Master Thesis, GEO 511

Flow patterns of a maritime-terminated outlet glacier in West-Greenland derived from oblique photography

Adrian Simon Peter
adrian.peter2@uzh.ch
Student Nr: 11-101-474

Glaciology and Geomorphodynamics Group, Physical Geography,
Department of Geography, Faculty of Science, University of Zurich, Switzerland

Supervised by:
Prof. Dr. Andreas Vieli (Faculty Member) and Dr. Martin Lüthi

20th of April 2017

Cover Page: Camp Eqi in front of Eqi Sermia on the 26th of August 2016 (Photo: R. Mercenier)

I Abstract

Climate change causes atmospheric and oceanic warming, two external forces affecting cryospheric change. The Greenland Ice Sheet is, in terms of potential as well as actual melt rate, a major contributor to global sea level rise. Up to half of its mass is lost through calving outlet glaciers. Still, the links are unclear between external forces and the interactions of the different mechanisms driving the calving process. Understanding glacial velocity patterns is therefore key for predicting contributions to seal lever rise. Yet, most of the seasonal data available is at best of weekly temporal resolution. A promising and cost-effective alternative method is motion detection derived from time lapse camera imagery. In this thesis, a total of three different methods are carried out by combining, applying and testing existing techniques to determine their suitability for velocity tracking. Eqi Sermia, a tide water glacier in West Greenland was photographed once a day for up to 10 months. Seasonal velocity patterns and relative motion are obtained by all three methods, leading to the classification of Eqi Sermia as a *marine-terminating glacier type 2*, according to the categories of Moon et al. (2014). Using a terrestrially-acquired Structure from Motion digital elevation model, the absolute glacier velocity, which ranged from 2 m/day to 13 m/day was calculated with an uncertainty of $\pm$ 1.4 m/day.

Keywords: Calving glacier, flow pattern, time-lapse camera, Structure from Motion

II Table of Content

I Abstract 3
II Table of Content 4
III List of Figures 6
IV List of Tables 7
1 Introduction 9
1.1 State of Research and Motivation: 9
1.1.1 Greenland's Maritime-Terminated Calving Glaciers 9
1.1.2 Time Laps Cameras in the Cryosphere 10
1.2 Research Gap and Motivation 11
1.3 Aim of the Thesis 12
1.4 Study Site: 12
2 Methods 15
2.1 Dataset, Data Management and Data Processing 15
2.1.1 Datasets 16
2.1.2 Software 18
2.1.3 Coordinate System 18
2.2 Template Matching (2D-Displacement) 19
2.2.1 The Principles of Template Matching 19
2.2.2 Application of Template Matching Method for Displacement Detection 20
2.3 Structure from Motion 23
2.3.1 The Principles of Structure from Motion 23
2.3.2 SfM Method Used to Generate a DEM 23
2.4 Template Matching and Georectification (3D-Velocity) 27
2.4.1 Template Matching and Image Georectification Using ImGRAFT 27
3 Results 35
3.1 Template Matching (2D-Displacement) 35
3.1.1 Image Pair Sample 35
3.1.2 Time Series 38
3.1.3 Source of Uncertainties 42
3.2 Structure form Motion Digital Elevation Model 45
3.2.1 Presenting the Data 45
3.2.2 Uncertainties 47
3.3 Template Matching and Georectification (3D-Velocity) 49
3.3.1 Image Pair Sample 49
3.3.2 Time Series 51
3.3.3 Sources of Uncertainties 57
3.4 Further Observations 58
4 Comparison 61
4.1 Comparison of the DEMs 61
4.2 Comparing Flow Patterns 63
4.3 Comparing Time Series 67
5 Discussion 71
5.1 Methodological Limitations 71

5.1.1 Template Matching (2D) 71
5.1.2 Georectification and Template Matching (3D) 72
5.1.3 Structure from Motion derived Digital Elevation Model 73
5.2 Interpretation of flow pattern 74
5.3 Restrictions and Improvements for further Research 76
6 Conclusion 79
7 Bibliography 81
8 Appendix 85
8.1 Terminology and Abbreviations 85
8.2 Additional Information 86
8.2.1 A Brief Introduction into MATLAB 86
8.2.2 Further Attempts to Register Images 86
8.2.3 The Theory Behind Structure from Motion 88
8.2.4 The Principle of Image Georectification 91
8.3 Qualitative Image Description 92
8.3.1 Sample Images for Structure from Motion 95
8.4 Additional Data 97
8.4.1 Reconstructed Timestamps 97
8.4.2 Spatial Patterns Averaged per Month 99
8.4.3 Example of Applied Smoothing 101
8.4.4 Further Figures 102
8.5 Code Sample 104
8.6 Acknowledgement 108
8.7 Personal Declaration: 109

III List of Figures

Cover Photo: Camp Eqi in front of Eqi Sermia (Photo: R. Mercenier) 1
Fig. 1: Three types of maritime-terminated outlet glaciers (Moon *et al.*, 2014, Figure 2) 10
Fig. 2: Study Site 13
Fig. 3: Possible Positions of Pinning Points 13
Fig. 4: Flow Chat of Methodology 15
Fig. 5: Time-Lapse Cameras Location (Photos: A. Vieli, L. Moreau) 16
Fig. 6: Principle of Template Matching (Messerli and Grinsted, 2015) 19
Fig. 7: Sample of Masked Images with Ground Control Points 25
Fig. 8: Sample of Sparse and Dense Point Cloud Using PhotoScan Pro (Screen Shot) 26
Fig. 9: East-west profile through the drone derived DEM 28
Fig. 10: Sample Image for the location of the initial offset and the rectification grid 30
Fig. 11: Sample of Template Matching (2D- Displacement) 36
Fig. 12: Sample of Peak Correlation Ratio (2D- Displacement) 36
Fig. 13: Sample of Mean Absolut Correlation Ration (2D- Displacement) 37
Fig. 14: Sample of Signal to Noise Ratio (2D- Displacement) 37
Fig. 15: Position of Tracking Boxes (2D- Displacement) 38
Fig. 16: Mean Values of Tracked Displacement (2D- Displacement) 39
Fig. 17: Displacement of Upper Camera (2D- Displacement) 40
Fig. 18: Displacement of Lower Camera (2D- Displacement) 40
Fig. 19: Mean Values of the Tracked Displacement (2D- Velocity) 41
Fig. 20: Camera Shake (2D- Displacement) 42
Fig. 21: Signal to Noise Ration (2D- Displacement) 42
Fig. 22: Digital Elevation Models using Structure from Motion Technique 46
Fig. 23: Sample of Template Matching and Georectification (3D- Velocity) 49
Fig. 24: Sample of Peak Correlation, Mean Absolut Correlation and Signal to Noise Ratio (3D) 50
Fig. 25: Data Availability and Position of Tracking Boxes (3D- Velocity) 51
Fig. 26: Means of Tracked Velocity using Drone derived Digital Elevation Model 52
Fig. 27: Tracked Velocity Upper Camera using Drone derived Digital Elevation Model 53
Fig. 28: Tracked Velocity Lower Camera using Drone derived Digital Elevation Model 53
Fig. 29: Camera Shake using Drone derived DEM (3D- Velocity) 54
Fig. 30: Signal to Noise Ration using drone derived DEM (3D-Velocity) 54
Fig. 31: Means of Tracked Velocity using Structure from Motion derived DEM 55
Fig. 32: Tracked Velocity Upper Camera using Structure from Motion derived DEM 55
Fig. 33: Tracked Velocity Lower Camera using Structure from Motion derived DEM 56
Fig. 34: Camera Shake using Structure from Motion derived DEM 56
Fig. 35: Signal to Nosie Ration Using Structure from Motion derived DEM 57
Fig. 36: Calving Front Positions over Time (Lower Camera) 59
Fig. 37: Digital Elevation Models in Comparison 62
Fig. 38: Spatial Distribution of Surface Velocity using Drone derived DEM 64
Fig. 39: Spatial Distribution of Surface Velocity using Structure from Motion derived DEM 65
Fig. 40: Velocity Comparison of the Findings with Complementary Data (Upper Camera) 68
Fig. 41: Velocity Comparison of the Findings with Complementary Data (Lower Camera) 68
Fig. 42: Lowe's Scale-Space Concepts (Lowe, 2004) 89
Fig. 43: Sample Images of Time-Lapse Imagery (Lower Camera) 95

Fig. 44: Sample Images used for Structure from Motion (28th of August 2016) 96
Fig. 45: Monthly Averaged Spatial Flow Patterns Using the Drone DEM 99
Fig. 46: Monthly Averaged Spatial Flow Patterns Using the SfM DEM 100
Fig. 47: Visualisation of DEM Smoothing 101
Fig. 48: Sample of Unsuccessfully Calculated Point Cloud (SfM) 102
Fig. 49: Determination of the Ideal Window Size for Template Matching 102
Fig. 50: Sample of a Calculated View Shed 103

IV List of Tables

Table 1: Parameters of the Cameras 16
Table 2: Data Availability for Displacement Tracking (2D) 38
Table 3: Multipliers for the Pinhole Camera Projection Model 41
Table 4: Parameters of the Structure from Motion Methods 47
Table 5: Basic Statistics of the Digital Elevation Models 47
Table 6: Data Availability for Velocity Calculation (3D) 51
Table 7: Projection Error of the GCPs 57
Table 8: Root-Mean-Squared Error and Standard Deviation of the Different Calculations 69
Table 9: Reconstructed Timestamps for the Upper Camera 97

(blank page)

1 Introduction

1.1 State of Research and Motivation:

1.1.1 Greenland's Maritime-Terminated Calving Glaciers

The role of Greenlands Outlet Glaciers

The Greenland ice sheet (GrIS) is the second biggest fresh water reserve on earth, storing about 7.36 meters of sea level equivalent (SLE) (IPCC, 2013). From 1992 to 2012, a rapid, non-linear acceleration of mass loss has been monitored in Greenland resulting in a loss of 2700 $\pm$ 930 Gt of ice, which is about twice as much as Antarctica over the same time period (Shepherd et al., 2012). The annual contribution from GrIS is estimated to be between ~0.7 mm to 1.1 mm SLE (Moon, Joughin and Smith, 2015). About a third to a half of Greenland's mass loss is due to ice discharge by maritime-terminated outlet glaciers (Moon et al., 2014), making Greenland's outlet glacier one of the largest contributor to sea level rise (SLR). Yet, rapidly changing velocity patterns of Greenland's maritime-terminated outlet glaciers are not fully understood.

Estimations suggest that outlet glaciers in Western and North-Western Greenland could contribute up to an additional 33 mm to global SLR by 2100, if the current acceleration since 2000 continues (Enderlin *et al.*, 2014). Ocean and atmospheric warming, caused by climate change (IPCC, 2013), is understood to be the two main external forces driving this process.

Consensus

In the scientific community there is consensus that the marine-terminated calving glaciers are responding differently to climatic changes than land terminated glaciers, in regard to the terminus position, the magnitude of the response as well as the temporal aspect (Benn and Evans, 2010, p. 177). Furthermore, it is known that seasonally returning cycles of advancement and retreat are evident (Ahlstrøm *et al.*, 2013). For calving glaciers this cycle is asynchronous, the advance is usually slower than the retreat.

As far as it is understood, advance and retreat is caused by two kinds of drivers.

On one hand the dynamic processes are 'pushing' the ice into the water. This driver is mostly linked to meltwater-bedrock interactions.

On the other hand, the calving process reduces drag, leading to changing ice dynamics and therefore 'pulling' the terminus along by (Benn and Evans, 2010, p. 177). A glacier's velocity increases with the decreasing ratio of calving front to water depth, as basal drag is reduced. This explains why outlet glaciers tend to increase speed towards the front (Benn and Evans, 2010, p. 165). Because of the same principle, changes in bathymetry strongly influence the calving rate. An outlet glacier retreating from its terminal moraine shoal into an overdeepened basin is likely to retreat rapidly until the next stable position is reached (Benn and Evans, 2010, p. 179). Further processes enhancing the drag, such as sea ice or ice mélange (buttressing effect), are likely to reduce the calving rate and tend to lead to an advancement of the terminus position (Moon, Joughin and Smith, 2015).

Restraints

Flow velocity patterns of Greenland's tide water outlet glaciers are not fully understood. Especially, the link between the external forcing (atmospheric and oceanic warming) and the driving mechanisms of outlet glaciers is still a topic of research (Moon and Joughin, 2008). The rapid dynamic changes, such as retreat, acceleration and thinning, of tidewater outlet glaciers, is

applied to selected examples (Vieli and Nick, 2011). Still, they are not implemented successfully on large scale model calculations. A better understanding of these processes is crucial in order to gain the ability to predict dynamic mass losses successfully. A major challenge is the vast abundance of Greenland's outlet glaciers and the variation of behaviour even on a regional scale (Moon *et al.*, 2012). Large scale satellite-based observation made it possible to observe three general categories (Fig. 1), according to their seasonal velocity pattern (Moon *et al.*, 2014). Yet, the mechanisms driving the different patterns are not determined.

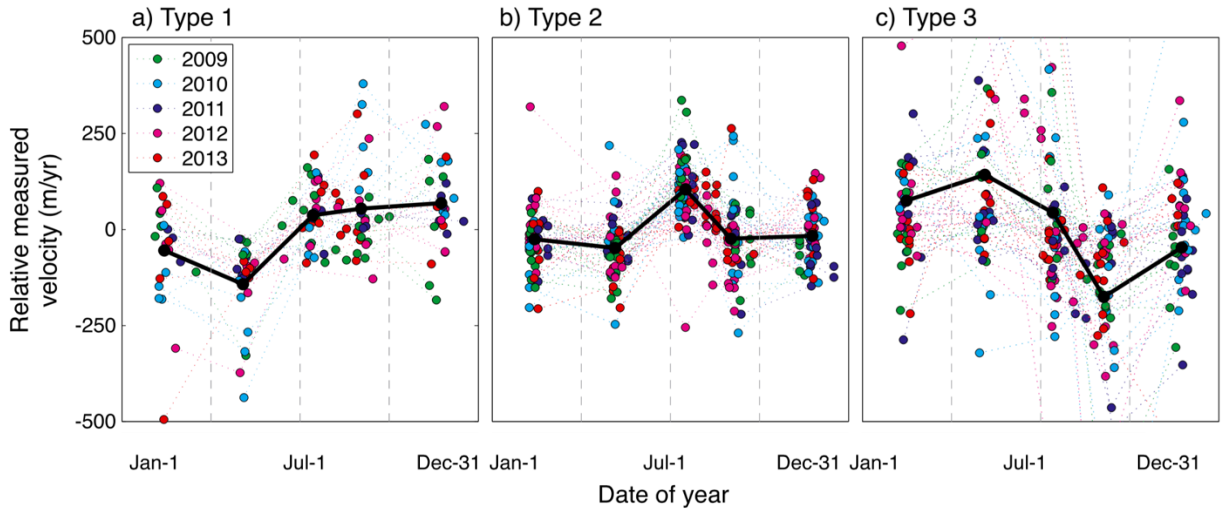


Fig. 1. The three identified types of maritime-terminated outlet glaciers in Greenland (Moon *et al.*, 2014, Figure 2).

In order to understand the link between the processes on a local scale, further investigations are needed. The study sites however, are mostly in remote areas and therefore hardly accessible, particularly during wintertime. Thus, field campaigns with accurate instruments are expensive, non-trivial and likely to be carried out during summer time only.

As a feasible alternative, satellite-based investigations are often used. Advancements in recent years have made high quality satellite data increasingly available for civilian use, some freely available, with ever improving temporal and spatial resolution (Pope *et al.*, 2014). Nevertheless, satellite data may still be too coarse in temporal or spatial resolution, allowing investigations at best with a footprint of several meters on a weekly scale.

1.1.2 Time Laps Cameras in the Cryosphere

Examples of Application

Time lapse cameras from a mono-optic camera perspective have been used lately within the Cryosphere for many different application, such as the investigations of supraglacial lake drainage processes (Danielson and Sharp, 2013), for quantifying calving events (Adinugroho, 2015) or for displacement detection:

Effective motion detection can be calculated from photo sequences, even by digitalizing analogue image negatives. Evans (2000) tracked glacial displacement successfully, without the attempt of conversion into velocity. Ahn and Box (2010) quantified glacier velocity by projecting the image displacement onto an digital elevation model (DEM) based on ASTER-satellite imagery. However, time lapse photography is not used in many studies as a basis of scientific research. This is probably due to the fact that the projection of an image into world coordinates is work intensive,

to say the least. Airborne images have the huge advantage that they are looking down onto the scenery, where as terrestrial based photography, with oblique viewing angles, has added complications to achieve image rectification (Fusiello, Trucco and Verri, 2000).

The Field of Computer Vision

The principles of feature tracking and template matching are however well known and widely used in airborne remote sensing. The aim of gaining information from images finds various applications, from closed circuit television (CCTV) monitoring to autofocus functions by facial recognition in the latest digital cameras. The entire field of *computer vision* is dedicated to the question of how to detect feature in images.

A cornerstone was laid by Lowe's (1999) scale invariant feature transformation (SIFT) approach (Priese, 2015, p. 302). This technique allows automatic identification and detection of features independent of their size and somewhat invariant of the viewing angle. The *SIFT* technique allows a relative movement of either the camera or the feature, or both. Combined with the principles describe by Ullman in 1979 it was now possible to not only reconstruct or rectify the scenery based on image information (Ullman, 1979), but also to successfully track features within scenery while motion occurs. The logical combination of both methods lead to Structure from Motion (SfM) (see Chapter 2.3), the ability to reconstruct a three dimensional relief of a scenery using images acquired from different locations (Westoby *et al.*, 2012). With the use of unmanned aerial vehicles (UAV), also called drones, this technique has been used for a variety of applications, including the observation of calving glaciers (Ryan *et al.*, 2015).

Provided Software

Until recently there was no software available for motion detection from oblique photography in glaciology. This possibly led to few studies using time lapse cameras for this purpose in the science community. A major contribution to this field of science was made by Messerli and Grinsted (2015) with the first open source toolbox called *ImGRAFT*, a MATLAB® based function bundle which stands for Image Georectification and Feature Tracking. *ImGRAFT* offers a fully operable script with several application examples, open for individual adaptation.

Following the creation of *ImGraft*, other similar software packages have been developed. About one year later James, How and Wynn (2016) presented their free-to-use software called *pointcatcher*. As the name implies, the software is meant to manually select distinctive points to be tracked, which they applied to the tracking of dirty cones. As two high- resolution DEMs are used, the advantage is a redefined three dimensional position of the projected features.

1.2 Research Gap and Motivation

Even with the potential of a high temporal and spatial as well as all-year observation technique, such as time-lapse camera (TLC), all reviewed publications are concentrated on summer periods only. Moreover, most are dependent on digital elevation models of high resolution, such as LIDAR DEMs.

This leaves two major advantages of TLC unrecognised. First, TLC-based methods can achieve greater temporal and spatial resolution than satellite images, granted with smaller coverage.

Second, they perform better in terms of cost-efficiency compared to most available observation systems. Time- lapse derived velocity calculations might not be as high-precision as other

techniques. Thus, is seems to some extent pointless to make them dependent on expensive, high definition DEM.
Furthermore, neither *ImGRAFT* nor *Pointcatcher* has been used with oblique photography for Greenland's maritime-terminating outlet glaciers, unrecognised the potential of Greenland's role in terms of contribution to SLR.

1.3 Aim of the Thesis

My master thesis investigates the benefit of oblique, mono-optic time-lapse photography for glacial flow detection. Using time lapse cameras allows for seasonal observations with a high spatial as well as temporal resolution while being a relatively low-cost and easy method. Therefore, the overall aim of this thesis is to test the feasibility of oblique photography for glacial flow detection of the 'Eqi' glacier in West-Greenland.

Previous studies used time-lapse cameras in combination with precise and therefore sophisticated as well as expensive technique for topography scanning (Messerli and Grinsted, 2015; James, How and Wynn, 2016). However, it is the goal to combine time- lapse images with an as easily applicable and low-cost method for terrain modelling. Such a method could be the Structure from Motion technique for digital elevation model calculation, based on a series of pictures from the scenery. In this context, my investigations focus on the following questions:

What level of information can be gained from:
Q 1: … two dimensional displacement tracking within the image plane?
Q 2: … three dimensional displacement tracking using
    Q 2.1: … an existing, UAV derived digital elevation model?
    Q 2.2: … the Structure from Motion technique to create a digital elevation model?
Q 3: … the different techniques in terms of better understanding the processes on the Eqi glacier?

The level of information is understood as the spatial and temporal resolution of motion detection as well as the capability of detecting motion patterns.
To the best of my knowledge, such an approach of combining Structure form Motion with time lapse imagery has not been published yet. Thus, for uncertainty estimation I compare my results to complementary data.

1.4 Study Site:

Eqi Sermia (60°48'N, 50°13'W), often also called Eqip Sermia, is a medium sized tidewater outlet glacier in West Greenland, 90 km north of Ilulissat and Jakobshavn Isbræ (Fig. 2). Coming from the GrIS, the outlet glacier describes a curve to the north before entering a shallow fjord of Ata Sund (Rignot, Koppes and Velicogna, 2010). The 3.5 km wide concave-shaped calving front is oriented in south-west direction and flanked by two mountains. The terminus is anchored on both sides by pinning points in the fjord (see Fig. 3). In 2011 the first time lapse camera was mounted on the lateral moraine in the south of Eqi Sermia, taking daily pictures from the terminus. In the next year, a second camera was installed further up-glacier. The imagery of these camera provide the data basis for this thesis.

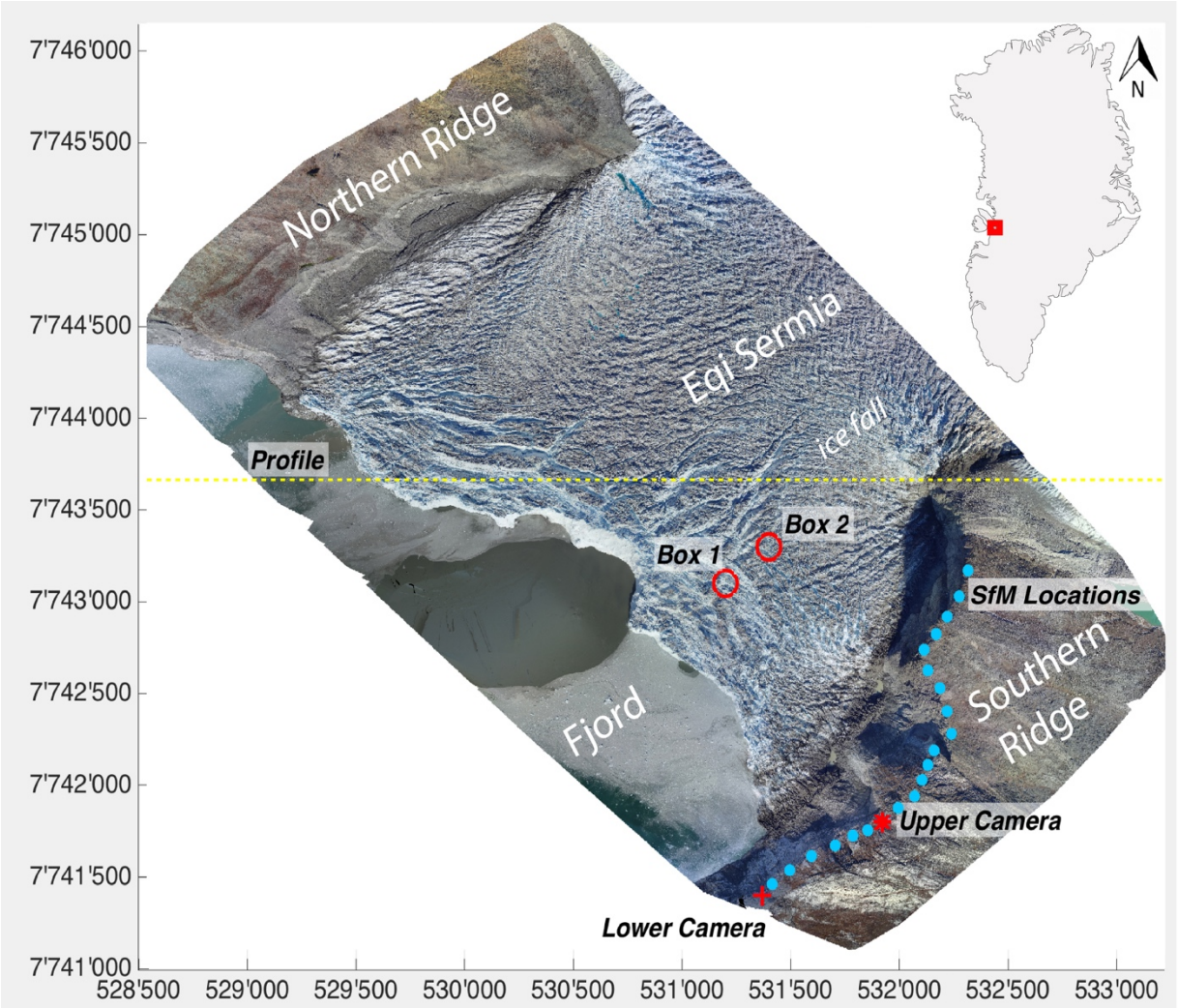


Fig. 2 Orthophoto of Eqi Sermia (21th of August 2016) and the positions of the two time lapse cameras (red asterisk and and plus) and the approximate positions of the photo locations for Structure from Motion (SfM) dotted blue. Red circles indicate the boxes used for the time series (see Chapter 3). The yellow dashed line shows the location of the profile in Fig. 9. The inset at top right shows the approximate location within Greenland.

Fig. 3 shows the location of possible surfaced bedrock, acting as pinning point (red circles). Image from lower camera on the 3rd of August 2016.

The extraordinarily long history of observations shows an advance terminus position from 1912 to 1920, followed by a retreat up to the 1950s. Here it advanced and retreated for the next 50 years within approximately 1km. After 2000 the terminus retreated in an asynchronous behaviour of the two lobes for more than 2.5 km up fjord (Lüthi *et al.*, 2016).
The terminus in 2016 can be described in two parts. The southern lobe is characterized with a ~50 m ice front. The northern lobe shows a ~200 m high and inclined calving front of approximately 45°. These steep ice cliffs are likely to favour larger calving events of the order of 10^5 m^3, which can lead to severe tsunami waves (Lüthi and Vieli, 2015).

In summer 2014, the flow velocity was measured to 12 m/ day, which is double the rate of the long term average. The maximum flow speed observed was 15m/ day in 2014 (Lüthi *et al.*, 2016). The submarine melt is estimated to be between 1m/ day and 1.8 m/ day (Rignot, Koppes and Velicogna, 2010; Lüthi *et al.*, 2016).

2 Methods

Two different approaches are pursued to detect glacial motion, both using template matching on image series from time-lapse cameras.

The chapter is divided into three subchapters. The first subchapter describes the available datasets, and the equipment and software used. Then, the first method is described, detecting displacement in the image plane only. By using a distance estimate this displacement can be converted into an assumed 2D- velocity. The subsequent subchapter describes the Structure from Motion technique used for digital elevation model calculation. The last subchapter uses the created DEM of the second method by applying a more sophisticated approach, allowing projection and reprojection of image points into world coordinates. See Fig. 4 for clarification of how the methods are related to each other.

