

Master's thesis

Snow depth estimation in forested terrain using the ICESat-2 space laser

Simen Dalseid Aune

Geomorphology and Geomatics
60 ECTS credits

Department of Geosciences
Faculty of Mathematics and Natural Sciences

Spring 2024



Simen Dalseid Aune

Snow depth estimation in forested
terrain using the ICESat-2 space
laser

Supervisors:
Désirée Treichler
Clare Webster

Abstract

NASA's Ice, Cloud and Land Elevation Satellite-2 (ICESat-2) provides a way to measure terrain heights at a global scale with 91-day repeat intervals in polar regions. ICESat-2's ATLAS instrument is a photon counting, visible green light (532 nm) Light Detection and Ranging (LiDAR) sensor which measures the Earth surface in six separate beams. An ongoing project at the University of Oslo, SNOWDEPTH, aims to use these measurements to model snow depths at large scale. This thesis will build on — and hopefully contribute to — that work, and aims to assess whether ICESat-2 is able to accurately measure snow surface heights in a boreal forest in Southern Norway. LiDAR is particularly effective at ground surface measurements in forests, due to it's ability to penetrate canopies and reliance on just a single laser beam to determine positions. While this is a common use case for Airborne Laser Scanning (ALS) and Uncrewed Aerial Vehicle (UAV)-borne LiDAR sensors, space based laser altimetry is less explored. In this thesis, UAV-borne LiDAR has been used to generate control data from a total of four field sites, with a spatial and near-temporal overlap with ICESat-2 measurements. The ICESat-2 data is normalised against Digital Terrain Model (DTM)s from the Norwegian Mapping Authority (NMA), and the resulting snow depths are compared to the UAV control data. To ensure calculations are done on photons reflected from the snow surface, filtering is necessary. Forest canopies are also represented in the data, along with stray photons and photons which have penetrated the snow surface before reflecting. Three methods for photon filtering are compared for their ability to help determine the actual snow surface height, and filter noise. Two of the methods are developed specifically during the work with this thesis, while the third is developed by NASA. The correlation between the canopy cover and height measurement bias within each photon footprint is also examined, to determine the impact canopy cover has on the bias and assess the presence of a causal relationship. Overall, across all the field sites, the NASA developed filtering method was very efficient with a Root Mean Square Error (RMSE) of 0.59 m. The second method was slightly more accurate, with an RMSE of 0.48 m, but relies on additional input data. The third method performed the worst, with an RMSE of 0.72 m. This is an order of magnitude larger than what has been achieved with ICESat-2 in unforested flat terrain, but comparable to at least one study looking at snow depths in sloped, forested terrain.

Acknowledgement

First off I would like to thank my supervisors at the Department of Geosciences, Désirée and Clare, for your guidance, support and input while working on this project. Your help has been highly valuable, and I can safely say that I would not have finished the thesis without you.

I would also like to thank my employer, who has generously allowed me the time and resources to attain more knowledge within the field of Geosciences.

My family deserves my gratitude as well. My mother and father for their constant encouragements, and also for providing a place to retreat when the need for isolation arrived. And of course my daughter, Josefina, for always keeping my mood up and motivation high.

Last, but not least, I would like to thank my lovely wife Tirill, for your support and comfort throughout these past four years. At times I am certain that this work has demanded more from you than me, and I will forever be grateful for your seemingly never-ending patience with my constant ups and downs. I love you.

Acknowledgement

Contents

1	Introduction	1
1.1	Background	1
1.2	Research question	2
1.3	Theoretical framework	2
1.3.1	Snow mapping	3
1.3.2	Forest mapping	3
1.3.3	Light Detection and Ranging (LiDAR)	4
1.3.4	Geodesy	4
2	Data acquisition	7
2.1	Instrumentation	7
2.1.1	ICESat-2	8
2.1.2	DJI Zenmuse L1	10
2.2	Data acquisition	11
2.2.1	The actual site selection process	12
2.2.2	UAV data acquisition	13
2.2.3	ICESat-2 data acquisition	13
2.3	Field sites	13
2.3.1	Drammen	16
2.3.2	Hof	17
2.3.3	Jevnaker	18
2.3.4	Vikerfjell	19
2.4	Data	21
2.4.1	ICESat-2 products	21
2.4.2	UAV-borne LiDAR point clouds	21
2.4.3	Norwegian Mapping Authority (NMA) elevation data	22
3	Data processing	23
3.1	Software	23
3.1.1	DJI Terra	23
3.1.2	R	23
3.1.3	Python	24
3.2	Processing	25
3.2.1	UAV data	25
3.2.2	Data structuring and metrics extraction	27
3.2.3	Canopy height model	28
3.2.4	Photon filtering methods	28
3.2.5	Snow depth calculations	33
3.2.6	Accuracy metrics	33
3.2.7	Canopy cover vs. bias correlation	34

