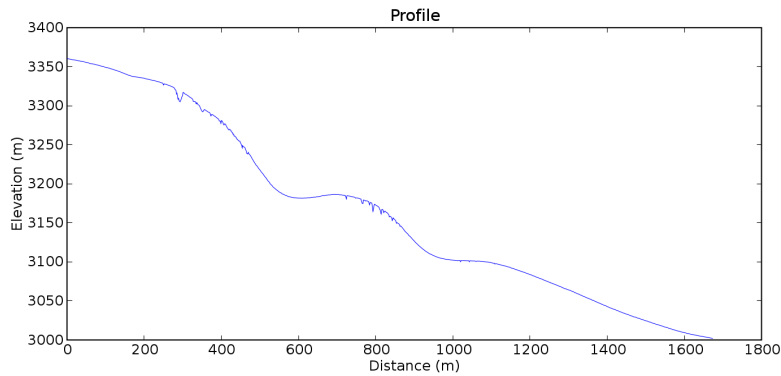


Figure 5.5 Profile intersecting approximately 1.5 kilometres of the glacier showing large scale relief and some crevasses.

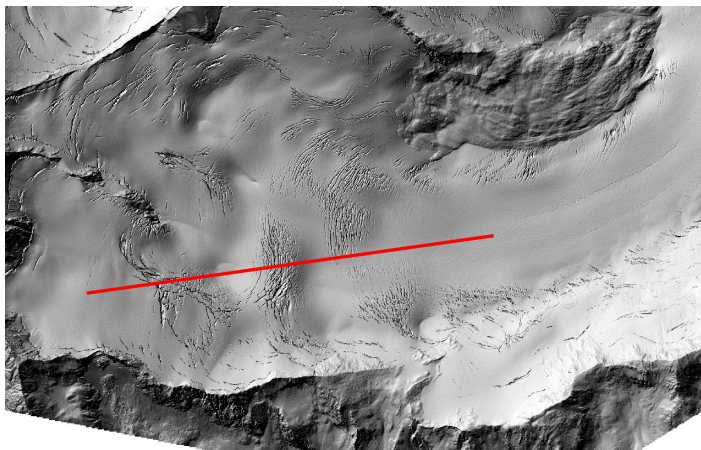


Figure 5.6 Location where the profile was taken.

If recognising an object is equivalent to removing everything else from the scene, then this means that a flat surface is to be created with only non-zero values at crevasse locations. The part that has to be removed from the original DEM is the glacier surface as well as the elevation of underlying bedrock. The physical meaning of these elevations would be a glacier in which no crevasses were formed. In order to obtain this surface, the previously introduced morphological closing filter can be used. Provided that the structuring element is larger than the width of the crevasse, the closing filter will close all crevasses, effectively removing them from the glacier surface. Figure 5.7 shows a profile of the glacier after performing the closing filter.

Having generated this surface of a glacier without crevasses, we know exactly what to eliminate from the scene. This simply means that the closed surface is subtracted from the original data, an operation that is known as Black Top Hat filtering:

$$BTH(\mathbf{X}) = \phi(\mathbf{X}) - \mathbf{X} \quad (35)$$

Applied to the DEM, the resulting dataset will be zero over the whole terrain, except for the locations with a crevasse. Given the DEM \mathbf{H} , the operation is now defined as:

$$\mathbf{H}_{\text{crev}} = BTH(\mathbf{H}) = \phi(\mathbf{H}) - \mathbf{H} \quad (36)$$

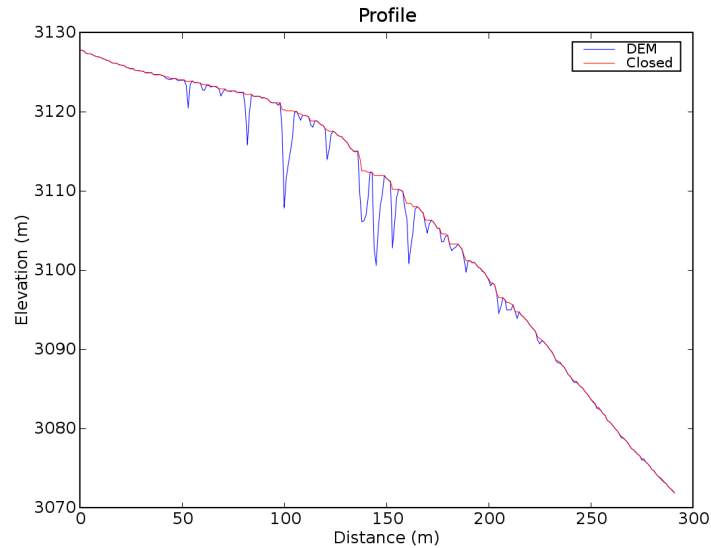


Figure 5.7 Profile of the Hintereisferner DEM and the result after closing. The crevasses have been closed.

Figure 5.8 shows the result of the Top Hat operation applied to the glacier dataset. The large scale topography has been removed from the data, giving the glacier a continuous elevation of zero. The crevasses stand out in this profile with their non-zero values. A binary image of the crevasse locations, i.e. a classification between crevasse and non-crevasse pixels, can now be made by applying a threshold to these values. For all detected crevasses, the depth can be found in \mathbf{H}_{crev} .

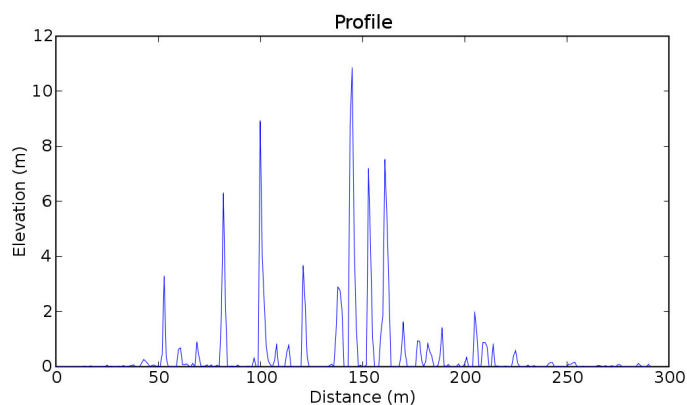


Figure 5.8 The result of the Top Hat operation. After thresholding the crevasse locations are known.

5.2 Setting the parameters

The whole method described in the previous sections relies on two variables. These are the structuring element used in the morphological

filtering and the threshold for the crevasse selection. Any shape can be used as a structuring element, but in practice there is only a limited set of elements that is used very often. This forms actually one of the weak points in morphology as it is hard to make a solid decision upon which structuring element to use. For the parameters that constitute the element, a rationale is given for the selected value.

5.2.1 3D structuring element values

The structuring element is used to define the shape and size of the morphological operations, i.e. to define the structure of the operation. In the described closing operation, the goal is to find the surface of the glacier without crevasses. Therefore, the structuring element should be set in such a way that it resembles the structure of the crevasse. There are three parameters to set for defining the structuring element: 3D values, size and shape.

A structuring element is often seen as a binary element using ones and zeros to define whether the part of the element belongs to the neighbourhood. However, the element can also be assigned values, making it essentially a 3D structuring element, in image processing also known as a grey-scale structuring element. The morphological filtering is used to extract the glacier surface from the DEM. The structuring element should therefore be designed in such a way that it resembles the morphology of the glacier. In chapter 4 it was postulated that a glacier can be recognised by several aspects, the most important one being its smoothness. If we hold this assumption and for the time being forget aspects concerning the slope of the terrain, one can state that the morphology of the glacier is just a flat plane. This can be resembled with a simple flat structuring element comprising ones and zeros to define a planer patch.

5.2.2 Structuring element size and variogram

The next aspect is the size of the structuring element, which is closely related to the consideration regarding the shape of the structuring element. The size of the element can be seen as a definition of how long (or how far) the morphology in the structuring element holds. In this specific case that can be translated to the question how "how large is the area in which the flat plane assumption holds?" After all, if we take a very large structuring element that exceeds the boundaries of the plane assumptions, small relief changes in the glacier will be detected as a crevasse too. This is because these relief changes appear like pits in the surface, when compared to a very large plane. The morphological closing filter will detect any pit in the surface, being a crevasse or not.

If we see the structuring element size as a way to express the frequency that is filtered, we can also say that a larger filter will filter lower frequencies. As a result, it could be that the shape of the glacier will be filtered too. On the other hand, a structuring element that is too small won't help us much either, as the filtered frequency will be so high that we run the risk of filtering some crevasses too. More concrete this means that the structuring element should be larger than the widest crevasse, because otherwise the crevasse will be filtered and thus not be detected. This is best seen in Figure 5.9 where a part of the crevasse remains undetected because of a structuring element that was smaller than the crevasse itself.

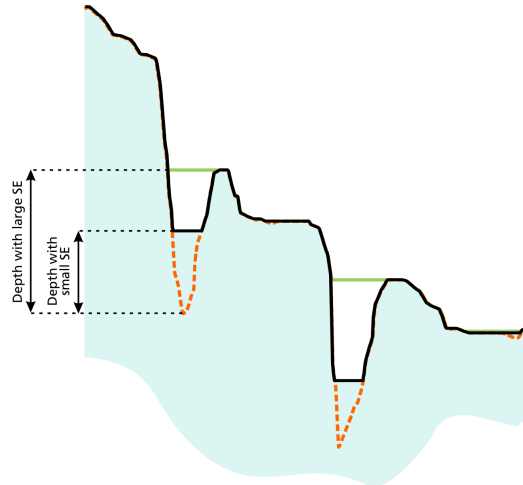


Figure 5.9 Effect of a small structuring element (SE) on the detected crevasse size. Compare with Figure 5.4.

Often, the correct filter size is determined by a well-educated guess or a simple trial-and-error approach. In this work, an attempt was made to formalise the selected structure size. In this new method, the variogram of the terrain is used to find a correct filter size.

A variogram is a measure of the variance between data as a function of distance, it is one of the second order moments besides variance and covariance. The theoretical variogram is defined as (Kitanidis 1997):

$$\gamma(d) = \frac{1}{2} E \left\{ [h(p+d) - h(p)]^2 \right\}$$

Where p is a point in the DEM, $h(p)$ the height of that point and d the distance from that point. A scatter variogram can be computed from the data by evaluating the variance between all pairs of points in the dataset. An example is shown in Figure 5.10 for a small part of the bedrock besides the glacier. The blue dots represent the combination of points with the variance. An experimental variogram can then be calculated by taking the mean of these dots in fixed bins.

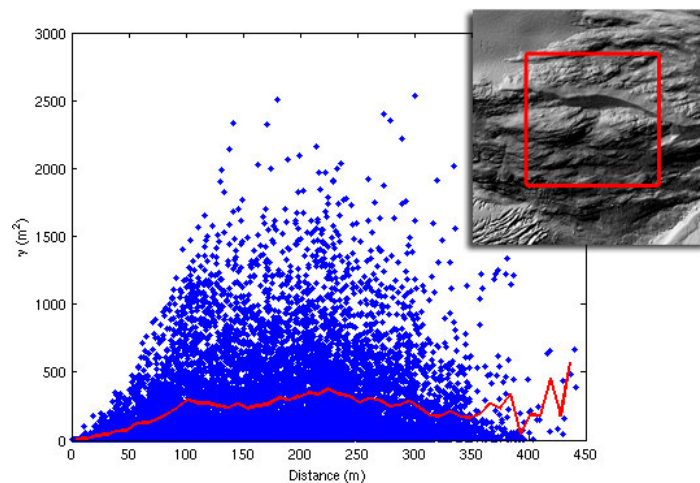
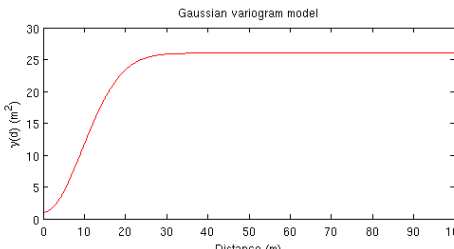
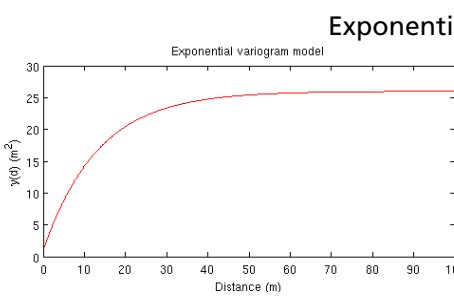
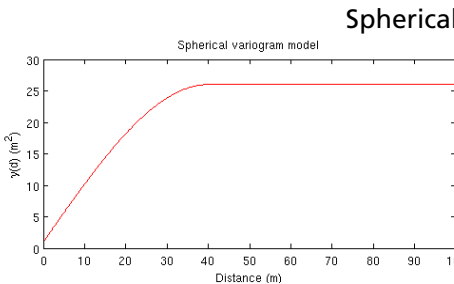


Figure 5.10 Experimental variogram for a part of the bedrock. Blue dots represent the variance between two points. The red line is the mean and represent the experimental variogram.

Similarly, a variogram can be calculated for a part of the glacier surface. Naturally, the variances will lay in a completely different order compared to Figure 5.10 as the surface of a glacier is much smoother.

Additionally, a theoretical variogram can be calculated from the scatter variogram represented by the blue dots. A theoretical variogram is based on a pre-defined model that is fitted through the blue dots. The advantage of a theoretical variogram is that it can be reconstructed with only three parameters and that it is by definition positive definite. This means that when the values are put in a matrix, this matrix is positive definite too and can therefore be inverted. There are three well known variogram models that are most used in practice, which are shown in Table 5.1.

Table 5.1 Common used variogram models. Calculated with range $R = 40\text{ m}$ and $\sigma = 5\text{ m}$. The range is the distance where the variance becomes constant.