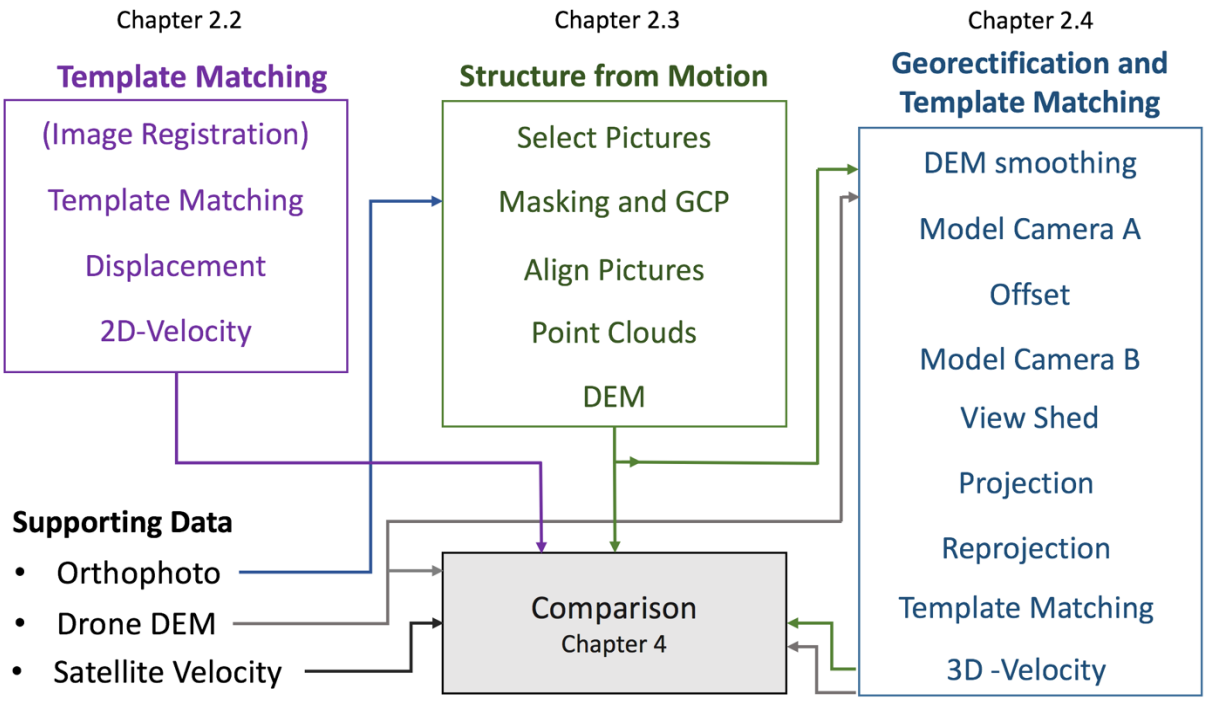


Fig. 4 Flow chart of different methods in use and how they are interconnected.

All of these subchapters contain a general overview, followed by a brief introduction on the principle theories and ending with a detailed description of the method and its implementation.

2.1 Dataset, Data Management and Data Processing

First, the four different kinds of datasets used in this thesis are presented: Time-lapse imagery, oblique photographs, digital elevation models and velocity calculation for comparison. Following this is a list of the software used and its implementations, and finally, a short definition of the used coordinate system.

2.1.1 Datasets

Image Series from Time Laps Cameras

In 2011 a time-lapse camera was installed on the lateral moraine near the calving front of Sermia. Pointed towards the fjord, it captured daily images of the calving front from slightly behind. Due to the rapid retreat of the outlet glacier's terminus, the camera had to been reoriented and then repositioned again on the 31st of October 2015 as the glacier retreated out of the picture. Since then it is looking inland, facing the calving front (Fig. 5, left). The camera, a Panasonic Lumix DMC-LX 3, is still in place and hopefully still taking images. The data series was last retrieved on the 3rd of September 2016 (see Table 1). This continuous succession of daily images, taken at midday, is one of the main data series used in this thesis.

Table 1 Camera parameters. From left: Camera model, Location in UTM 22N Coordinates (see Fig. 2), start and end date of used time series in [MM.dd.yyyy], time laps rate per day, count of retrieved images from camera, sensor size in mm, effective focal length in mm, pixel resolution of images.

Camera	Location	Time Series	Image Rate	Images retrieved	sensor size [mm]	focal [mm]	resolution [pixel]
Panasonic DMC-LX 3 "Lower"	530870 / 7741400	10.31.2015 – 09.03.2016	1/day	306	8.070 x 5.56	5.1	3648x2736
Leica D-LUX 3 "Upper"	531422 / 7741798	(03.04.2016) – 08.22.2016	> 1/day	233	8.498 x 4.780	6.3	4424x2376
Brinno TLC 200 Pro "Brinno"	531764 / 7742950	08.21.2016 - 08.28.2016	> 145/day	1000 +	5.376 x 3.024	2.8	1280x720

Fig. 5 shows the used cameras. Left: The lower camera (circle) on the lateral moraine, facing the calving front (Photo by A. Vieli in July 2016). Right: The upper camera looking onto the lower terminus of Eqi Sermia (Photo by L. Moreau in July 2015).

In fall 2015 a second camera was mounted on the southern ridge, flanking the outlet glacier. The camera's perspective is from higher up so that the lower parts of the terminus are within its field of view (Fig. 5, right). The camera's internal clock stopped working and therefore, the date and time was not recorded along with the images. This Leica D-LUX 3 camera captured pictures irregularly and the retrieved images were black for a long time in winter. By visual comparison of images from both cameras, the dates were reconstructed. Such a visual comparison was possible

by comparing the shape of the calving front with the weather conditions and the sea ice pattern. The exact time of the day the pictures are taken is unknown but it seems to be after noon or even early evening. The data series recovered with a successful reconstruction of the timestamp reaches from the 3rd of April to the 22nd of August 2016 (see Appendix, page 97).

During the field season in 2016 a third camera was installed, covering the upper parts of the Eqi Sermia. After seven days, this camera was taken down and brought home again. Fast changing weather conditions during the field season as well as the wide angle and the comparably low pixel resolution of the *Brinno TLC 200 Pro* Camera make the images unsuitable. With only 10 minutes or less between pictures, the temporal resolution would however be favourable. In the end, this time series was not taken into account.

Pictures for Structure from Motion

Within the goals of this thesis is the calculation of a digital elevation model using Structure from Motion technology, as described in Chapter 2.3. For this reason, photographs of the terminus were taken every other day between the 21st and 28th of April 2016, with an image collection of 429 to 553 pictures per day (see Table 4 on page 47). As it was aimed to gain as many view angles as possible, the spacing of the photo location was approximately 100m along the southern ridge of the outlet glacier (see Fig. 2). Attention was paid that the time used for the data collection was as short as possible in order to prevent errors due to glacier flow during the time of recording. One run along the nearly 2 km long ridge, including picture acquisition and note taking took approximately 1.5 hours.

For the data acquisition, a customary Nikon D200 with a 23.6 x 15.8 mm CCD sensor and an image resolution of 3872 x 2592 pixels was used. Attached to the camera was a Nikon AF-S DX Nikkor 18-200 mm 1:3.5 - 5.6 G ED VR II lens with a maximum field of view of 76° (horizontally). Mounted on top of the camera was the global positioning system Nikon GP-1 with a horizontal accuracy of 10 m residual mean square under open sky.

Digital Elevation Models (DEM)

During the field season, fellow researchers from the University of Zurich, Switzerland, used the *senseFly's eBee* drone for airborne data acquisition. The flight on the 21st of August 2016 resulted in an orthomosaic and a digital elevation model of the outlet glacier's terminus, both with a resolution of less than 0.2 m (see Fig. 37 on page 62). This *drone DEM* is a backbone dataset used throughout the entire thesis, whether it is for retrospective Ground Control Point (GCP) acquisition, image georectification or graphic representation of the finding.

The recently released *Arctic DEM* (see Fig. 37 on page 62) of the Polar Geospatial Centre of the University of Minnesota is used for further assessment and as second input for evaluation. The data used is a compilation of Landsat 8 satellite imagery between 2011 and 2015 with a spatial resolution of ~0.5 m.

Alternative Surface Velocity Calculations

Two datasets are used as ground truth in order to compare the calculated surface velocity and velocity patterns. One dataset are velocity calculations derived from the *eBee*-drone flyovers on the 21st, 23rd and 25th of August 2016. The calculated surface velocities available are averaged over these time periods. The other calculations are based on satellite imagery (Sentinel A-1) since November 2015, with an averaged surface velocity over twelve days and a spatial resolution of 20 m. Both datasets are currently unpublished. The methodology to the one used for the satellite derived data is described in Strozzi et al. (2002).

2.1.2 Software

Image Series from Time Laps Cameras

The algorithm used for image derived motion detection as well as velocity calculation are based on *ImGRAFT* (Messerli and Grinsted, 2015), a free MATLAB®-based toolbox for image georectification and template tracking. Therefore, all calculations were done in MATLAB® (R2015b). *ImGRAFT* provides multiple samples on their webpage for download (www.imgraft.glaciology.net). The demonstration script *Engabreen* is underlying groundwork for this thesis.

Pictures for Structure from Motion

The digital elevation models were compiled in *PhotoScan Pro (1.2.3)* using the Structure from Motion technique (Agisoft LLC, 2016). Further processing of the DEM was completed in either ESRI*'s ArcGIS* (Version 10.4.1) or the open source QGIS (2.14.3-Essen).

2.1.3 Coordinate System

Throughout the entire thesis the coordinate system WGS 84/ UTM Zone 22 North (EPSG:32622) is used. Coordinates originally in other systems were converted. For example, the GCP's were imported into Agisoft using Longitude-Latitude coordinates system and the Mean Sea Level (MSL) geoid. The same is true for the camera's GPS position as this was not collected in UTM 22N. The coordinates are then transformed into UTM 22N either directly in Agisoft or in one of the used GIS software programs.

2.2 Template Matching (2D-Displacement)

2.2.1 The Principles of Template Matching

Template matching is understood as the detection of displacement of template features and is therefore, closely linked to *feature tacking*, *image registration* and *feature detection* in multiple images.

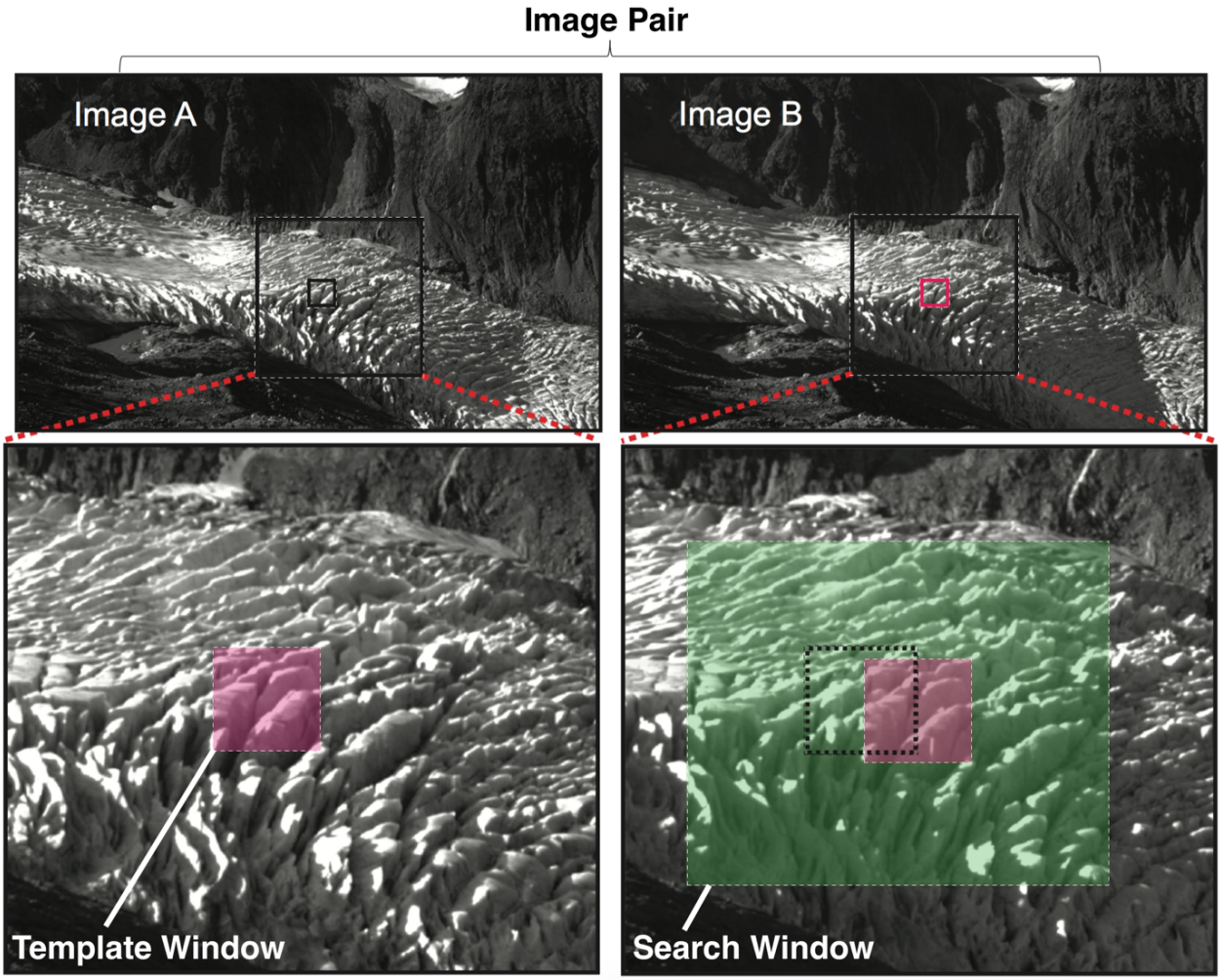


Fig. 6 Illustrative example of a possible template window (magenta) and search window (green). The pair of images (A and B) from a time-lapse camera at Engrabreen (northern Norway) is shown, taken in summer 2013. (Messerli and Grinsted, 2015, Figure 4).

Feature detection describes a method often used in computer vision and image processing. The method uses abstract imagery derived information to determine features at pixel level. *Key features* are distinct pixel patterns within a picture. These features can reach from pixel to pixel comparison to complex pixel cluster matching (Szeliski, 2010, p. 207f.).

For example, a blue pixel in a satellite image could be of interest in order to detect possible water ponds. Or as a more sophisticated example, smart phone cameras automatically search for human faces (facial recognition) in order to pull the focus onto the right image plane. In the first example, every single pixel is compared to the given value, for instance blue. In the second example, an advanced version of the key feature is used. Here distinct pixel-clusters have to be

detected such as eyes, eyebrows, nostrils or a combination of them. Therefore, the result will most likely be an indication of the similarity.
In this case, no specific key features are searched for, instead a template is given as an input. This has the advantage of increasing the robustness of the method, as multiple *key features* are detected. The software identifies patterns of key features within the *template window*. Once patters are detected, the pattern itself as well as the location in image coordinates (u, v) are saved. The algorithm then scans another images for this pattern, within the search window, and saves the location again (Fig. 6). In case of multiple possible matches, a cross-correlation function is used to determine one single candidate.

Now, as the locations of the pattern within the images are identified, the displacement can be calculated. By subtracting the coordinates of the feature in the first image A (u_A, v_A) with the coordinates in the second image B (u_B, v_B).

$$\Delta u = u_B - u_A \; ; \; \Delta v = v_B - v_A$$

$$|\Delta d| = \sqrt{\Delta u^2 + \Delta v^2}$$

The result of the *template matching* is the displacement of the pattern from image A to image B. The displacement $|\Delta d|$ can be calculated using Pythagoras.
Therefore, template matching alone is only able to detect motion in the image plane, which is a two dimensional displacement. Velocity measurements are thus not yet the product of this method.

Understandably, the matter becomes more complex as soon as the conditions changes from one image to another, for example by changing the view angle or the illumination (Messerli and Grinsted, 2015). For further information on how key features are identified and how they become to a degree invariant to scale and illumination, see Appendix on page 88.

2.2.2 Application of Template Matching Method for Displacement Detection

Template Matching

ImGRAFT's templatematch.m function was used for template matching. Instead of tracking only one template, a regular grid of template window is laid over the image. Every template is then searched for in the succeeding image. In order to limit the possible matches, the search radius is defined as a maximum distance from the central grid point (consequently, the search window has to be larger than the template).
In MATLAB the functions command looks like:

```
[du,dv,C,Cnoise,pu,pv]= templatematch(A,B[,pu,pv][,parameters])
```

with the output on the left side and the input variables on the right side (for 'a brief introduction into MATLAB' see Appendix, page 86). Where *du* and *dv* are the tracked position of the search window in image *B*. The initial template grid points are *pu* and *pv*, from image *A*. Further outputs

are: *C*, the peak of the normalized correlation coefficient, and *Cnoise*, the mean absolute correlation coefficient over the search area. These two obtained variables are used to judge the quality of the matches. In the parameters the width of both the template and the search window can be determined. Furthermore, an initial offset estimation in both directions as well as a super sampling factor for additional subgrid search can be specified (Messerli and Grinsted, 2015).
Here, the widths for the template window is set to to 150 pixels and the search window to 240 pixels. A super sampling factor of 2 is chosen. The window widths are determined by running and comparing the cross-correlation coefficients of different window sizes (see Appendix, page 102). The grid spacing is approximately 52 to 58 pixels with respect to the image ratio.

Image Registration

Prior to templates matching the pictures have to be aligned as good as possible. Weather, animals and humans can cause vibrations of the camera casing, called camera shakes. The alignment of the two pictures to be compared has to be nearly perfect, otherwise not the actual motion of the features may be measured, but instead the shake of the camera. This image correction method is called *image registration*. The concept of *image registration* works similarly to the *template matching* method. The difference is, however, that now stable elements in the images are compared, as rock faces or mountain tops instead of the glacier surface in motion. Any tacked motion on a stable object is an indication of camera shake.

The initial offset is estimated by using the *templatematch.m* function on a single window of 110 pixels' width at the northern ridge. The search windows width was 220 pixels. The detected offset was then accounted for by using it as input for the template matching of the ice motion. If the initial offset value exceeds 5 pixels, the image pair is rejected.

The above described image registration is far from perfect. Being a vector it accounts only for yaw and pitch motion. Neither roll nor translations can be detected, as this would require a rotation matrix. During the time of this master thesis several approaches for image registration techniques have been tackled, and no attempt was fully satisfactory. The failed attempts are described in the Appendix (page 86).

2D- Velocity Assumption

Until now only a relative displacement on the image plane is detected. The result is therefore a motion in u and v direction in pixels. Keep in mind that each pixel has the exact same size on the camera sensor but is likely to cover an area of different extent in the scenery. While in the foreground one pixel covers the equivalent of some millimetre it can be scaled up to metres or even tens of meters in the far background.
By estimating the distance between the camera and the object depicted, a bold assumption can be made on the effective displacements. This, however, is likely to work only at the given distance and if the main displacement is perpendicular to the line of sight. Any motion directly towards or

away from the camera cannot be detected. Therefore, ideally the camera should orientated perpendicular to the glacier flow (Ahn and Box, 2010).

To calculate the 2D-velocity, the tracked displacement $|\Delta d|$ has first to be divided through the elapsed time (Δt) between image acquisitions. By visual comparison of the oblique images from the camera with the drone derived orthophoto, the position to the window of interest on the ice is estimated. With the Pythagoras the Euclidean distance between the box and the camera can be calculated

Furthermore, the intercept theorem defines the ratios of similar triangles. The size of the line segment of two lines crossing in a two dimensional space is in the same ratio as the segment of two intercepting parallels. Applied for the purpose here, this means that the width of a pixel (x) to the focal length (f) is the same ratio as the depicted object size (X) in the real world to the distance (D) between the focal point and the object of interest (Szeliski, 2010, p. 27f.; Adinugroho, 2015).

$$\frac{x}{f} = \frac{X}{D} \;\Rightarrow\; X = \frac{x \times D}{f}$$

$$d = \frac{|\Delta d|}{\Delta t}$$

$$s = d \times X$$

As the size of a pixel in [m] and the focal length in [m] is usually provided by the manufacturer (see Table 1) and the distance can be calculated, the size represented by one pixel on the sensor can be calculated. Now only the tracked displacement (d) in pixel units had to be multiplied with X in order to get an estimation of the velocity (s).

In order to smooth the results, the mean for a total of nine neighbouring template grid points is calculated.

2.3 Structure from Motion

2.3.1 The Principles of Structure from Motion

Structure from Motion (SfM) is a rather new method used for acquisition of topographic data. It includes the advantages of remote sensing techniques without being as cost intensive (Westoby *et al.*, 2012). In principle the technique works like other optical photogrammetric approaches: Generating a three dimensional (3D) model out of two dimensional (2D) pictures by identifying key features and matching them to the collection of overlapping pictures. Multiple images form different locations and angles allow to reconstruct the geometry of the object of interest. Ideally pictures for SfM are taken in circular pathways of different altitudes around the object. As every feature is visible in several offset images the angle and distance from the cameras to the feature can be determined. Furthermore, the camera positions are estimated, adjusted and refined in an iterative process. As a result, a high resolution 3D- point cloud is generated using low-cost customer grade cameras (Javernick, Brasington and Caruso, 2014; Carrivick, Smith and Quiencey, 2016). For a more detailed description of the SfM technique, consult the Appendix (page 88).

SfM is even applicable in remote and hardly accessible regions. Neither narrow valleys nor high cloud sealing affects SfM data acquisition, when using a hand-held camera. Therefore, SfM is not dependent on flyable weather conditions or terrain, nor UAV battery life. Unlike photogrammetry, SfM does not require the absolute camera position nor the orientation. In theory, SfM is not necessarily dependent on satellite coverage for photo flight navigation nor on image coordinates for post processing; the GPS signal can be weak. Consequently, the 3D- model produced lacks an internal scale and orientation. SfM alone can only lead to a random reference system which is not to scale. Thus, ground control points (GCPs) are favourable. Without at least three GCPs and/or the world coordinates of the cameras positions, the georeference in global coordinates can not be achieved. Additionally, GCPs can be used as initial parameters to link multiple images and to reduce the uncertainty of the resulting point cloud.

2.3.2 SfM Method Used to Generate a DEM

Here, SfM is used to acquire multiple digital elevation models (DEM) of the terminus region of the glacier. The aim is to create a DEM which then can be used to project the tracked ice movement calculated from time-lapse imagery (2D) into real world coordinates (3D). This allows terrain mapping independent of airborne system.

Agistoft PhotoScan Pro, a well established, commercial software (Javernick, Brasington and Caruso, 2014; Ryan *et al.*, 2015) was used for processing. The only available literature for the black box like software is the manual with little information about the algorithms in use (Agisoft LLC, 2016).

Data Collection

Ideally, the lower terminus, a nearly 10km^2 area, was mapped. The challenge for this thesis is that only one side, the southern ridge of the outlet glacier, was accessible for photo acquisition (see Fig. 2). The northern ridge was inaccessible. The crevassed and therefore rugged as well as instable ice surface of the terminus is not crossable by foot. Alternative routes would be by sea or air. Unfortunately, no such undertaking was planned in the field work in August 2016. Thus, the view angles onto the glacier terminus are limited. Steep cliffs and flat ridge tops are also limiting the range of elevation images could be taken from.

Image Selection

The images were then manually filtered for processing. Pictures not in focus or with large obstacles were removed. Flies and mosquitos in front of the lens turned out to be a bigger problem than expected. In between 408 and 443 pictures remained after such manual filtering. Soon it became clear that too many pictures are likely to create an abundance of erroneous key points (see Appendix, page 102). Only the best shot from every photo location was selected, limiting the data to less than 40 pictures.

Image Alignment

The images were imported into the software *PhotoScan Pro*. Several options for alignment can be chosen. The best result are found using the *referenced method*, where the known camera locations are used for initial pair selection (Agisoft LLC, 2016). The resulting model is sparser with many false positives when compared with other methods. However, the geometry, scale and the internal dimensions are closer to reality.

Three levels of accuracy can be selected, determining different, to the user unknown, thresholds of the correlation match correlation. Experience showed that the best results for this study site were achieved by running the alignment with no higher than medium accuracy levels. Additionally, the key point limit and tie point limit can be set. Experience shows that the best result is gained with 40 000 key and 4 000 tie points.

Masking and GCPs

By masking the pictures and determine GCPs, a higher quality of point cloud was achieved. This is manifested through fewer outliners, a general smoother appearance, and a reduction of the projection errors. Images were masked manually so that only the main scenery is visible (Fig. 7). The foreground as well as the air and sea patches in the images have been masked out. The foreground is likely to be alike and therefore the programs finds key features in pictures which are not overlapping in reality. Clouds in the sky and sea ice/mélange are source of false positives: Icebergs might give a distinct feature, same as clouds but as they move too fast between the acquisition of two images they lead to distortion of the 3D model.

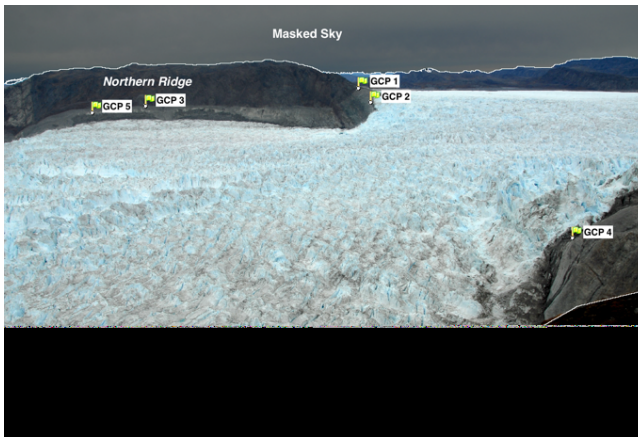


Fig. 7 Sample pictures showing of the masking of pictures and the position of the ground control points (GCP) 1 to 4, displaying the outlet glacier and the northern ridge used for Structure from Motion. Also showing an additional marker (GCP 5) without georeferencing, unique to day 2. Grey shaded areas such as the sky, sea and the foreground are masked out.

GCPs where collected during the field work on the southern ridge using differential GPS with an accuracy of up to 1 cm. They could not be used, as they turned out to be too close to the camera. Therefore, a new set of GCPs had to be created by comparing good visible features in the pictures to the drone derived orthophoto. The GCP's were manually placed in every image (Fig. 7). Finally, with this new input, the images were aligned again.

Calculation of Point Clouds

After the images have been realigned a sparse *point cloud* is generated using the known camera locations, GCPs, and the masked images. The sparse point cloud combines all the key features found in different image pairs to a single 3D representation of the scenery (Fig. 8, left).

Next, a dense point cloud is generated (Fig. 8, right). A dense point cloud enables the construction of a DEM. While generating a dense point cloud *PhotoScan Pro* let the user choose the quality of the result again. High quality was chosen, reducing the image resolution by a factor 2 (Agisoft LLC, 2016). This quality serves the purpose of this thesis, experience shows that the difference between high and ultra high resolution is negligible. The time savings however are significant.

The *depth filtering* provides a tool for culling outliers, by determining the distance of the keypoints to the camera in every picture. By choosing the approach of depth filtering the user can decide about the level of detail of the scene to be reconstructed. In this thesis a moderate filtering was applied.

Digital Elevation Model

Next a DEM was generated. According to the developers, the most accurate results are generated from dense point cloud data (Agisoft LLC, 2016). The interpolation option was enabled and the

A *model camera A* (`camA`) was constructed using the given inputs:

```
camA = camera(camera location, image size, view direction, focal length)
```

and the camA is then optimized by using the world (`XYZ`) and image coordinates (`uv`) of the GCPs

$$[\texttt{camA,RMSE,AIC}] = \texttt{camA.optimizecam(XYZ, uv, CamParam}_{7/20}\texttt{)}$$

Camera.m's *optimizecam* aims to reduce the projection error as far as possible. The error is measured by calculating the offset after a back and forth projection. For manual supervision the root- mean- squared error (`RMSE`) and the Akaike information criterion (`AIC`) are returned. The CamParam, represented by a 20-element long vector of logics (containing a string of 0 and 1's), indicates which parameters are sought to be improved. For this thesis, a parameter-vector in the form of '11100111000010000000' is applied. The first three slots representing the X, Y and Z of the word coordinates, followed by the image size in u and v direction. Slot 6 to 8 representing the camera's yaw, pitch and roll. Followed by the internal parameters such focal length in pixels as two-element vector, pixel coordinates u and v of the model camera centre, as a six element vector full distortion model and a two element vector for the tangential distortion coefficients. Here, however, the pictures are optimized for only the world coordinates, the view direction angle and a simple single vector distortion model.
For insight on the theory of the projection, see Appendix (page 91).

Offset

One single non-glaciated patch was tracked in both pictures using templematch.m. As in the chapter earlyer, the movement of this fixed location gives a first estimate of any potential camera movement (Fig. 10, white asterisk). This patch is rather big as it provides a general understanding of the offset between these two pictures. In this thesis, the western side of the northern ridge was used, as the change in geology provides good edges for image feature recognition. Additionally, this offset was used as a threshold: If the offset is larger than 5 pixels in u and v direction together, this image pair was rejected and the next pair was matched. This helped to drastically improve calculation time of the time series by culling unfeasible image pairs. With the offset as initial input, the templematch.m is used again, with a much narrower template and search window. Instead of taking a 110 pixel template and a 220 pixel search windows, 50 pixel and 100 pixel are used, respectively. Now as input a mesh [pu pv] is inserted, covering the entire northern ridge in the picture, also a non-glaciated part of the picture (Fig. 10, red crosses). The result [du dv] is the offset of any given point of the grid between the two pictures in pixel. Therefore, a theoretical point pu_n, pv_n in picture A is described as $pu_n + du_n$,$pv_n + dv_n$ in image B.