Contents

4	Results	37
4.1	Snow depth estimates	37
4.1.1	Drammen	38
4.1.2	Hof	39
4.1.3	Jevnaker	40
4.1.4	Vikerfjell East	41
4.1.5	Vikerfjell West	42
4.2	Performance of filtering methods	43
4.3	Impact of canopy cover on measurement bias	44
4.3.1	Per field site	45
5	Discussion	51
5.1	Snow depth estimation in forested terrain	51
5.2	Filtering methods	52
5.2.1	Nearest neighbor	52
5.2.2	Threshold validation	52
5.2.3	Yet Another Photon Classifier (YAPC)	52
5.3	Workflow review	53
5.4	Control Data and Field Work	53
6	Conclusion	57
A	UAV LiDAR processing script	63
A.1	GNU bash script	63
A.2	R code	64
B	Python script	65
C	Complete errors table	71

List of Figures

2.1	Map from SeNorge.no showing the maximum snow depth for Southern Norway in 2023.	7
2.2	Global map showing where snow is expected to be present during winter, and the minimum and maximum latitudes of the GEDI mission for reference. (Snow data from MODIS/NASA.gov, August 2023 and January 2024).	8
2.3	Spatial pattern of the ATLAS beams and footprints. From T. A. Neumann et al. (2023)	9
2.4	Artist rendition of the ICESat-2 satellite platform. ©NASA	9
2.5	DJI Matrice 300 RTK UAV with DJI Zenmuse L1 on a gimbal mount. (Image from DJI.com, accessed on 2024-04-09)	10
2.6	Overview map of Southern Norway, with each field site represented by red dots. The purple lines show the ICESat-2 ground tracks used in the thesis, while the blue lines show the other tracks in a single cycle.	15
2.7	Upper: An overview map of the Drammen field site. The boundaries of the study area is shown with a dark blue line, while the ICESat-2 photons are shown in red. The green raster overlay shows the canopy height model, from the UAV data. Lower: Sample image from the UAV flight. The canopy structure is covered in snow.	16
2.8	Upper: An overview map of the Hof field site. The boundaries of the study area is shown with a dark blue line, while the ICESat-2 photons are shown in red. The green raster overlay shows the canopy height model, from the UAV data. Lower: Sample image from the UAV flight. The canopy structure is covered in snow.	17
2.9	Upper: An overview map of the Jevnaker field site. The boundaries of the study area is shown with a dark blue line, while the ICESat-2 photons are shown in red. The green raster overlay shows the canopy height model, from the UAV data. Lower: Sample image from the UAV flight. The canopy structure is not covered in snow.	18
2.10	Upper: An overview map of the Vikerfjell field site, which had intersecting ICESat-2 tracks. The eastern is shown here. The boundaries of the study area is shown with a dark blue line, while the ICESat-2 photons are shown in red. The green raster overlay shows the canopy height model, from the UAV data. Sample image from the UAV flight. The canopy structure is not covered in snow.	19
2.11	An overview map of the Vikerfjell field site, which had intersecting ICESat-2 tracks. The western is shown here. The boundaries of the study area is shown with a dark blue line, while the ICESat-2 photons are shown in red. The green raster overlay shows the canopy height model, from the UAV data.	20

List of Figures

2.12	Sample point cloud from one of the field sites, showing canopy structure rising vertically, and half of a road passing through. Blue points are first returns. Green are second and red third returns.	22
3.1	Flow chart describing the main workflow of the data processing.	25
3.2	Example DTM output from the lidR processing, showing the Drammen field site from a top-down angle.	27
3.3	The effect of the nearest neighbour filtering method is shown in this plot. The ICESat-2 ATL03 photons (orange) show a large vertical spread, while the filtered photons (red) appear much closer to the UAV measured snow surface. The UAV canopy surface (green) is also displayed.	30
3.4	A visualisation of how the maximum snow depth in a subset increases when more photons are included, when applying the <i>threshold validation</i> filtering method to the photons from the Drammen and Vikerfjell East field sites. The red line shows where the slope increases beyond the defined limit (0.15 m / 5 % photons), and what the maximum snow depth in the subset is at that point. Snow depths below this threshold are considered valid by the function..	31
3.5	The effect of the threshold validation filtering method on a sample subset from the Drammen field site. The ICESat-2 ATL03 photons (orange) show a large vertical spread, while the filtered photons (red) appear much closer to the UAV measured snow surface. The UAV canopy surface (green) is also displayed..	32
3.6	The effect of the filtering by applying the YAPC algorithm on a sample subset from the Hof site. The ICESat-2 ATL03 photons (orange) show a large vertical spread, while the filtered photons (red) appear much closer to the UAV measured snow surface. The UAV canopy surface (green) is also displayed..	33
3.7	Scatter plots showing different distributions of points along both axes, and their corresponding Pearson correlation coefficients (r). The scatter resembles a straight line when r approaches either $ 1 $. In sub-figure E, the effect of the chosen range is visible, with an r of 0.34 for the data within the shaded box, and 0.60 when all the data is counted. (From Schober, Boer and Schwarte (2018))	35
4.1	Snow depths as measured by the UAV and the two beams of ICESat-2 at the Drammen site. ICESat-2 photons have been filtered with three different methods. The DTM from NMA is shown in solid grey.	38
4.2	Snow depths as measured by the UAV and the two beams of ICESat-2 at the Hof site. ICESat-2 photons have been filtered with three different methods. The DTM from NMA is shown in solid grey.	39
4.3	Snow depths as measured by the UAV and the two beams of ICESat-2 at the Jevnaker site. ICESat-2 photons have been filtered with three different methods. Note the strong beam being about 1/4 the length of the weak beam, due to low spatial overlap with the UAV data. The DTM from NMA is shown in solid grey.	40
4.4	Snow depths as measured by the UAV and the two beams of ICESat-2 at the eastern part of the Vikerfjell site. ICESat-2 photons have been filtered with three different methods. The DTM from NMA is shown in solid grey.	41
4.5	Snow depths as measured by the UAV and the two beams of ICESat-2 at the western part of the Vikerfjell site. ICESat-2 photons have been filtered with three different methods. The DTM from NMA is shown in solid grey.	42