<p style="text-align: center;">Gaussian Model</p>  <p>The plot shows the Gaussian variogram model. The y-axis is labeled $\gamma(d) \text{ (m}^2\text{)}$ and ranges from 0 to 30. The x-axis is labeled 'Distance (m)' and ranges from 0 to 100. A red curve starts at (0,0) and rises to a plateau of approximately 25 m^2 at a distance of about 30 meters, remaining constant thereafter.</p>	$\gamma(d) = \sigma^2 \cdot \left(1 - e^{-\frac{(3d)^2}{R^2}}\right)$
<p style="text-align: center;">Exponential Model</p>  <p>The plot shows the Exponential variogram model. The y-axis is labeled $\gamma(d) \text{ (m}^2\text{)}$ and ranges from 0 to 30. The x-axis is labeled 'Distance (m)' and ranges from 0 to 100. A red curve starts at (0,0) and rises to a plateau of approximately 25 m^2 at a distance of about 40 meters, remaining constant thereafter.</p>	$\gamma(d) = \sigma^2 \cdot \left(1 - e^{-\frac{3d}{R}}\right)$
<p style="text-align: center;">Spherical Model</p>  <p>The plot shows the Spherical variogram model. The y-axis is labeled $\gamma(d) \text{ (m}^2\text{)}$ and ranges from 0 to 30. The x-axis is labeled 'Distance (m)' and ranges from 0 to 100. A red curve starts at (0,0) and rises to a plateau of approximately 25 m^2 at a distance of about 40 meters, remaining constant thereafter.</p>	$\gamma(d) = \sigma^2 \cdot \left(\frac{3d}{2R} - \frac{d^3}{2R^3}\right), d \leq R$ $= \sigma^2, \quad d > R$

The theoretical variogram is now calculated by fitting the model through the scatter variogram, i.e. the blue dots, using the linearised least squares adjustment. Figure 5.11 gives the theoretical variogram based on the Gaussian model for a selected small part of the glacier surface. For comparison, the scatter- and experimental variograms are displayed as well. The values found after fitting the Gaussian model were $R = 369\text{ m}$ and $\sigma^2 = 1.3\text{ m}^2$.

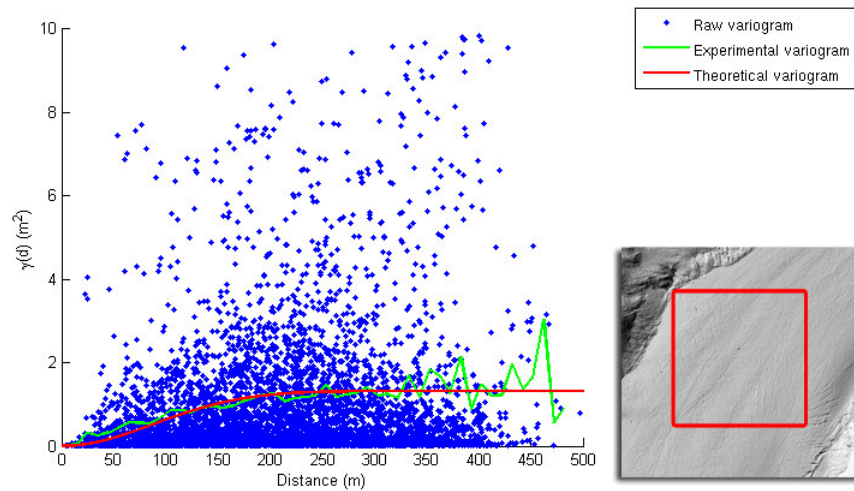


Figure 5.11 Theoretical variogram for the glacier surface.

The variogram presented above is a useful tool for analysing the morphology of a terrain. A comparison with the experimental variogram from Figure 5.10, taking scale differences on the axis into account, shows for instance a clear difference between the relief on a glacier and that on the bedrock. Apart from that, we can use the variogram to find a size for the morphological structuring element. The problem of choosing a size can now be translated to the problem of choosing the maximum variance that can be allowed on a glacier surface.

A value for the variance might be supplied manually to the program. This is still not much more reliable as giving a filter size. However, for glacier experts, the intended end-users of this method, giving this variance might be more natural than given a filter size. For instance, Figure 5.12 features a photograph that was taken on the Hintereisferner. A good window size may be hard to select from this view, but the variance of the surface can be inferred from minimum and maximum heights seen on the surface.



Figure 5.12 Surface of the Hintereisferner

From chapter 4 we found that the variance of a 11 by 11 meter part of a glacier is about 0.06 m^2 . Surface with a higher variance is most likely not part of the glacier. If we want to use the filter to reconstruct a glacier surface without crevasses, the largest variance that is acceptable is this 0.06 m^2 . If we take a maximum variance of 0.06 m^2 , the half filter size can be read from the variogram.

5.2.3 Structuring element shape and anisotropy

The variogram gives information on the variance between points in the terrain given a certain distance. The variogram, as it was shown in Figure 5.11, was based on the assumption of absence of anisotropy. Anisotropy is the difference in the property of a signal with changes in direction. In a perfect isotropic surface, the variogram will be equal for all directions. The appropriate choice for the structuring element will then be a disk, because the size that was derived with the variogram applies to all directions.

However, if the surface is anisotropic, the disk shaped structuring element fails to represent the anisotropy in the terrain. The anisotropy of the terrain can be quantified with a directional variogram. This means that a number of directions are fixed with a certain tolerance. For each of these directions a variogram is calculated. Kitanidis (1997) proposes to search for the two main directional axis that form the anisotropy. For these axes a structuring element size can be calculated from the variogram, which will result in an ellipse-shaped structuring element.

5.2.4 Threshold value

A low threshold value will result in a lot of noise added to the plot. Small scale relief in the surface will show up as speckle in the classified image (Figure 5.13). Because these small wells are usually not deep, a higher threshold will resolve this problem.

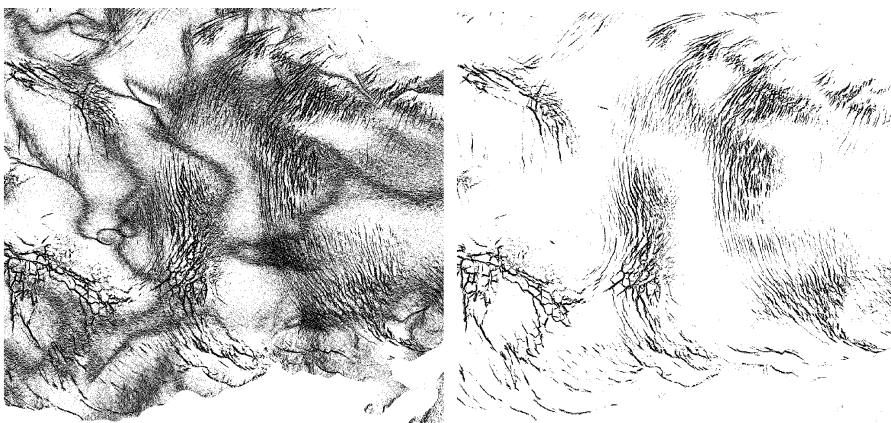


Figure 5.13 Selecting crevasses. Left: threshold = 0m; Right: threshold = 0.3 m.

Apart from being a necessary tool to distinguish noise and crevasses, the threshold value is at the same time also a generalisation parameter. If a full overview of all crevasses is required, the threshold value can be set to the lowest value that gives a reasonable result. This value is approximately the square root of the variance that was selected when choosing the structuring element size, giving all crevasses deeper than

30 centimetres. However, for many applications, such as glacier cartography, that amount of detail is unnecessary. In that case the parameter can be set higher, for instance to 1 meter or more. Because crevasses are approximately V-shaped, setting the crevasse depth linearly influences the width of the crevasses that is displayed.

5.2.5 Effects of glacier slope

Temperate glaciers like those in the Alps are located in the high mountain areas. It is therefore not surprising that the glacier surface will have a certain inclination. This inclination is not taken into account in the structuring element for the morphological filtering, which results in a bias in the detected crevasse depth. Because the filter closes the crevasses horizontally, the filtered surface is not exactly a surface without crevasses. This is depicted in Figure 5.14 where the red line shows the horizontal closing of a crevasse. Although definitions differ, the depth of a crevasse is usually related to the volume a crevasse in the glacier could have. This volume is represented with the dotted orange line.

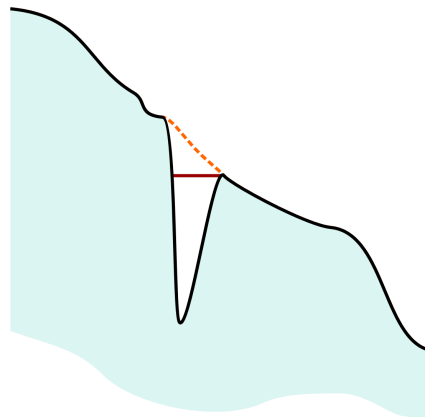


Figure 5.14 Crevasse closing under inclination. The red line is the result, the orange line is what it should be.

One solution is to detrend the data first, so that the horizontal closing gives the correct result. This detrending of the DEM, i.e. removing the large scale relief features, can for instance be done by top-hat filtering with a very large window size. Another method is to partition the DEM in a grid with cells of 10 by 10 pixels and search for the highest point in every cell. A simple interpolation method that transforms these points back to a surface with the same resolution as the original DEM, results in a trend surface. This trend surface can then be subtracted from the DEM after which only small scale relief and crevasses remain for the morphological filtering. When this approach is used, the crevasses will be filled in the correct way and give the correct depth after the top-hat filtering.

5.3 Limitations of ALS based crevasse detection

With Airborne Laser Scanning only visible crevasses can be detected. Often crevasses are covered by snow, making them invisible for the observer on the glacier, but also for the laser scanning system. The crevasses with a so called snow bridge are actually the most dangerous ones. Figure 5.15 features an extreme example of a snow bridged

crevasse at Antarctica. Alternatively, the crevasses can be filled with snow up to the bottom of the crevasse. For mountaineering these are logically much less dangerous. Sometimes these crevasses can be recognised by intensity differences in the data. The problem of snow cover for glacier monitoring is not new for Airborne Laser Scanning. Glaciologists have struggled with the problem of snow cover ever since measurements on glaciers are made. Also with other techniques like photogrammetry, or field work, these problems of snow cover existed.



Figure 5.15 A snow bridged crevasse at Antarctica. This crevasse was 10 meters wide and 25 meters deep. (Photograph by: Russ Alger, U.S. National Science Foundation)

Apart from the snow cover, the crevasses can be hidden to the observer in other ways too. Although most crevasses have a V-shape, i.e. wide opening on top and small on the bottom, some crevasses are formed like an A. This can happen on the concave parts on hills where the upper part of the ice is compressed, while the lower parts will be stretched. Because of their small opening at the top they can not be measured.

5.4 Results

Within the GRASS GIS a program was written that performs the detection of the crevasses. As input it takes a raster based DEM and values for the structuring element size and threshold. The output is a binary raster map that displays the locations of the crevasses. The structuring element shape and values have been fixed to a disk and a flat plane respectively. The program is based on the morphological operations that have been described in earlier sections of this chapter. The steps executed by the program are visualized in Figure 5.16. The identified sub programs, usually starting with the letter 'l' as prefix, were written to extend the functionality of GRASS.

The crevasse detection algorithm was tested on the ALS dataset of the Hintereisferner. The epoch, dated of August 2003, was used to test the method. This epoch has the nice property that most of the snow had melted at the time of acquisition. Many crevasses in the glacier are

therefore revealed compared to other datasets. The area of interest was selected from the glacier by calculating the glacier boundary with the method that was explained in chapter 4.

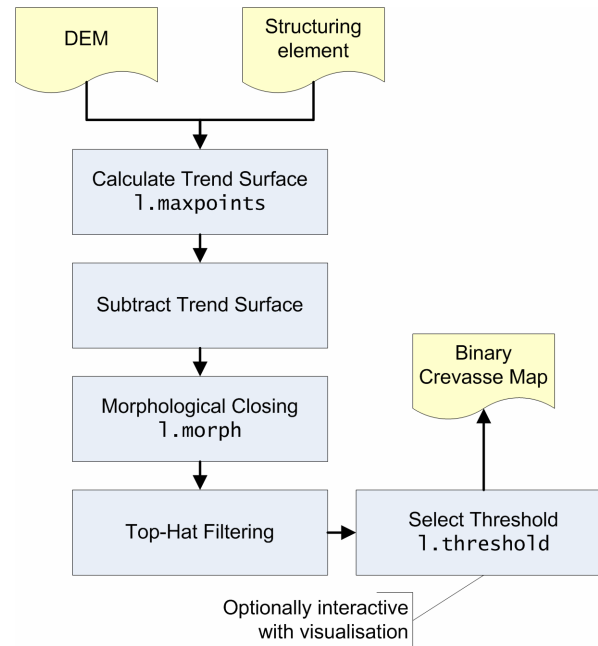


Figure 5.16 Flow chart of crevasse detection method.

For the calculations, a filter size of 10 meters was chosen. The threshold value was selected interactively with a selection program. This program lets the user supply a threshold value and shows the result, after which the user can correct the value in order to see any potential improvements. Eventually, the best results were obtained with a threshold of 0.4 meter.

Figure 5.17 shows the result after Top Hat filtering. The locations of the crevasses can clearly be distinguished from this plot. The next figure (Figure 5.18) shows the result after applying the threshold.

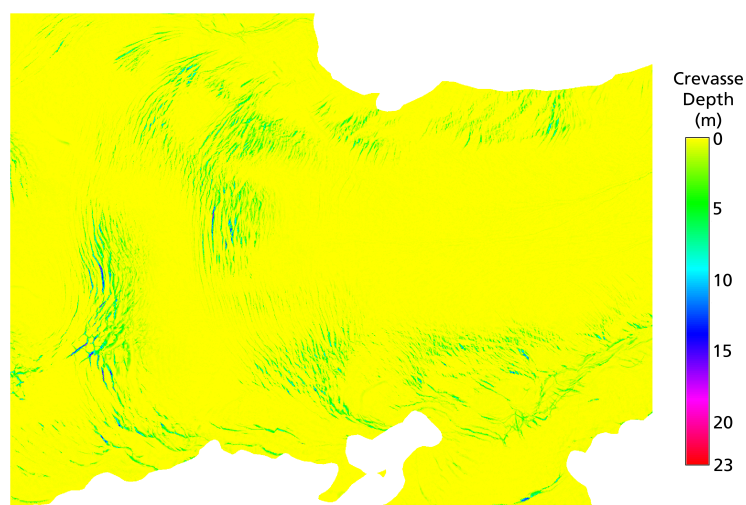


Figure 5.17 Result after top hat filtering. (Closed Surface –Surface)

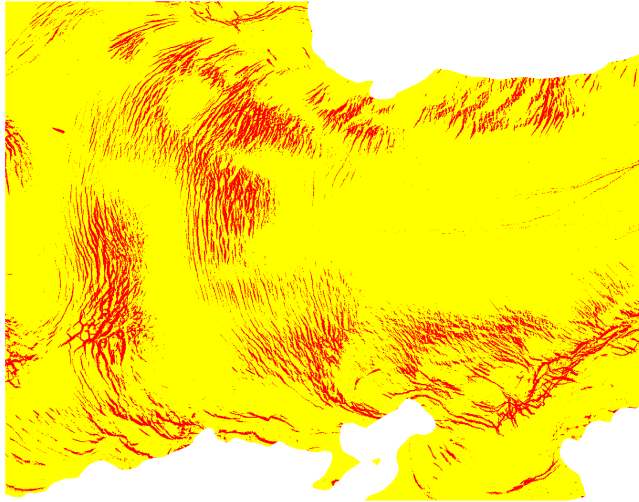


Figure 5.18 Detected crevasse locations

In the absence of any reference data, it is virtually impossible to assess the quality of the results. Two aspects should be considered here. The first aspect is the geometrical accuracy, i.e. the correctness of the delineation and depth of the individual crevasses. In chapter 6 a more elaborate analysis of this accuracy can be found. The second aspect is the errors in the classification. In other words, how many type I errors, crevasse classified as glacier surface, occur in the result? Contrary to that, how many type II errors, glacier surface classified as crevasse, occur? An explanation cannot be obtained by a field visit either as the situation will have been changed in the course of years.