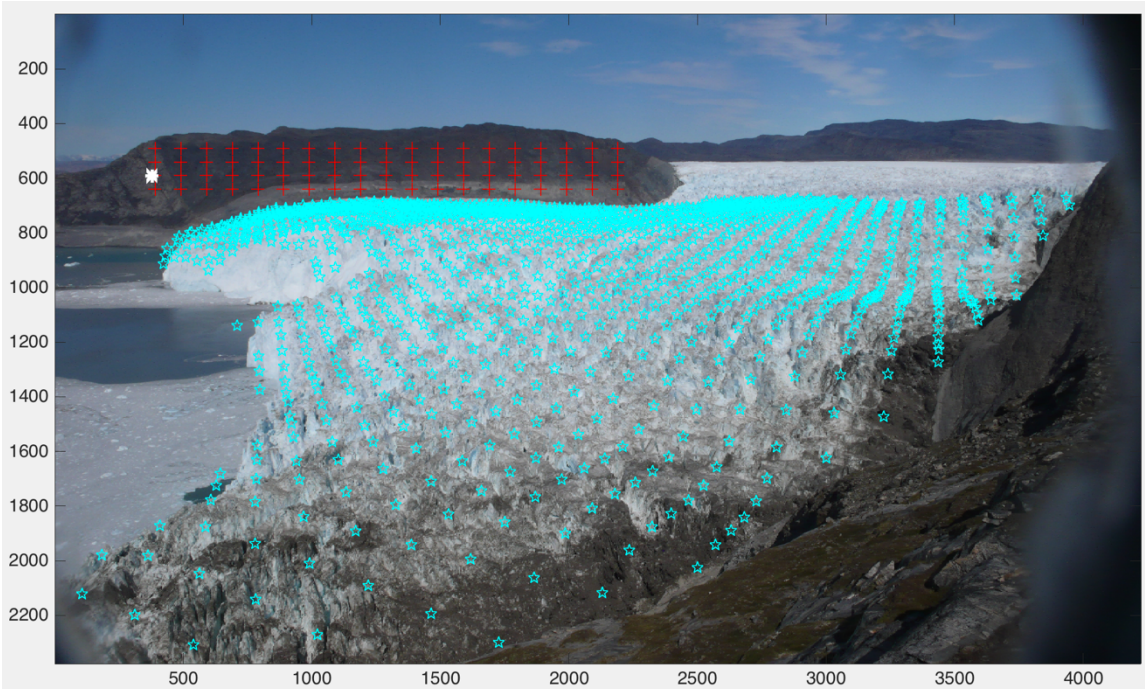


Fig. 10 Image L114686 from the upper camera as sample of the location of the initial offset point (white asterisk) and the registration grid (red crosses) on the northern ridge of Eqi Sermia. The uvA (cyan stars), the point grid where the velocity is calculated is displayed as well.

Camera Model B

The view direction of model camera B is determined, which is the projection model of the second image. In principle the cameras positions should be the same as before, as it is mounted on a tripod. However, strong winds, snow on top of the case, the self-weight or even humans, for instance changing memory cards, could change the cameras orientation.

By knowing the 3D-world coordinates (XYZ) of every point of the 2D pu, pv -mesh using *camera.m*'s *inversprojection*

```
xyz=camA.invproject([pu pv])
```

the model camera A can be optimized in such a way, that it fits for picture B

```
[camB,rmse]=camA.optimizecam(XYZ,[pu+du pv+dv], CamParam3/20))
```

The model camera B's parameters are only optimized for view direction, namely for yaw, pitch and roll. The remaining parameters are the same as in model camera A, such as the world coordinates, the distortion model, etc. Thus, the CamParam vector is '00000111000000000000'. The result is a *model camera B* (camB), which is quite similar to the previous camera model but now optimized for image B.

Note that by optimizing the model camera only the focal point of the projection is altered. This underlines a major difference between *ImGRAFT* and other image rectification algorithms, including the approach undertaken in the previous chapter (2.2.2, page 20). *ImGRAFT* does not calculate a rectified image B to compare to image A, giving two main advantages. Scaling, rotating and re-saving images can lead to lost quality. And using this method, only a text file with the

parameters have to be stored (Messerli and Grinsted, 2015), providng a huge advantage in terms of memory saving.

View Shed

A view shed was calculated using the smoothed DEM, making a binary map of which parts of the DEM can be seen from the camera location. This was carried out by *voxelviewshed.m,* another important function of the toolbox, by solving linear functions of projection vectors. The output (vis) is a logical matrix representing in bird's eye view of which point can or cannot be seen (see Appendix, page 103), based on the world coordinates (XYZ) and the camera position (Camxyz).

```
vis=voxelviewshed(XYZ,Camxyz)
```

Next a grid was laid out over the scenery, produced within the geographic boundaries of the provided DEM. Grid points outside the view shed or the glaciated area were culled out. The spacing of the grid was chosen to be at least 30 meters. The points corresponding to the criteria, so called candidate points (xyzA), are in world coordinates and are the points on the glacier where velocity can be detected.

Projection 3D to 2D

The candidate points in world coordinates were projected into pixel coordinates of picture A, as if no ice motion occurred. The *project.m*, a mirror-inverted function of *ImGRAFT*'s *invproject.m*, was used.

```
[uvA,~,inframe]=camA.project(xyzA)
```

The output (uvA) is the information where the points of the 3D-grid are situated in the 2D- picture coordinates for the model camera A (see Fig. 10, cyan stars). The *view depth* is here neglected by not using a variable name but a tilde-sign (~), as for this purpose the depth map is not of any concern. *Inframe* was used again to cull the points which were not within the viewfinder of the camera. Remember that the view shed is just a theoretical approach which points can be seen from the camera position, using linear projection, not a representation of what the camera actually captures.

Reprojection 2D to 3D

The next step aims to quantify the offset between both cameras. In principle the same step as described in the prior paragraph is now made for *model camera B*. But instead of taking xyzA and projecting them with the already elaborated model camera B parameters, the just gained uvA output are inversely projected. This has the advantage that unlike the previously constructed xyzA only the points within the frame are used to construct uvB.

Furthermore, by subtracting the uvB from uvA, the real camera shake can be determined, using the world coordinate offset instead of image based feature recognition.

Template Matching

The georeferenced grid points were then tracked. Using the *templatematch.m* function again, features around the determined grid uvA were tracked.

Experimental runs for the upper TLC showed that template and search window best fit is achieved by a width of 150 pixels and 240 pixels (see Appendix, page 102). The centre of the template windows is determined through the generated grid. The centre of the search window changes with every iteration, as another of the 150 x 150 pixels of the template window is used as a centre point. For each grid point the best match is then evaluated using a normalized cross correlation. The result is the final displacement of the tracked features (du_{final} , dv_{final}). Additionally, the mean and peak value of the correlation ratio are returned. The grid uvB is newly constructed using the refreshed uvA plus du_{final} , dv_{final}.

Velocity Calculation

By projecting both picture grids (uvA and uvB) into world coordinate system using *invproject.m*, the points A(a_x,a_y,a_z) and B(b_x,b_y,b_z) can be derived.

The three dimensional velocity can be calculated using

$$\begin{pmatrix} s_x \\ s_y \\ s_z \end{pmatrix} = \frac{\left[\begin{pmatrix} b_x \\ b_y \\ b_z \end{pmatrix} - \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix}\right]}{t_b - t_a} = \frac{\begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}}{\Delta t}$$

where the velocity s is calculated from the difference in all three coordinates coordinates $\Delta x, \Delta y, \Delta z$ over the time span between image acquisitions Δt. *ImGRAFT* allows to automatically extract Δt from the exif-data. For the upper camera a manually reconstructed file with time and date information is used (see Chapter 2.1.1).

As s_{xyz} is a vector, s_x and s_y can be extracted. The velocity, in X- and Y-direction s_n is calculated by adding the values of $|s_x|$ and $|s_y|$:

$$s_n = \sqrt{{s_x}^2 + {s_y}^2}\ .$$

The Z-axis is ignored due to changes < 0.1 m only.

Loop

In order to make the code feasible for a time laps series, a for loop has been created, encompassing the method described. An image pair of subsequent images is taken for each iteration. Theoretically, for a series of 91 pictures from the upper camera, a total match of 90 pictures is possible.

As the lower camera was readjusted in order to get more of the glacier into the field of view, some parameters have to be changed within the calculation process. Here, image number 71 (6th June 2016) is the first picture after the camera has been adjusted. Thus, several if, else expressions are required with image count 70 as threshold.

Quality and Uncertainty

The results are filtered using a Singal to Noise (SNR) and Peak Correlation Ratio (C) threshold of SNR>2 & C > 0.6. Camera shake as well as Signal to Noise ratio are used for quality assessment, and are presented in the chapter *Results* (Chapter 3.3, page 49). A subchapter is devoted to possible sources of uncertainty and error estimation. Further, the findings will be compared in Chapter 5.1.2 (page 72) to satellite derived velocity calculations.

(blank page)

3 Results

The organisation of this chapter will be as follows: In three main parts the results of I) template matching (2D) method, II) the DEMs obtained by using the Structure from Motion technique and III) the template matching including image georectification (3D) method will be described. Each of the subchapters dealing with tracking are split into three sections. The first section shows exemplary results from a chosen image pair to cover the spatial dimension of the calculations. The second section presents time lines, covering the temporal dimension of the calculations of both time- lapse cameras (TLC) used. In the third section, the uncertainties are estimated and possible sources of error identified.

The findings are compared to alternative velocity data from the same location and time period in Chapter 4. The discussion of all results can be found in Chapter 5.

3.1 Template Matching (2D-Displacement)

Results from the template matching method are presented here using only one out of three main algorithms of the ImGRAFT toolbox. As described in Chapter 2.2.2, template matching allows only a detection of the movement in the two dimensional image plane, referred to as *displacement*. With an estimation of the distance the velocity can be assumed as is done in subchapter 'Time Series with assumed 2D- Velocity' (page 41).

3.1.1 Image Pair Sample

This subchapter aims to give the reader an understanding of the possible results of template matching in the spatial dimension. Therefore, the results of one single image pair are presented. As a sample, the images L114685 and L114686, captured by the upper TLC on the 10th and 11th of June 2016, around midday, are presented. The four figures below show the same image pair sample with its grid of 67x43 template windows.

Fig. 11 presents the tracked displacement, which is the detected motion within the timespan of the two pictures in pixel units. As pixel units the value of $\overrightarrow{uv}$- vectors is understood.

$$\overrightarrow{uv} = \sqrt[2]{u^2 + v^2}.$$

A large range of different displacement magnitudes is visible downslope of the outlet glacier, with a tendency to increase towards the calving front. Little to no movement is detected on the ridges. Seemingly unstructured motion with heterogeneous direction and displacement detection is detected in the upper parts and left edge of the image pair, covering the sea and the sky.

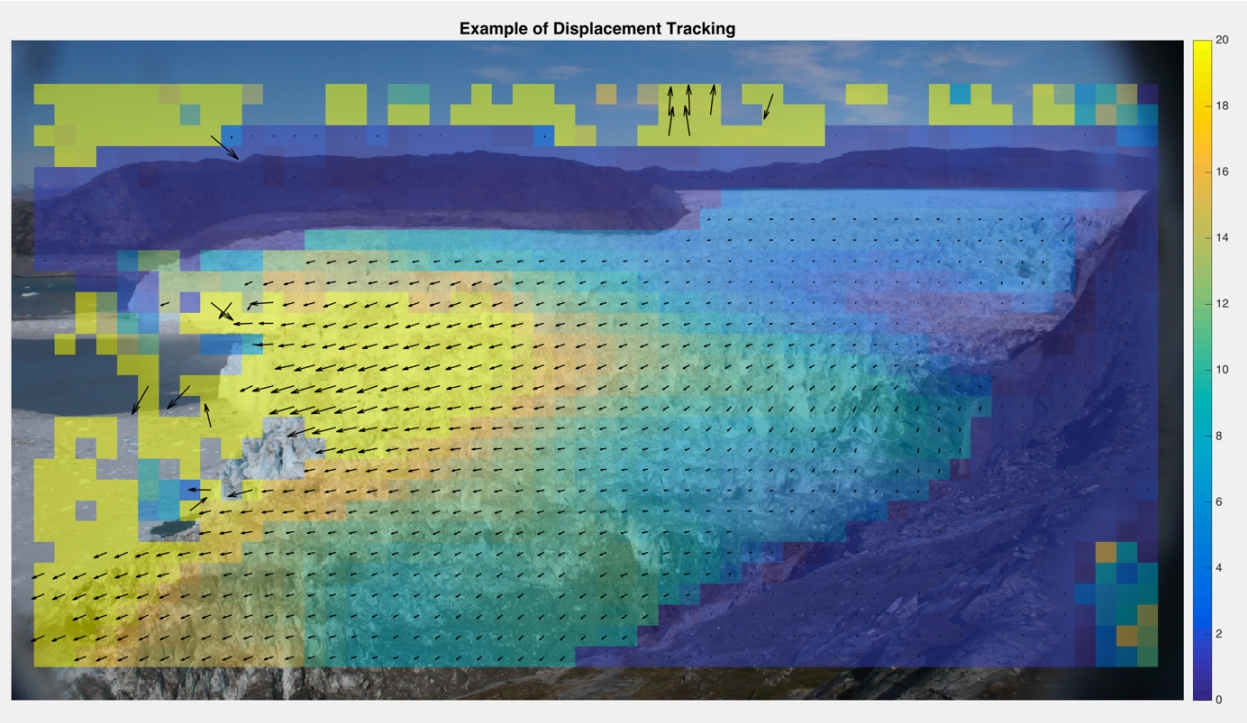


Fig. 11 Displacement Tracking. Image pair from the 10th and 11th of June 2016 (upper camera). The colour bar shows the displacement in pixel units for each of the windows. The black arrows are showing the direction. The length of the arrows is showing the normalized displacement. Highly opaque colours indicate a qualitatively decent match.

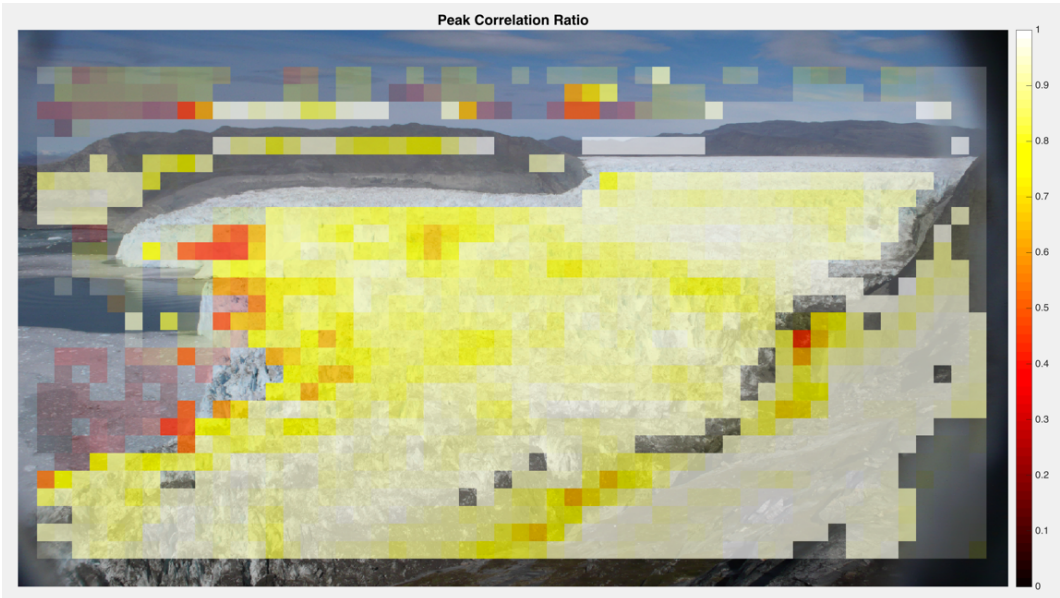


Fig. 12 Peak Correlation Ratio of the image pair from the 10th and 11th of June 2016 (see colourbar). Opaque colours indicate a qualitatively decent match.

Fig. 13 Mean absolute correlation ratio of the image pair from the 10th and 11th of June 2016. Red colours are indicating a high noise where as blue colours are representing a relative low noise. Opaque colours indicate a qualitatively decent match.

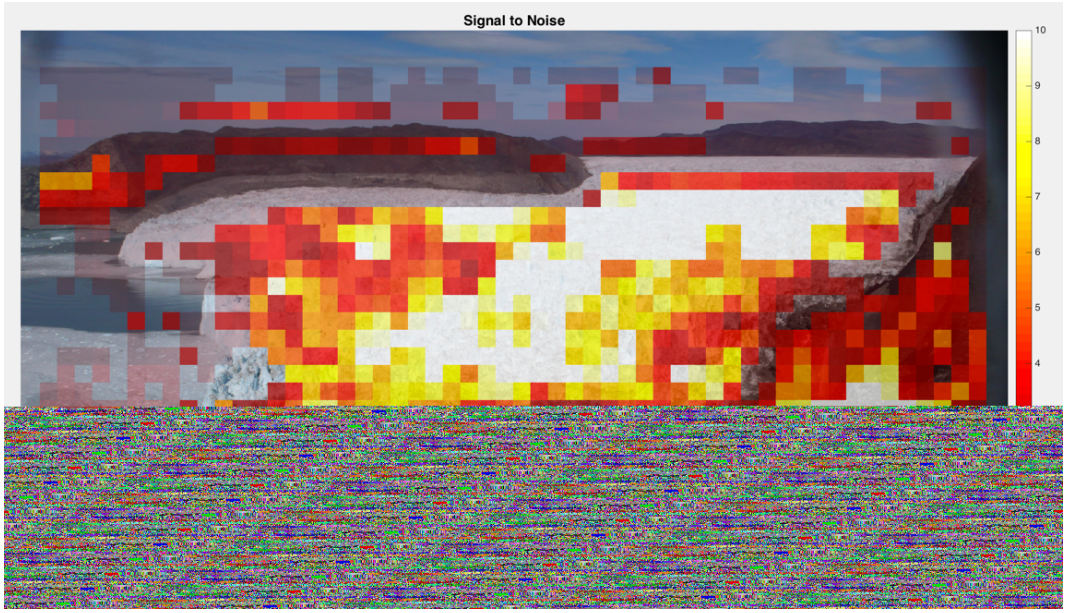


Fig. 14 Signal to noise ratio of the image pair from the 10th and 11th of June 2016. The brighter the colour the more unique a match. Opaque colours indicate a qualitatively decent match.

Correlation describes here the similarity of the template window to the search window using a normalized cross correlation. Peak correlation ratio is the highest match found (Fig. 12). The mean absolute correlation ratio (Fig. 13) describes the similarity of the template window to all possible search windows. Fig. 14 combines these two pieces of information into the signal to noise ratio (SNR) by dividing the signal (the peak correlation) to the noise (the mean absolute correlation ratio). In other words, the SNR provides information about the reliability of the found matches. This information is shown with the transparency of the windows: In all four figures below high opacity indicate a qualitatively decent match SNR > 2 as well as a peak correlation ratio > 0.6. For

application purposes this means that opaque coloured windows can be trusted more. These are mainly the areas nearer to the camera as well as the part after the icefall (see Fig. 1).

The initial offset, which is the camera shake, detected and corrected for this image pair is +2.01 pixels in u (x-axis) and +0.43 pixels in v (y-axis) direction.

3.1.2 Time Series

In this subchapter time series for the template matching results are presented for both the upper and lower TLC. The depiction of the entire data over the survey period, as it has been done in the image pair sample, is illegible to the reader's eye. Therefore, the results of two spots on the glacier surface are presented. Over time the glacier moves beneath these spots, while the spots remain on their position at 530700/7743100 and 530900/7743300 (UTM 22N). A snapshot of the ice below this spot is observed at every point in time when a picture is taken. These spots have the size of 3x3 template windows within the image plane. The location of these *boxes* are shown in Fig. 15 and are the same for every method compared for the entirety of this thesis.

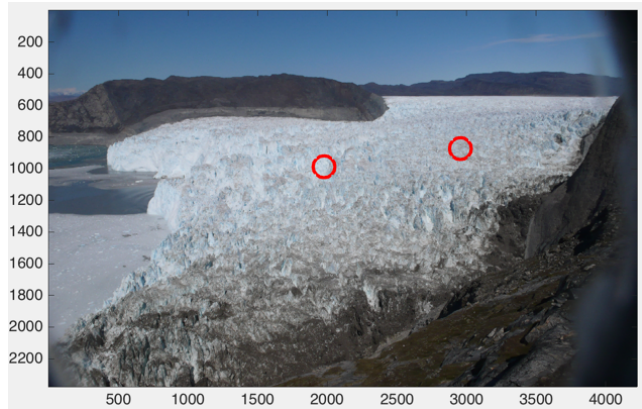


Fig. 15 shows the position of the boxes (red circles) within the image of the upper camera after the turn.

Four filtering conditions are used to cull outliners:

A) A threshold of displacement < 40 pixels.

B) Only displacements with SNR > 2 and peak correlation ration > 0.6.

C) Only displacement within 2σ (twice the standard deviation) of the day's mean.

D) No zero values are allowed, as it is assumed that there is always motion.

Table 2: Summary of data availability for both the upper and the lower TLC time series. Showing: The retrieved image from the cameras, the images with / or successfully assigned to a point in time, image pairs leading to a successful displacement calculation, data points available in the time line plots and the time span covered by the plots for the upper and the lower time laps camera (TLC).

	Images retrieved from cameras	Images w/ time (image pairs)	Image pairs calculated	Images points remaining after filtering*	Time period of successful calculations*
Upper TLC	219	91 (90)	55	38	2016-03-04 2016-08-22
Lower TLC	300	300 (299)	232	177	2015-10-31 2016-09-03

* averaged value of the two boxes, as depending on individual template windows.

Of the collected images, 17% from the upper TLC and 59% from the lower TLC were successfully used in the corresponding figures (Table 2, Fig. 16). The low success rate of the upper TLC can be

explained through the loss of time information, as well as 93 photos which were retrieved as black images. Including the culled pictures, the time series would reach back to fall 2015. Now the time period of the upper camera covers only 4 months. Additionally, the many gaps within the data series are noticeable. A much longer timer period with less data voids is collected and calculated from the lower camera. Note that the displacements are corrected for the lapsed time between the two images.

3.1.2.1 Time Series in Pixel Units

Fig. 16 shows the mean values of the two boxes over time for both cameras. More detail about the individual values can be gained from Fig. 17 for the upper and Fig. 18 for the lower camera.

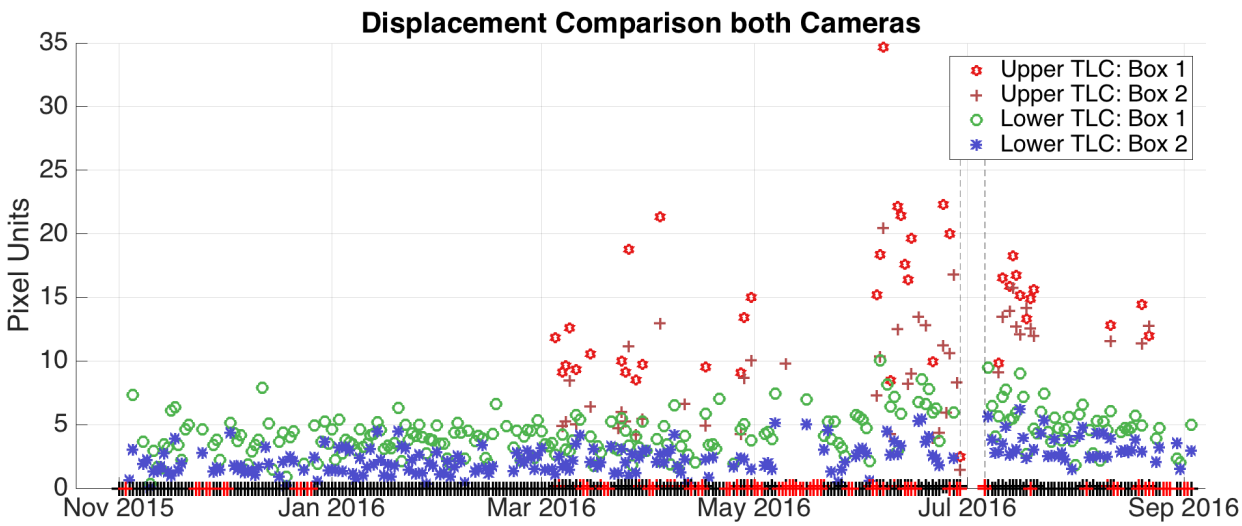


Fig. 16 shows the mean values of the tracked displacement of both boxes in [pixel units] over time for the upper (red hexagon and brown cross) and the lower TLC (green circle and blue asterisk). Two vertical dashed lines marking the time period of camera repositioning. The bottom line (around zero at X-axis) shows the image availability (red crosses). Images successfully used for template matching are marked in black. The upper row corresponds to the upper TLC and the lower row for the lower TLC.

One obvious difference of the two cameras is the data availability and continuity. Box 1 shows generally larger displacements than box 2. Thus, box 1 is seemingly moving faster.
For all time lines an increase is evident with a peak in June and July and a decrease towards September. This is less clearly visible for the Lower TLC, especially on the compressed y-axis as in Fig. 16. Therefore, compare with Fig. 18.

Generally, the upper TLC shows higher values than the lower one. The distribution of the data points can be seen for a single box to a certain point in time, as intended to be presented with the small symbols in Fig. 17 and Fig. 18. A difference of +/- 4 pixel units to the mean is not exceeded for the vast majority of the data points.

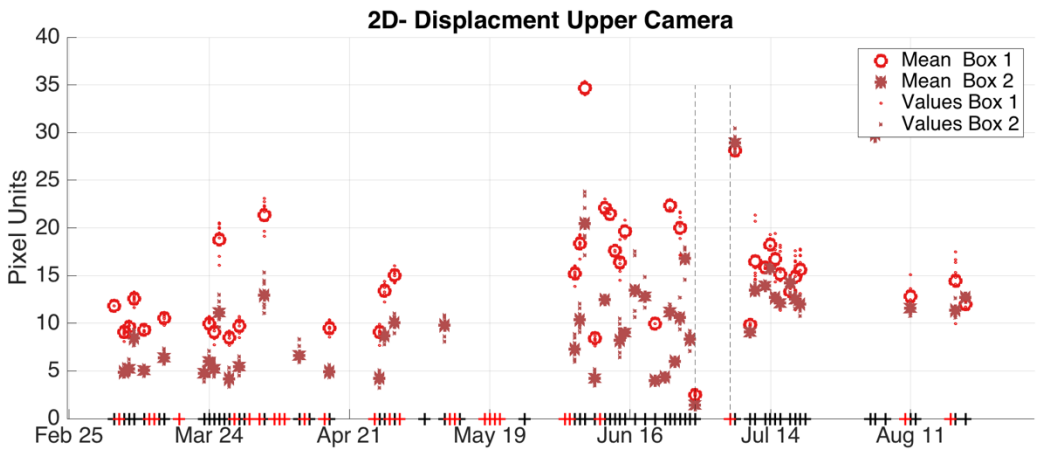


Fig. 17 Tracked displacement of the upper TLC for both boxes in [pixel units] over time. Red circles correspond with Box 1, brown asterisks with Box 2. The mean values are shown as bold, large symbols whereas the individual values of a box are shown as small symbols. Two vertical dashed lines marking the time period of camera repositioning. The bottom line (around zero at X-axis) shows the image availability (red crosses). Images successfully used for template matching are marked in black.