- 4.6 The distribution of the vertical measurement bias of the photons after each filtering method is applied, per field site. Median bias is indicated by the lines. 43
- 4.7 Upper: Scatter plot with measurement bias from all the filtering methods used, shown in individual colours, versus the canopy cover inside the photon footprint. A clear linear correlation would be present if the points aligned to a 45°line ($|x| = |y|$). No such pattern is visible for any of the methods. Lower: The distribution of snow depths in the field site, as estimated after each filtering method has been applied. The snow depths from the UAV data is shown in grey. 46
- 4.8 Upper: Scatter plot with measurement bias from all the filtering methods used, shown in individual colours, versus the canopy cover inside the photon footprint. A clear linear correlation would be present if the points aligned to a 45°line ($|x| = |y|$). No such pattern is visible for any of the methods. Lower: The distribution of snow depths in the field site, as estimated after each filtering method has been applied. The snow depths from the UAV data is shown in grey. 47
- 4.9 Upper: Scatter plot with measurement bias from all the filtering methods used, shown in individual colours, versus the canopy cover inside the photon footprint. A clear linear correlation would be present if the points aligned to a 45°line ($|x| = |y|$). No such pattern is visible for any of the methods. Lower: The distribution of snow depths in the field site, as estimated after each filtering method has been applied. The snow depths from the UAV data is shown in grey. 48
- 4.10 Upper: Scatter plot with measurement bias from all the filtering methods used, shown in individual colours, versus the canopy cover inside the photon footprint. A clear linear correlation would be present if the points aligned to a 45°line ($|x| = |y|$). No such pattern is visible for any of the methods. Lower: The distribution of snow depths in the field site, as estimated after each filtering method has been applied. The snow depths from the UAV data is shown in grey. 49
- 4.11 Upper: Scatter plot with measurement bias from all the filtering methods used, shown in individual colours, versus the canopy cover inside the photon footprint. A clear linear correlation would be present if the points aligned to a 45°line ($|x| = |y|$). No such pattern is visible for any of the methods. Lower: The distribution of snow depths in the field site, as estimated after each filtering method has been applied. The snow depths from the UAV data is shown in grey. 50

List of Figures

List of Tables

2.1	Orbital parameters of ICESat-2	10
2.2	DJI Zenmuse L1 sensor properties (From dji.com)	10
2.3	A comparison of the specifications of both the ATLAS and Zenmuse L1 LiDAR sensors.	11
2.4	The main criteria affecting the field site selection for ICESat-2 and UAV data acquisition are listed in this table.	11
2.5	The field sites with their geodetic coordinates and times of ICESat-2 and UAV data acquisitions. The NMA DTM is based on ALS surveys (NDH), and the time periods for these surveys are also added. (All times are UTC.)	14
2.6	Descriptive statistics of the field sites, and dates of the NMA NDH data collection.	14
4.1	Quality metrics per filtering method, with both beams, across all the field sites. r (canopy) is the correlation coefficient between measurement bias and canopy cover. Photons kept after filtering is also shown (not applicable for YAPC).	44
4.2	Quality metrics per filtering method, with only strong beams, across all the field sites. r (canopy) is the correlation coefficient between measurement bias and canopy cover. Photons kept after filtering is also shown (not applicable for YAPC).	44
4.3	Quality metrics per filtering method, with only weak beams, across all the field sites. r (canopy) is the correlation coefficient between measurement bias and canopy cover. Photons kept after filtering is also shown (not applicable for YAPC).	44
4.4	Comparisons of the RMSE values on the snow depth measurements within the canopy and in vegetation-free terrain, on photons filtered with the threshold validation method.	45
4.5	Quality metrics of the filtering methods, for each individual site. Photons kept after filtering is also shown (not applicable for YAPC).	45
C.1	Complete errors data for all methods and both beams, in the Drammen and Hof field sites. (PART I)	71
C.2	Complete errors data for all methods and both beams, in the Jevnaker and Vikerfjell field sites. (PART II)	72

List of Tables

Acronyms

ALS Airborne Laser Scanning.

AP Aerial Photogrammetry.

ATLAS Advanced Topographic Laser Altimeter System.

CHM Canopy Height Model.

DEM Digital Elevation Model.

DTM Digital Terrain Model.

GEDI Global Ecosystem Dynamics Investigation.

GLAS Geoscience Laser Altimeter System.

GNSS Global Navigation Satellite System.

HDF Hierarchical Data Format.

ICESat Ice, Cloud and Land Elevation Satellite.

ICESat-2 Ice, Cloud and Land Elevation Satellite-2.

IMU Inertial Measurement Unit.