However, a visual inspection can be made if we overlay the detected crevasses over an orthophoto. These orthophotos have been made independently but approximately at the same time. Figure 5.19 shows a small part of the orthophoto and the same orthophoto with the detected crevasses overlaid. A visual inspection revealed no important missing crevasses. Note however, that it is hard to find crevasses from aerial photographs because of the lack of contrast.

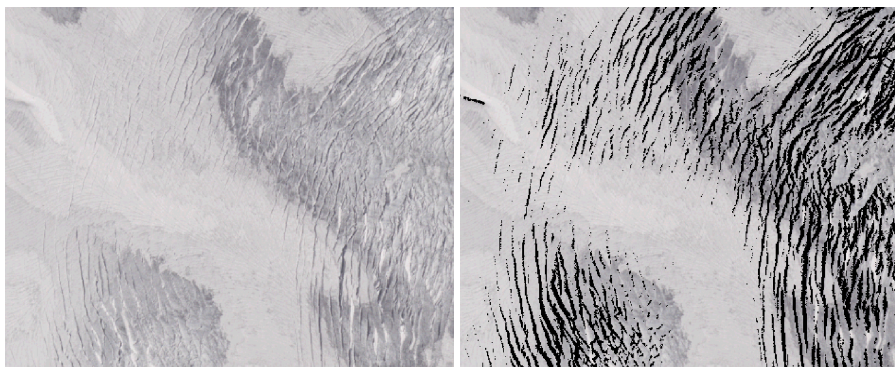


Figure 5.19 Verification with orthophoto.

With the program I.feature, which was developed by the LiSA team (Rutzinger et al. 2006) one can also calculate specific properties for each individual crevasse like maximum depth, volume or shape index (ratio of perimeter and area). Figure 5.20 shows the detected crevasses, coloured with respect to their depths.

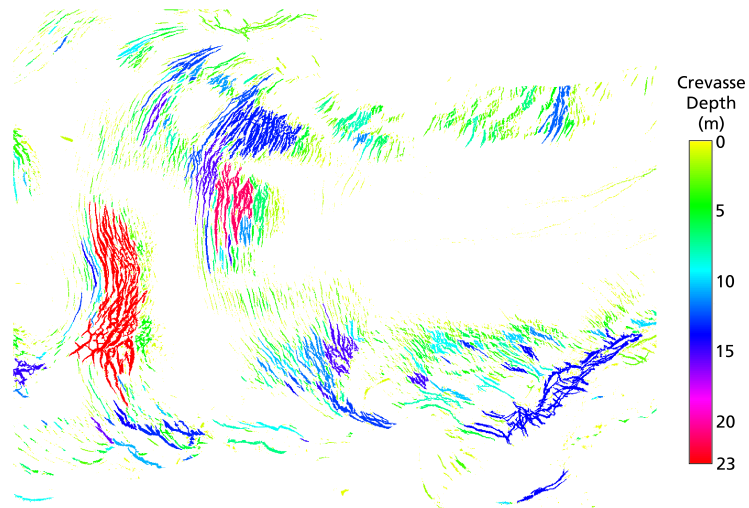


Figure 5.20 Figure showing maximum depth of crevasses.

The detected crevasses can be converted to vectors and stored in any ordinary vector format. A Douglas-Peucker line simplification was implemented to reduce the data load of the converted vectors.

5.5 Reconstruction of crevasses

5.5.1 Introduction

In the previous sections work was done to detect the locations of the crevasses and assign a depth to them. This data can be used for numerous applications. However, it doesn't tell anything about the exact shape of the crevasse. A method to derive the shape of a crevasse from ALS data will be described in this section.

Figure 5.21 shows a cross-section of a crevasse in a glacier. The V-like shape of this crevasse is typical for most crevasses that appear on a glacier.



Figure 5.21 Side view of a crevasse (© Corbis)

Whilst the delineation of the glacier and detection of crevasses was solely performed on height measurements that were interpolated to a grid, the reconstruction of crevasses will be done on the unprocessed point data. The reason is that during the interpolation from points to a DEM surface, information gets lost. This is not so relevant for small scale operations that were described before, but on this more detailed level the effects are notable. Figure 5.22 shows what the effect of interpolation is. The interpolated surface does not reach to the deepest points of the crevasse, showing a clear loss of information.

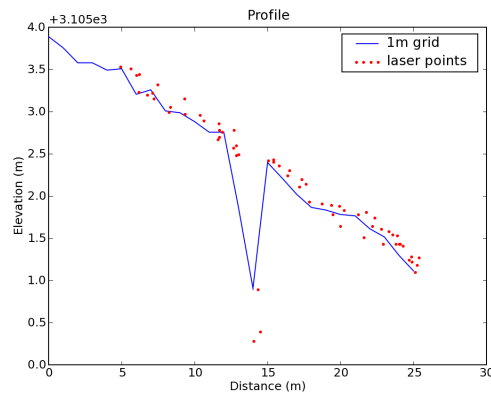


Figure 5.22 Laser points interpolated to a 1m grid.

The problem with reconstructing crevasses is the low number of points that is available from the ALS measurements. This requires a method that can fill these gaps using a couple of assumptions. The following three different methods were identified as possible ways to do this:

- Fitting a parameterised crevasse
- Adjusting spline interpolation
- Boundary representation

The fitting of a parameterised crevasse seemed to be a quite promising method for reconstruction. Crevasses are generally believed to have a regular V-like shape. If the shape of a crevasse is that regular, it should be possible to parameterise this shape into a geometrical object. This object can then be fit into the point cloud, giving values for the parameters of this object. The question of concern is now how to parameterise the crevasse. A simple parameterisation would consist of parameters for depth, width and length. Unfortunately, crevasses are not that simple: they are usually bended and don't have a constant depth. Describing this in parameters is quite cumbersome if not infeasible. One could wonder whether it makes sense to fit something with the best unbiased estimator if the parameterisation is most likely to be imperfect. The two other methods were therefore selected to reconstruct the crevasse. These are described in the subsequent sections.

5.5.2 Adjusting spline interpolation

The problem of crevasse reconstruction is that the number of measured points is too small to see all details of the crevasse. Assumptions are required to find values for the in-between point locations. A well known method for this purpose is interpolation with splines. The mathematical definition of splines is analogous to the wooden splines as they were used in ship building. Wooden boards were fixed at a number of points,

so that the board would bend towards a situation where the overall tension on all points was minimal.



Figure 5.23 Wooden splines on the hull of a Viking ship. (Source: Oslo Viking Ship Museum)

Exactly this can be done with mathematical splines. A bivariate spline, i.e. a 2-dimensional spline surface, can be seen as one plate that is laid over a series of fixation points. These fixation points are the ALS points in this case. The plate will be fixed on these points and bend to a shape that gives minimal tension, i.e. a smooth surface. The surface does not have to touch the points, it is not an exact interpolator. Because of this smoothing behaviour, the spline interpolation is very useful for interpolating natural phenomena. For this purpose the Regularised Splines with Tension implementation was used (Mitas and Mitasova 1999). This program allows the fitting of a bivariate spline with parameters for tension and smoothing.

The tension parameter changes the surface from a stiff plane to a flexible plate. The role of the tension parameter can be seen as the distance over which a point influences the interpolation. With interpolations with a very high tension, the points influence the shape over a long distance. The smoothing parameter on the other hand tells how much the surface may deviate from the observations. If a larger deviation is allowed, the resulting surface will be smoother compared to the case where the surface has to go through exactly all points.

The reconstruction was performed by interpolating the bivariate spline through all the points using a high value for the tension, as points were expected to have a far reaching influence. The smoothing parameter was set relatively low, so that the surface follows the available observations closely. For the example presented in this section, the tension was set to 160, the smoothing to 10. However, there is no physical quantity related to these values.

From the interpolated surface, it is easy to generate a 3D model of the crevasse. This is done in Figure 5.24 where a 3D view is generated from the interpolated grid. Additional values, like volume and length of the crevasse, can easily be derived from this data.



Figure 5.24 Model of a crevasse, interpolated with splines.

5.5.3 Boundary representation

Another method is what we refer to as a boundary representation. Sticking to the similarity to a V-character, we can build a crevasse from two boundaries, the left and right side of the V-shape. This boundary is formed by three lines: a bottom line and two surface edges. These three lines are connected at the beginning and the end of the crevasse. Now, a crevasse can be reconstructed by deriving these three lines and assuming that the surface that connects the bottom with the edge lines is linear.

The bottom line is the first line to extract, using a process that is split in two steps. The first step constitutes the determination of the horizontal position of the line. In the second step, z-coordinates are calculated for this line. It is unlikely that there are any laser points that lie exactly on the bottom line of the crevasse. Even if there were points that do so, it would be impossible to know. The location of the bottom line must therefore be derived from the surrounding pixels. It seems reasonable to assume that the horizontal position of the bottom line lies in the middle of the crevasse. The program selects the lowest 25 percent of the points in the crevasses and calculates the centre by fitting a line through these points. This line is an 8th-order polynomial, but could just as well have been a spline or any other model. After fitting, the program removes points with a large residual and tries to fit the line again, giving an improved position. The fitted line through the points is shown in Figure 5.25.

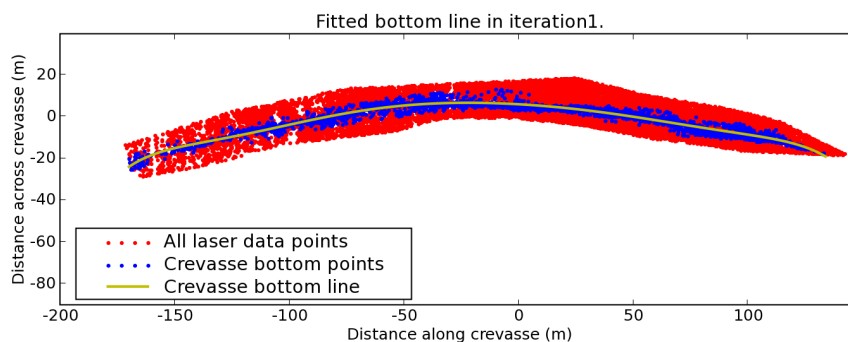


Figure 5.25 Fitted line representing the bottom of the crevasse.

The result of the fitting of the polynomial can further be improved by giving weights to the points used in the adjustment. Points with a large depth are given a higher weight in the adjustments, while points near the surface, get a lower weight. Though the result might be improved slightly, the processing time for the adjustment grows considerably. We call the estimated polynomial coefficients \hat{x} , the weight matrix W and the design matrix A . The stochastic x and y coordinates are combined in the random vector y . The Least Squares Adjustment is then given by:

$$\hat{x} = (A^*WA)^{-1} A^*Wy$$

Then the weights are derived as:

$$w_{ii} = \left(\frac{h_i - h_{\min}}{h_{\max} - h_{\min}} \right)^2 \quad i = 0, 1, \dots, n-1$$

If the horizontal position of the line is known, the vertical elevation can be attached to the line. Figure 5.26 shows a plot with the distance along the line and the height of the selected 25% lowest points. From these points the height of the bottom should be derived. There are many different ways to do this, all with completely different results:

- always taking the lowest point;
- fitting a smooth spline;
- take the convex hull;
- etc.

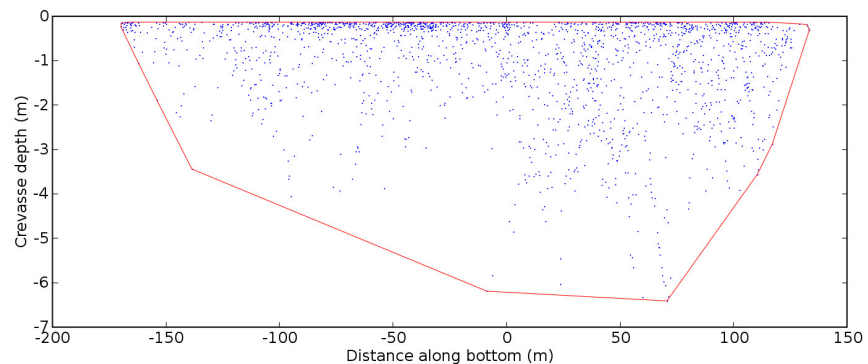


Figure 5.26 Laser points in the crevasse. Bottom is modelled as the convex hull.

The figure shows that in this case the depth of the crevasse is modelled as the convex hull of all the points. This is based on the assumption that missing deep points are due to snow bridges in the crevasse. However, there is no way to verify if the convex hull is indeed the best solution. This overall ambiguity of the correct crevasse bottom model is actually the most important disadvantage of reconstruction from ALS data.

The remaining step comprises the modelling of the edge of the crevasse, the line where the glacier stops and the crevasse starts. Like the bottom line, this feature is hard to find from the sparse number of laser points. For finding the edge, profiles of points were generated perpendicular to the bottom line that was found before. For the selected test crevasse, 152 profiles were made with an in-between spacing of 2 meters. In each of these profiles, the locations of the left and right edge were searched independently. Figure 5.27 displays one of these profiles.