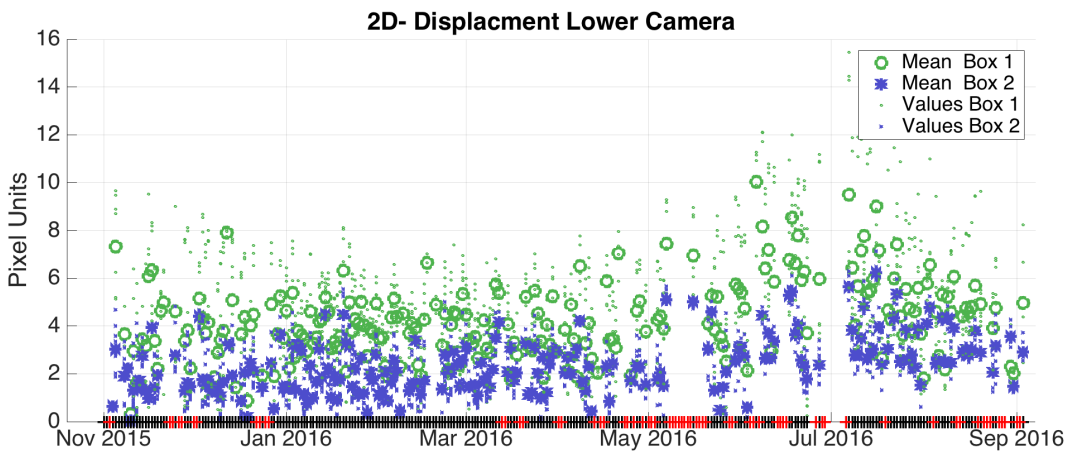


Fig. 18 Tracked displacement of the lower TLC both boxes in [pixel units] over time. Green circles correspond with Box 1, blue asterisks with Box 2. The mean values are shown as bold, large symbols whereas the individual values of a box are shown as small symbols. Two vertical dashed lines marking the time period of camera repositioning. The bottom line (around zero at X-axis) shows the image availability (red crosses). Images successfully used for template matching are marked in black.

3.1.2.2 Time Series with assumed 2D- Velocity

In principle the same graph as in Fig. 16 is shown also in Fig. 19. However, the y-axis is altered. Now it is showing assumed two dimensional (2D) velocity in meter per day instead of displacement in abstract pixel units. The approximation is made by a distance to pixel ratio. The further an object is away from the observer, the smaller it is (angular size). By connecting the distance between the camera and the object of interest, one can calculate the size that one pixel would have if projected into the real world (Table 3). This enables the velocity calculation as the displacement in pixel unit and the lapsed time are known.

Table 3*: shows the used multiplier for the corresponding boxes and camera.*

	Box 1	*Box 2*
Upper TLC	0.4795	0.5088
Lower TLC	0.5458	0.6070

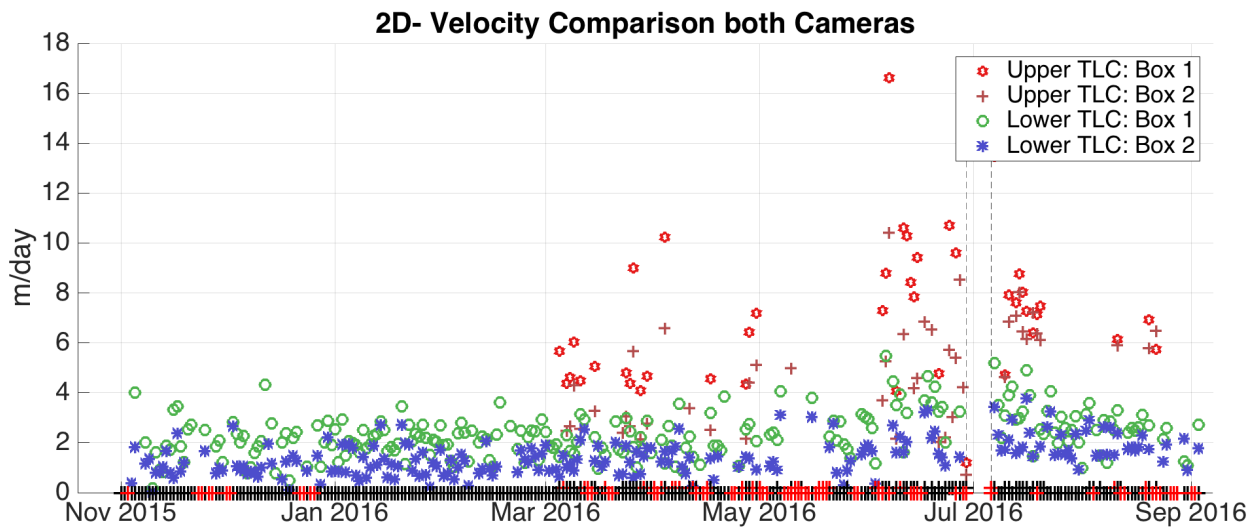


Fig. 19 Tracked displacement means of both boxes in 2D-Velocity [m/day] over time for the upper (red hexagon and brown cross) and the lower TLC (green circle and blue asterisk). The bottom line (around zero at X-axis) shows the image availability (red crosses). Images successfully used for template matching are marked in black. The upper row corresponds to the upper TLC and the lower row for the lower TLC.

Again, the same pattern is visible with an increase in June and July. This is expected as the underlying data is identical, but the units and the ratio from one time series to the other is slightly altered. However, two advantages can be gained from this graph: On one hand the reader gets a sense of the velocity when using SI-units, which is not granted if pixel units are used. On the other hand, 2D-velocity assumption makes it comparable with other findings. As the individual graphs for are similar to those already shown, they have been excluded.

Fig. 20 shows the initial offset in both u and v direction, which is treated as camera shake. Clearly visible are the set boundaries of +/- 5 pixel units for the maximum offset in pixel units. The initial offset stays around the image centre, as can be seen on the right hand side. Thus, slow but constant camera movements, such as a gradually nose down pitch of a badly fastened tripod, can be excluded. Fig. 21 shows the mean signal to noise ration of the time series. The SNR is generally

higher for the upper camera. If the SNR is compared to the findings presented above, whether a correlation to the camera shake nor to the displacement is obvious.

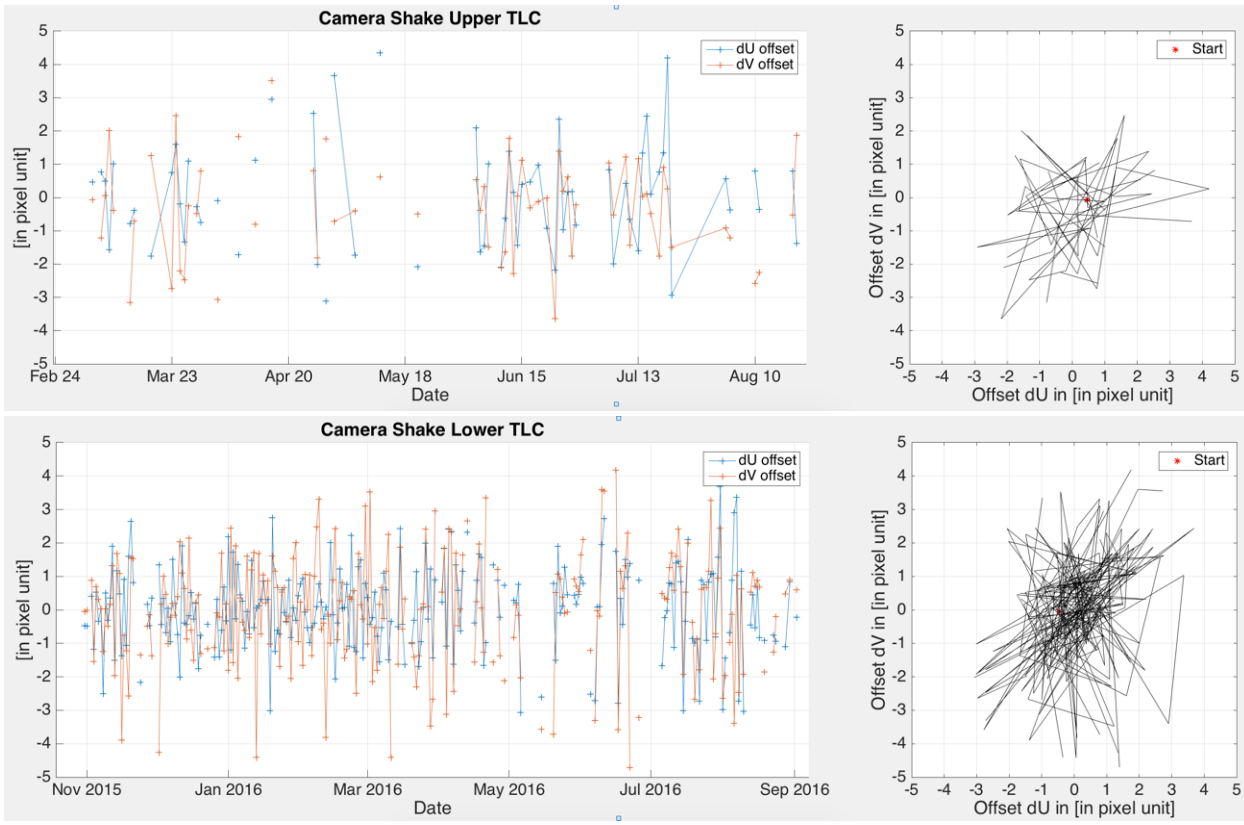


Fig. 20 shows the camera shake as offset in dU and dV direction over time (left) and mean displacement in dV over dU (right) for the Upper (above) and lower (below) TLC.

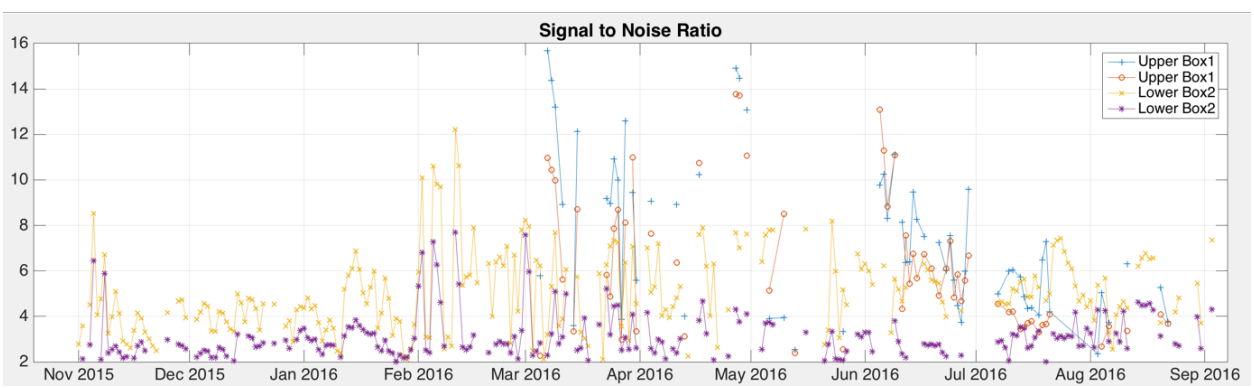


Fig. 21 shows the signal to noise ratio for both boxes and both cameras over the corresponding timeline.

3.1.3 Source of Uncertainties

There are five possible sources of error specific to this method. They will be described hereafter, despite the fact that they are nearly impossible to quantify.

A) Image Registration

The first potential source of error is insufficient image registration. As already described in the method chapter, only a rough estimation of the camera shake could be estimated as a single reference point is used for alignment. This leads to an inability to detect roll and translation movements or combinations of multiple motions of an affine transformation. Thus, the insufficient image registration is likely to be a source of error. This potential error is reduced by using a threshold of 5 pixels. The average value (root-square) of the initial offsets (du, dv) in pixel units for the upper camera is du=1.37 and dv= 1.87. For the lower camera it is substantially lower with du= 0.22 and dv= 0.61. In the worst case this corresponds to an initial offset of up to 1.2 m at the given distance.

B) False Positives

A second source of error are false positives: Instead of tracking the actual patch of ice that moved, a similar looking patch could by mistake be tracked. This leads probably to most of the outliner. Therefore, some effort was put into the optimisation of the size of both the template and search window. The size is important as a too small search window restricts the actual match. A too large window increases the possibility of false positives.

It was tried to cull most outliner effectively using the filters mentioned. As evident in the figures above, this worked probably more reliably for the lower camera. Additionally, false positives could also lead to too small results. A lower margin, however, could not be determined but by omitting zero values. More over, the size of search and template windows are optimized for a time step of roughly 1 day. The larger the time gaps between an image pair, the more likely is it to create false positives.

C) Foreshortening

Third, probably the largest source of error is the inability to track any motion vectors towards the camera position or away from it. That is why only a 2D- velocity within the image plane can be tracked. The result is a likely an underestimation of the actual flow velocity, as the third dimension is missing. This undercatch can be estimated with simple geometry given the camera's orientation by assuming the general flow direction of the glacier. The fraction of the movement vector towards the camera can thus be estimated, which is proportional to the undercatch. Thus, the undercatch for the upper TLC is between 5° and 17° depending on the box. The undercatch for the lower TLC is up to 44%.

D) Geometry of the Boxes

A fourth source of error is the geometry of the templates on glacier surface. A square-shaped template in the image plane, if projected onto a flat surface, will be a trapeze-shaped area. Thus, the term *box* is used as the shape within the 3x3 templates the image plane, and does not refer to the covered area on the glacier. Additionally, the shape of the projected box will vary with the

topography. This leads to different distances and displacements within the projected boxes. The effect is minor due to the abundance of pixels on modern DSLRs sensors, and therefore the influence of this problem is ignored.

3.2 Structure form Motion Digital Elevation Model

An alternative approach for DEM acquisition is carried out in order to get as independent as possible form any sophisticated terrain mapping techniques, by using Structure from Motion on imagery captured hand-held with an off-the-shelf camera. In the following chapter the resulting DEMs are presented.

3.2.1 Presenting the Data

All DEMs show the same general patterns (Fig. 22): The outlet glacier is flanked by two steep ridges, with their tops some hundred meters higher than the glacier surface. The glacier itself is represented with a higher elevation at the rear (northeast) than at the calving front (southwest). Along the flow line, which is estimated to be the line connecting these two boundaries, a convex raise is stretching out. The terrain falls to both side of it, towards the ridges. Eye-catching are the two plateaus of same elevation formed on both side of the middle raise, down-stream after the icefall.

At a closer look Day 1 (top left) seems to be a quite good representation of the outlet glacier. Crevasses and seracs can be seen along the flow line of the glacier. Additionally, few voids, sinks and an overall smooth appearance is noticeable. However, a big part of the southern end of the calving front is calculated to be lower than 60m. This altitude is absolutely possible but due to the value ranges of the other DEMs displayed transparent.
Day 2 (top right) stands out with a large part in the fjord of an elevation between 100 m to 120 m. Also its southern ridge is seemingly too low in elevation. In general, the entire DEM is up to 40 meters lower than the others.
Day 3 (middle left) looks in many regards similar to Day 1, such as distribution pattern, elevation and extent. Two blobs of elevation are visible on the western side of the outlet steam, near the calving front, which are unique to this calculation.
Day 4 (middle right) calculates the sea relatively high. Apart from this it matches up with the DEMs of day 1 and 3 well in terms of elevation.
The *Assembled DEM* (bottom) is the average of every grid point of the four DEMs including a general offset in Z-axis direction of 40 m. In a first impression, the calving front as well as the ridges on both side of the outlet glaciers are seemingly nicely represented.

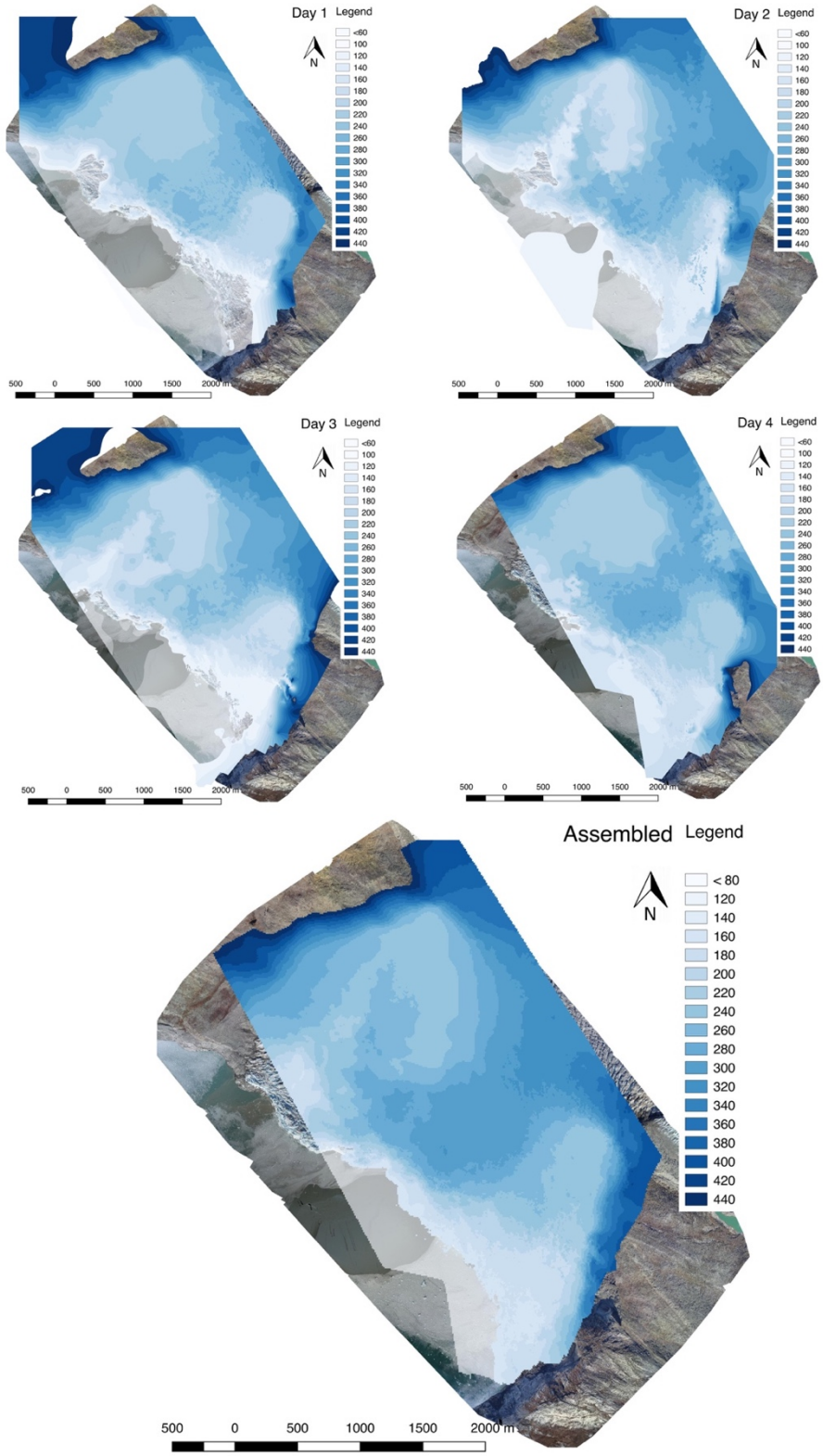


Fig. 22 shows the elevation in [m] of the four calculated DEMs on the 21st (top left), the 23rd (top right), the 25th (left), the 28th of August 2016 (right) as well as the assembled DEM (bottom). The orthophoto of the drone-DEM helps in terms of orientation. The legend shows the elevation in meters. Transparent are areas below 80m.

3.2.2 Uncertainties

It becomes evident from Table 4, that day 2 is the only run where an additional marker had to be added in order to get the error down to a reasonable level. This additional marker however was not georeferenced. The four common markers are all at the same positions in every scenery and picture (see Fig. 7). Out of two reasons the count of markers is intentionally low: Firstly, a larger marker count does not automatically increase the accuracy, and may in fact do the opposite.

More markers seemingly led the program to projection failure. This is probably due to the uncertainty of the redone markers. Secondly, only a few markers in total are uniquely distinguishable in both satellite images and oblique photographs. The projection error of the markers is in general below 6.5 m and the pixel error below 2.3 pixel offset. Note, that the highest projection error [in meters] (day 2) is not aligning with the highest projection error [in pixels] (day 3). This can be explained, as a pixel in the foreground covers a much smaller area than a pixel in the background. This measure can be described as angular size. The projection error is the offset of an object's known position and its reprojection. The known position can be derived from manual entry, such as the ground control points, or from other measures such as triangulation and projection in regards to other points. The sparse point error [in meters] is the root-mean-square error of the projection error averaged over all tie points in all images. The ground resolution of the DEM is an averaged value of all the pixels. The true resolution varies due to the effect of the angular size, as described above. Neither the error of the dense point cloud nor the error of the DEM are provided by the program. However, they are closely linked to the sparse point cloud.

Table 4*: Table of parameters from the SfM-Calculation is displayed for the four survey days. Columns: First column indicates the date of collection, e.g. 2016 August the 23rd.Images, count indicates the number of images used for the SfM. Markers, count indicates the number of Markers used where the brackets indicate how many of these mark of these markers are GCP's, i.e. including GPS coordinates. The averaged error in meters and in pixels of the markers after the projection. Followed by projections per marker (i.e. matches in pictures). Marker1/ Marker2/ Marker3/ Marker4. Sparse Point indicates the amount of tie points and the projection error in meter (RMSE). Dense Point shows the count (#) of key points in millions. The last column shows the resolution of digital elevation model in in meter per pixel (average). Values are extracted form Agisoft PhotoScan.*

	2016	Images	Markers				Sparse Point		Dense Point	DEM
	Aug.	count	# (GPS)	Error in m	Error in pixel	Projections	Count	Error m	# mio.	m/pix
Day 1	21st	39	4 (4)	4.23	1.89	39/38/38/13	32'457	130.79	3.93	0.98
Day 2	23rd	31	5 (4)	6.49	2.07	28/28/28/08	24'850	158.60	3.55	0.96
Day 3	25th	35	4 (4)	2.26	2.21	44/34/30/15	26'130	161.47	2.79	0.95
Day 4	27th	30	4 (4)	0.88	1.27	26/27/21/12	23'478	129.78	4.53	0.73

Table 5 *shows basic statistics of the two DEM used for projection. With mean height (Mean), standard deviation of the mean (Std), minimal (Min) and maximal (Max) altitude in [m] of elevation (WGS 84 geoid) for the Structure from Motion and the drone derived DEM*

DEM	Mean	Std	Min	Max
SfM	248	103	0	547
Drone	221	139	18	533

As already mentioned, the the error margins of the GCPs location is set to an uncertainty of 0.005m whereas the camera position is set to 10m. The uncertainty of the camera positions is supported by the error margin of the mounted GPS in use. In contrast the 5 mm for the GCP uncertainty. Such an unrealistic low value is set in order to force the program to pin the point clouds to these positions and weighted them more than the camera position form the GPS.
In reality the uncertainty of the GCPs lies probably in the meters and is therefore of the same magnitude as the camera GPS-coordinates. This estimation is based on the calculation that the width of one pixel on the camera sensor depicting the northern ridge, at a distance of 3.5 km to 4 km, represents approximately 1.12 m to 1.27 m of displayed scenery. To contain internal integrity, the very same GCPs are used every time. The coordinates of the GCPs are therefore left unchanged. An averaged of less than 3x3 pixel offset accuracy is estimated by manually pinning the GCPs to a pixel in the photographs. Be aware that here only vertical structures and distances are mentioned as the effective inaccuracy would have to be projected onto the slope angle.
By stacking these resulting DEMs into an assembled DEM the RMSE of the projection is estimated to be < 162m. This value is for the total count of projected key points. The error of the scenery of interest is very likely lower. Additionally, the DEM was rescaled to a resolution of 10m.
Basic statistics of the two DEMs used further for projection, can be extracted from Table 5.

3.3 Template Matching and Georectification (3D-Velocity)

The following subchapter deals with the calculation of 3D-Velocity using oblique photography and the full capacity of the *ImGRAFT* toolbox. A DEM is used, in order to successfully project a point from the two dimensional image plane into three dimensional world coordinates. In the first section the results of a sample image pair are presented, covering the spatial dimension of the calculations. The temporal dimension is covered in section three, showing time series. Here, the data is shown twice as two different DEMs are used. Finally, the uncertainties are discussed.

The structure of this chapter is similar to Chapter 2.2. And as the results are similar, some repetition can not be avoided. Note that presented data, while based on the same imagery as used for the prior method, are recalculated from scratch. This is due to the method of *ImGRAFT*.

3.3.1 Image Pair Sample

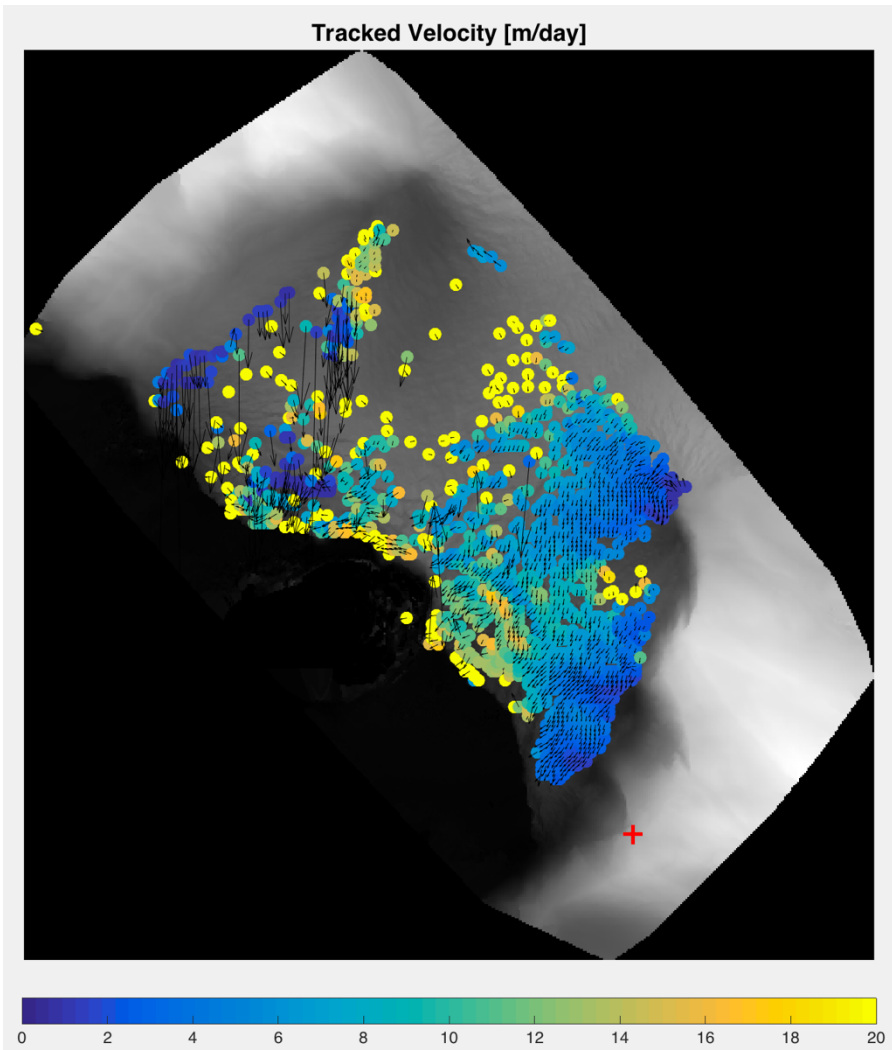
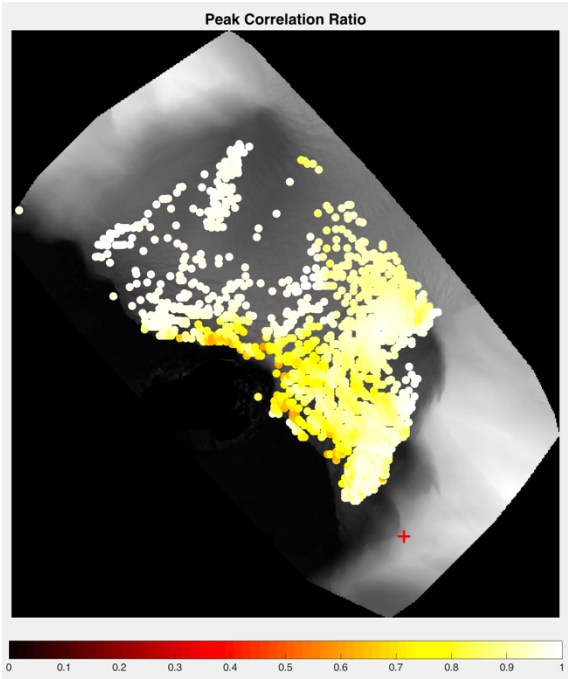
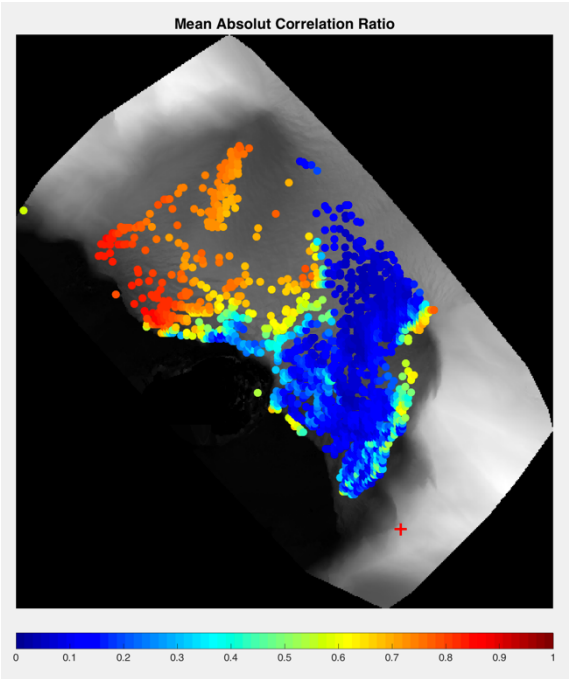


Fig. 23 shows tracked 3D- Velocity in [m/day]. Colours correspond with velocity (see colour bar) in meters per day. Thin black arrows are showing the direction of the movement. A red cross is indicating the camera position. The Drone DEM is underlaid in grey for orientation.