InSAR Interferometric synthetic aperture radar.

ISS International Space Station.

LiDAR Light Detection and Ranging.

MAE Mean Absolute Error.

NASA National Aeronautics and Space Administration.

NDH Nasjonal Detaljert Høydemodell.

NIR Near-Infrared.

NMA Norwegian Mapping Authority.

NSIDC National Snow and Ice Data Center.

NTRIP Networked Transport of RTCM.

PPK Post-Processed Kinematic.

Acronyms

RMSE Root Mean Square Error.

RTK Real-Time Kinematic.

TIN Triangulated irregular network.

UAV Uncrewed Aerial Vehicle.

YAPC Yet Another Photon Classifier.

Chapter 1

Introduction

1.1 Background

Repeated measurements of snow depths – at large scale – can serve a multitude of purposes in Geosciences. They can provide estimates of available hydropower and drinking water resources; As time series spanning several years they can serve as an indicator of climate change; Given high spatio-temporal resolution they can even aid prediction of snow avalanche risk. However, gathering the data from the field is resource demanding. Remote areas, in countries with increasing energy demand, often combined with challenges brought on by climate change, generate the need for low-cost methods of supervising the amounts of snow in large catchments.

Remote sensing provides a predictable and reliable way of measuring snow depths over time. Whether borne by Uncrewed Aerial Vehicle (UAV), aircraft or satellites, the sensors will be able to provide spatially dense measurements over large swaths of terrain. Satellite-borne sensors in particular are predictable, and the output properties near-uniform over time. LiDAR has the ability to penetrate canopy cover to also register several returns per laser beam, thus allowing calculation of two surfaces per survey: The top-level surface, such as canopies and buildings; and the ground surface, referred to as the terrain. While airborne Light Detection and Ranging (LiDAR) has become the standard way of mapping the ground surface for use in digital terrain models (DTM) in local and regional surveys, Radar and stereo-photogrammetry dominate among satellite-borne sensors, surveying at global or inter-regional scale (Wilson 2012). Airborne LiDAR has a vertical accuracy of up to 0.15 m, but data acquisition is expensive and the post-processing requirements are high (Wilson 2012).

Measuring snow depths realistically requires a sub-metre level of vertical accuracy, as 1 m is already beyond the annual snow depth maximum for many regions, for example in Southern Norway (See fig. 2.1). Interferometric synthetic aperture radar (InSAR) sensors, such as the ones carried by the TerraSAR-X/TanDEM-X twin satellites, are capable of vertical accuracy of down to 2 m (Faller, Weber and GmbH 2007). Even still, these sensors are not suitable for measuring the ground surface in dense vegetation and forests. Both photogrammetry and SAR rely on sensing a point from several angles to establish its position, and the vertical structure of forests makes this difficult. The low-contrasting snow surface also makes the tie points necessary for photogrammetry sparser.

Satellite-borne LiDAR sensors may provide a solution to this problem. The Ice, Cloud and Land Elevation Satellite-2 (ICESat-2) carries an instrument called the Advanced Topographic Laser Altimeter System (ATLAS) — a photon-counting LiDAR

sensor with a near-global coverage (88° N- 88° S) and 91-day repeat period (Thomas A. Neumann et al. 2019). While photon-counting sensors do not provide the capabilities of a full-waveform LiDAR, such as the structure and properties of the back-scattering from the illuminated surface (Mallet and Bretar 2009), properly filtered individual photons can enable height measurements down to sub-meter accuracy while penetrating the forest canopies.

The ATLAS instrument has six beams of visible green laser, using the two-way travel time for each photon to calculate the distance to the surface. With a wavelength of 532 nm, the laser beams have the capability to penetrate shallow water, snow and ice bodies.

One project aimed at modelling snow with the use of ICESat-2 is the SNOWDEPTH project at the University of Oslo. By combining the measurements from the ATLAS instrument with other satellite data, climate reanalyses, elevation data and statistical methods, the project aims to determine how much snow there is on the ground, at scale (Forskningsrådet 2022). For this to work in mountainous and forested terrain, it is necessary to determine if and how forest canopies affect the surface measurements.

1.2 Research question

Several studies have been made into the ICESat-2 ATLAS instrument's ability to measure terrain heights in forests (Neuenschwander et al. 2020; Feng et al. 2023; Pronk, Eleveld and Ledoux 2023), and some studies have looked into its ability to estimate snow depths (Deschamps-Berger et al. 2023; Hu et al. 2021). Few studies examine both aspects combined: How well does ATLAS perform in estimating snow surface heights — and thus snow depths — in forested terrain?

Regardless of the answer, filtering methods may improve that ability, and identifying the most efficient filtering methods can aid future use of the ATLAS instrument for snow depth measurements in forested terrain.

This thesis will address these knowledge gaps, and use control data from a UAV borne LiDAR sensor to assess ATLAS' ability to measure snow surface heights in dense vegetation and forest canopies. It will also explore three methods for filtering noise from the data, with the purpose of increasing vertical accuracy. The research questions for the thesis are therefore:

1. *How does the ATLAS instrument on ICESat-2 perform when measuring snow surface heights in forested terrain?*
2. *Which filtering methods can improve the accuracy of these measurements?*

1.3 Theoretical framework

This section will provide the theoretical background for the thesis. The thesis combines the fields of forest mapping with snow mapping, by use of two different types of LiDAR sensors, from altitudes differing in the order of magnitudes. This section introduces the previous research in all of these fields, which will serve as the theoretical framework for the thesis.