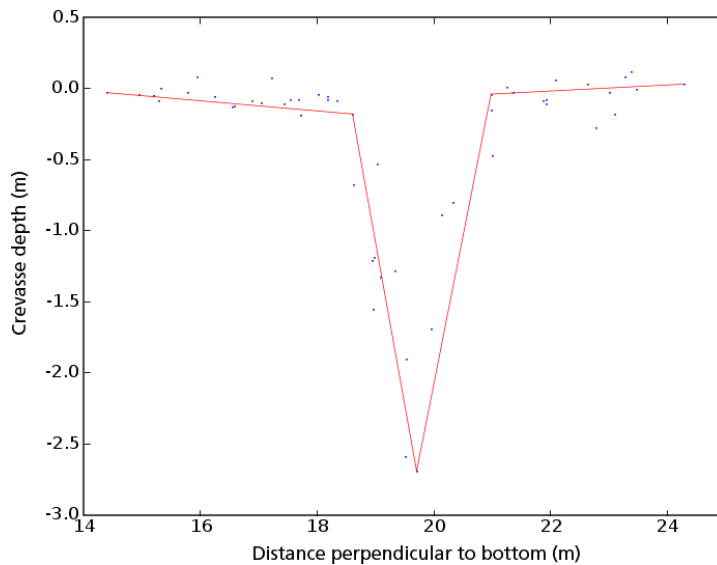


Figure 5.27 Points in profile perpendicular to bottom line. The red line is the result of the Douglas-Peucker algorithm.

The edge points should now be found from this profile. Different methods were considered. One successful one was calculating a line through the points using the Douglas-Peucker algorithm (Douglas and Peucker 1973). This algorithm is a line simplification algorithm, it reduces the number of points using a given tolerance. Applied to the crevasse data, it will return only the interesting points of the crevasse: the two edges and a bottom point.

For all profiles, the edge points can now be found by taking the second and one but last point from the simplified line. If this is done for all profiles across the bottom line, a large number of points left and right of the bottom line is found. These points can be connected using an interpolating spline. This gives the three lines of the crevasse and with these the crevasse is fully reconstructed. The points that are found on the three lines can be used for generating a Triangulated Irregular Network (TIN) as a first step to a 3D model. From the TIN, it is possible to calculate values for the volume and shape of the crevasse.

5.5.4 Reconstruction implementation

Like the other methods presented in this thesis, the crevasse reconstruction was implemented as a module for GRASS GIS. The outline of the implementation is depicted in Figure 5.28.

The programs reconstruct one crevasse at a time. The crevasse locations can be identified from the binary crevasse locations map that was generated with methods described earlier in this chapter. Each crevasse is buffered with two meters to include additional points around the crevasse. This area forms the boundary in which points can be extracted from the database.

These points are fed into the reconstruction program. However, this program works by taking the lowest 25% of points to find the location of the bottom. An inclined glacier surface would bias this selection of the lowest points. Therefore these points have to be detrended first, which can be done with the trend surface that was introduced earlier in

this chapter for the crevasse detection. From the pixels in the trend surface a correct trend value for every individual laser point is calculated using bilinear interpolation.

After that, the program runs the boundary representation algorithm described above. It results in a file with the coordinates of the three lines. These can be read into any program for further processing.

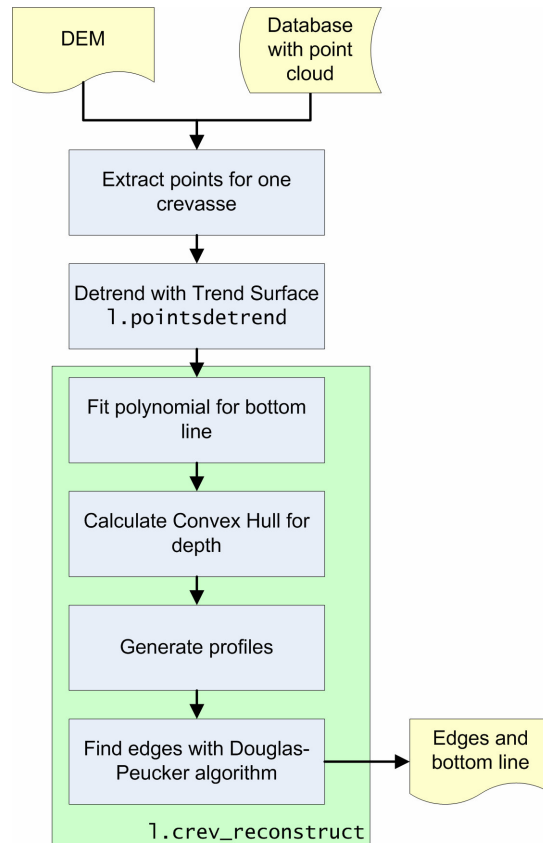


Figure 5.28 Flow chart of the crevasse reconstruction process.

6 Quality Analysis

In chapter 4 a methodology was developed for delineating the glacier surface. In chapter 5 the process was described for detecting and reconstructing crevasses in the glacier surface. However, until now no investigation was done in the suitability of the laser scanning data for this purpose. In this chapter, the quality of the crevasse reconstruction will be analysed.

When modelling the crevasse, many a-priori assumptions concerning the shape of the crevasse can be made in order to improve the result. Assuming a perfect V-like shape for instance, will give good results in most cases. In any case, the depth of the crevasse is an important parameter that has to be retrieved from the data. It is rather questionable whether the real depth of a crevasse can be assessed with Airborne Laser Scanning.

First of all, natural conditions such as snow filling up crevasses can prevent laser signals to reach the ice-bottom of the crevasse. The dangerous snow bridges are examples where a crevasse is completely hidden due to snow. Basically, this is a limitation of any sensor system operating in the visible or infra-red portion of the electromagnetic spectrum. The effect of snow cover and snow filling in crevasses can potentially be predicted using meteorological data and specialist's experience. For now it is assumed that the glacier is snow-free, which is a valid assumption for some of the datasets that were acquired in the summer of 2003.

6.1 Sampling interval

More fundamental is the problem concerning the sampling frequency. With a resolution of on average only one point for every square meter, the chance that a crevasse cannot fully be reconstructed is quite high. One way to analyse the correctness of the reconstruction is by using the Nyquist-Shannon sampling theorem, also known as the Kotelnikow theorem after the one who invented the theorem independently at the same time (Kotelnikow 1933). This theorem describes the process of sampling a continuous signal by a finite number of points, and the process of reconstructing the continuous signal again from these points. A practical consideration that follows from this theorem involves the Nyquist rate, which is defined as twice the bandwidth of the continuous signal. In order to reconstruct the signal, the sampling frequency should be greater than the Nyquist rate.

6.1.1 Effects of a low sampling interval

We know that the sampling frequency used in the measurement campaigns on the Hintereisferner is too low for fully capturing all the details from a crevasse. The effect of the too low sampling rate is depicted in Figure 6.1. As can be seen, the under sampling causes the deepest point of the crevasse to remain undetected. Additionally, an

apparent shift of the crevasse with a magnitude of at maximum half the sampling interval is caused.

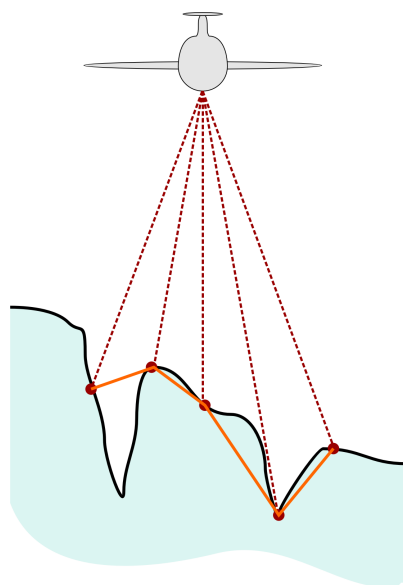


Figure 6.1 The effect of a low sampling interval. Note that the resulting orange line is exaggerating because of the differences in scale.

The obvious solution would be to increase the sampling rate. However, as explained in chapter 3, the measurement resolution is a trade off between available financial resources, operational constraints such as the airplane's flying height and technical limitations of the instruments. At this moment, the optimal sampling interval is at about 1 point per meter, which is most likely to be the case for the coming years as well.

As explained, due to the under sampling, there is a chance that the lowest point was missed during scanning. It is interesting to quantify the greatest error that can occur due to under sampling. For the purpose of quantifying this effect, a standard crevasse will be defined. The standard crevasse is a geometrical simplified shape that is representative for most crevasses in the dataset. Figure 6.2 shows the shape of the crevasse.

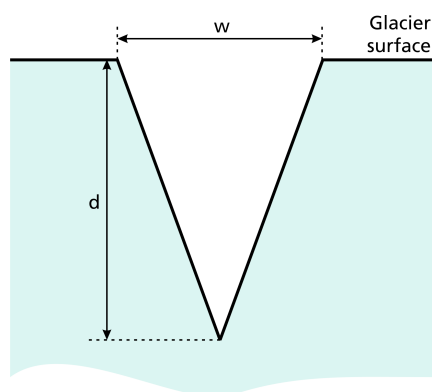


Figure 6.2 Shape of the standard crevasse

One way to quantify the error is by looking what the maximum error due to under sampling will be. In the case of a 1 meter sampling interval, the largest error occurs when both points lay on a distance of half a meter from the crevasse bottom. The line that is derived after

interpolation has in this case the largest error. The measured depth can be expressed as a percentage of the real depth. This percentage can easily be derived with:

$$\text{depth fraction} = \left(1 - \frac{s}{w}\right) \cdot 100\% \text{ for } w > s \quad (37)$$

where w is the width of the crevasse and s the spacing between samples. For a sampling interval of 1 meter, Figure 6.3 shows the maximum possible error.

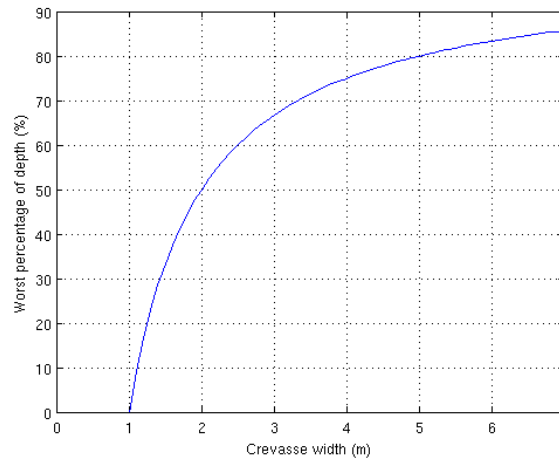


Figure 6.3 The measured depth of a crevasse in the worst case.

Of course, the situation sketched here is a pessimistic one; there is an equal chance that the samples are taken on any other location in the crevasse, giving a deeper depth. Additionally, because crevasses are elongated features, more samples will be taken inside the crevasse increasing the chance that at least at one point the correct depth is measured.

6.1.2 The optimal sampling interval

As the ALS technology improves, more possibilities will be available in the future to improve the resolution, i.e. the sampling interval, of the laser data. With a higher resolution, a better reconstruction of the crevasse with a smaller error can be made.

What is this optimal sampling interval? According to Shannon-Nyquist theorem, the sampling frequency should be twice the frequency of the signal. Determining the double frequency of the crevasses in a glacier surface is a cumbersome task and not very helpful to solving the problem. The discrete Fourier transform of a V-shape gives an incomprehensive collection of frequencies that are difficult to use for finding a correct sampling frequency. In fact, an infinite high frequency is required to accurately reconstruct the V-shape.

However, the exact reconstructing of the V-shape is not required, because the shape is already known. Reconstructing the shape is possible when at least two points are available at both sides of the crevasse, but for a reliable reconstruction three or more points are necessary. Clearly, the required interval is related to the steepness of the

crevasse, which is on its turn related to width and depth. The question on the required point density can now be answered by considering the width of the crevasses that one wants to reconstruct. A minimum width of 2 meters is a safe assumption in this case. That implies a point density of 9 points per meter squared. In some areas of the glacier this density is achieved because of the overlap of adjacent strips. Additionally, the point density can be increased by using both the first and last pulse measurements.

6.2 Effects of scan angles

As explained in the previous chapter, the sampling interval is one property that may cause error in the DEM. Something else is the orientation of the crevasses. The laser pulses are transmitted in a swath of 20 degrees in both directions. Consequently the pulses can make an angle with the surface of 70 degrees. In practice that means that parts of the glacier remain hidden for the ALS system when the walls of the crevasse are steeper than 70 degrees. It was unclear what the influence of this scan angle is on the maximum derived depth. Figure 6.4 shows the situation where the deepest point of the crevasse is occluded at scan angle θ_s .

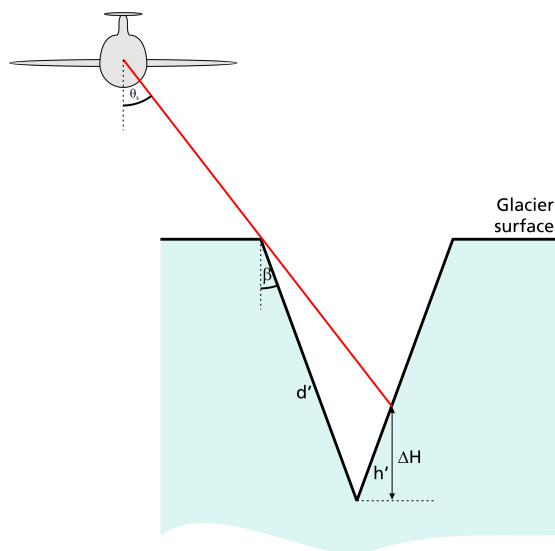


Figure 6.4 An error of ΔH as a result of the scan angle θ_s .

Assume that a laser pulse is emitted in such a way that it just enters the crevasse like in Figure 6.4. Unless the crevasse will be measured again from another airplane position, this is the deepest point of the crevasse that will be acquired. Given the width w and depth d of a crevasse, the error in height can now be calculated using simple geometrical relations.

Given that

$$\beta = \arctan \frac{\frac{1}{2}w}{d}$$

then the diagonal error h' is

$$h' = \frac{d' \sin(\theta_s - \beta)}{\sin(180^\circ - \theta_s - \beta)} \quad (38)$$

Now the vertical error follows easily from the conformity rules.

$$\Delta H = \frac{h'}{d'} \cdot d = \frac{\sin(\theta_s - \beta)}{\sin(180^\circ - \theta_s - \beta)} \cdot d \quad (39)$$

Assume a crevasse of 3 meters wide and 10 meters deep so that β is 8.5 degrees. Additionally we assume that the crevasse will be measured in the extreme situation where the scan angle is 20 degrees. The depth error is now 4.1 meter. However, this number should be mitigated immediately. The overlap between two strips in the Hintereisferner project is 155 meters which is almost a quarter of the full strip width. As a result, the extreme scan angle will seldom be more than 10 degrees. Using the same crevasse again, the height error is now only 0.8 m; an error of 8 percent of the depth.