A sample image pair is presented, to give the reader an idea of the possible results from template matching and georectification using ImGRAFT. For consistency, the same image pair as in Chapter 3.1.1 is displayed (images L114685 and L114686, captured by the Upper TLC on the 10th and 11th of June 2016 around noon). For this example, the drone DEM is used for projection.
Fig. 23 shows the glaciers three dimensional velocity (colour) and its vectors (arrows) in map view. The three graphs in Fig. 24 enable a quality assessment of the data.





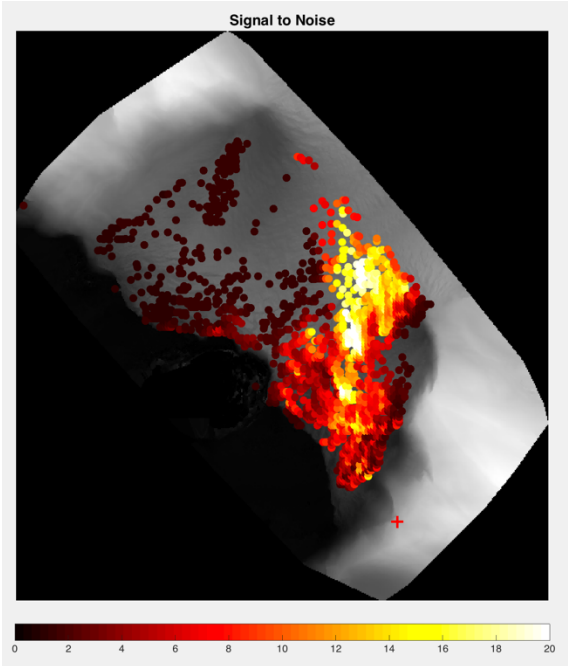


Fig. 24 shows three plots for quality assessment of the tracked points in Fig. 23, on top of the grey shaded DEM. Successful matches are indicated as coloured dots. The red cross indicates the camera position. Top left: Peak correlation ratio (Signal) of the normalised cross correlation for the best match between the template and the search window (candidate). Top right: Mean absolute correlation ratio (noise) describing how alike the candidate is to the remaining possible matches. Bottom left: Signal to noise ratio as indication of the qualitatively high matches.

In total 1401 matches were found, of which 1260 pass the filtering of a SNR > 1 and Signal > 0.6. As is visible in the figures above, the majority of the matches are on the southern half of the outlet glacier's width. The high-quality matches, with high SNR, are mainly located to the centre of the outlet glacier. Neither the ice margin nor the calving front stands out with high SNR ratios.
The average camera shake detected is -1.68 in u and +1.51 in v direction, with and standard deviation of 0.32 pixel and 0.98 pixel and a slight, counter clockwise rotation.

3.3.2 Time Series

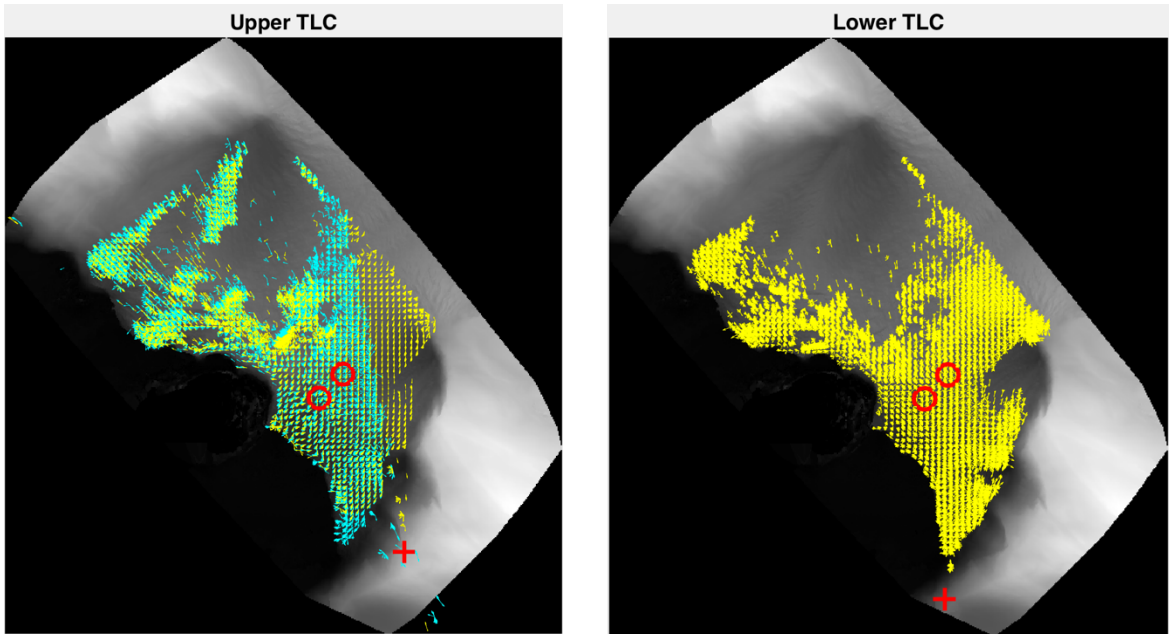


Fig. 25 shows the data availability of both TLC. In red the positions of the boxes (circles) where box1 is further south-east at 530700/ 7743100 and box 2, further north-east at 530900/ 7743300. The cross marks the camera Positions. Upper TLC (left) shows the data prior to the repositioning (cyan) and after (yellow). The position remaining for the Lower TLC (right) throughout the entire survey.

Over time the ice flow is tracked on two positions. This time the boxes are averaging windows within a 75 m radius of the centre point of the boxes (Fig. 25, red circles). This is applicable as now coordinates are available. Nine values are returned, given a mesh with a spacing of 30 m. Similar filtering criteria are used as in the prior subchapter:
A) A threshold of velocity < 20 [m/day].
B) Only displacements with SNR > 1 and a peak correlation ration > 0.6.
C) Only displacement within 2σ deviation (twice the standard deviation), of the day's mean.
D) No zero values are allowed. It is assumed that there is always motion.

An overview of the spatial and temporal distribution of the data available can be gained from Fig. 25 and Table 6. Note that in Table 6, column one and two are the same as Table 2 (page 38), as the same input data is used.

Table 6: Summary of data availability for both the upper and the lower TLC time series while displacement calculations. Showing: The retrieved image from the cameras, the images with / or successfully assigned to a point in time, image pairs leading to a

successful displacement calculation, data points available in the time line plot and the time span covered by the plots for the upper and the lower camera.

	Images collected from cameras	Images w/ time (image pairs)	Image pairs calculated	Data points remaining after filtering*	Time period of successful calculations*
Upper TLC	219	91 (90)	63	46	2016-03-04 2016-08-20
Lower TLC	300	300 (299)	234	158	2015-11-04 2016-09-03

* averaged value of the two boxes, as depending on individual template windows.

3.3.2.1 Time Series Using the Drone DEM

Fig. 26 shows the resulting 3D-Velocity of boxes 1 and 2 over time for both cameras, using the drone derived DEM (see Fig. 37, upper right). Note that only mean values are shown, for legibility. For more insight on the individual values, see Fig. 27 for the upper TLC, and Fig. 28 for the Lower TLC.

Again, it is evident, that box 1 shows higher velocities than box 2. Also, both boxes are showing more or less the same pattern: End of 2015 and beginning of 2016 a decrease in velocity for box 1 and a stagnation in velocity for box 2 is evident. Around May both data series show an increase, peaking in June/July, followed by a deceleration towards August with an additional peak at the end of the month.

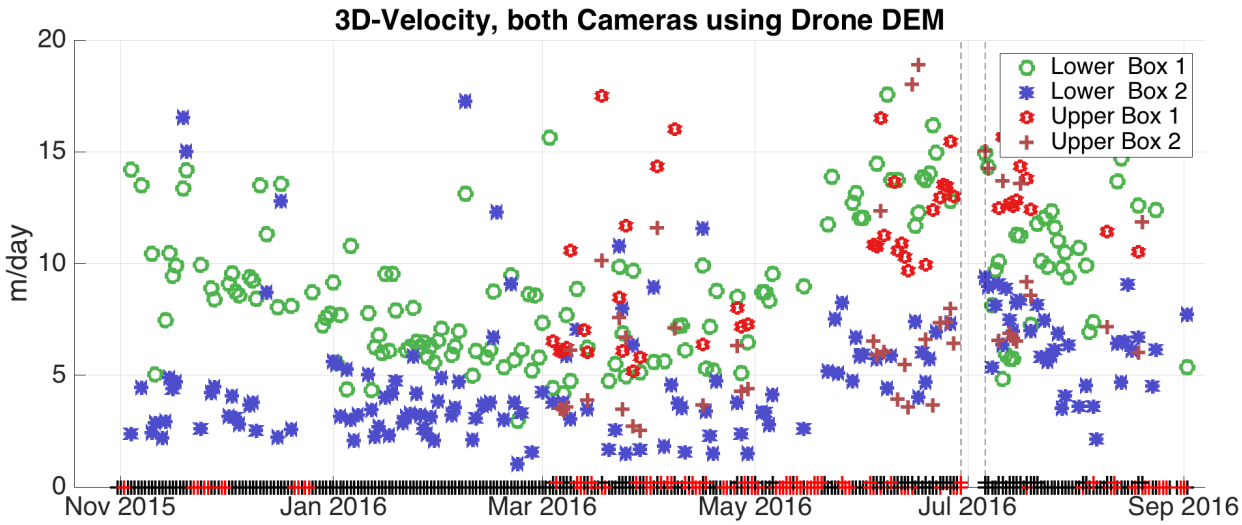


Fig. 26 Tracked 3D-Velocity means of both boxes in [meter per day] over time for the upper (red hexagon and brown cross) and the lower TLC (green circle and blue asterisk). Black-dashed lines are indicating repositioning of the upper camera. The bottom line (around zero at X-axis) shows the image availability (red crosses). Images successfully used for template matching are marked in black. The upper row corresponds to the upper TLC and the lower row for the lower TLC.

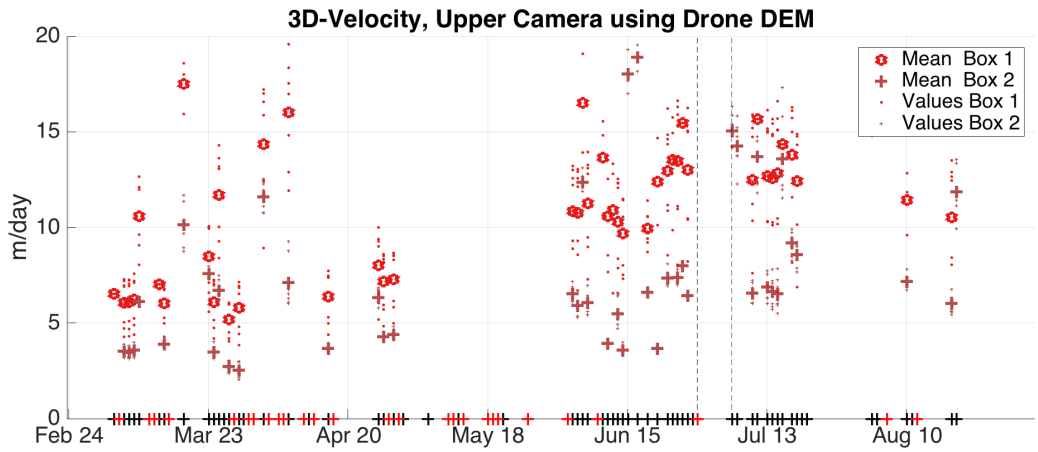


Fig. 27 Tracked 3D-Velocity of both boxes in [meter per day] over time for the upper TLC. The mean values are displayed in large symbols where as the individual data points are plotted as minor symbols (red hexagon and brown cross). Black-dashed lines indicate repositioning of the camera. The bottom line (around zero at X-axis) shows the image availability (red crosses). Images successfully used for template matching are marked in black.

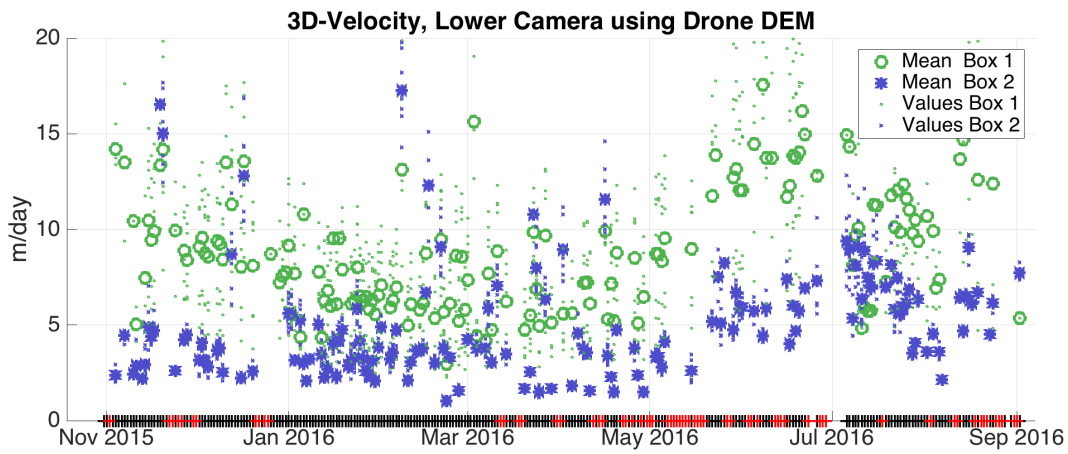


Fig. 28 Tracked 3D-Velocity of both boxes in [meter / days] over time for the lower TLC. The mean values are displayed in large symbols where as the individual data points are plotted as minor symbols (green circle and blue asterisk). The bottom line (around zero at X-axis) shows the image availability (red crosses). Images successfully used for template matching are marked in black.

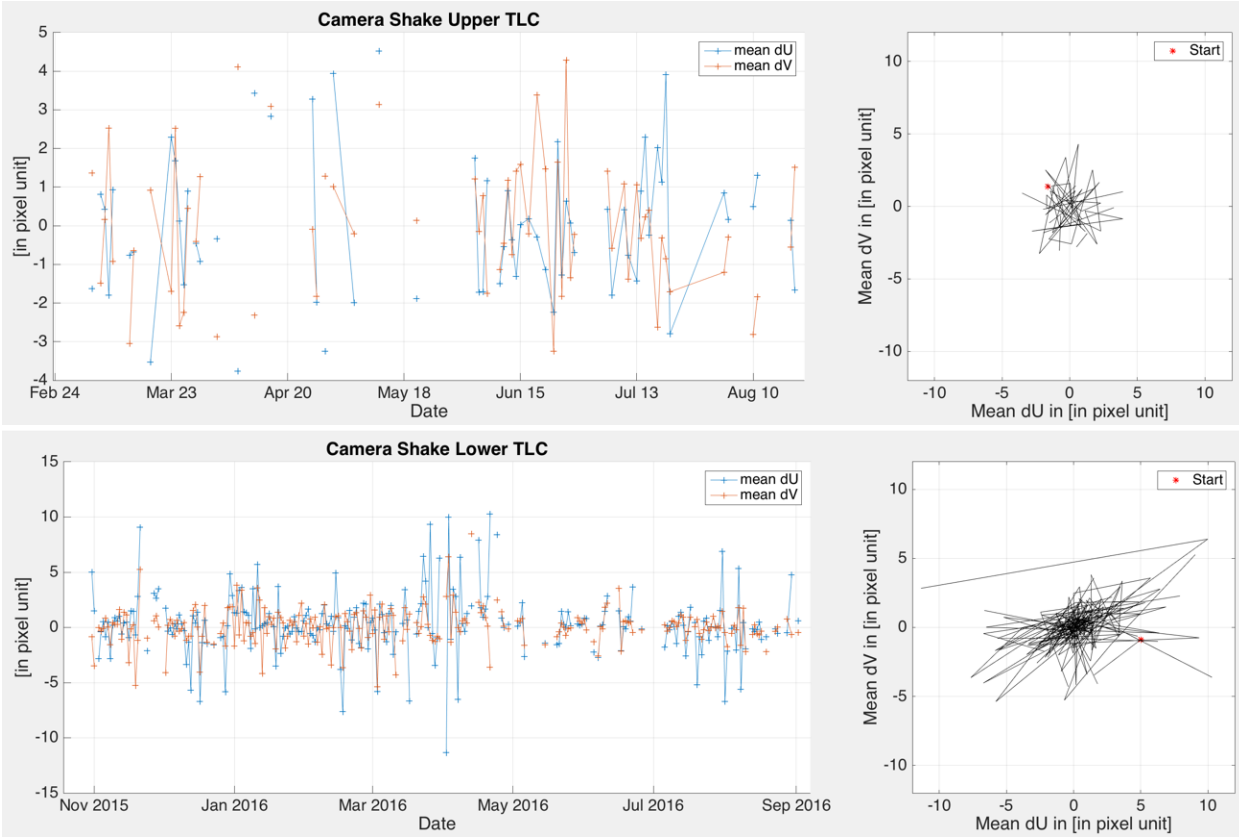


Fig. 29 shows the mean camera shake in dU and dV direction over time (left) and mean displacement in dV over dU (right) for the Upper (above) and lower (below) TLC.

Fig. 29 shows the detected camera shake for both TLC's over time. Remember, here the initial offset is only used to cull image pairs which are far off or too similar, as with completely black images. The camera shake is detected by running the projection and reprojection process on a rock face. The difference in the camera models are then accounted for. Thus, the velocity calculations are already corrected for camera shake. The mean signal to noise ratio is displayed in Fig. 30. By comparing, it can be said, that high camera shake values are not obviously linked to the SNR curve, indicating a qualitatively high camera shake correction.

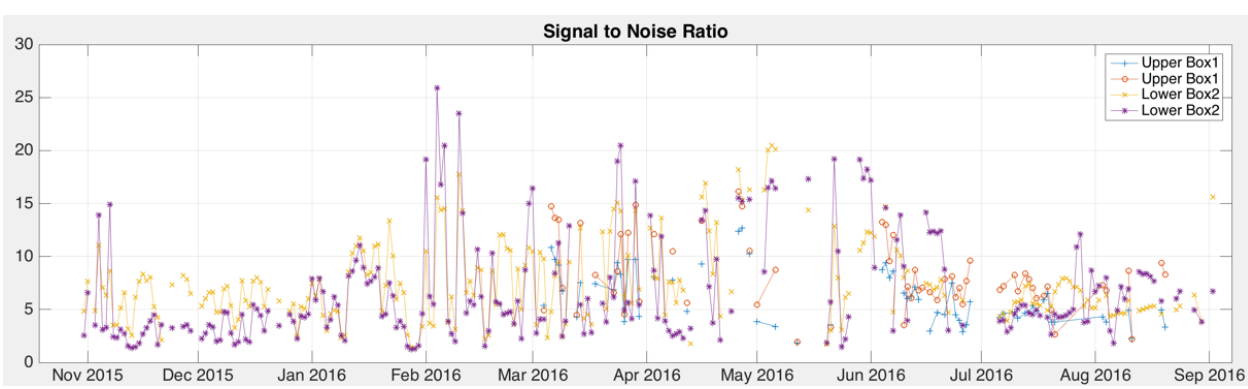


Fig. 30 shows the signal to noise ratio for both boxes and both cameras over the corresponding timeline.

3.3.2.2 Time Series Using the SfM DEM

Fig. 31 shows more single high values prior to the month of May for the lower camera. In fact, twenty-seven readings are exceeding 10 m/day in the months November and May. The same threshold is surpassed by only three readings for the rest of the observation period, all detected in box 1. When consulting Fig. 33 it can be seen, that every one of this three values are containing one reading only (indicated as dot in the middle of the symbol). This could lead to the interpretation of outlier, also questioning the values of the twenty-seven readings before.
Fig. 32 shows many alternation in magnitude between the months of June and July, with velocity changes of more than 4m/day.

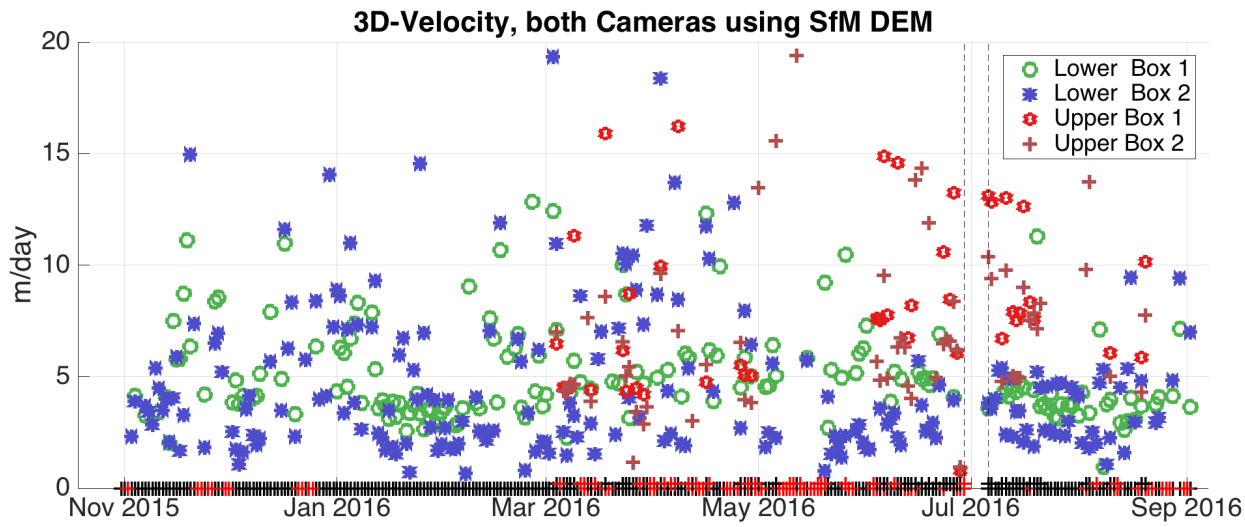


Fig. 31 Tracked 3D-Velocity means of both boxes in [meter per day] over time for the upper (red hexagon and brown cross) and the lower TLC (green circle and blue asterisk). Black-dashed lines are indicating repositioning of the upper camera. The bottom line (around zero at X-axis) shows the image availability (red crosses). Images successfully used for template matching are marked in black. The upper row corresponds to the upper TLC and the lower row for the lower TLC.

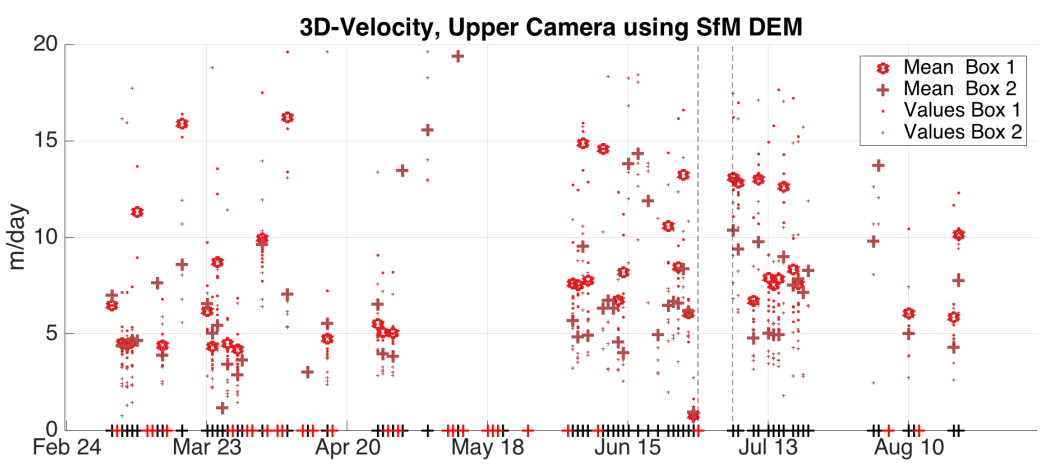


Fig. 32 Tracked 3D-Velocity of both boxes in [meter per day] over time for the upper TLC. The mean values are displayed in large symbols where as the individual data points are plotted as minor symbols (red hexagon brown cross). Black-dashed lines indicate repositioning of the camera. The bottom line (around zero at X-axis) shows the image availability (red crosses). Images successfully used for template matching are marked in black.

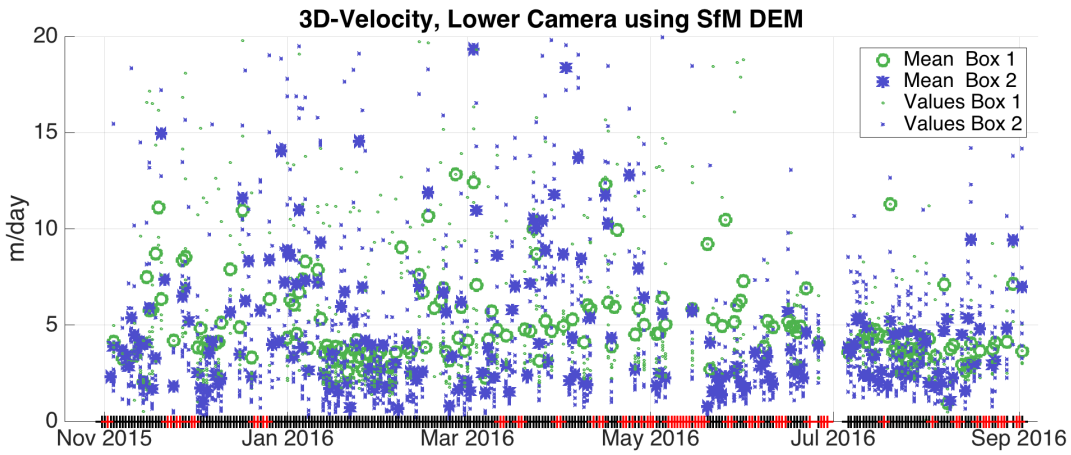


Fig. 33 Tracked 3D-Velocity of both boxes in [meter / days] over time for the lower TLC. The mean values are displayed in large symbols where as the individual data points are plotted as minor symbols (green circle and blue asterisk). The bottom line (around zero at X-axis) shows the image availability (red crosses). Images successfully used for template matching are marked in black.