1.3.1 Snow mapping

Snow is an important factor for how we understand our global environment, in many different aspects (Treichler and Andreas Kääb 2017). Snow depth measurements at a large scale is complex: Snow can hide vegetation and small-scale topographic features, making it difficult to estimate from the surface height alone how much snow there is above a terrain surface (Jost et al. 2007; Komarov and Sturm 2023); Snow is a solid material which exists relatively close to its melting point, which means that slight variations in meteorological conditions can alter its properties significantly, in short time (Adams et al. 2011; Pomeroy and Brun 2001); The low-contrasting surface makes it more difficult to measure using airborne and satellite photogrammetric and Interferometric synthetic aperture radar (InSAR) techniques. Even more so in forested and mountainous terrain, which significantly increases the probability of occlusion (Frey et al. 2018).

The traditional way of estimating snow depths is one of either direct in-situ measurements, remote sensing or climate reanalyses (Treichler and Andreas Kääb 2017). In-situ measurements requires the physical presence of equipment and/or personnel, which also means that the measurements are most often done in vicinity of man-made infrastructure. However, when considering snow as a hydropower reservoir, having snow depth estimates of catchment-scale, remote areas is significantly more valuable. Remote sensing can make this possible (Finger 2018).

1.3.2 Forest mapping

As mentioned in section 1.3.1, forests introduce several problems in remote sensing, which are not just related to snow mapping. In photogrammetry, the vertical structure of the forest introduces the problem of occlusion, due to the reliance on tie points on the surface which must be visible from several angles. Pearse et al. (2018) has found that points clouds derived from satellite photogrammetry provided an accuracy proximate to Airborne Laser Scanning (ALS) when conducting forest inventory in New Zealand, i.e. measuring the canopy itself, but that analysis relied on normalised heights using a DTM derived from the ALS data. The mean top height of the canopy was estimated with an RMSE of 2.1 m with the stereo satellite imagery, while the ALS data yielded an RMSE of 1.7 m.

A study by Rahlf et al. (2014) looked at four types of 3D data for timber volume estimation in Southern Norway: ALS, Aerial Photogrammetry (AP), Interferometric synthetic aperture radar (InSAR) and satellite radargrammetry. It found that ALS estimated the inventory with a plot-level relative RMSE of 19 %. AP yielded a relative RMSE of 31 %, InSAR 41 % and radargrammetry 44 % (Rahlf et al. 2014). Again, these analyses were all using a DTM based on ALS data for height normalisation. All of the other sensors struggle with penetrating the canopy, which makes them less useful for mapping the ground and snow surface.

Under-story vegetation is another problem affecting terrain model generation. Particularly when the canopy is dense, and e.g. 2 out of 3 returns are already registered in the canopy, the last return will potentially represent under-story vegetation, and not the ground. With ALS, these errors can amount to 0.20 m in leaf-off conditions (Simpson, Smith and Wooster 2017). Forest mapping with laser scanning has been conducted in Scandinavia for over 30 years (Næsset et al. 2004), and with the use modern UAVs, the cost of these operations has decreased significantly in the last decade.

1.3.3 Light Detection and Ranging (LiDAR)

LiDAR is an active sensor system which uses the two-way travel time of a beam of light to determine distances. Combined with the exact location and direction of the sensor, this distance can be used to calculate the location of the back-scattered surface. Each light pulse can have several returns, where each return corresponds to a different surface. E.g. canopy crowns, terrain or buildings. Because LiDAR relies on just a single beam to determine a three-dimensional position, it is especially valuable for mapping of the terrain surface in forests with its ability to penetrate the canopy. Two different types of LiDAR are used in this thesis: *Discrete return* and *photon counting*. The DJI Zenmuse L1 is discrete return, while the ATLAS instrument is a photon counting sensor. Full-waveform Light Detection and Ranging (LiDAR) sensors also exist, which record the distribution of the returned light. These data are more complex, but also yield more information about the illuminated surface.

Discrete return sensors are the most common modality of LiDAR instruments. As the name suggests, these sensors register individual returns from each beam. When the signal intensity exceeds a predefined threshold, it is registered as a return. The three-dimensional coordinates for the return is stored, along with the intensity. In other words, discrete return LiDARs detect individual returns representing the peaks in a waveform curve. The system normally records between 1 and 5 returns, depending on the sensor capability. (The Zenmuse L1 can register 3 returns.)

ICESat-2 is a satellite platform carrying the ATLAS instrument, which is a photon counting LiDAR sensor. Its predecessor ICESat carried the Geoscience Laser Altimeter System (GLAS), which was a full-waveform LiDAR (Liu et al. 2018). ICESat ended its operations in 2009, and today ICESat-2 and Global Ecosystem Dynamics Investigation (GEDI) remain the only operational laser altimeter sensors in orbit (Pronk, Eleveld and Ledoux 2023). GEDI is mounted on the International Space Station (ISS), which also means that its orbit is between 51.6°N and 51.6°S. This orbit makes it less interesting for snow depth retrieval, because it does not cover the polar or near-polar regions.