The example above was made with a crevasse of 10 meters deep. This seems to be representative, because there are hardly crevasses deeper than 10 meters in the dataset. The histogram in Figure 6.5 shows the number of crevasses with a certain depth.

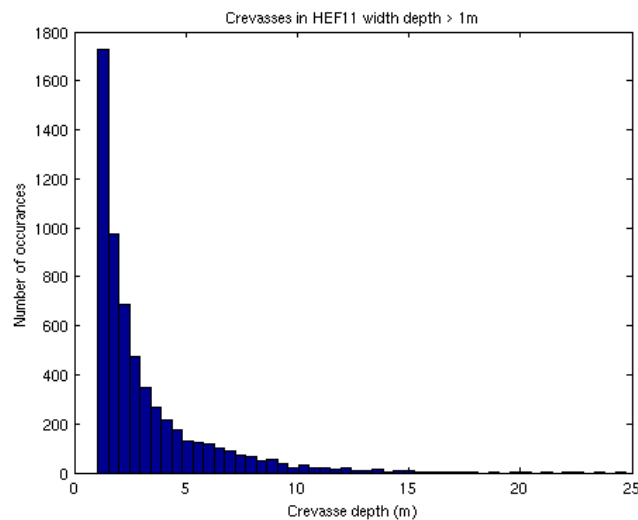


Figure 6.5 Occurrence of crevasses in 2004 dataset.

6.3 Laser intensity

One problem with laser scanning on glaciers is the reflectivity of ice. Although one often thinks of ice as a very white surface which should therefore reflect very well, this is not necessarily the case in practice. Figure 6.6 shows the spectral reflectivity curves for ice and snow. The reflectance of visible light lies between 400 nm and 800 nm. The wavelength of the infrared beam of the laser scanner is 1024 nm. From the left plot in Figure 6.6, it is clear that the overall reflectivity of ice is not good for both visible and infrared light. In the case of a real icy surface, this means a reflectivity between 1% and 2%.

Glacier surface analysis

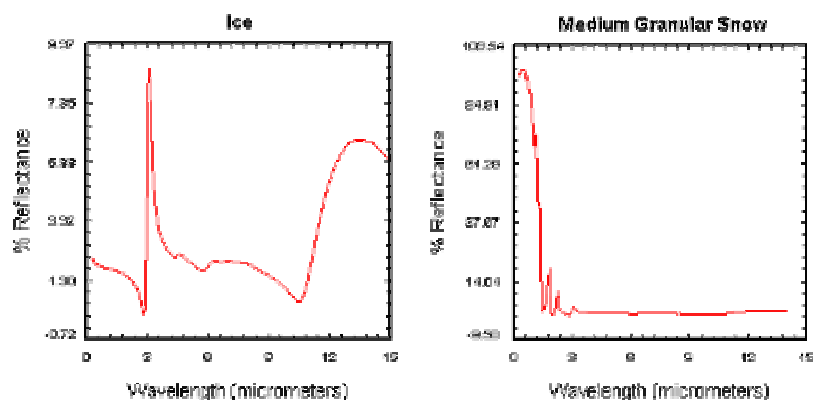


Figure 6.6 Spectral reflectance curves for Ice and Snow. (Reproduced from the ASTER Spectral Library, California Institute of Technology)

A surface covered by snow has a much higher reflectance, especially in the visible light, but for infrared light with a wavelength of $1 \mu m$ the reflectivity is still 31%. It is exactly for this reason that the Laser instrument of the ICESat mission uses two wavelengths: infrared and green visible light (CSR GLAS 2006). These two wavelengths can be used for a good distinguishing between ice and snow. It would be advantageous to use visible light for the laser scanning because of the better reflectance. However, the propagation through the atmosphere is worse for these signals compared to infrared.

These graphs can be compared with the spectral reflectance curve for rocks. There are different types of rocks around the glacier and some layering of rocks is apparent too. According to some geological maps, one of the rocks found is Gray Slate, with a spectral reflectance curve depicted in Figure 6.7. It shows that the reflectivity of rock is much better than the reflectivity of ice and similar to that of snow.

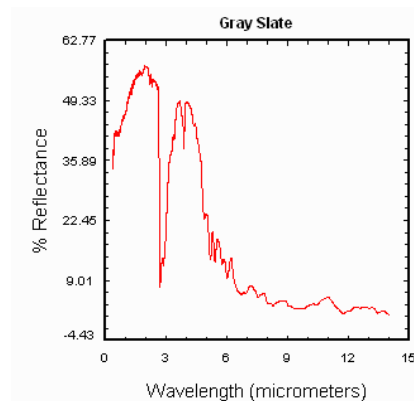


Figure 6.7 Spectral reflectance curve for Gray Slate, one of the rock types found around the glacier.

This is also visible from Figure 4.13 which shows the corrected image with intensity values from the laser scanner. A large part of the glacier was covered by snow at the time of acquisition, causing the glacier to show up brighter than the surrounding rock in the intensity image. Near the snout of the glacier, the area lies below the equilibrium line. Below this equilibrium line, the air temperature is above pressure freezing point, thus causing melting of the glacier. In this area there is no snow

and possibly also a thin layer of water covering the glacier. Clearly, the reflectivity of this area is lower than the surrounding bedrock.

Lutz et al. (2003) also investigated the reflectivity of ice in ALS data, but on another glacier: the Svartisheibreen glacier in Norway. They found similar values for the reflectivity of ice, i.e. the received laser intensity after applying a simple correction. See Figure 6.9 on the next page for the results of reflection on different types of surfaces.

The difference in reflectance has a couple of implications. One important one is that one should indeed be careful with using intensity images for classification. There are too many factors that influence the intensity in the image. Another implication is that mere ice surfaces can give problems because only little energy will be returned. As a result, the Signal to Noise Ratio (SNR) will be worse for good reflecting objects. In case of a bad SNR, the measurement will not be reliable and disregarded in the ALS post processing. This effectively decreases the number of available points. Especially in crevasses a very pure ice surface is visible, mountaineers know this as the very bluish colour that is visible when looking in a crevasse (Figure 6.8). This means that it is likely that there is a higher chance of missing points in a crevasse than on other parts of the surface.



Figure 6.8 Blue colours in a crevasse. (©Martin Harvey/Corbis)

Glacier surface analysis

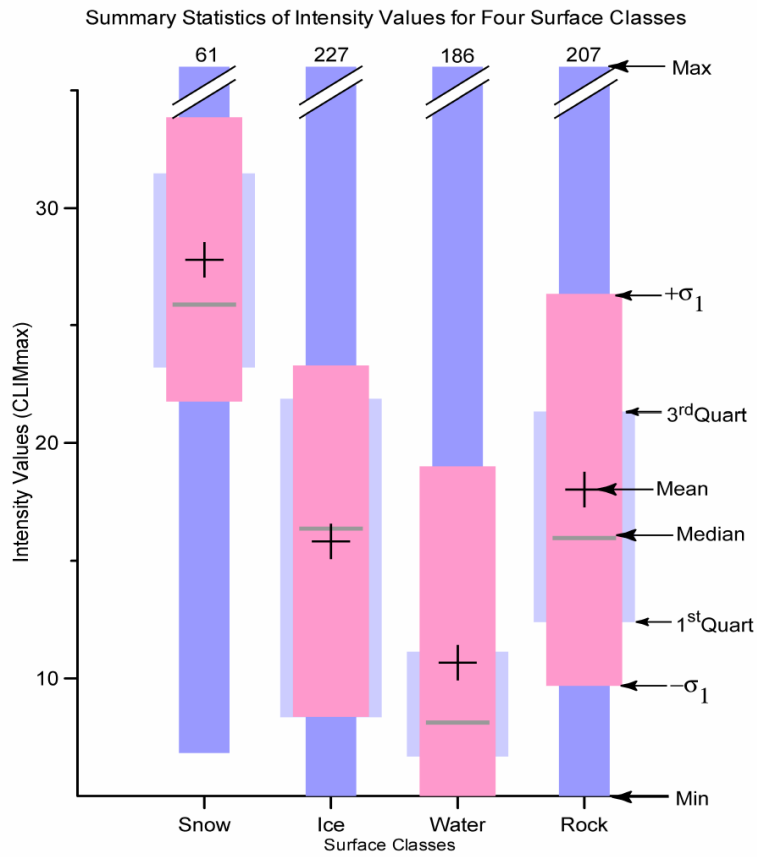


Figure 6.9 Intensity for different surface classes on the Svartisheibreen in Norway.
(Source: Lutz et al. 2003)

7 Conclusions and future work

In the previous chapters methods were presented for processing Airborne Laser Scanning data for the delineation of glaciers and the detection and reconstruction of crevasses. A number of quality considerations were given too. This chapter completes the thesis by giving final conclusions concerning the presented work.

7.1 Conclusions

The hypothesis that Airborne Laser Scanning is an accurate and reliable tool for monitoring glaciers and crevasses has to be answered for the delineation of glaciers and detection and reconstruction of crevasses separately.

- Glaciers can be delineated from ALS data at an accuracy of the pixel size. The delineation used smoothness as well as some other classification criteria to find the outer extends of the glacier. The delineation faces problems at locations of crevasses and therefore still needs human supervision.
- It is possible to detect the location of crevasses from ALS data, provided that the glacier surface is not covered by snow so the crevasses are visible for the laser beam. In the resulting map, the depth for each crevasse can be displayed.
- Reconstructing crevasses is cumbersome because of the low sampling interval. The number of data points is too little to reliably reconstruct the crevasse without making many unmotivated assumptions. Additionally, specific situations, such as snow bridges, make the reconstruction even more unreliable.

Other conclusions that can be drawn from the thesis are:

- A classification of glacier surface based on a smoothness property alone is not accurate enough. Other criteria should be added, including connectivity and hydrological constraints. Although there are several methods for calculating smoothness, for instance from variance or gradients, the results are strikingly similar.
- The intensity data returned from the ALS-system is useful additional information. Using the range from the airplane to the ground point, it is possible to correct the intensity for energy loss during travel. The resulting image is similar to a black-white photograph and can be used for such purposes.
- Intensity data is not suitable for classification purposes, because of the uncertain reflection on glaciers. The ice itself on a glacier has bad reflectance properties, possibly deteriorated by a thin layer of melting water on the ice surface. On the other hand, snow cover does have good reflectance properties, even in the infrared part of the spectrum. These physical conditions change over time and are

unpredictable. A general classification method for delineating glaciers or detecting crevasses can therefore not be based on the intensity data. However, it is useful for human supervision.

- Mathematical morphology is a useful tool to extract specific data from a dataset. In this work, it is used to remove everything from the data except for the crevasses. The problem is to find correct values for the structuring element. Using a directional variogram, the choice of appropriate values can be simplified for the operator.
- Both the detection as well as the reconstruction of crevasses is hindered by objects that block the view into the crevasse. Often these are snow bridges, or even crevasses completely filled with snow. The obvious way to circumvent this is to acquire the data at the moment with the lowest snow cover. This problem is not only typical for ALS; all other methods for measuring crevasses, like photogrammetry or field measurements, suffer from the same problem.
- The quality of the DEM is influenced by aspects like sampling interval and laser inclination. Together, these effects may cause an error in the crevasse depth of 20% in the worst case scenario.

7.2 Future work

Although the thesis gives an answer to the postulated hypothesis, some questions remain open or occurred during the work. Future work could involve more research into these questions.

- The exact interaction of the laser beam with the earth's surface was not investigated. As a result, the intensity correction might be improved by incorporating more knowledge about the type of scattering at the surface.
- The exact reconstruction of crevasses was difficult due to the high number of ambiguities in the modelling. The acquisition of a DEM with a higher point density might solve this problem. With a terrestrial laser scanner, the exact reconstruction of a crevasse could be made in order to gain more knowledge about the crevasse shape.
- In this work, only data from the last reflected pulse is used. A combination of first and last pulse data increases the point density and could be useful for finding the edges of a glacier.
- The methods presented in this thesis and implemented in GRASS GIS have been tested on a single dataset at different epochs. Although the methods are meant to be general, this can only be proven if the programs are applied to other datasets. For instance, the recently released DEM covering Südtirol in Italy could be used for such a test.
- The work in this thesis has proven that several methods for DEM analysis can reveal a lot of additional information from a dataset. However, the power of morphological filtering and frequency filtering has not been used to its full extends in this thesis nor in the GRASS GIS in general. New applications should be developed to give easy access to these tools for the inexperienced user.

- The processes in the current LiSA framework will be of greater interest if they can be accessed more easily by non-experts, for instance using the internet. A faster processing and distribution is required to make this work. Some of the programs developed in this thesis can be adapted for faster and more user friendly processing. Additionally, the GRASS software should be updated with faster displaying facilities such as pyramid layering.

7.3 Discussion

In this thesis a method is presented for the classification, detection and reconstruction of objects on glaciers. It constitutes the full process from acquisition of data to the final delivery of the product. In particular, the method is fully operational as a set of Python scripts within the framework of GRASS GIS.

The results of the program can be used for the applications identified in the introduction of this thesis. However, some of the methods presented here appeared to be interesting to researchers for other reasons. For instance, the detection of crevasses also gives the area of crevasse openings and the full volume of crevasses. This is useful information for research on the velocity change of glaciers. The detection of crevasses can also help the cartographers of the Alpenverein, who are currently upgrading their maps in terms of accuracy and actuality. Due to the modular approach it is easy to answer these questions with the scripts developed.

One drawback of the presented work is that it has not been tested thoroughly on other glaciers. For this reason, the generality of the method cannot be proven. In particular a classification parameter such as smoothness might prove to be less valuable on other glaciers with another type of topography. If more data becomes available, the programs can be tested with this new data and the method can be improved if necessary.