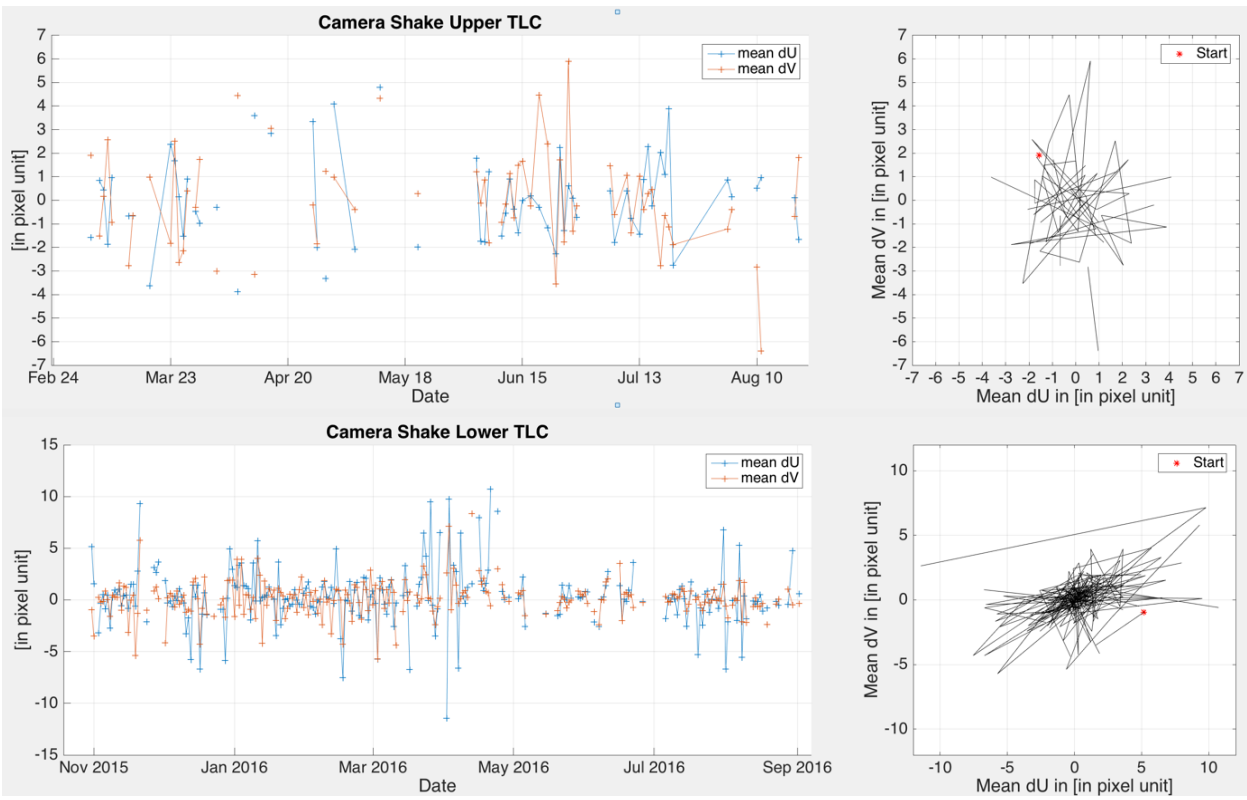


Fig. 34 shows the mean camera shake in dU and dV direction over time (left) and mean displacement in dV over dU (right) for the upper (above) and lower (below) TLC.

Fig. 34 shows considerably higher camera motion for the upper than for the lower camera. The algorithms are programmed to account for such movement, however the potential source of error is increased with the higher camera motion.

Remarkable is the relative low but consistent level of SNR for Box 2 of the lower camera, getting declining time (Fig. 35).

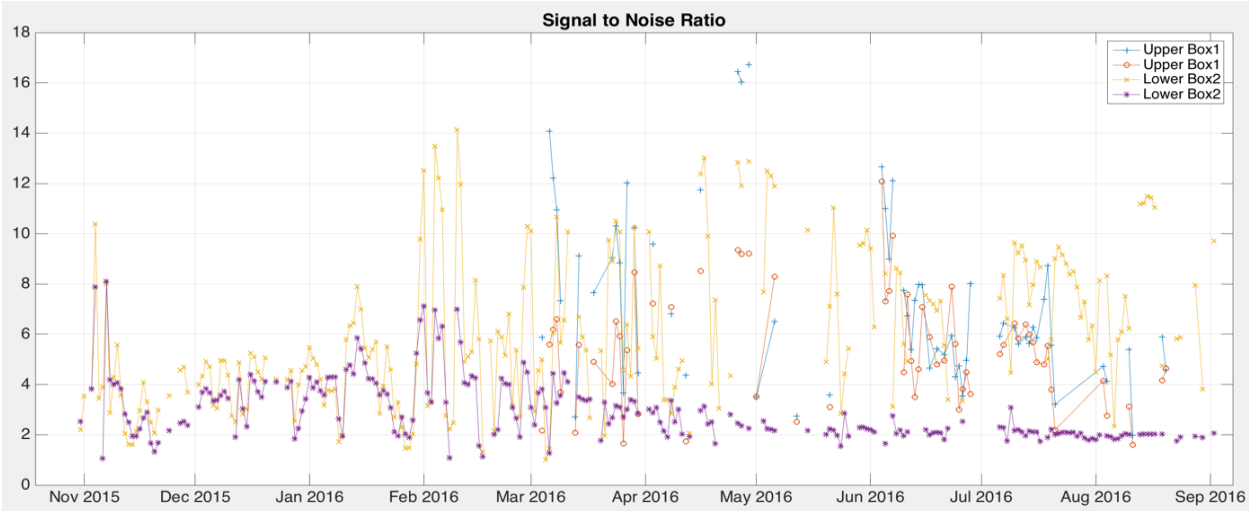


Fig. 35 shows the signal to noise ratio for both boxes and both cameras over the corresponding timeline

3.3.3 Sources of Uncertainties

Possible uncertainties and sources of error are described below. As the same template matching technique is used here, some uncertainties are consistent as in the earlier chapter (3.1.3) described. However, with the third dimension available by projecting into world coordinates, the main focus lies here on different uncertainties occurring when projecting. It can be noted, that the initial projection error, determined by reprojecting the GCPs, is remarkably larger from the lower camera than from the upper camera, despite which DEM was used (Table 7).

Table 7 shows the root-mean-squared projection error (RMSE) and the Akaike information criterion (AIC) of the ground control points in [pixels] for the corresponding Camera and its orientation for both DEMs.

RMSE (AIC) [pixel]	Upper Camera Prior to Turn	Upper Camera After Turn	Lower Camera
Drone DEM	6.9 (63)	6.5 (61)	29.5 (249)
SfM DEM	7.4 (79)	11.7 (86)	36.8 (201)

A) False Positives

Again, the size of search and template windows are optimized for a time step of roughly 1 day. The larger the time gaps between an image pair, the more likely is it to create false positives. Yet, false positives can also occur in different situations and can lead to outlier. Also changes in illumination are revealed as a complication. This can be opposed by extensive filtering. Here, the filtering is intentionally moderate in order to give the reader the possibility to evaluate for themselves.

B) Changing Elevation

Glacier surfaces change over time. Having the DEM from the end of the surveyed time period is suboptimal. It can be assumed that the uncertainty increases with increasing time between the matched image pair and the lapsed time. This is one reason why smoothed DEM's are used. Potential ice thickening in winter would lead to a distance shortening from the camera's point of

view. This could lead to a potential overcatch during the winter moths, if the topography model remain unchanged.

C) Ground Control Points

For projection and reprojection to work an image point to world coordinate relation has to be known. Therefore, few GCP's are used. If the initial points are inaccurate, the entire projection can be distorted. Not having the possibility to spread such points evenly over the surface of observation area, increases the remaining uncertainty. More over, if these points are not accessible and their coordinates have to be read out by visual comparison of photograph and orthophoto.

D) Camera Position

As the camera was recorded using handheld GPS only, a horizontal uncertainty of +/-10m is possible. The vertical uncertainty is not determined by the provider, but likely to be larger. For projection purposes a precise camera position in all three dimensions is crucial. Abundance can lead to distortion of the projection model.

3.4 Further Observations

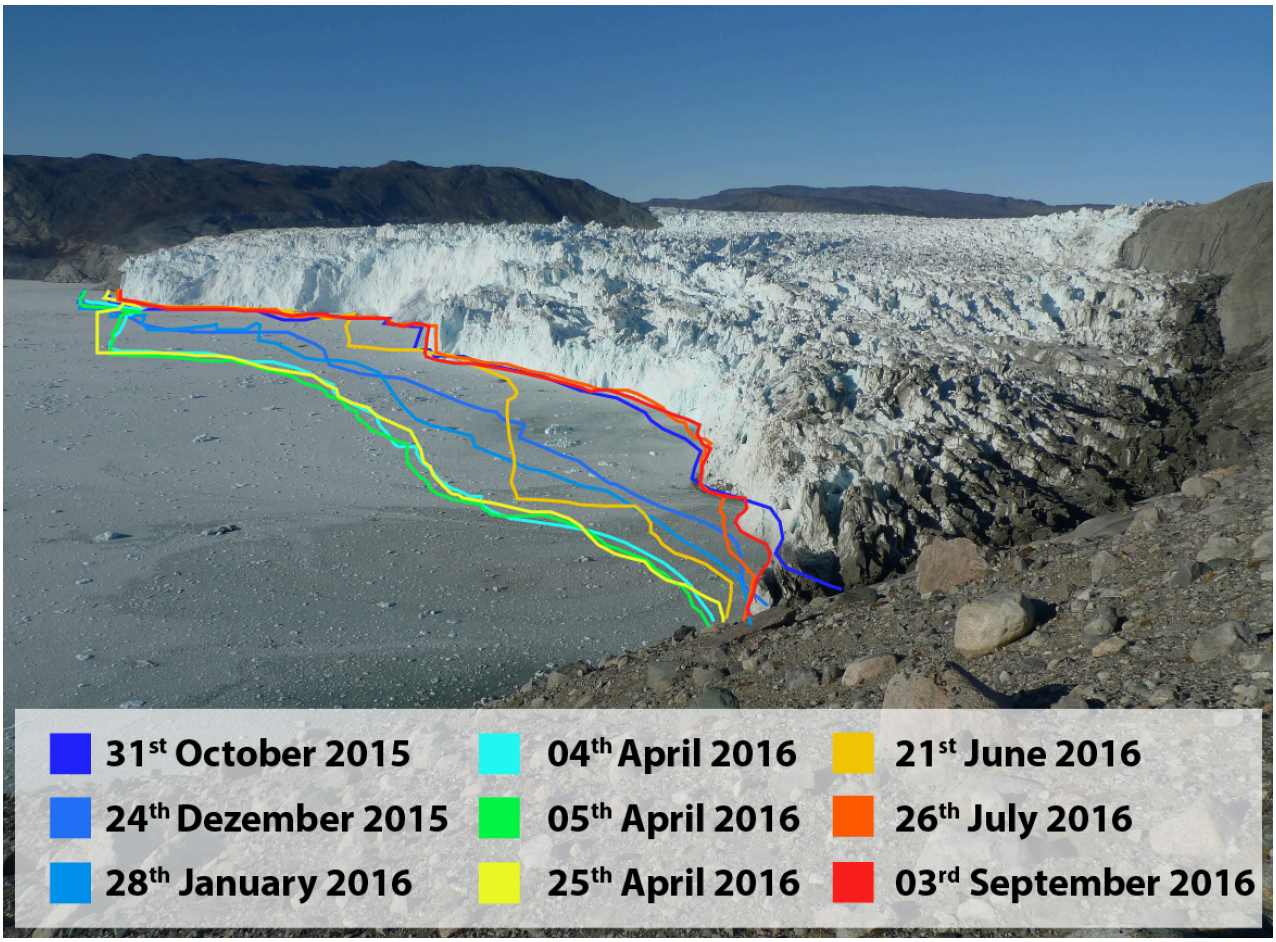


Fig. 36 shows the manually tracked the contours of the glacier terminus at different points in time (see legend insight) from the view of the lower time-lapse camera. The image showed is from the 3rd of September 2016. For the individual pictures including a brief description see Appendix (page 95f.).

An interesting observation beyond the scope of the motion detection, was made by visual inspection of the dataset from the lower time-lapse camera. By comparing the contours of the calving front over time, insight on the dates of advance and retreat could be obtained (Fig. 36): The terminus advances from the beginning of the series until the 4th of April 2016. One day later the calving front starts to retreat. On this very day the fjord is free of ice mélange for the first time since November 2015. In the month between the 26th of July and the 3rd of September the front seemingly remained on the same position, possibly on a position supported by surfaced bedrock (see Fig. 3 on page 13). At the end of the dataset the frontal position of the terminus is about on the same location as at the beginning of the dataset. Be aware that the contour visible in the images are drawn and not the actual waterline tracked. For the individual images see Appendix (page 95f.).

(blank page)

4 Comparison

4.1 Comparison of the DEMs

In Fig. 37 the DEM of the Polar Geospatial Centre of the University of Minnesota, called *Arctic DEM* (top left) as well as the *drone* derived *DEM* of the University of Zurich (top right) are displayed. For reasons of comparison the *Arctic DEM* (middle) and the *Drone DEM* (bottom) are subtracted from the *Assembled DEM.* The *Arctic DEM* is a mashup of satellite data from 2011 to 2015 and is only illustrated as a second guidance. The drone DEM was collected on 21st August 2016 and is used successfully as high resolution DEM for the template matching and georectification purposes in the chapter 3.3.2 (page 51).

All DEMs are sampled to a 10 m pixel resolution. Nevertheless, the drone derived DEM shows a more detailed picture, visible for example in the highly crevassed areas. The *Assembled DEM* seemingly undercatches the western part of the calving as well as parts of the southern ridge. Also the southern part of the calving front is displayed too low in the *Assembled DEM*. However, the differences are in the same magnitude: Where the *Arctic DEM* is representing this part up to 40 meter higher in elevation, the *Drone DEM* represents it in average only 2 m higher. Comparing the plateau in the north, the situation is vice-versa: The Arctic DEM shows it around -5 m lower where as the *Drone DEM* up 40 meters higher. This leads to the impression as if the *SfM DEM*s are tilted along a North- Northwest to south-Southeast axis.

The southern plateau is in both comparisons positive with much less undercatch by the SfM DEM. The slope itself, however, has an offset in Z-axis by 20m and 90m, respectively.
In general, it can be stated that the variations are inhomogeneous but for the western part of the calving front. The amplitudes, however, seem to be less with the *Drone DEM* than the *Arctic DEM*. This effect was intended as – in terms of glacier surface – of the *Drone DEM* is more likely to be up to date. Thus the *Assembled DEM* was lifted in order to fit better to the *Drone DEM*.

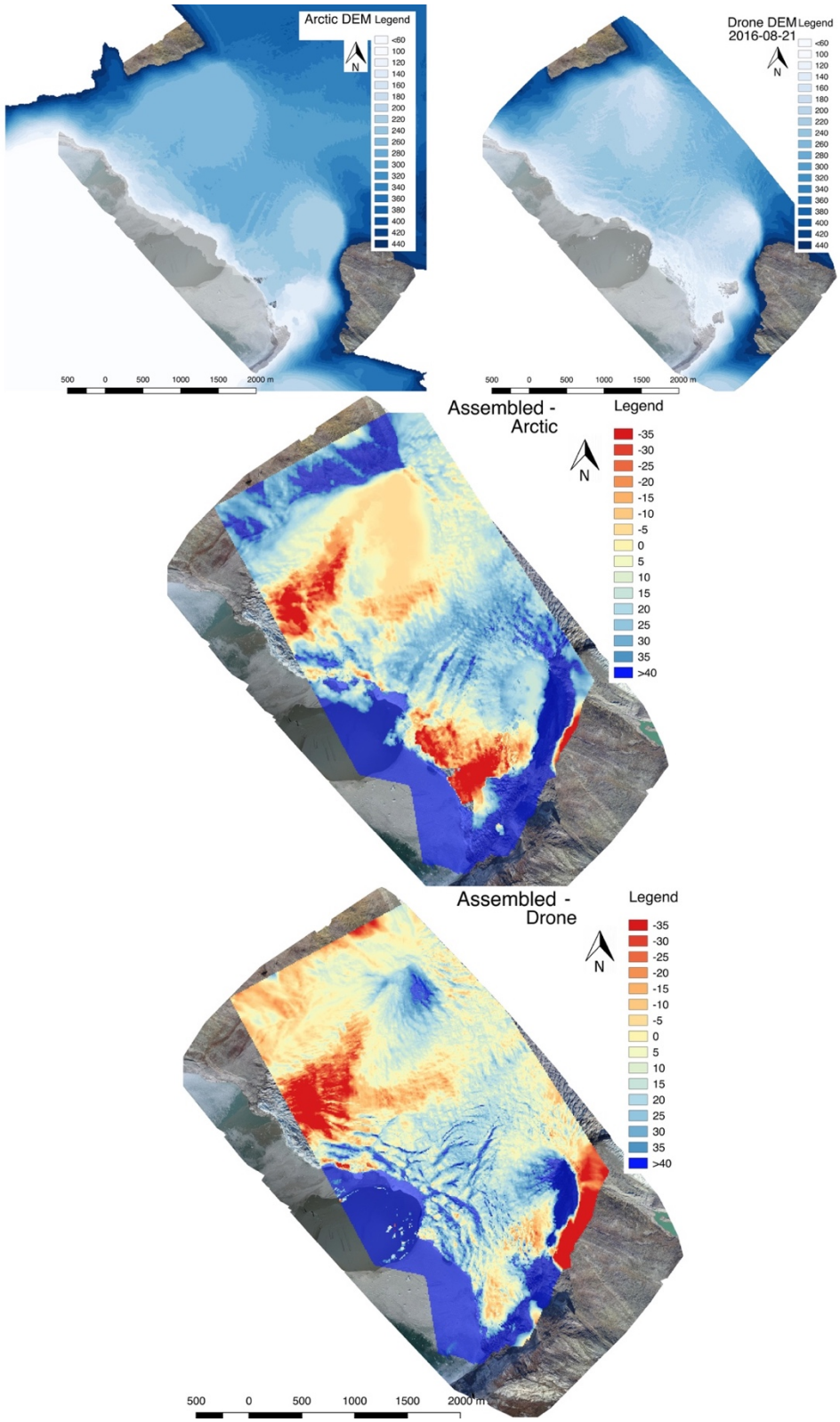


Fig. 37 Representation of the two DEMs for comparison: (Top left) Arctic DEM (2011-2015) and (top right) the Drone derived DEM (collected the 21st of August 2016). Below the differential image of the SfM derived Assembled DEM with Arctic DEM (middle) and the Drone DEM (bottom), both at same scale. The legend shows the elevation difference in meters. All DEMs are resampled to a 10m resolution.

4.2 Comparing Flow Patterns

Flow patterns are displayed for both cameras using the drone derived elevation model (Fig. 38) as well as the Structure from Motion DEM (Fig. 39). For visual comparison two complementary velocity calculations are displayed: First, the available satellite derived data since fall 2015 was averaged over 12 days with a spatial resolution of 10 m. Second, the drone data was averaged over two days but was only available twice, in late August 2015. Its spacial resolution is approximately 0.5 m. For the sample purpose, a four-day average of the drone DEM was taken, covering the entire time interval available. Due to the temporal resolution of the complementary calculations, the findings of this thesis are averaged over the same time periods.

Complementary Data

The drone data displays an increasing velocity gradient towards the front and the centre of the glacier. The satellite data does not represent the latter pattern as clearly, especially as patches of higher velocity are placed within the glacier surface and are obstructing the view. The very same patches are unrecognisable in the drone data and therefore less likely to be accurate measurements. However, the satellite data does shows plainly an increase towards the protruding parts.

Cameras

The upper camera shows generally higher velocities than the lower camera. In fact, the upper camera shows higher velocities than all the other presented data. Here, an increase towards the calving front is clearly shown. And velocities decrease towards the southernmost tip of the terminus, pushing on the lateral moraine. Slower velocities are also visible further back, towards the icefall. Nevertheless, the overall velocity represented is presumably 2 to 3 times higher than in the other calculations. Note that the count of image pairs for averaging is low for the upper camera.

The lower camera is showing more consistent velocity patterns. It can be said, that the velocities are ranging over large parts within the velocity range of both the satellite and the drone data. But, unlike those two data series, nearly no gradient is visible. There are fewer spots of higher velocity, about 400m from the calving front. Interestingly these highlight points are show in every single one of the four figures depicted regardless of the underlying DEM. The location of these spots if found to be where the large semicircle-shaped transversal crevasses are, near to the calving front.

DEMs

The results are very similar, whether the *drone DEM* or the *SfM DEM* are used. However, a difference in resolution is evident when the *SfM DEM* and the *Drone DEM* are compared. The *Drone DEM* based data has a spatial resolution of 25 m whereas the *SfM DEM* data is based on a 30 m grid. The positioning of the *SfM DEM* is obviously not perfect. It extends over the outlet

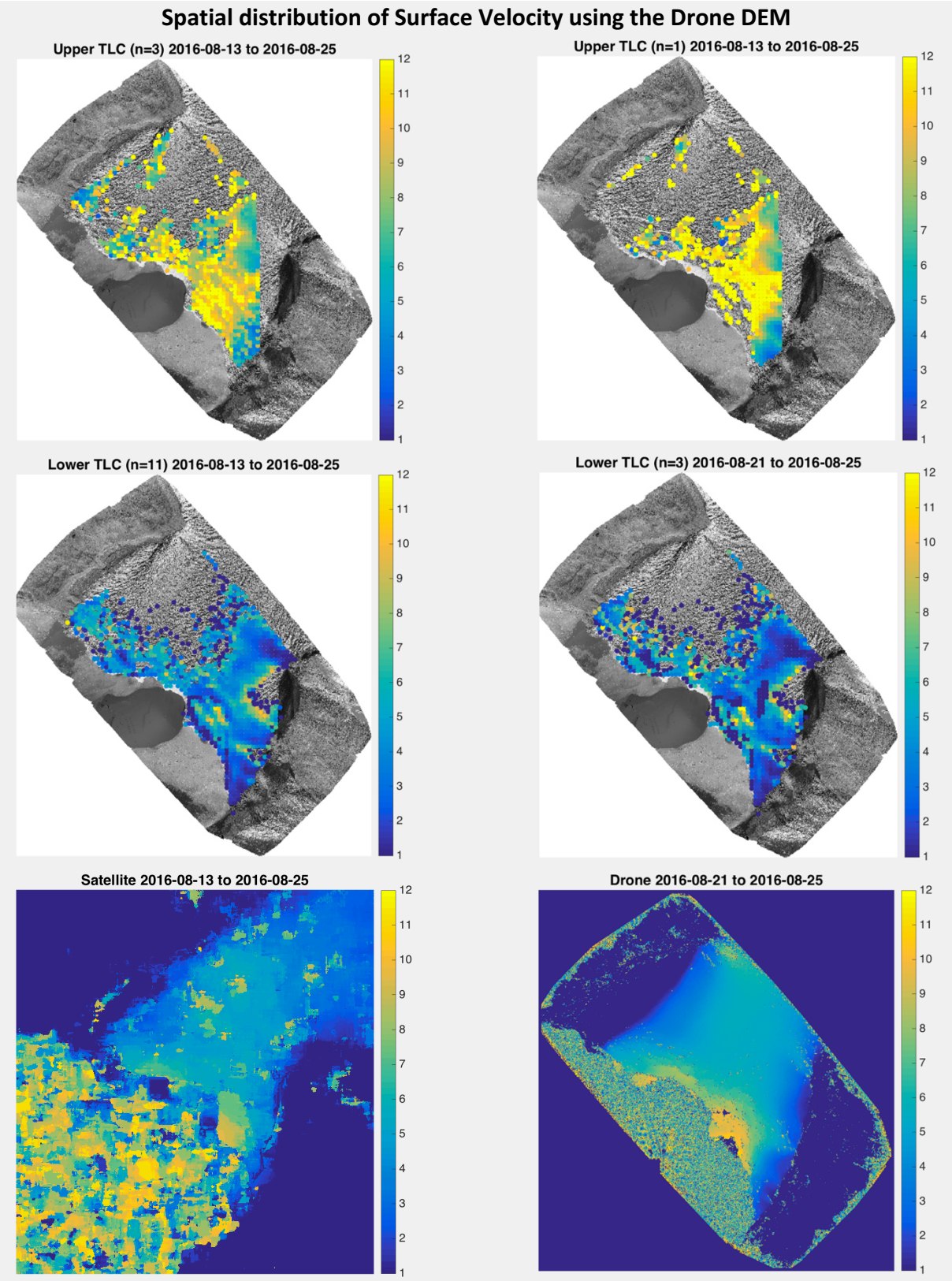


Fig. 38 shows the spatial distribution of velocity calculation using the drone derived DEM for two different time periods in [m/day] for the upper (left) and lower (middle) camera. The time periods are equal to the temporal resolution of the satellite (top right) and the drone imagery (bottom right) derived velocity calculations. The top row shows the data averaged between the 13th and 25th of August 2016. The bottom row shows the data averaged between the 21th and 25th of August 2016. 'n' indicates the number of image pairs available over the time period.

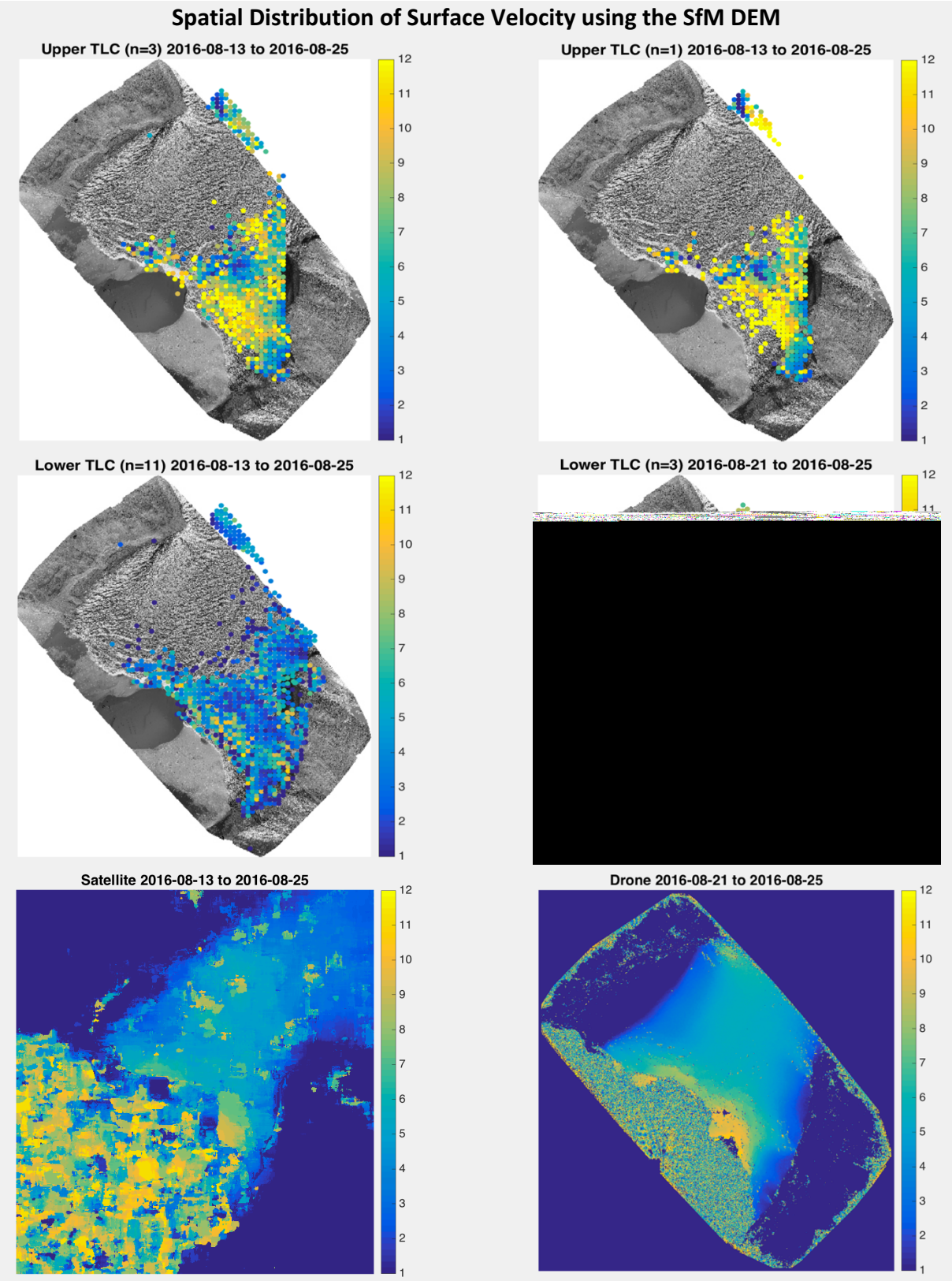


Fig. 39 shows the spatial distribution of velocity calculation using the Structure from Motion derived DEM for two different time periods in [m/day] for the upper (left) and lower (middle) camera. The time periods are equal to the temporal resolution of the satellite (top right) and the drone imagery (bottom right) derived velocity calculations. The top row shows the data averaged between the 13th and 25th of August 2016. The bottom row shows the data averaged between the 21th and 25th of August 2016. 'n' indicates the number of image pairs available over the time period.