The most common wavelength for terrestrial mapping with LiDAR systems is 1064 nm (NIR), while 532 nm (visible green light) is common in bathymetric and glacial mapping (Szafarczyk and Toś 2023). The latter wavelength has less attenuation than the Near-Infrared (NIR) wavelength, and is therefore able to penetrate shallow water and ice bodies, making it able to map rivers and near-coastal water bodies (Szafarczyk and Toś 2023).

(Hu et al. 2021) has examined the ability of ICESat-2 to measure snow depths in the Chinese region of Altay, which consists of both flat and mountainous terrain with mostly grass and cropland land-cover types. Their findings suggest that mountainous terrain, with its steep slopes, is a difficult environment for snow depth measurements at scale with the ATLAS sensor. The best results are achieved in flat terrain, where the MAE and RMSE are 3.19 cm and 4.20 cm, respectively. These errors are very low — but can be explained by the low terrain and absence of tall vegetation/forests — and the authors state that further research is needed into snow depth estimation in forested terrain (Hu et al. 2021).

1.3.4 Geodesy

The radius of the Earth is approximately $6,372 \pm 15$ km, and when adding satellite orbital altitudes of several hundred kilometres to that, it becomes clear that aiming to achieve a vertical accuracy at the decimetre-level requires at least a rudimentary knowledge

of geodesy. Geodesy, or geodetics, is the science of measuring and representing the geometric shape, orientation and gravity field of the Earth. The analyses of this thesis combines data from a sensor at a distance from 500 km above the Earth's surface — moving in relation to the Earth's centre — to data collected at an altitude of 100 m above the terrain. The instruments are so far apart that relative, internal positioning is not possible without the use of GNSS systems, which can introduce errors of several decimetres (King 2009).

Chapter 1. Introduction

Chapter 2

Data acquisition

The thesis focuses on forested areas in Southern Norway, which is mostly covered in snow during winter (fig. 2.1). The winter season normally lasts from late November to early April, with January to March being the period with the most snowfall. The data for the thesis was acquired during the winter season of 2023, between January and April (table 2.5).

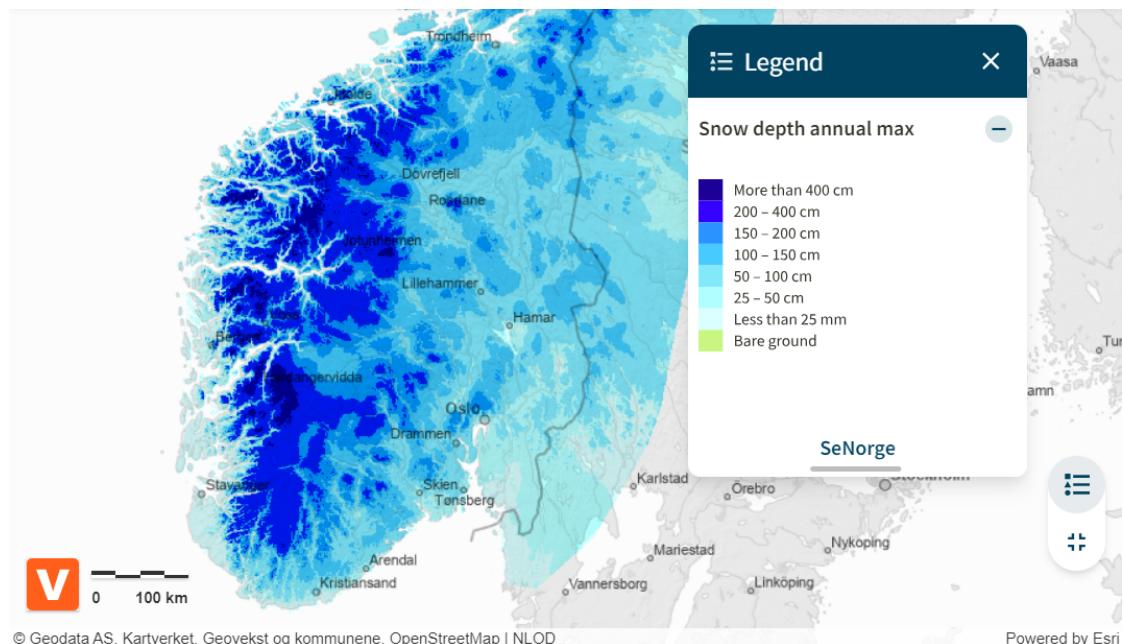


Figure 2.1: Map from SeNorge.no showing the maximum snow depth for Southern Norway in 2023.

2.1 Instrumentation

The following section will describe the instruments used in the acquisition of data for this thesis — The ICESat-2 system and the DJI Zenmuse L1 LiDAR device. A short description of the platforms are included, but the main focus will be on the sensors themselves. Data from other providers are described in section 2.4.