Bibliography

- Baltsavias, Emmanuel P., et al. (2001), 'Digital Surface Modelling by Airborne Laser Scanning and Digital Photogrammetry for Glacier Monitoring', *The Photogrammetric Record*, 17 (98), 243-73.
- Besl, P.J. and McKay, H.D. (1992), 'A method for registration of 3-D shapes', *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 14 (2), 239 - 56.
- Brügelmann, Regine (2000), 'Automatic Breakline Detection from Airborne Laser Range Data', *International Archives of Photogrammetry and Remote Sensing*, XXXIII.
- Byers, H.R., et al. (1951), *Compendium of Meteorology*, ed. Thomas F. Malone (Boston, USA: American Meteorological Society).
- CSR GLAS (2006), 'ICESat/GLAS Web site', <<http://www.csr.utexas.edu/glas/>>, accessed September.
- Delaney, Allan J., et al. (2004), 'Crevasse detection with GPR across the Ross Ice Shelf, Antarctica', paper given at Tenth International Conference on Ground Penetrating Radar, Delft, The Netherlands, 21-24 June.
- Douglas, D. H. and Peucker, T. K. (1973), 'Algorithms for the reduction of the number of points required to represent a line or its caricature', *The Canadian Cartographer*, 10 (2), 112-22.
- Forbes, James D. (1845), *Travels through the Alps of Savoy and other parts of the Pennine Chain with observations on the phenomena of glaciers* (Edinburgh: Adam and Charles Black).
- Free Software Foundation Inc. (1991), 'GNU General Public License', (Version 2 edn.).
- Geist, Th. and Stötter, J. (2003), 'First Results on Airborne Laser Scanning Technology as a Tool for the Quantification of Glacier Mass Balance', *EARSel eProceedings*, 2 (1).
- (2004), 'Airborne Laser Scanning Technology for Glaciological Applications - Results from the OMEGA Project', in M. Thies, et al. (eds.), *Proceedings of the ISPRS working group VIII/2 'Laser-Scanners for Forest and Landscape Assessment'* (XXXVI, PART 8/W2; Freiburg, Germany).
- Geist, Th., Lutz, E., and Stötter, J. (2003), 'Airborne Laser Scanning Technology and its Potential for Applications in Glaciology', in H.-G. Maas, G. Vosselman, and A. Streilein (eds.), *Proceedings of the ISPRS working group III/3 workshop '3-D reconstruction from airborne laserscanner and InSAR data'* (XXXIV, PART 3/W13; Dresden, Germany).

Glacier surface analysis

- Gonzalez, Rafael C. and Woods, Richard E. (2002), *Digital Image Processing* (2nd edn.: Prentice Hall).
- Gorte, B.G.H. (1996), 'Multi-spectral quadtree based image segmentation', *International Archives of Photogrammetry and Remote Sensing*, 31, 251-56.
- GRASS Development Team (2006), 'Geographic Resources Analysis Support System (GRASS) Software', (Trento, Italy: ITC-irst).
- Haala, Norbert, Brenner, Claus, and Anders, Karl-Heinrich (1998), '3D Urban GIS from Laser Altimeter and 2D Map Data', *International Archives of Photogrammetry and Remote Sensing*.
- Hambrey, Michael and Alean, Jürg (2004), *Glaciers* (2nd edn.; Cambridge, UK: Cambridge University Press) 376.
- (2006), 'Glaciers Online', <<http://www.glaciers-online.net/>>, accessed July.
- Höfle, B., et al. (2006), 'Using airborne laser scanning data in urban data management - set up of a flexible information system with open source components', *Urban Data Management Symposium 2006* (Aalborg, Denmark).
- Hoover, Adam, et al. (1996), 'An Experimental Comparison of Range Image Segmentation Algorithms', *IEEE Trans. Pattern Anal. Mach. Intell.*, 18 (7), 673-89.
- ITC 'ILWIS User Documentation', <<http://www.itc.nl/ilwis>>.
- Jähne, Bernd (2005), *Digital Image Processing* (6th edn.; Berlin Heidelberg, Germany: Springer Verlag).
- Jelalian, Albert V. (1992), *Laser Radar Systems* (Norwood, MA, USA: Artech House).
- Khosravi, Mehdi and Schafer, Ronald W. (1996), 'Template Matching Based on a Grayscale Hit-or-Miss Transform', *IEEE Transactions on Image Processing*, 5 (6).
- Kitanidis, P.K. (1997), *Introduction to Geostatistics; Applications in Hydrogeology* (Cambridge: Cambridge University Press).
- Klostermann, Josef (1999), *Das Klima in Eiszeitalter* (Stuttgart, Germany: Schweizerbart'sche Verlagsbuchhandlung).
- Kotelnikow, V. A. (1933), 'On the transmission capacity of "ether" and wire in electrocommunications', *Izd. Red. Upr. Svyazzi RKKA*, Reprint in: *Modern Sampling Theory: Mathematics and Applications* (2002).
- Kraak, M.J., Sliwinski, Adam, and Wytzisk, Andreas (2005), 'What happens at 52N? An Open source approach to education and research', *Internet-Based Cartographic Teaching and Learning: Atlases, Map Use and Visual Analytics* (Madrid), 16-20.

- Kraus, K. and Pfeifer, N. (1998), 'Determination of terrain models in wooded areas with airborne laser scanner data', *ISPRS Journal of Photogrammetry & Remote Sensing*, 53, 193-203.
- Lillesand, Tomas M., Kiefer, Ralph W., and Chipman, Jonathan W. (2004), *Remote Sensing and Image Interpretation* (5th edition edn.; New York, US: John Wiley & Sons).
- Luethya, J. and Stengeleb, R. (2005), '3D Mapping of Switzerland - Challenges and Experiences', *ISPRS Workshop Laser scanning 2005* (Enschede, NL).
- Lutz, E., Geist, Th., and Stötter, J. (2003), 'Investigations of Airborne Laser Scanning Signal Intensity on Glacial Surfaces - Utilizing comprehensive Laser Geometry Modeling and Orthophoto Surface Modeling (A Case Study: Svartiseibreen, Norway)', *International Archives of Photogrammetry, Remote Sensing and Spatial Information Science*, XXXIV (3/W13), 143-48.
- Maas, Hans-Gerd (1999), 'The Potential of Height Texture Measures for the Segmentation of Airborne Laserscanner Data', *Fourth International Airborne Remote Sensing Conference and Exhibition* (Ottawa, Ontario, Canada).
- Mitas, L. and Mitasova, Helena (1999), 'Spatial Interpolation', in P. Longley, et al. (eds.), *Geographical Information Systems: Principles, Techniques, Management and Applications* (New York: Wiley), 481-92.
- Naegel, Benoît, Passat, Nicolas, and Ronse, Christian (2006a), 'Grey-level hit-or-miss transforms--Part I: Unified theory', *Pattern Recognition*, In Press, Corrected Proof.
- (2006b), 'Grey-level hit-or-miss transforms--part II: Application to angiographic image processing', *Pattern Recognition*, In Press, Corrected Proof.
- Oerlemans, J. (1994), 'Quantifying Global Warming from the Retreat of Glaciers', *Science*, 264, 263-45.
- Olhoeft, Gary R. (1996), 'Application of Ground Penetrating Radar', *Conference on Ground Penetrating Radar* (Sendai, Japan), 1-4.
- Passat, N., et al. (2005), 'Automatic Parameterization of Grey-Level Hit-or-Miss Operators for Brain Vessel Segmentation', *Conference on Acoustics, Speech, and Signal Processing* (2; Philadelphia, USA), 737-40.
- Paterson, W.S.B. (1994), *The Physics of Glaciers* (3rd edn.; Oxford, UK: Butterworth Heinemann).
- Rees, W.G. (2001), *Physical Principles of Remote Sensing* (Second Edition edn.; Cambridge, UK: Cambridge University Press).
- Ronse, C. (1996), 'A lattice-theoretical morphological view on template extraction in images', *Journal of Visual Communication & Image Representation*, 7, 273-95.

Glacier surface analysis

- Rutzinger, Martin, et al. (2006), 'Object-Based Building Detection Based on Airborne Laser Scanning Data within GRASS GIS Environment', *Urban Data Management Symposium* (Aalborg).
- Serra, J. (1982), *Image analysis and mathematical morphology* (London, UK: Academic Press).
- (1988), *Image analysis and mathematical morphology; Volume 2: theoretical advances*, ed. J. Serra (2; London: Academic Press).
- Soille, P (1999), *Morphological Image Analysis: Principles and Applications* (Berlin: Springer Verlag).
- Stern, W. (1930), 'Über Grundlagen, Methodik und bisherige Ergebnisse elektrodynamischer Dickenmessung von Gletschereis', *Zeitschrift für Gletscherkunde*, 15, 24-42.
- Strahler, Arthur Newell (1952), 'Dynamic basis of geomorphology', *Geological Society of America Bulletin*, 63 (9), 923-38.
- Tarboton, D. G., Bras, R. L., and Rodriguez-Iturbe, I. (1991), 'On the Extraction of Channel Networks from Digital Elevation Data', *Hydrologic Processes*, 5 (1), 81-100.
- Tyndall, John (1860), *The Glaciers of the Alps* (Elibron Classics Series: Adamant Media Corporation (July 19, 2002)).
- U.S. Geological Survey (2005), 'Glaciers around the world'.
- Vosselman, George (2000), 'Slope Based Filtering of Laser Altimetry Data', *International Archives of Photogrammetry and Remote Sensing*, XXXIII.
- Vosselman, George, et al. (2004), 'Recognising Structure in Laser Scanner Point Clouds', *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*.
- Wack, R. and Stelzl, H. (2005), 'Laser DTM Generation for South-Tyrol and 3D-Visualization', *ISPRS Workshop Laser scanning 2005* (Enschede, NL).
- Wagner, W., Ullrich, A., and Briese, C. (2003), 'Der Laserstrahl und seine Interaktion mit der Erdoberfläche', *Österreichische Zeitschrift für Vermessung und Geoinformation*, 91 (4), 223 - 35.
- Wagner, W., et al. (2004), 'From Single-Pulse to Full-Waveform Airborne Laser Scanners: Potential and Practical Challenges', *International Archives of Photogrammetry and Remote Sensing*.
- WGMS (2006), 'World Glacier Monitoring Website', [website], (updated 20-12-2005) <<http://www.geo.unizh.ch/wgms/>>, accessed March, 15.
- Whillans, Ian M. and Tseng, Yi-Hsing (1995), 'Automatic tracking of crevasses on satellite images', *Cold Regions Science and Technology*, 23 (2), 201-14.

Appendix A: Alpenvereinskarte



Cut-out from the Alpenvereinskarte covering the Hintereisferner. Original scale 1:25 000

© 2003 Österreichischen Alpenverein
© 2006 CIA World Fact Book

Appendix B: Research Area



© 2006 TerraMetrics
 © 2006 TeleAtals

Appendix C: Open Source GIS

The work in this report relied heavily on the use of Open Source components for the implementation of the methods for ALS data processing. Although the project could have been implemented in any proprietary program as well, it was convenient to implement the methods in the existing LiSA framework of the University of Innsbruck, which happened to be based on Open Source software. As such, in parallel to the research work, the project functioned as a small experiment on the use of open source GIS.

GRASS GIS

The geospatial open source program used in this thesis was GRASS GIS. GRASS (Geographic Resources Analysis Support System) is an Open Source GIS program that was originally developed by the U.S. Army Construction Engineering Research Laboratory (USA CERL) as from 1982 as a response to the lack of descent raster based GIS programs at that time. By the mid-1990s the development at CERL was stopped and the source released as Open Source. From then on, the development was coordinated by different Universities with input from programmers from all over the world. Currently it is hosted at ITC-irst in Italy.

Up to now, this is the only real open source GIS program that features both raster and vector support. The program has a devoted group of users and developers, both of which are important for the continuation of the project. Due to the collective effort put into the project, the functionality has been extended so that it includes almost any feature that can be found in mainstream GIS packages. The user interface is mainly based on the command line, like ESRI ArcGIS once did, but a graphical user interface called QGIS is in development. Actually, the command line appears to be one of the more powerful features of the GIS program for the more experienced user.

Concerning the work flow and functions the program is quite similar to ILWIS, the GIS program developed by ITC which is no longer maintained at this time. For instance, both have a strong mapcalculator and neighbourhood functions. The user interface of ILWIS, which combined command line and windows based use in one environment, could be an example for GRASS.

The future of GRASS

However, there are some important disadvantages in GRASS. One of them is the GRASS source that is written in Ansi C. Although the source is open, it is difficult to access because of the complexity of both the language and the development structure. The initial advantage of allowing any user to develop new functions is therefore not so significant, because only a few users will be able to do so. In many ways it is also clear that the package has become quite old by now. For instance, it lacks support for the OGC standards and is not internet-ready. There is a long way ahead for this all to be developed.

This results in the problem that only few companies will invest in using GRASS as they face the risk that a new project could emerge which gathers the group of developers, leaving the GRASS system unmaintained. However, due to this the developing group will not grow fast enough to make required changes. The commitment of some companies or universities is therefore required to bring the open source system to a new phase.

One example is the 52°North organisation, a partnership between ITC, the University of Münster and others, is developing open source software for this OGC implementation. (Kraak et al. 2005) With this new software, a layers architecture for open source GIS can be build. The GRASS

Glacier surface analysis

systems should function as the system for the complex data analysis and data storage in an open standard. The OGC chain takes care of the distribution of the data to any viewer. This viewer can then be any program the end user would like to use, for instance QGIS, a commercial GIS viewer or even better a Web 2.0 based viewer.

Appendix D: Module documentation

Within GRASS a couple of modules were developed with in the LiSA framework for glacier surface analysis. This appendix contains the user documentation of all these programs. The following is a list of modules developed during the course of this thesis.

Main programs

<code>l.crev_detect</code>	Detect crevasse locations from raster based DEM
<code>l.crev_reconstruct</code>	Reconstruct a single crevasse from detrended point data
<code>l.delineation</code>	Delineate the glacier outline from a raster based DEM
<code>l.int_correct</code>	Correct a laser scanning intensity map for elevation influence

Helper programs

<code>l.fitplane</code>	Fit plane trough raster DEM, gives variance, slope and aspect
<code>l.maxpoints</code>	Selects the highest points in a grid from a raster DEM
<code>l.morph</code>	Morphologic filtering on raster data
<code>l.pointsdetrend</code>	Detrend point data using a trend surface
<code>l.simplify</code>	Apply Douglas-Peucker line simplification to vector map
<code>l.threshold</code>	Interactively determine a suitable threshold value

Data analysis

<code>l.drawprofile</code>	Draw a height profile from raster and/or vector data
<code>l.featurespace</code>	Show the feature space of two raster maps

Python Modules

<code>douglaspeucker.py</code>	Functions for Douglas-Peucker line simplification
<code>GRASStools.py</code>	Classes and functions for GRASS module handling
<code>Mathtools.py</code>	Classes and functions for common mathematical operations
<code>profile.py</code>	Functions for calculating profiles from point data

l.crev_detect

Detect the locations of crevasses from raster data

Synopsis

```
l.crev_detect
l.crev_detect help
l.crev_detect [-ir] input=string prefix=string [filtersize=value]
               [threshold=value]
```

Flags

-i Interactively choose threshold
-r First remove ALL files starting with prefix

Dependencies

- Lisa module GRASSTools

Parameters

input=string	Input raster map
prefix=string	Prefix for output raster map names
filtersize=value	Size for the morphological closing filter (default: 5)
threshold=value	Threshold for detecting crevasses (default 0.3)

Description

The `l.crev_detect` program works like a batch program that executes the subsequent GRASS commands to detect glacier crevasses in a raster map. The required input is a DEM stored as a standard GRASS raster. All the intermediate results of the process are stored as raster in the current region with filenames starting with prefix.