4. Reconstruction the Scenery

In this step, the geometry of the scenery is reconstructed. This part was originally described as *Structure-from-Motion* by Ullman in his book *The Interpretation of Visual Motion* (1979). The outstanding aspect of this process is the simultaneous reconstruction of three elements: I) The three-dimensional structure of the scenery, II) the position and orientation of the cameras i.e. the location where the pictures have been taken from and III) the intrinsic camera parameters i.e. focal length, sensor size etc (Westoby *et al.*, 2012).

This is an iterative process and every iteration needs some assumption to be based on. Therefore, an initial picture pair is selected. Most commonly it is the pair with many tracks and a large offset angle. For the 3D structure (I), the tracks are used. For the external camera orientation (II), a *rigid body transformation* is carried out. The initial parameters are derived from the tracks. This is commonly done by using the pinhole camera perspective projection model (see Chapter 2.2.2). The internal camera orientation (III) is defined by 3x3 matrix called *camera calibration/resectioning matrix*. To optimize the parameters a *bundle adjustment* method can be used. The frequency of model fitting errors, defined as values of the cost function are hereby minimized. The result of the iteration process is a sparse point cloud of the scenery. This is the very product many other applications in geoscience are based on (Westoby *et al.*, 2012; Carrivick, Smith and Quiencey, 2016).

5. Dense Point Cloud

A dense point cloud is generated. The original *Structure-for-Motion* (4th step of the process, described) generates with the spare point cloud already a 3D illustration of the scenery. Using *Multi- View Stereo* (MVS) algorithms this illustration can be improved. The resulting *dense point* cloud is likely to have a point density of at least two orders in magnitude (Carrivick, Smith and Quiencey, 2016:77), which adds much more detail to the illustration of the scenery. In general, there are four kinds of MVS; (I) the *voxel based method*, (II) *the surface evolution-based method*, (III) *the depth-map merging method* and (IV) *the patch-based method*. The difference between the methods reach from simple interpolation of the existing sparse point cloud by a grid with limitation of accuracy (I) to a polygonal mesh in need of an initialisation in order to minimize a cost function (II). Latter struggles with large-scale scenes. This issue is resolved by the rather complicated approach of individual *depth maps* calculation for every single picture describing the distance from the viewpoint to the scenery (III). Broadly used, this is the last method which uses matching features to identify blobs and corners on multiple images(IV). Then the especially sparse patches of the point cloud are tackled and filled with additional grid points. Afterwards the last step filters remove possible outlier again.

No matter which approach is pursued, the result is a dense point cloud with a point density similar to that of terrestrial laser scanner (Carrivick, Smith and Quiencey, 2016, p. 77). Note, the fact that *depth filtering* is available is an indication that *PhotoScan Pro* (Agisoft LLC, 2016) uses the *surface evolution-based method* as *Multi- View Stereo* (MVS) method.

It is worth noting, that SfM alone can only lead to an arbitrary reference system, not to scale. Without at least three *Ground Control Points* (GCP) and/or the world coordinates of the cameras positions the georeference in global coordinates can never be achieved. Additionally, GCP can be used as initial parameters as they are set manually and link multiple images. GCPs can therefore increase the accuracy of the point cloud.

Bibliography

Agisoft LLC (2016) 'Agisoft PhotoScan User Manual. Professional Edition, Version 1.2', p. 97. Available at: www.agisoft.ru.

Carrivick, J. L., Smith, M. W. and Quiencey, D. J. (2016) Structure from motion in the geosciences. Chichester: Wiley-Blackwell.

Fusiello, A., Trucco, E. and Verri, A. (2000) 'Compact algorithm for rectification of stereo pairs', Machine Vision and Applications, 12(1), pp. 16–22. doi: 10.1007/s001380050120.

Javernick, L., Brasington, J. and Caruso, B. (2014) 'Modeling the topography of shallow braided rivers using Structure-from-Motion photogrammetry', Geomorphology. Elsevier B.V., 213, pp. 166–182. doi: 10.1016/j.geomorph.2014.01.006.

Lowe, D. G. (1999) 'Object recognition from local scale-invariant features', Proceedings of the Seventh IEEE International Conference on Computer Vision, 2(8), pp. 1150–1157. doi: 10.1109/ICCV.1999.790410.

Lowe, D. G. (2004) 'Distinctive image features from scale-invariant keypoints', International Journal of Computer Vision, 60(2), pp. 91–110. doi: 10.1023/B:VISI.0000029664.99615.94.

Priese, L. (2015) Computer Vision. Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-662-45129-8.

Szeliski, R. (2010) 'Computer Vision: Algorithms and Applications', Computer, 5, p. 832. doi: 10.1007/978-1-84882-935-0.

Ullman, S. (1979) The Interpretation of Visual Motion, The Interpretation of Visual Motion. doi: 10.1016/0013-4694(80)90055-3.

Westoby, M. J., Brasington, J., Glasser, N. F., Hambrey, M. J. and Reynolds, J. M. (2012) '"Structure-from-Motion" photogrammetry: A low-cost, effective tool for geoscience applications', Geomorphology. Elsevier B.V., 179, pp. 300–314. doi: 10.1016/j.geomorph.2012.08.021.

8.2.4 The Principle of Image Georectification

Image rectification can be described as a method to project an oblique image point onto a horizontal reference plane i.e. a world coordinate system, and vice versa (Fusiello, Trucco and Verri, 2000).

The location of a point in the world P (X, Y, Z) is defined by the three coordinates X, Y, and Z. It can be projected or inverse projected onto a point p(u,v) on the camera sensor by solving

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \times \left(R \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + T \right)$$

where K is the intrinsic camera parameters

$$K = \begin{pmatrix} f d_u & 0 & u_0 \\ 0 & -f d_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

with d_u and d_v is the width of a single pixel in the corresponding direction. u_0 and v_0 are the origins of the pixel coordinate system, conveniently, the middle of the image plane. f represents the focal length. R as well as T are the orthonormal rotation and the translation matrix to move from one coordinate system into the other (Szeliski, 2010:27f.).

$$R = \begin{bmatrix} cos\,\theta & -\,sin\,\theta & 0 \\ sin\,\theta & cos\,\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$T = \begin{bmatrix} t_u \\ t_v \\ 0 \end{bmatrix}$$

With the angle θ and the translation in u and v direction.

The parameters in K are initially defined by the camera specifications. By using GCPs with corresponding point in p(u,v) and P(X,Y,Z) both R and T can be solved. This enables solving as long as either camera coordinates or world coordinates are put in.

Bibliography

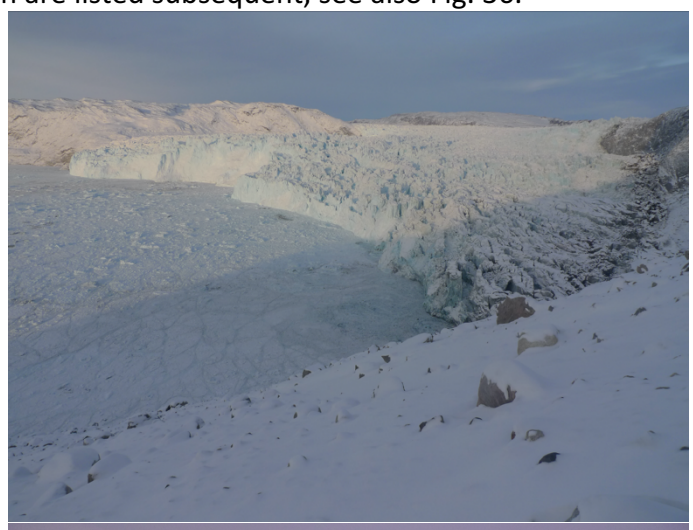
Fusiello, A., Trucco, E. and Verri, A. (2000) 'Compact algorithm for rectification of stereo pairs', Machine Vision and Applications, 12(1), pp. 16–22. doi: 10.1007/s001380050120.

Szeliski, R. (2010) 'Computer Vision : Algorithms and Applications', Computer, 5, p. 832. doi: 10.1007/978-1-84882-935-0.

8.3 Qualitative Image Description

The time series of the lower camera shows an impressive 10 moths archive of the Eqi Sermia's calving front. This data is qualitatively assessed by flicking manually through the time lapse images. The findings of the visual comparison are listed subsequent, see also Fig. 36.

31st of October 2015: Beginning of data series of the lower time laps camera. The fjord is covered in melange, broken up sea ice. The calving front is precipitous. From now on the days are getting notably longer.



24th of December 2015: The outlet glacier slows down in a steady deceleration. The calving front has advanced and changed its profile into a much flatter shape.

A: The satellite based calculations are showing a decrease.*

28th of January 2016: For the first time the sun is visible again on the pictures. The flow is slowly accelerating.

A: The satellite based calculations are levelling out at the seasonal minimum.*

4th of April 2016: For the first time since the data series, the calving front is not any longer advancing. One of the last pictures with snow cover.

5th of April 2016: The fjord is ice free for the first time since the data series. Additionally, a very pronounced calving front is visible.

A: The satellite based calculations are accelerating.*

25th of April 2016: The pinning point starts to collapse over the next nine days. After this event, the calving front is retreating fast in the middle. After some days, the calving rate stabilizes again.

21st of June 2016: Another, less pronounces pinning point collapsed succeeded by a further step-like retreat of the calving front.

A: The satellite based calculations presumably velocity-peak.*

26th of July 2016: The retreat of the middle part seemingly caused the southern wing of the calving front to retreat as well. Now the calving front is forming a curved arch, further back than before. A larger calving event reveals the bright face of newly exposed ice. The shadow of the time laps camera is visible in every picture and is getting longer every day.
A: The satellite based calculations already decreasing.*

3rd of September 2016: End of the data series. Red circles are showing possible positions of surfaced bedrock, supporting the position of the terminus. This image identical to Fig. 2. Red circles are indicating surface bedrock and possible pinning points.

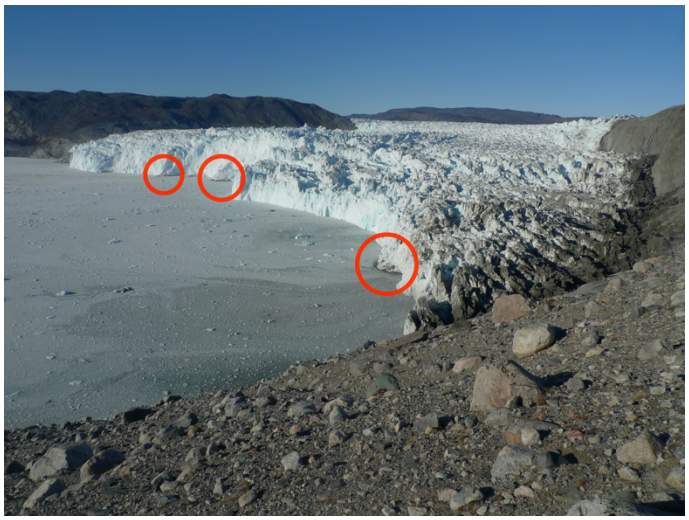
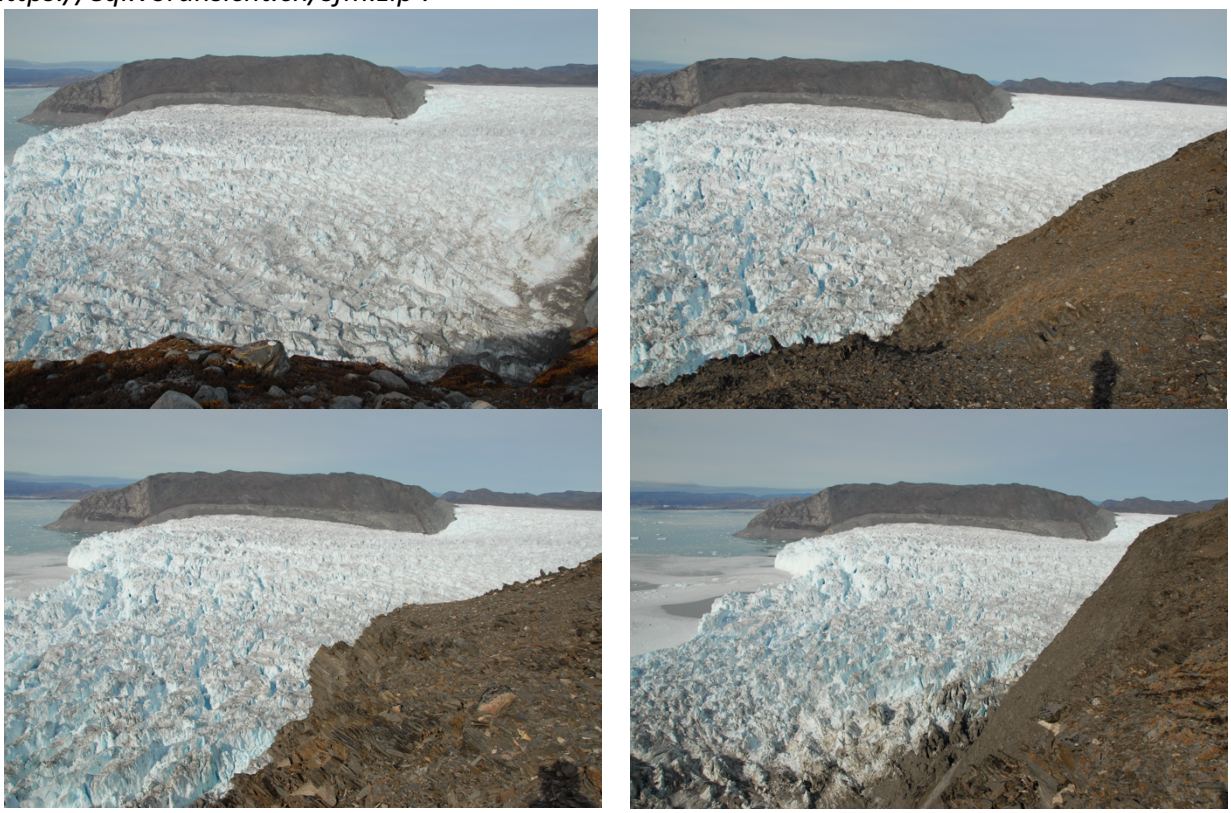


Fig. 43 shows a total of nine sample images from the lower time-lapse camera. A*: are annotation concerning satellite series plots in Fig. 40 Fig. 41 on page 68.

8.3.1 Sample Images for Structure from Motion

Following sample images used for Structure from Motion are presented. All Images have been taken on the 28th of August 2016. The calculations (PhotoScan Pro -files) can be downloaded from *https://eqi.voransicht.ch/sfm.zip* .



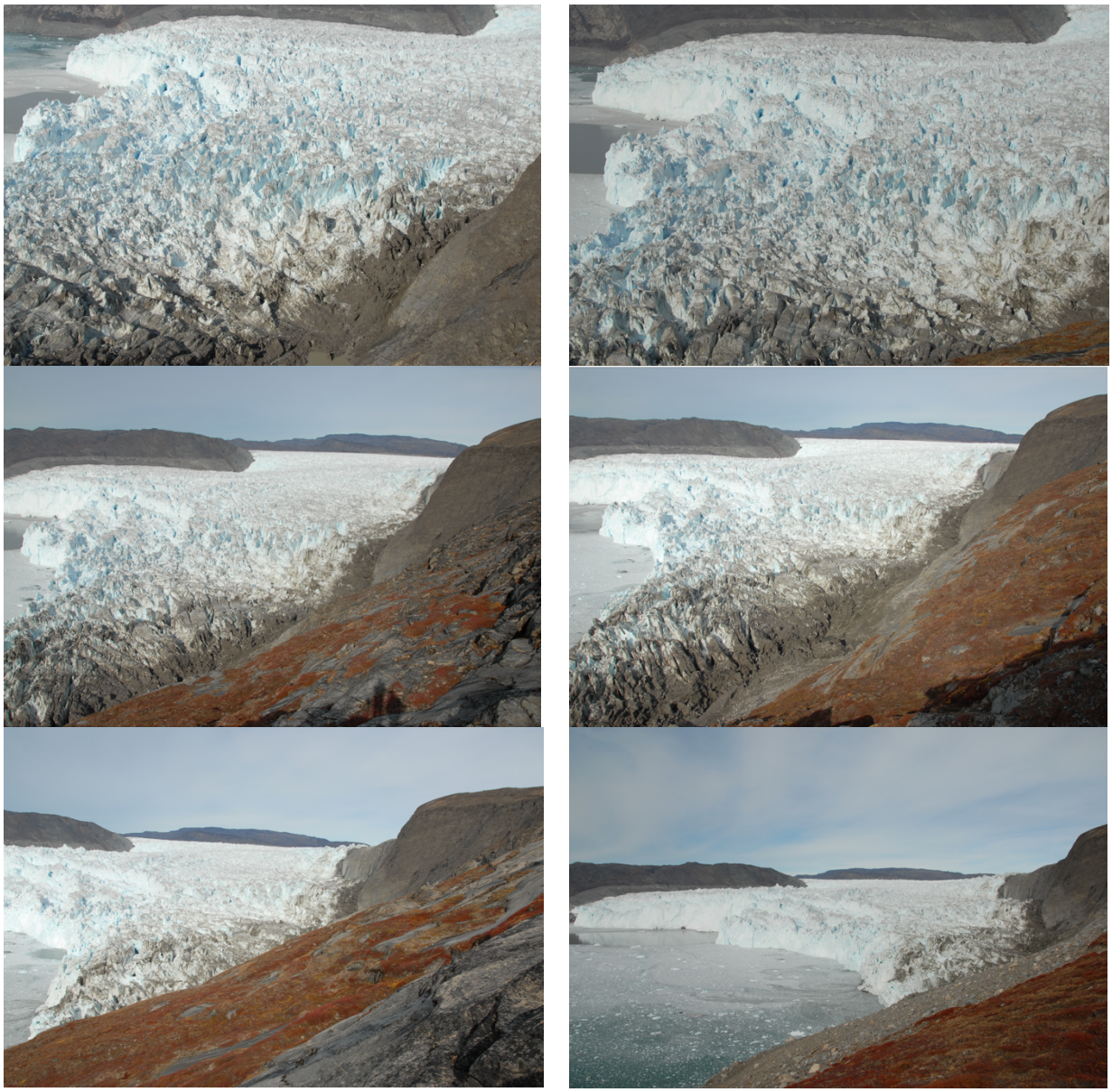


Fig. 44 Sample images of Day 4 (28th of August 2016) used for the Structure from Motion technique.

8.4 Additional Data

8.4.1 Reconstructed Timestamps

Table 9: The timestamps for the upper time-lapse camera had to be reconstructed, as the internal clock failed. By visual comparison with images from the lower camera the possible data was assumed. Weather, snow cover, sea ice/ice mélange and the shape of the calving front was used for orientation. Coloured boxes are indicating obvious jumps in time evident by the glacier movement. Bold numbers are indicating matches with a very high certainty.

Picture Number # Lower Camera P1070###	Data Obtained from Lower Camera	Picture Number # Upper Camera L1140###	Picture Number # Lower Camera P1070###	Data Obtained from Lower Camera	Picture Number # Upper Camera L1140###
126	04 March 2016	**595**	171	18 April 2016	
127	05 March 2016	**596**	172	19 April 2016	
128	06 March 2016	**597**	173	20 April 2016	
129	07 March 2016	**598**	174	21 April 2016	
130	08 March 2016	**599**	175	22 April 2016	
131	09 March 2016	**600**	176	23 April 2016	
132	10 March 2016		177	24 April 2016	
133	11 March 2016	601	178	25 April 2016	
134	12 March 2016	602	**179**	26 April 2016	**627**
135	13 March 2016	**603**	**180**	27 April 2016	**628**
136	14 March 2016	604	**181**	28 April 2016	**629**
137	15 March 2016	605	**182**	29 April 2016	**630**
138	16 March 2016		**183**	30 April 2016	**631**
139	17 March 2016		**184**	01 May 2016	**632**
140	18 March 2016	**606**	185	02 May 2016	
141	19 March 2016		186	03 May 2016	
142	20 March 2016		187	04 May 2016	
143	21 March 2016		188	05 May 2016	
144	22 March 2016		189	06 May 2016	**633**
145	23 March 2016	**607**	190	07 May 2016	
146	24 March 2016	**608**	191	08 May 2016	
147	25 March 2016	**609**	192	09 May 2016	
148	26 March 2016	**610**	193	10 May 2016	
149	27 March 2016	**611**	**194**	11 May 2016	**634**
150	28 March 2016	**612**	**195**	12 May 2016	**635**
151	29 March 2016	**613**	**196**	13 May 2016	**636**
152	30 March 2016	**614**	197	14 May 2016	
153	31 March 2016	**615**	198	15 May 2016	
154	01 April 2016	**616**	199	16 May 2016	
155	02 April 2016	**617**	200	17 May 2016	
156	03 April 2016		**201**	18 May 2016	**637**
157	04 April 2016	**618**	**202**	19 May 2016	**638**
158	05 April 2016	**619**	**203**	20 May 2016	**639**
159	06 April 2016		**204**	21 May 2016	**640**
160	07 April 2016	**620**	205	22 May 2016	
161	08 April 2016	**621**	206	23 May 2016	
162	09 April 2016		207	24 May 2016	
163	10 April 2016		208	25 May 2016	
164	11 April 2016	**622**	**209**	26 May 2016	**641**
165	12 April 2016	**623**	210	27 May 2016	
166	13 April 2016	**624**	211	28 May 2016	
167	14 April 2016		212	29 May 2016	
168	15 April 2016		213	30 May 2016	
169	16 April 2016	**625**	214	31 May 2016	
170	17 April 2016	**626**	215	01 June 2016	

216	02 June 2016		286	23 July 2016	
217	03 June 2016	**642**	287	24 July 2016	
218	04 June 2016	**643**	288	25 July 2016	
219	05 June 2016	**644**	289	26 July 2016	
220	06 June 2016	**645**	290	27 July 2016	
221	07 June 2016	**646**	291	28 July 2016	
222	08 June 2016		292	29 July 2016	
223	09 June 2016	647	293	30 July 2016	
224	10 June 2016	648	294	31 July 2016	
225	11 June 2016	**649**	295	01 August 2016	
226	12 June 2016	650	296	02 August 2016	
227	13 June 2016	651	297	03 August 2016	**695**
228	14 June 2016	**652**	298	04 August 2016	**696**
229	15 June 2016	**653**	299	05 August 2016	
230	16 June 2016		300	06 August 2016	697
231	17 June 2016	654	301	07 August 2016	
232	18 June 2016		302	08 August 2016	
233	19 June 2016	655	303	09 August 2016	
234	20 June 2016		304	10 August 2016	698
235	21 June 2016	656	305	11 August 2016	**699**
236	22 June 2016		306	12 August 2016	700
237	23 June 2016	**657**	307	13 August 2016	
	24 June 2016	658	308	14 August 2016	
	25 June 2016	659	309	15 August 2016	
238	26 June 2016	660	310	16 August 2016	
239	27 June 2016	**661**	311	17 August 2016	
240	28 June 2016	**662**	312	18 August 2016	
243	29 June 2016	664	313	19 August 2016	**701**
DATA GAP	**DUE TO FIELD**	**WORK**	314	20 August 2016	**702**
269	06 July 2016	683	315	21 August 2016	
270	07 July 2016	684	316	22 August 2016	703
271	08 July 2016				
272	09 July 2016				
273	10 July 2016	685			
274	11 July 2016	686			
275	12 July 2016				
276	13 July 2016	687			
277	14 July 2016	**688**			
278	15 July 2016	**689**			
279	16 July 2016	**690**			
280	17 July 2016				
281	18 July 2016	**691**			
282	19 July 2016	**692**			
283	20 July 2016	693			
284	21 July 2016	694			
285	22 July 2016				
Picture Number # Lower Camera P1070###	**Data Obtained from Lower Camera**	**Picture Number # Upper Camera L1140###**	**Picture Number # Lower Camera P1070###**	**Data Obtained from Lower Camera**	**Picture Number # Upper Camera L1140###**

8.4.2 Spatial Patterns Averaged per Month

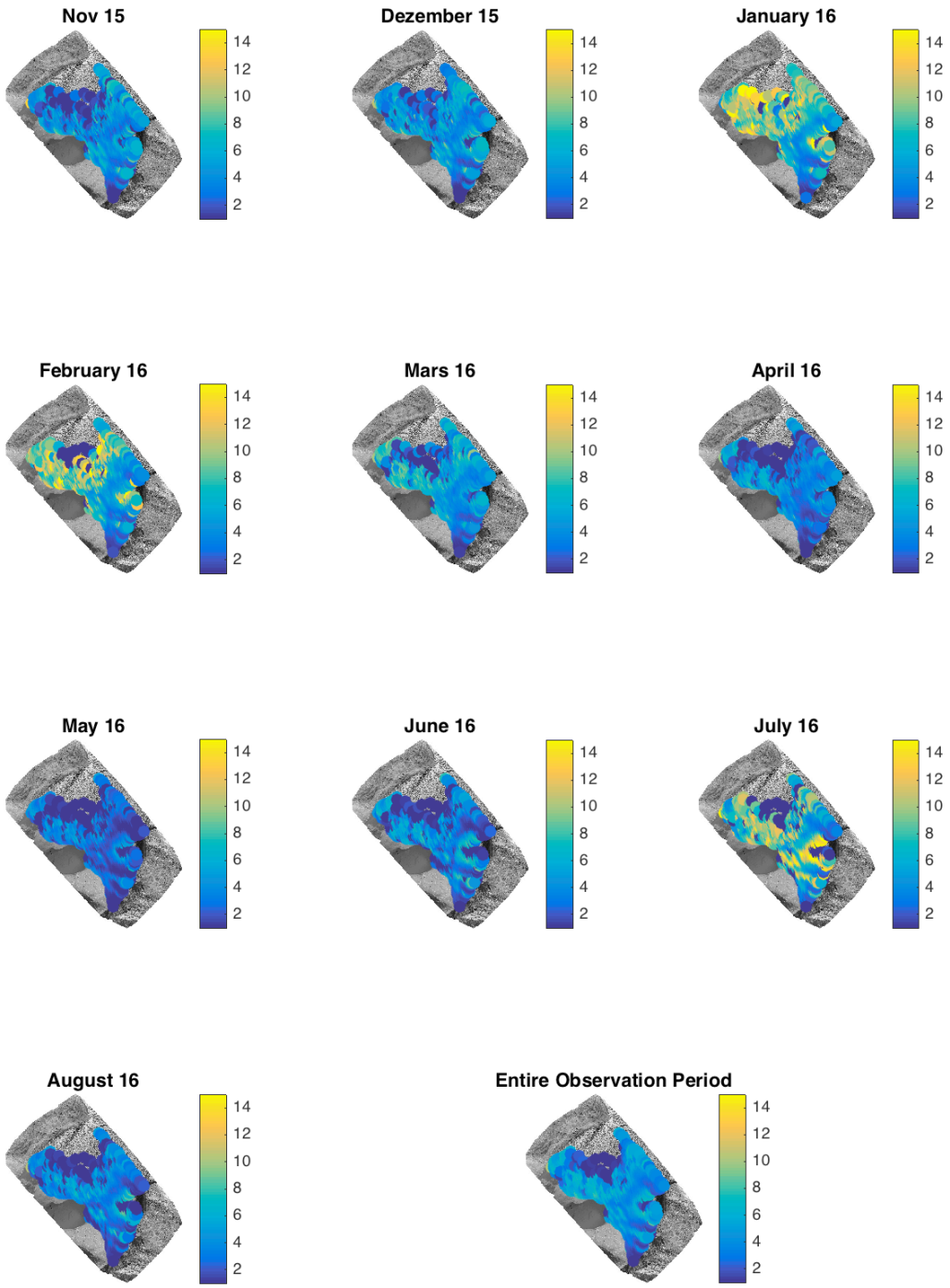


Fig. 45 Monthly average showing spatial patterns for the lower camera using the drone derived DEM.