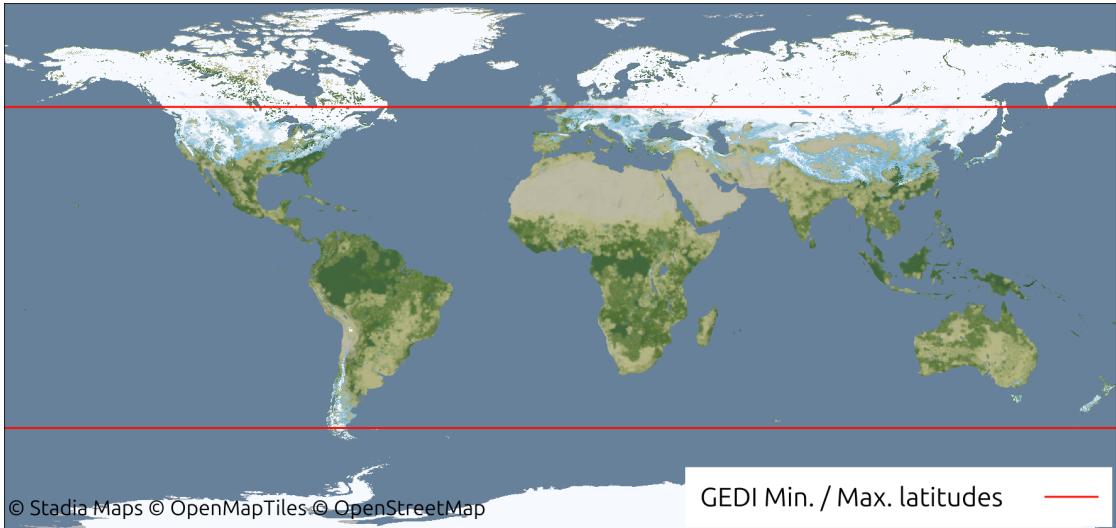


Figure 2.2: Global map showing where snow is expected to be present during winter, and the minimum and maximum latitudes of the GEDI mission for reference. (Snow data from MODIS/NASA.gov, August 2023 and January 2024.)

2.1.1 ICESat-2

All platform and sensor information in this subsection is referenced from the ICESat-2 system description (Thomas A. Neumann et al. 2019), unless otherwise specified. The Ice, Cloud and Land Elevation Satellite-2 (ICESat-2) mission is a part of *NASA’s Earth Observing System* (2024). As the number suggests, it is the second iteration of a mission for this purpose, with the first mission (ICESat) lasting from 2003 to 2009. ICESat-2 consists of a single satellite in a near-polar orbit, carrying one instrument — the Advanced Topographic Laser Altimeter System (ATLAS), which is a photon-counting laser altimeter. Laser altimetry from space-born sensors is not widely used, with the Global Ecosystem Dynamics Investigation (GEDI) mission being the only other operational system besides ICESat-2.

Sensor properties

In the ATLAS instrument, a single pulse of green (532 nm) laser is split into a total of six beams — in three pairs — which are directed toward the Earth surface. The pairs are separated by approximately 3.3 km, in cross-track distance.

In each beam pair, one beam is stronger than the other, meaning that it has more energy. The ratio of energy between then two beams is 1:4. The lower-energy beam is thus referred to as the *weak beam*, while the other is the *strong beam*. The different energy output in turn means that the strong beam nominally yields four times as many photon returns as the weak beam. The beams in each pair are separated with a cross-track distance of 90 m.

The entire sensor can be rotated, and the beams hit the surface with an along-track distance of approximately 2.5 km. The beam tracks can be adjusted laterally when needed, and for the first two operational years of the mission, this was done actively to map as much of the Earth’s land surface as possible. The satellite is still occasionally rotated, which also introduces a possible delta between the planned ground track and the one it actually uses.

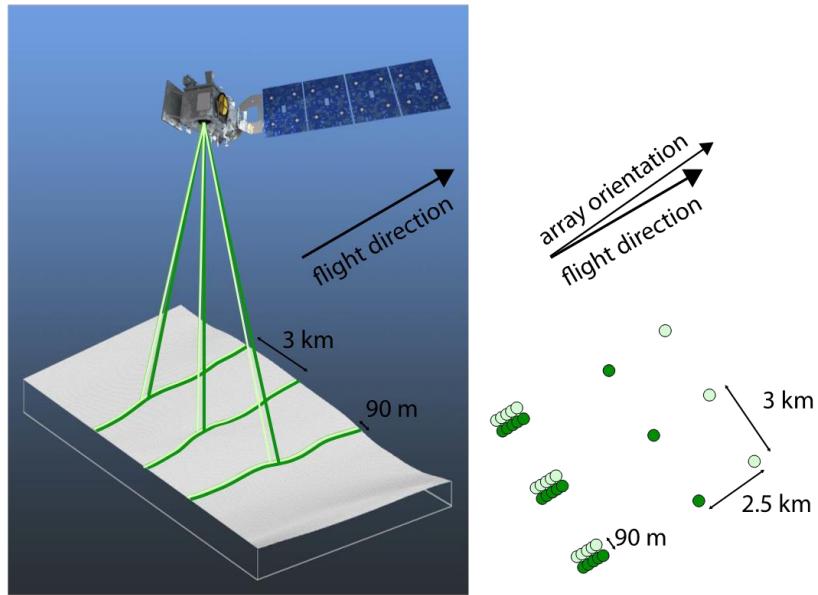


Figure 2.3: Spatial pattern of the ATLAS beams and footprints. From T. A. Neumann et al. (2023)

Platform properties

The ICESat-2 platform is a LEOStar-3 satellite bus, carrying the ATLAS as its sole instrument. It was launched on 15 September 2018 from Vandenberg Space Force Base in California, USA. The planned mission duration was for 3 years, but it is still fully operational (NASA 2022).