The filtersize should be given in pixel units, the threshold value in map units, usually meters. The filter size determines the sensitivity for the steepness difference in the terrain. It is also the maximum width that will be given to each crevasse. The threshold value is used to select the minimum crevasse depth. If the interactive threshold selection is used, the operator can try different threshold values and assess them visually before selecting a final one.

An elaborate description of the implementation can be found in chapter 5 of the thesis.

Output files

The following files are stored by the program. All these files normally start with prefix.

-	_trend	Trend surface calculated with l.maxpoints
-	_detrend	Detrended glacier surface
-	_closed	Surface closed with morphological closing, i.e. surface without crevasses
-	_diff	Difference between _detrend and _closed
-	_crevasses	Binary map showing crevasses after applying threshold

Author

Martin Kodde for LiSA
Last changed: 2006-09-06

l.crev_reconstruct

Reconstruct a single crevasse from point data

Synopsis

```
l.crev_reconstruct
l.crev_reconstruct help
l.crev_reconstruct [-pq] input=string
```

Flags

-p Plot intermediate results to display (requires interactive use)
-q Run quietly

Parameters

input=string Input vector point map covering a single crevasse

Dependencies

- Python modules Numeric and Matplotlib
- Lisa modules GRASSTools, MathTools, profile and douglaspeucker

Description

This program takes the point data of a single crevasse and tries to reconstruct the crevasse from these points. It does this by assuming that the crevasse can be reconstructed from three lines: one bottom line and two edge lines. The procedure is described in section 5.5 of the thesis. The required input is a detrended GRASS vector map with the 3D point coordinates. Detrending of points can be performed with the program `l.pointsdetrend`.

The output of this program is an ASCII file with the XYZ coordinates for the points on the lines.

Notes

The current implementation of the program is a prototype for real crevasse reconstruction. It has not yet been tested thoroughly and some functions of the reconstruction are not yet implemented in this program.

Author

Martin Kodde for LiSA
Last changed: 2006-09-06

l.delineation

Delineate the glacier surface from a raster DEM

Synopsis

```
l.delineation
l.delineation help
l.delineation input=string prefix=string [thresh=value]
```

Parameters

input=string	Input raster map
prefix=string	Prefix for intermediate results
thresh=value	Variance threshold value in m ² . (default: 0.06)

Dependencies

- Lisa module GRASSTools

Description

The `l.delineation` program works like a batch program that executes the subsequent GRASS commands to detect glacier crevasses in a raster map. The required input is a DEM stored as a standard GRASS raster. All the intermediate results of the process are stored as raster in the current region with filenames starting with `prefix`.

The threshold value should be given in map units squared, this is usually m². The threshold value is used to classify between smooth pixels (belonging to the glacier) and non-smooth pixels (not belonging to the glacier). An elaborate description of the implementation can be found in chapter 4 of the thesis.

Output files

The following files are stored by the program. All these files normally start with `prefix`

- `_variance` Variance map calculated with `l.variance`
- `_threshold` Binary map with smooth surface. Result of thresholding variance map
- `_label` Result of connected component labelling
- `_glacier` Binary map showing glacier surface based on smoothness and connectivity
- `_morph` Result after morphological closing to smooth result
- `_area` Same as above but converted to vector

Notes

This program still misses the possibility to intersect the result with the border of a catchment area. This can be done manually afterwards using standard GRASS functionality like `v.select` or `v.overlay`.

Author

Martin Kodde for LiSA
Last changed: 2006-09-06

l.int_correct

Correct intensity map for elevation differences

Synopsis

```
l.int_correct
l.int_correct help
l.int_correct input=string output=string [range=string] [elev=string]
    [height=value] [alfa=value] [--overwrite]
```

Flags

```
--o    Force overwrite of output files
```

Parameters

input=string	Intensity map
output=string	Corrected intensity map
range=string	Map with the range for each pixel
elev=string	Elevation map (Required if no Range Map available)
height=value	Mean flying height above terrain (Required if no Range Map)
alfa=value	Attenuation coefficient [dB/km] (default: 1)

Dependencies

The program `l.int_correct` only depends on default Python and GRASS modules. Currently it uses deprecated the module `r.mean` which is supplied in the same directory.

Description

`l.int_correct` corrects a Lidar intensity raster file for influence from height differences. The travel distance of the laser signal is used to correct the intensity. The travel distance can be given as input in a raster map, but it can also be estimated from the mean flying height and an elevation model. The intensity values should be stored on a scale from 0 to 255. A correction is applied for the loss due to spreading and for the loss due to absorption. Additionally, histogram equalisation is applied for improving the contrast in the image. `l.int_correct` doesn't give fully corrected values; it rather eliminates the largest error sources. For the algorithm of this program, see section 4.5 of the thesis.

Author

Martin Kodde for LiSA
Last changed: 2006-09-06

l.fitplane

Fit a plane through raster points within a given window. Calculate slope, aspect and variance for the fitted plane.

Synopsis

```
l.fitplane
l.fitplane help
l.fitplane [-q] input=string [slope=string] [aspect=string]
           [variance=string] size=integer
```

Flags:

-q Quiet

Parameters:

input=string	Input raster (elevation) map
slope=string	Name for the slope map
aspect=string	Name for the aspect map
variance=string	Name for the variance map
size=integer	Window size (must be odd) (default: 3)

Dependencies

The program `l.fitplane` has dependencies to the following c-modules:

- GRASS libraries
- GNU Scientific Library (GSL), <http://www.gnu.org/software/gsl/>

Read INSTALL for compilation instructions

Description

`l.fitplane` fits a plane through raster points within a moving window of a given size. The program calculates the slope, aspect and variance for the centre pixel using the parameters of the fitted plane. The window can be a square of any given (odd) size. This is where it distinguishes itself from other modules such as `r.slope.aspect`, which only operate within a 3 by 3 window.

Algorithm

`l.fitplane` moves a window of size by size pixels of the rastermap. For each step, it fits a plane through all the raster points within the window. This plane can be written as $z = a \cdot i + b \cdot j + c$ with z the elevation values in the raster and i, j the row and column locations. Within the window, all points can be described using matrix notation: $z = Ax$. With z as a vector with the elevations, A as a matrix with the row and column coordinates, and x a vector with the parameters a, b and c . Using the least squares adjustment algorithm, weight factors for the elevations will be determined from the A matrix. Because A is constant for a given window, a large part of the calculations, including the inversions, have to be done only once. With the resulting weight factors, the unknown parameters a, b and c can be calculated for each plane.

The plane normal vector is defined as $n = (-a \quad -b \quad 1)^*$. From this normal vector the slope and aspect can be calculated as:

$$slope = \arctan\left(\sqrt{a^2 + b^2}\right)$$

$$aspect = \arctan\left(\frac{a}{b}\right)$$

Here, the slope equation gives a 360 degrees outcome.

The variance is determined from the residuals after fitting the plane. Assume the residuals to be $\hat{e} = z - \hat{z}$, with \hat{z} the elevation of the pixels according to the fitted plane, then variance = $(\hat{e}^2)/n$, i.e. the squared sum of the residuals. This definition implies that the residuals are calculated as the vertical differences and not as the orthogonal difference to the plane. The latter would usually be slightly smaller.

Notes

1. `fitplane` does take the grid size into account for the calculations, which is mainly relevant for determining the slope. However, until now the module has only been tested on a grid with a 1m resolution.

Due to the window-based processing it is impossible to determine values at the outer edges of the region. These pixels will be filled with NULL values. One way to circumvent this is by temporarily extending the region size with half the window size in each direction. It would be nice to do this automatically in the program.

See also

`r.mapcalc`
`r.slope.aspect`

Author

Martin Kodde for LiSA
Last changed: 2006-05-28

l.maxpoints

Find the highest points within a dist x dist grid

Synopsis

l.maxpoints

l.maxpoints help

l.maxpoints input=string output=string dist=integer

Parameters:

input=string

Input raster (elevation) map

output=string

Output point vector map

dist=integer

Distance of grid for searching highest points (default: 10)

Description

l.maxpoints can be used to derive the highest points in a DEM. It can be used as a filtering tool for filtering out cracks or pits in the terrain by interpolating the highest points to a grid again. A dist parameter defines the spacing in pixels of a grid that is overlaid over the raster map. Within each cell of dist by dist pixels the highest point is selected. The highest points are exported as a 3D Vector point map.

Algorithm

l.maxpoints loops over all the points in the raster map. For each grid point it determines to which cell it belongs and whether it is the highest point within the cell. If so, it is stored internally in a raster with cell size dist. After evaluating all raster points, the highest points are converted to a 3D Vector file.

Notes

Currently, all points receive the same category value. Maybe, this should be changed.

Author

Martin Kodde for LiSA

Last changed: 2006-05-28

l.morph

Perform morphological processing on a binary or grey scale raster map.

Synopsis

```
l.morph
l.morph help
l.morph input=string output=string shape=string size=value operation=string
  [--overwrite]
```

Flags:

--o Force overwrite of output files

Parameters:

input=string	Binary or grayscale input raster map.
output=string	Output map name
shape=string	Shape of structure element options: square,circle default: square
size=value	Size of the structure element. Must be odd for 'square'. default: 3
operation=string	Morphological operation options: dilation,erosion,closing,opening default: dilation

Description

l.morph can be used to apply morphological filters to binary or greyscale raster maps. The operations dilation and erosion are implemented. Opening and closing is also possible.

Features

With l.morph, morphological filters can be run over a raster file. One can choose between erosion or dilation, or the combined operations opening and closing. The shape of the window can be selected, currently the possibilities are a square and a circle, but additional shapes can easily be added to the source. The size of the window can freely be set, but is limited to odd values for the square shape.

Algorithm

The principle behind dilation on greyscale data is that the centre pixel of a window receives the highest value of all pixels within that window. As an opposite to dilation, the centre pixel get the minimum value of all pixels.

Author

Martin Kodde for LiSA
Last changed: 2006-06-03

l.pointsdetrend

Detrend points in vector file

Synopsis

```
l.pointsdetrend
l.pointsdetrend help
l.pointsdetrend [-qr] input=string trendsurf=string [output=string]
                  [interp=string] [--overwrite]
```

Flags

-q Run quietly
-r Do not change region, but leave current region
--o Force overwrite of output files

Parameters

input=string	Input vector point map covering a single crevasse
trendsurf=string	Input raster map with trend values
output=string	Output vector map with detrended values (leave empty for stdout)
interp=string	Interpolation method options: neighbour,bilinear default: neighbour

Dependencies

The program `l.pointsdetrend` has dependencies to the following Python libraries

- Numeric
- GRASStools
- Mathtools

Description

`l.pointsdetrend` detrends vector points using data from a raster based trend surface. It can be used if the effect of slopes in the data has to be removed before further processing. The elevations in the trend surface are subtracted from the vector points. The result can either be written to stdout or a new GRASS vector map.

Trend surface

The trend surface is an essential input for this module. The trend surface should contain the large-scale relief features, but not the small scale features. There are different ways to derive such a surface, some examples are:

- Let a mean or median mask with a large window size run over the data set using `r.neighbors`
- Apply a Kriging interpolation with adapted covariance function
- Extract maximum points with `l.maxpoints` and interpolate these with `v.surf.rst` to a raster.

Notes

The program tries to find a trend value from the trend surface for each vector point. By default it takes the nearest neighbour in the raster map, but the value can also be derived using bilinear interpolation from the 4 surrounding raster points. The latter should give better (smoother) results. However, if the vector point density is approximately equal to the raster point spacing, the difference is usually very small. (up to 5 cm) Therefore the faster nearest neighbour should be preferred in those cases.

If the flag `r` is not used, the region is temporarily changed so that it fits exactly around the vector data. Note that it is not advised to have a mask activated when detrending.

See also

`r.neighbors`

l.maxpoints
v.srf.rst
v.in.ascii
v.out.ascii

Author

Martin Kodde for LiSA
Last changed: 2006-06-03

l.simplify

Simplify a vector map using the Douglas-Peucker algorithm

Synopsis

```
l.simplify  
l.simplify help  
l.simplify [-q] input=string output=string threshold=value [--overwrite]
```

Flags

```
-q    Run quietly  
--o   Force overwrite of output files
```

Parameters

```
input=string      Input vector map  
output=string     Output vector map  
threshold=value   Threshold value
```

Dependencies

The program `l.simplify` depends on these python modules

- GRASSTools
- MathTools
- DouglasPeucker

Description

The `l.simplify` program can be used to simplify vector maps with lines or polygons. It removes insignificant vertices from the lines using the Douglas-Peucker algorithm. This is useful after raster to vector conversion where a vertex is added at every pixel location.

The program reads in all the lines and polygons from a GRASS vector map and applies the simplification algorithm. The results are stored as an ASCII vector map that is imported back into GRASS. For more information about the Douglas-Peucker algorithm, see the documentation of the `douglaspeucker` python module.

Notes

Because the data is temporarily stored as an ASCII vector and then imported back into GRASS, any attribute data will get lost in the current implementation. For polygons, this procedure is not topological correct.

See also

```
douglaspeucker  
v.clean
```

Author

Martin Kodde for LiSA
Last changed: 2006-09-07

l.threshold

Interactively select a threshold value

Synopsis

```
l.threshold
l.threshold help
l.threshold [-b] [input=string] [threshold=value] [operation=string]
            [output=string] [command=string] [--overwrite]
```

Flags

```
-b    Set black/white colormap
--o   Force overwrite of output files
```

Parameters

input=string	Rastermap to threshold
threshold=value	Initial threshold value default: 0
operation=string	Operation to perform, eg: 'operation=\<' (include quotes!) options: >,<,>=,<= default: >
output=string	Optionally save thresholded map
command=string	File with command to run instead of thresholding. Use %f for threshold and %s for output map

Dependencies

The program `l.simplify` depends on these python modules

- GRASSTools
- MathTools

Description

With the `l.threshold` program an operator can easily select a suitable threshold for his application. The program displays the result after thresholding visually in the active GRASS monitor. The operator can either agree with the current threshold or select a new threshold after which the display is updated with the new results. After selecting the threshold, the results can optionally be written to a new GRASS raster map.