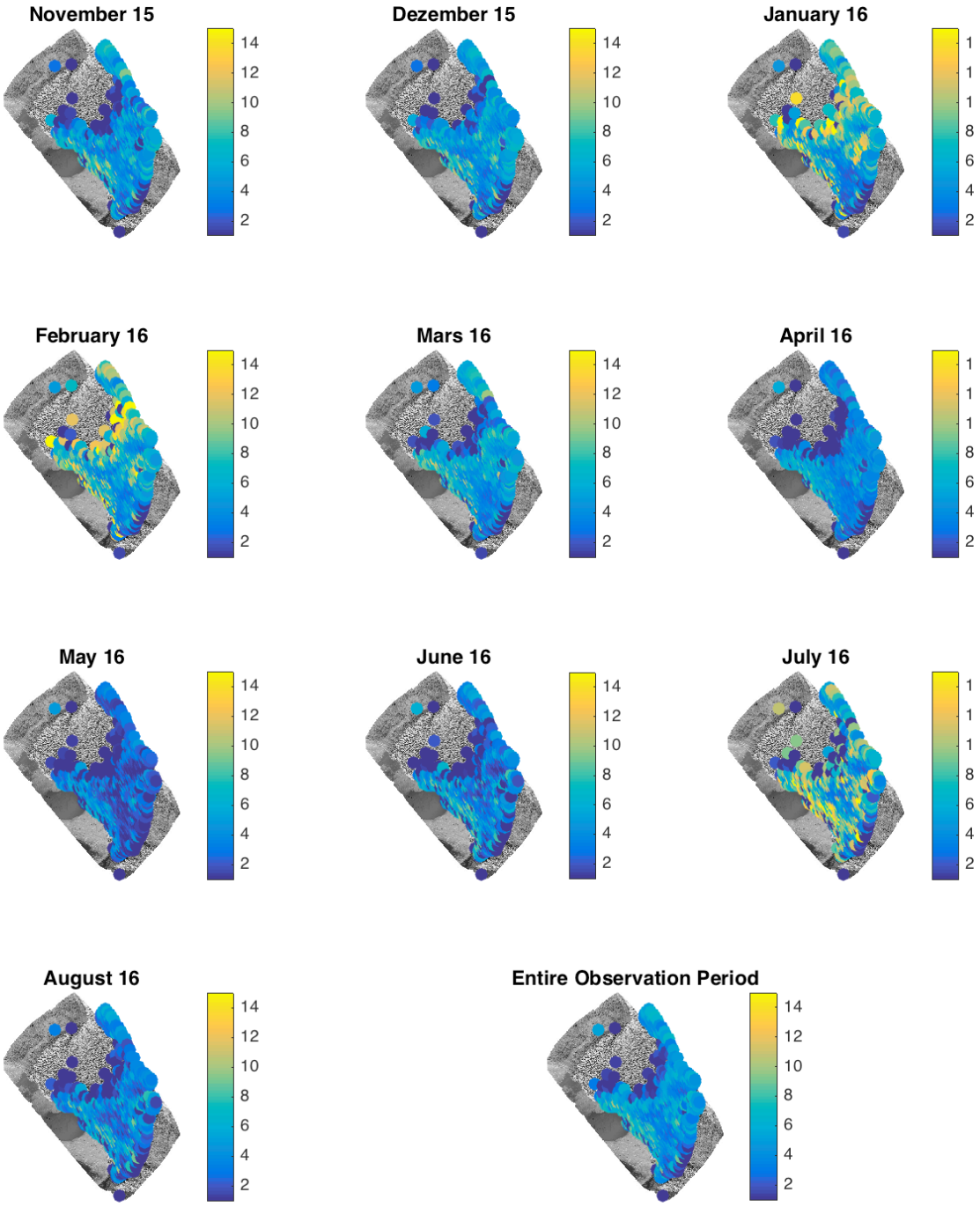


Fig. 46 Monthly average showing spatial patterns for the lower camera using the SfM derived DEM.

8.4.3 Example of Applied Smoothing

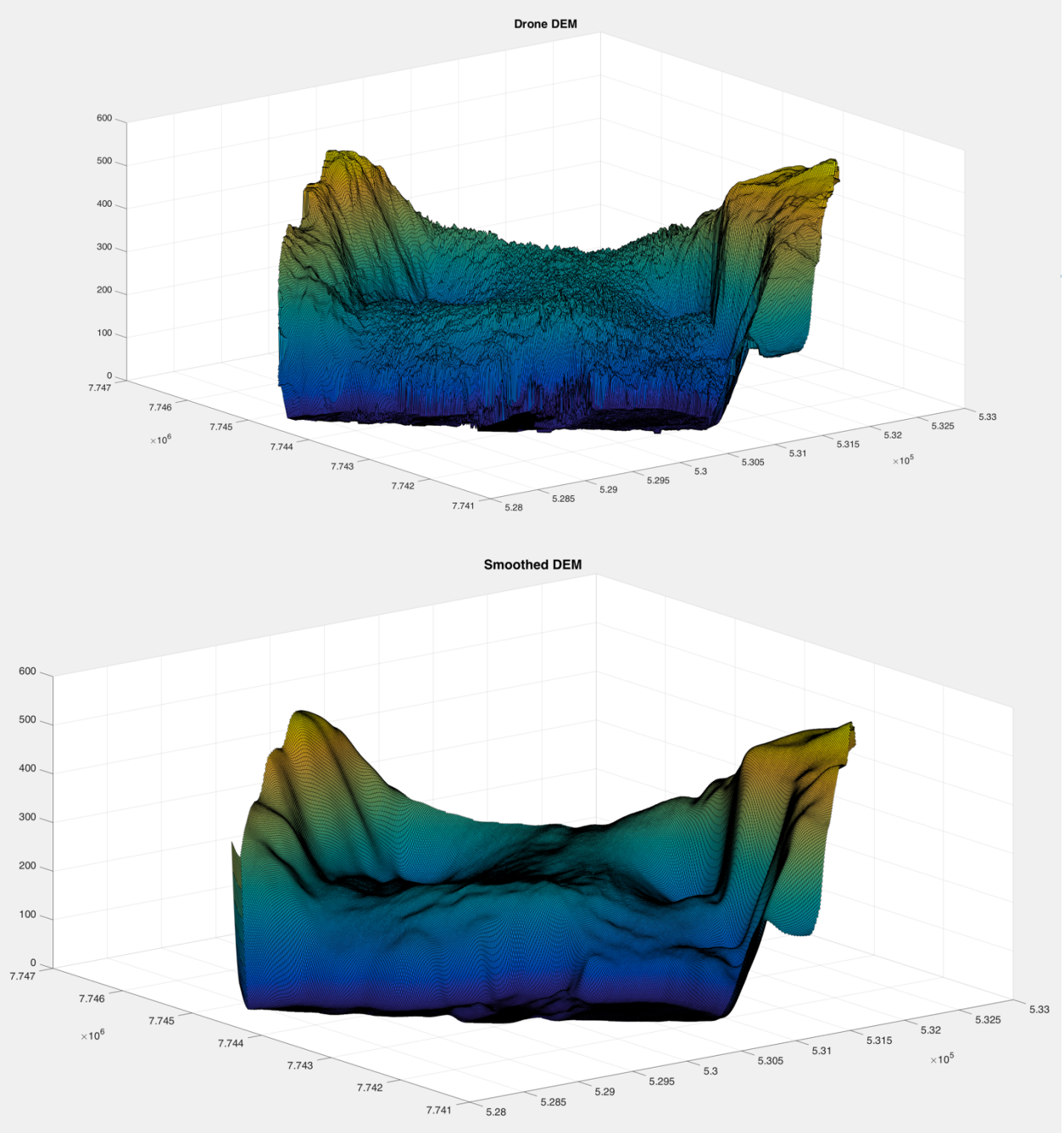


Fig. 47 shows the smoothing applied on the sample of the drone derived digital elevation model. Above the original, below the smoothed DEM used for georectification.

8.4.4 Further Figures

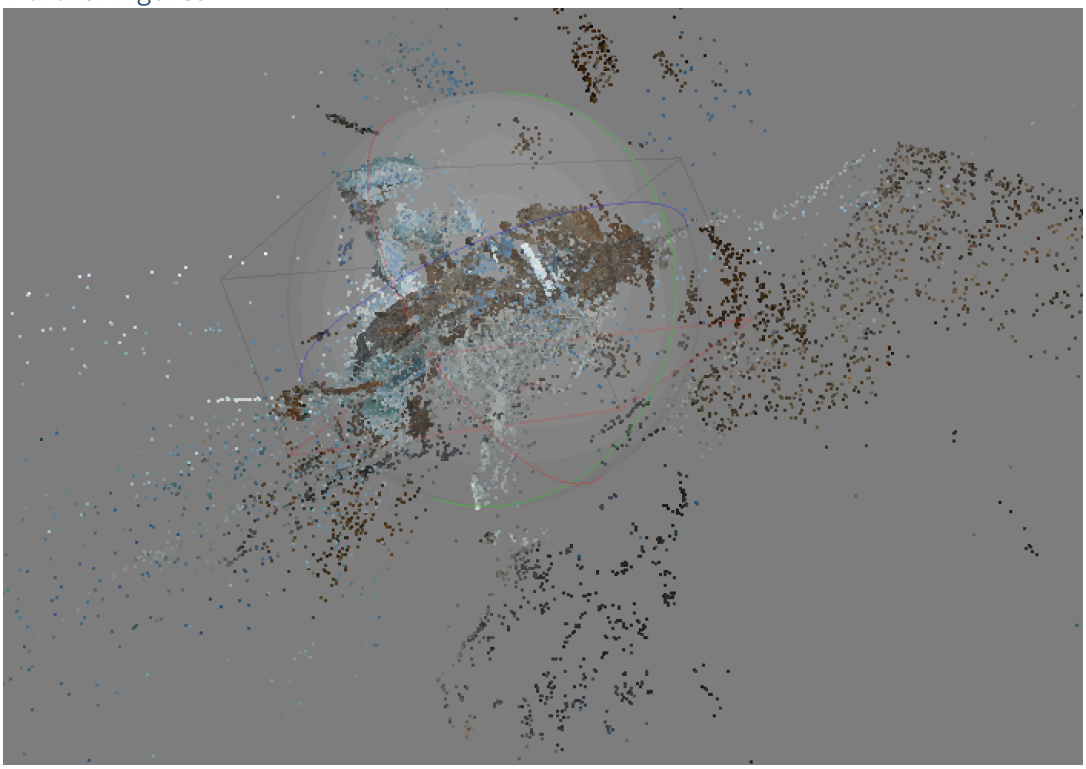


Fig. 48 sample of an unsuccessful point cloud using PhotoScan Pro.

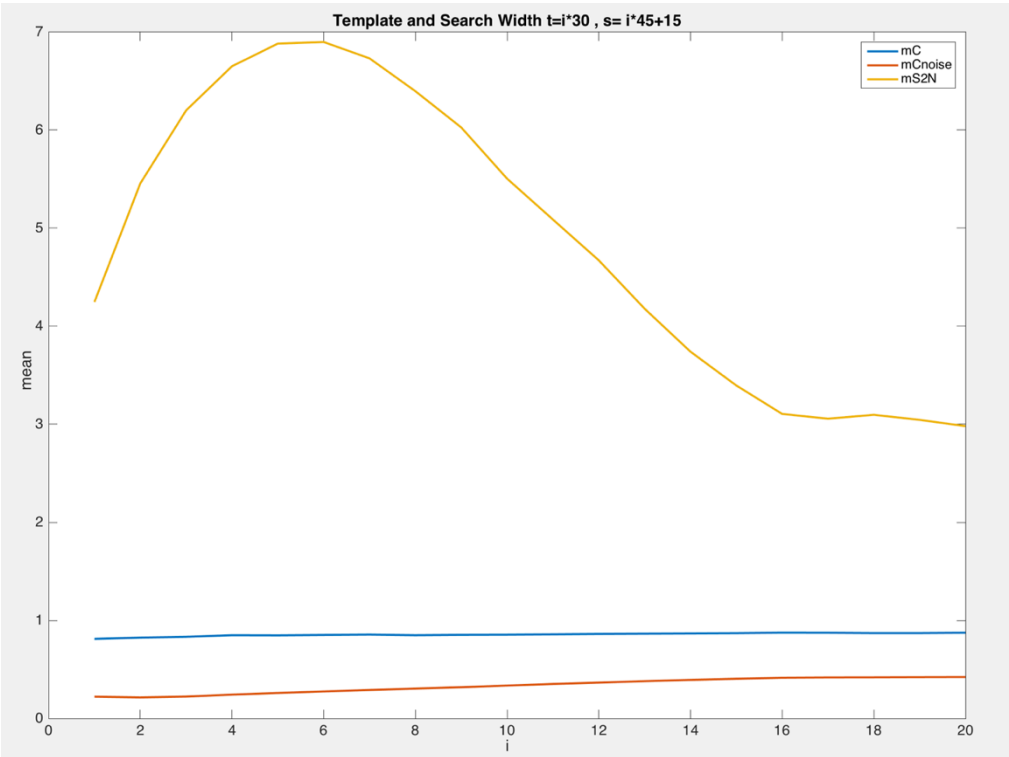


Fig. 49 shows the signal to noise ration for different sizes of template and search window.

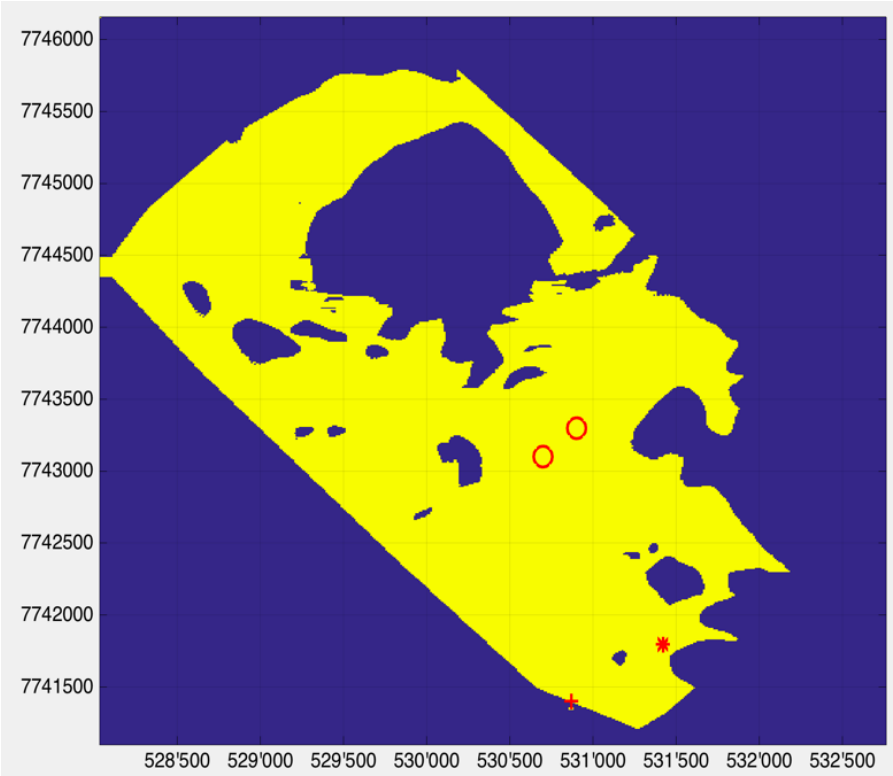


Fig. 50 Sample of the view shed calculated by the ImGRAFT's own voxelviewshed.m function.

8.5 Code Sample

The MATLAB code sample subsequent is tailored for the lower camera. The actual code, ready to run, including the all functions and some data is available for download from *https://eqi.voransicht.ch/flow.zip* .

```
%% --Using Time Laps Camera (TLC) imagery for velocity displacement of Eqi Sermia----
%  a maritime-terminated outlet glacier, West Greenland.
%  This script is heavely based on Image GeoRectification and Feature
%  Tracking (ImGRAFT) by Messerli and Grinsted (2015) example Engabreen.
% ---------------------------------------------------------------------------------
Master Thesis 'Flow velocity patterns of a tidewater glacier in West-Greenland derived
from oblique photography ',Adrian Simon PETER, Ralligweg 8, 3012 Bern,
adrian.peter2@hotmail.com, Student Nr: 11-101-474

Glaciology and Geomorphodynamics, Physical Geography,Department of Geography,
Faculty of Science, University of Zurich, Switzerland
Supervised by: Prof. Dr. Andreas Vieli, Dr. Martin Lüthi
% April 2017
%%
clear all;
close all;
cd('INSERT DIRECTORY');
datafolder=('INSERT DIRECTORY');
%% Setup file locations and load images & data
fC=fullfile(datafolder,sprintf('GLACIER MASKES OF DEM.jpg'));
C=imread(fC);
[dem.x,dem.y,dem.Z] = createAxisFromGeoTiff(fullfile(datafolder,' DEM .tif'));
dem.Z = double(dem.Z);
dem.Z(dem.Z>1e35 | dem.Z<-1e35) = NaN;
dem.rgb = rgb2gray(imread(fullfile(datafolder,'DEM RGB.jpg')));
[dem.X, dem.Y] = meshgrid(dem.x,dem.y);
dem.mask = C<200; % create binary mask from glacier from above
GCP=load(fullfile(datafolder,'GCP.txt')); %load ground control points
%% Prepare DEM by filling crevasses.
% Smoothing of DEM using Gaussian Disk Filter
%
% -----------   THIS PART IS ------------
% ----------- HAS BEEN REMOVED ----------
% ----------- DUE TO COPYRIGHT ----------
%%
img = ['######.jpg']; % Add list of images
%% Add Time Stamps if EXIF failed
Time.Strg=load(fullfile(datafolder,'CameraTime.txt'));
Time.Date= datetime(Time.Strg,'ConvertFrom','yyyymmdd');
%%
tic % Start Stopwatch
% ---------- Initial Camera Settings ------------
cameralocation=[530870 7741400 230]; %known camera location:
%crude estimate of look direction.
j= length(img(:,1))-1; % Anzhal Testbilder -1
for i= i:j;
disp({ 'calculate image' (i) 'of' (j)}) % the value i is indicating the progress
    imgA = img(i,:); % image A
    ij= i+1; % indices for Image Selection and Time Samp.
    imgB = img(ij,:); % image B
    A=imread(fullfile(datafolder,'Pix/',imgA));
    B=imread(fullfile(datafolder,'Pix/',imgB));

% ------------------- Determine camera parameters for image A ------------------------
FocalLength=5.1; %in mm
SensorSize=[8.07 5.56]; %in mm
imgsz=size(A);
f=imgsz([2 1]).*(FocalLength./SensorSize);
camA=camera(cameralocation,size(A),[81 0 -2]*pi/180,f,fliplr(size(A(:,:,1))./2));
%insert yaw,pitch,roll in∞, where 0 is east, 90 is north, 180 is west.
```

```
%in Camera die Bildgr^sse ebenfalls angegeben.

%Use GCPs to optimize the following camera parameters:
%view dir, focal lengths, and a simple radial distortion model
[camA,rmse,aic]=camA.optimizecam(GCP(:,1:3),GCP(:,4:5),'11100111110010000000'); %9
Parameters of full Vector
fprintf('reprojectionerror=%3.1fpx  AIC:%4.0f\n',rmse,aic);

% ----- Visually compare the projection of the GCPs with the pixel coords:
   %figure
        axes('position',[0 .1 1 .8]); hold on
        image(A)
        axis equal off ij tight
        hold on
        uv=camA.project(GCP(:,1:3));
        hold on;
        h=plot(GCP(:,4),GCP(:,5),'g+',uv(:,1),uv(:,2),'rx');
        hold off;
        legend(h,'UV of GCP','projection of GCPs','location','southoutside')
        title(sprintf('Projection of ground control points. RMSE=%.1fpx',rmse))
% ----------- Determine view direction of camera B. -----------
%  find movement of rock features be tween images A and B
%  determine camera B by pertubing viewdir of camera A.

qp=5; % set initial template and search window multiplier.
t= qp*30; %template
s= t*1.5+15; %search
% First get an approximate estimate of the image shift using a single large
% template
   % ------------------------------- Assume offset -------------------------------- %

[duoffset,dvoffset]=templatematch(A,B,1290,554,'templatewidth',110,'searchwidth',220,'s
upersample',0.5, 'showprogress',false);
    % --------- kill criteria --------- %
    if (sqrt((duoffset.^2)+(dvoffset.^2))) < 5; % if the inistial offset is greater
than 5
    elseif isnan(sqrt((duoffset.^2)+(dvoffset.^2))) > 0; % or equal to NaN
        continue % the loop skips to the conditions (below).
    else
        % if first image pair has a low match, take next (imgA = i, imgB=i+2).
        ij= i+2; % jump to the next image in line.
        imgB = img(ij,:); % image B

[duoffset,dvoffset]=templatematch(A,B,1290,554,'templatewidth',110,'searchwidth',220,'s
upersample',0.5, 'showprogress',false);
    end
         if (sqrt((duoffset.^2)+(dvoffset.^2))) < 5; % if the inistial offset is
greater than 3
         elseif isnan(sqrt((duoffset.^2)+(dvoffset.^2))) > 0; % or equal to NaN
                continue
         else continue % the loop skips to the next image pair.
         end

% Get a whole bunch of image shift estimates using a grid of probe points.
% Having multiple shift estimates will allow us to determine camera
% rotation.

[pu,pv]=meshgrid(50:200:3650,2320:200:2720); % creat a mesh on the northern ridge
pu = double(pu);
pv = double(pv);
pu=pu(:); pv=pv(:);
[du,dv,~]=templatematch(A,B,pu,pv,'templatewidth',50,'searchwidth',100,'supersample',3,
'initialdu',duoffset,'initialdv',dvoffset, 'showprogress',false);

% Determine camera rotation between A and B from the set of image
% shifts.
% find 3d coords consistent with the 2d pixel coords in points.
```

```
xyz=camA.invproject([pu pv]);
% the projection of xyz has to match the shifted coords in points+dxy:

[camB,rmse]=camA.optimizecam(xyz,[pu+du pv+dv],'00000111000000000000'); %optimize 3
view direction angles to determine camera B.
rmse;

%quantify the shift between A and B in terms of an delta angle.
DeltaViewDirection=(camB.viewdir-camA.viewdir)*180/pi;

% Generate a set of points to be tracked between images
% # Generate a regular grid of candidate points in world coordinates.
% # Cull the set of candidate points to those that are visible and glaciated
% The viewshed is all the points of the dem that are visible from the
% camera location. They may not be in the field of view of the lens.
dem.visible=voxelviewshed(dem.X,dem.Y,dem.smoothed,camA.xyz);

[X,Y]=meshgrid(min(dem.x):30:max(dem.x),min(dem.y):30:max(dem.y));%make a 30m grid
keepers=double(dem.visible&dem.mask); %visible & glaciated dem points
keepers=filter2(ones(1)/(1^2),keepers); %throw away points close to the edge of
visibility
keepers=interp2(dem.X,dem.Y,keepers,X(:),Y(:))>.99; %which candidate points fullfill
the criteria.
xyzA=[X(keepers) Y(keepers) interp2(dem.X,dem.Y,dem.filled,X(keepers),Y(keepers))];
[uvA,~,inframe]=camA.project(xyzA); %where would the candidate points be in image A
xyzA=xyzA(inframe,:); %cull points outside the camera field of view.
uvA=uvA(inframe,:); %round because template match only works with integer pixel coords
uvA(end+1,:)=[2585 2061]; %add a non-glaciated point to test for residual camera
movement (here a tunnel entrance)

%Note xyzA no longer corresponds exactly to uvA because of the rounding.
% Track points between images.
% calculate where points would be in image B if no ice motion.
% ( i.e. accounting only for camera shake)
camshake=camB.project(camA.invproject(uvA))-uvA;
options=[];
options.pu=uvA(:,1);
options.pv=uvA(:,2);
%options.showprogress=[imgA imgB]; % Display of progress. Turned-off in order to save
memory
options.searchwidth=s;
options.templatewidth=t;
options.supersample=5; %supersample the input images
options.initialdu=camshake(:,1);
options.initialdv=camshake(:,2);

[du,dv,C,Cnoise,pu,pv]=templatematch(A,B,options);
uvA=[pu pv]; %the centers of the templates may have been rounded to nearest pixel.

uvB=uvA+[du dv];
signal2noise=C./Cnoise;

% Georeference tracked points
% ... and calculate velocities
tB =datenum(Time.Date(ij)); % come up with a time intervall in days
tA =datenum(Time.Date(i));
xyzA=camA.invproject(uvA,dem.X,dem.Y,dem.filled); % has to be recalculated because uvA
has been rounded.
xyzB=camB.invproject(uvB,dem.X,dem.Y,dem.filled); % -dem.mask*22.75*(tB-tA)/365); %
impose a thinning of the DEM of 23m/yr between images% stimmt so nicht, wenn dann dazu
je weiter weg.
V=(xyzB-xyzA)./(tB-tA); % 3D velocity.

%figure;
%showimg(dem.x,dem.y,dem.Z);
%hold on
Vn=sqrt(sum(V(:,1:2).^2,2));
```

```
keep=signal2noise>1&C>.6;
%scatter(xyzA(keep,1),xyzA(keep,2),100,Vn(keep),'.')
%quiver(xyzA(keep,1),xyzA(keep,2),V(keep,1)./Vn(keep),V(keep,2)./Vn(keep),.2,'k')
%caxis([0 20])
%colormap jet
%hcb=colorbar('southoutside');
%plot(camA.xyz(1),camA.xyz(2),'r+')
%title('Velocity in metres per day')
%Project velocity onto downhill slope direction
% ----
% The largest error in the velocities will along the view direction vector.
% By projecting to the slope direction we strongly suppress errors arising
% from this.

[gradX,gradY]=gradient(dem.filled,dem.X(2,2)-dem.X(1,1),dem.Y(2,2)-dem.Y(1,1)); % Eine
Schrittweite zwischen dem Grid feststellen
gradN=sqrt(gradX.^2+gradY.^2); % den Betrag daraus addieren
gradX=-gradX./gradN;gradY=-gradY./gradN; % den Betrag in die jeweilige Richtung.
gradX=interp2(dem.X,dem.Y,gradX,xyzA(:,1),xyzA(:,2)); %Zwischen den Punkten
Interpolieren
gradY=interp2(dem.X,dem.Y,gradY,xyzA(:,1),xyzA(:,2));
Vgn=V(:,1).*gradX+V(:,2).*gradY; % X / Y -Geschwindigkeit mit dem Betrag
multiplizieren.
Vg=[Vgn.*gradX Vgn.*gradY]; % Erneut die jew. Geschwindigkeit mir dem Betrag
multiplizieren.

% Save Results
    Result(i).keep = keep;
    Result(i).Vn = Vn;
    Result(i).V = V;
    Result(i).Vg = Vg;
    Result(i).Vgn = Vgn;
    Result(i).xyzA = xyzA;
    Result(i).xyzB =xyzB;
    Result(i).peakCorr = C;  %peakCorr is the maximum correlation coefficient found at
the location of each match.
    Result(i).meanAbsCorr =Cnoise;  %meanAbsCorr is the average or typical correlation
coefficient over the entire search window.
    Result(i).SNR = signal2noise; % Signal-to-nosie ratio
    Result(i).time = (tA+tB)/2; % Calculates and Saves Time represented.
    Result(i).imgA= [imgA]; %Date & Time of fixed picture
    Result(i).imgB= [imgB]; %Date & Time of moving picture
    Result(i).camshake =camshake;
    Result(i).duoff =duoffset;
    Result(i).dvoff =dvoffset;

    save RunYYYYMMHH.mat
end
toc
% ------------------------------------- FINITO ----------------------------------------
```

8.6 Acknowledgement

I am grateful for all the help and support I got throughout this thesis. For the many inputs and lively discussions, mental support and the strengthening coffee breaks of various people. Foremost, my spacial thank goes especially to:

Andreas Vieli for the advice, the constructive discussions, the support and the opportunity to joint the field season at Eqi.

Martin Lüthi for the help and the support, as project manager in the field and supervisor in the office.

Rémy Mercenier for this help during the fieldtrip and his upbeat mood on down days in the camp.

Alexandra Messerli for the encouragement and time of consultation on *ImGRAFT* issues.

Christoph Rohner for letting me use his satellite derived data as well as the around the clock support (6 to 24 o'clock, and also on rainy weekends), solving various problems.

Allie Strel for proof reading and giving the thesis a more precise language by eliminating disturbing foregrounds.

Lisa Ziegler for the loving support on long MATLAB-y days full of frustration.

My family, for their support throughout the last 27 years.

April 2017

Adrian Peter

8.7 Personal Declaration:

I hereby declare that the submitted thesis is the result of my own, independent work. All external sources are explicitly acknowledged in the thesis.

Bern, the 20th of April 2016

Adrian Simon Peter