The default behaviour of the program is that the operator selects a map, a threshold value and one of the operators: >,<,>=,<=. For instance, the default values for the parameters make the program show everything above zero.

Alternatively, the program can run custom commands for special types of selections. The GRASS commands should be stored in a file, which is called with the `command` parameter. The GRASS commands in the file can contain any desired GRASS operation or sequence of operations. However, it should contain the `%f` and `%s` variables. The `%f` variable should be used as value for the threshold, the `%s` variable as name for storing the result.

Notes

Because the `'<'` and `'>'` characters have a special meaning in the Linux shell environment, they should be escaped with the backslash character. Additionally, the parameter name as well as value should be grouped with quotes, eg:

```
l.threshold input=diff threshold=0.2 'operation=\>'
```

Author

Martin Kodde for LiSA
Last changed: 2006-09-07

l.drawprofile

Output the raster map layer values that lie on user-defined line(s).

Synopsis

```
l.drawprofile
l.drawprofile help
l.drawprofile [-irhp] [raster=string[,string,...]]
                [vector=string[,string,...]] [output=string]
                [profile=east,north[,east,north,...]] [res=value] [null=string]
                [selwidth=value] [python=string] [style=string[,string,...]]
                [legend=string[,string,...]]
```

Flags

```
-i   Interactively select End-Points
-r   Repeat profile program
-h   Hide legend
-p   Print vertex coordinates
```

Parameters

raster=string	Names of existing raster maps
vector=string	Names of existing vector maps
output=string	Name of file for output
profile=east,north	Profile Coordinate Pairs
res=value	Resolution along profile (default = current region resolution)
null=string	Character to represent no data cell (default: *)
selwidth=value	Width around line for selecting vector points (default: 1)
python=string	Create Python script with name <python> for plot commands
style=string	Matlab plot style for each raster and vector map (e.g. 'b-')
legend=string	Legend text, use '_' for space. Leave blank for map names

Dependencies

The program l.drawprofile has dependencies to the following Python libraries:

- Numeric
- Matplotlib (pylab)
- GRASStools
- Mathtools

Description

l.drawprofile is a versatile program for creating profiles of raster maps and 3D vector point data. Its behaviour is in large part similar to r.profile, but it offers additional functionality. The program can plot profiles for different maps at the same time in the same window and offers full flexibility for assigning plotting styles. Besides showing the output in a plot, the data can be stored in a file or as a separate Python program.

Functionality

If you activate the i-flag, you can draw the profile line in the current monitor by pointing at locations in the current map. If this flag is not used, a list of point coordinates should be provided in the profile parameter. The profile line can contain multiple vertices.

It is possible to let the program draw multiple profiles at one time. Separate the map names with a comma. For each map a different plotting style can be supplied using the Matlab plot style parameters. Note that first all the raster maps are plotted, then the vector maps. If you have two raster maps and one vector map, the plot style for the vector comes at the third place.

The vector points that will be shown are collected in an area around the plot line. The size of this area is by default set to one meter, but can be changed using the selwidth parameter.

With the output and python parameters, the results can be stored for later processing.

Notes

Due to some recent changes in the program, there are a lot of small bugs and todos left.

- Storing the result to output or python for now only works if a profile is made of only one raster or vector map.
- NULL values in the data are now simply skipped during plotting. If a line plotting style is used, this might give wrong results. The line should be broken at NULL data like Matlab does with NaN values in its plot() command.
- In interactive mode, the profile line is only shown after finishing the complete profile. It would be nicer to see the profile line directly.
- If the profile is built from multiple line vertices, vector maps cannot be plotted.
- An input option should be made for the legend.
- The program should use the newly made profile module for selecting vector points. This would solve some of the above stated problems.

See also

r.profile
l.profile

Author

Martin Kodde for LiSA
Last changed: 2006-06-03

l.featurespace

Make a plot of the feature space of two raster maps.

Synopsis

```
l.featurespace
l.featurespace help
l.featurespace [-p] input1=string input2=string [cat=string] [n=value]
                [style=string] [legend=string] [xlabel=string] [ylabel=string]
                [title=string]
```

Flags

-p Print scatter data to stdout

Parameters

input1=string	First input map.
input2=string	Second input map.
cat=string	Category map.
n=value	Use only every nth line. (default: 1)
style=string	Matlab style parameters eg: "b.,r.,k-,ro" (default: b.,r.,g.,k.)
legend=string	Legend for values from cat
xlabel=string	Label for first raster map
ylabel=string	Label for second raster map
title=string	Title for the graph

Dependencies

The program l.featurespace has dependencies to the following Python libraries:

- Numeric
- Matplotlib (pylab)

Description

l.featurespace lets you draw the feature space that is constituted by two raster maps. The featurespace is a 2D plot where the values of the first map are placed along the horizontal axis and the values of the second map along the vertical axis. Additionally, the plotted points can be coloured using a category map, which essentially adds a third dimension to the feature space. The feature space plot can for instance be used to search for correlation between two maps or to find clusters of data points.

Functionality

If you supply a category map, you can supply a different plotting style for all category values. This allows you to give points with different category values different colours. By default the colours blue, red, green and black are used for the first four category values, after which the colour scheme repeats itself. A custom category map can be given using the Matlab plot style parameters.

See also

Matlab plot styles: <http://www.mathworks.com/access/helpdesk/help/techdoc/ref/linespec.html>

Author

Martin Kodde for LiSA
Last changed: 2006-06-04

douglaspeucker.py

Douglas Peucker line simplification module

Dependencies

This python module depends on the Numeric module.

Description

This program uses the Douglas-Peucker algorithm to simplify lines or polygons. Given the coordinates of the line or polygon, it searches for the points that can be left out given a certain threshold.

Note that the current implementation is not topologically correct.

Functions

`do_simplify(points, t, line, return_index=0)`

Do the line simplification and return simplified line

points is a n by 2 matrix with the point coordinates order by their position in the line.

t is the treshold value

line is the initial simplified line, e.g. the begin and end points of the line.

return_index=0: return the coordinates of the simplified line

return_index=1: return the point index values of the points in the simplified line.

The program looks for the point with the largest distance from line and adds this points if the distance is larger than t. The procedure is repeated for the newly formed first line segment. If no more segments are to be added, the program continues with the second segment, etc.

Available sub functions:

`addpoints(points, key, t)`

`points2line(p, x1, x2)`

Returns perpendicular distance from point to a line segment.

p is a n by 2 matrix with the x- and y-coordinate of the points

x1 is a vector with the coordinates of the begin point

x2 is a vector with the coordinates of the end point

`points2point(p, x)`

Returns the distance between two points.

p is a n by 2 matrix with the x- and y-coordinate of the points

x is a vector with the x- and y-coordinate of the reference point

`simplifyline(points, t, return_index=0)`

Use this function to simplify the shape of a line

points should by a Numeric array with the (x,y) coordinates that constitute the line. The order of the points in the array determines the connectivity between points.

t is the tolerance in coordinate units used for the simplification.

If return_index=0, the coordinates of the new line is returned. If return_index=1, the index values of the points in the line are returned.

`simplifypolygon(points, t)`

Use this function to simplify the shape of a polygon

Glacier surface analysis

points should be the points that form the shape of the polygon. The last point may or may not be the same as the first one. The order of the points determines the connectivity between points

t is the tolerance in coordinate units.

If return_index=0, the coordinates of the new line is returned. If return_index=1, the index values of the points in the line are returned.

Author

Martin Kodde

Last changed: 2006-09-07

GRASSTools.py

Python module for GRASS module development.

Dependencies

GRASSTools depends on the NumPy module

Description

This module can be used in a python script that acts as a module for GRASS. It contains several functions for simplifying several actions that should be done in almost all GRASS modules.

Classes

`class GRASSModule`

Class for basic GRASS module handling. Call this for each GRASS module.

Methods defined here:

`__init__(self, argv)`

Initialise GRASSModule module

During initialisation this function takes the following action

- check if the GRASS is environment set
- start g.parser
- initialises verbose option

`array2vector(self, name, data)`

converts (x,y,z) array to GRASS point vector

`checkNew(self, name, elem='cell')`

Check if the new map name is valid and writable

`checkOld(self, name, elem='cell')`

Check is the map exists and is readable

`e(self, s)`

Print error message and exit program

`flag(self, name)`

Retrieve flags from parser

`getCurrentRegion(self, appendRC=0)`

returns current region settings [n,s,w,e,nsres,ewres,nrows,ncols] (last 2 optional with `getCurrentRegion(1)`)

`getEnv(self)`

returns list [GISDBASE,LOCATION_NAME,MAPSET]

`getRasterRes(self, name)`

returns raster resolutions: (nsres,ewres)

`getTempFileName(self)`

creates temporary file in MAPSETs .tmp directory and returns filename (absolute path) as string

.tmp directory is cleaned during GRASS exit

`loadRasterPoints(self, name, nodata='', nv='nan')`

Load points from a raster map as a numeric array in x y z (z2 z3 ...) format supports query on multiple rasters: `rast1,rast2,rast3`

`nodata='n'`: ignore cell/s with occurrence of NODATA > 0

`nodata='N'`: ignore cells where all rasters have NODATA for one cell

Glacier surface analysis

nodata='': return all cells (with NODATA values)
nv='nan': string/value representing NODATA VALUE

loadVectorPoints(self, name)
Load points from a vector map as a numeric array

p(self, s)
Print text messages. Use instead of print

readTable(self, name, colnames=0, cols='*')
returns vector table as array: all fields have to be numeric

removeTempFile(self, name, q=True)
deletes temporary file: returns True if file was removed and False if file still exists
if q=True: quiet mode: no output

var(self, name, default='')
Retrieve parameter values from parser

vectoris3d(self, name)
Checks if the vectormap is 3D.

w(self, s)
Print warning message, don't exit program

Author

Martin Kodde

Bernhard Höfle

Last changed: 2006-09-07

MathTools.py

Python module for some selected mathematical operations

Dependencies

This module depends on the following other python modules:

- Numeric
- LinearAlgebra
- Matplotlib

Description

This module contains some functions and classes for performing some mathematical operations required for the crevasse reconstruction and other laser scanning data processing tasks.

Classes

`class LeastSquares`

Class for performing linear least squares adjustment.

Variables within this class:

`xhat`
`Qxhat`

Methods defined here:

`__init__(self, y, A, sigma=0)`
Initialise least squares adjustment

Supply observations (`y`) and designmatrix (`A`) as Numeric array. Optionally, give the variance in the vector `sigma`.

`class poly2fit`

Class for fitting a 2D polynomial (line) through a serie of points.

Methods defined here:

`__init__(self, data, order=2, sigma=0, translate=1)`
Initialise 2D polynomial fitting

`data` should be a Numeric array with two columns (`x`, `y`)
`order` should be the polynomial order (1=line, 2=parabolus, etc)
`translate` defines whether the points should be moved to the origin of the system
Least Squares adjustment is used for the fitting

`getehat(self, data)`

Get the residual for each point in data i.e. the (vertical) distance from the line.

`gety(self, data)`

Get the y-values of the polynomial for a given array of x.

`makerange(self, dist=1)`

Give the range over which the polynomial is defined.

`xtransform(self, data)`

Move data to the origin.

Functions

`barycenter(numbers)`

Returns the barycenter of a numeric list.

`bilinear(row, col, area)`

Calculate bilinear interpolation

Glacier surface analysis

row and col should be coordinates pointing to the position of the current point. Positive horizontal axis points to the right, positive vertical axis points down (row direction). The surrounding pixels lay at (-0.5,-0.5), (-0.5, 0.5), (0.5, 0.5), (0.5, -0.5). The values of these pixels should be supplied as a 2x2 Numeric matrix

`convexhull(data)`

Calculate the convex hull of a dataset.

This function uses the QHULL algorithm (www.qhull.org) to calculate the convex hull. Note that this program should be installed and available from default path

`mean(numbers)`

Returns the arithmetic mean of a numeric list.

`plot hull(vertices, data, style='b-')`

Plot the convex hull as a line to the current pylab plot

`point_inside_polygon(x, y, poly)`

Determine if a point is inside a given polygon or not.

Polygon is a list of (x,y) pairs. Returns true if the point lies within the polygon.

`randomdata(data, n)`

Select n random points from dataset

`selectpoints(condition, a)`

Returns the rows of a Numeric matrix for which condition is true.

I couldn't find any native function to do this, so I wrote one for the specific purpose. Use one of the UFuncs from Numeric to make a binary selection on one columns of interest. Supply the result as condition together with the full dataset to this function. It will return only those rows where the condition was true.

`var(z)`

Returns variance of a dataset.

profile.py

Profile extraction from point data

Dependencies

This module depends on:

- Numeric
- MathTools

Description

Given a point cloud and the coordinates of some vertices, this module can calculate the profile along a multi-segment line. It returns the x,y coordinates for plotting the profile and the indexes of the point in the profile.

Functions

Profile(data, vertices, w)

Return a profile from XYZ cloud

data is a n by 3 numeric array with the coordinates of the points.

vertices is a m by 2 numeric array with the coordinates of the line vertexes.

w is the width of the area around that line from where points should be selected.

Author

Martin Kodde

Last changes: 2006-09-07

Glacier Surface Analysis

Airborne Laser Scanning for monitoring glaciers and crevasses

Glaciers are and have always been objects of great interest to both researchers and tourists. In the past, glaciers were major factors in the formation of our geography. Nowadays, they are interesting as indicators for climate change and important for the local economy. Getting to know more information of glaciers and being able to closely monitor these is therefore of general importance. A contribution in the form of processing Airborne Laser Scanning data is given in this thesis. The thesis was written as the final part of the MSc. Geomatics at Delft University of Technology.



Master of Science Geomatics - Thesis

Martin Kodde
October 2006

Optical and Laser Remote Sensing
Department of Earth Observation and Space Systems
Faculty of Aerospace Engineering, Delft University of Technology

In cooperation with:
Department for Natural Hazards and Risk Research
Institute of Geography
Faculty of Geo- and Atmospheric Sciences, University of Innsbruck