Figure 2.4: Artist rendition of the ICESat-2 satellite platform. ©NASA

Table 2.1: Orbital parameters of ICESat-2

Reference system:	Geocentric
Regime:	Low Earth Orbit (LEO)
Nominal altitude:	500 km
Inclination:	92°
Period:	94.22 min
Velocity:	6.9 km/s
Repeat period:	91 days

2.1.2 DJI Zenmuse L1



Figure 2.5: DJI Matrice 300 RTK UAV with DJI Zenmuse L1 on a gimbal mount. (Image from [DJI.com](https://www.dji.com), accessed on 2024-04-09)

Sensor properties

The DJI Zenmuse L1 is a gimbal-mounted LiDAR sensor capable of high accuracy measurements. The laser beam has a wavelength of 905 nm, which is in the Near-Infrared (NIR) range.

Table 2.2: DJI Zenmuse L1 sensor properties (From [dji.com](https://www.dji.com))

Wavelength:	905 nm (NIR)
Maximum Returns Supported:	3
Ranging Accuracy (RMS 1σ) ² :	3 cm @ 100 m
Operating temperature:	-20° C to 50° C
Point rate:	Single return: max. 240,000 pts/s Multiple return: max. 480,000 pts/s

Platform properties

The Zenmuse L1 is only compatible with a specific platform: The DJI Matrice 300 RTK. As the name suggests, this UAV is capable of Real-Time Kinematic (RTK) (or PPK)

positioning. RTK enables up to a centimetre-level accuracy in real time by using either virtual or physical base stations. The flight time per battery pair is ≈ 25 minutes.

Table 2.3: A comparison of the specifications of both the ATLAS and Zenmuse L1 LiDAR sensors.

	ATLAS	Zenmuse L1
LiDAR type	Photon counting	Discrete return
Wavelength	532 nm (Green)	905 nm (NIR)
Beam footprint	13 m	0.04 m
...at altitude	500 km	100 m

2.2 Data acquisition

Near-simultaneous acquisition of data from both the UAV and the ICESat-2 satellite would yield optimal results for comparative analyses, but it also requires meticulous planning and constant monitoring of the weather. This section will describe the different factors affecting the acquisition, and how the actual acquisition was eventually conducted. The site selection was essentially a multi-criteria analysis, with the most important criteria shown in table table 2.4.

Table 2.4: The main criteria affecting the field site selection for ICESat-2 and UAV data acquisition are listed in this table.

Criterion	Type	Requirement	Affects
Wind speed	Hard	< 15 m/s	DJI M300
Temperature	Hard	> -20°C	DJI M300
Precipitation	Hard	Not ongoing	DJI M300 & ICESat-2
4G coverage	Hard	Present	DJI M300
Cloud cover	Hard	< 1 okta	ICESat-2
Snow depth	Hard	> 0.5 m	ICESat-2
Driving distance	Soft	< 3 hrs	Personnel
Daylight	Soft	Present	Personnel

The main challenge in selecting the field sites was timing. The ICESat-2 orbit yields a 91-day exact repeat interval, meaning that each potential site could only be used for one satellite overpass — or two if the ascending and descending tracks intersected there — per season.

The weather affects both instruments in different ways, and if the conditions are not right when the satellite passes the site, the data collection would not yield useful results. The beams of the ATLAS instrument could not penetrate clouds, but clouds are prevalent during the winter season. Meanwhile, the UAV was unable to operate if the wind speed surpasses 15 m/s. Even if it is capable of flying in wind speeds approaching that, it requires more energy to fly in headwind, which in turn reduces the flight time per battery pair.

The weather could potentially also alter the snow properties in just a few hours if the temperature approaches or exceeds the snow melting point at 0 °C. Ensuring similar snow properties in the two resulting datasets therefore depended on either simultaneous acquisition or stable, cold weather in the period in-between. In the latter case, wind could

be accepted at the ICESat-2 acquisition time and cloud cover at the UAV acquisition time. Low temperatures would in turn reduce battery life on the UAV, unless they are stored in a warm environment until used. Weather predictions with enough precision to sufficiently guarantee acquisition are only available a few days in advance.

The Zenmuse L1 sensor relies on either Real-Time Kinematic (RTK) or Post-Processed Kinematic (PPK) to achieve centimetre-level accuracy for the point cloud. Both solutions rely on a base station for corrections, but while PPK requires a physical base station, RTK can utilise a virtual base station through a Networked Transport of RTCM (NTRIP) connection. RTK was considered a far more practical option than PPK in the field work, because it requires virtually no time to set up and less equipment brought to each field site. However, the NTRIP connection relied on either 4G or Wi-Fi network coverage to receive the virtual base station data. This meant that confirmation of network coverage was required before choosing a field site. This was done by using Telenor coverage maps.

Another main factor was the logistics of the acquisition. The UAV and its pilots were located in Oslo, and manually moving the equipment over distances was cumbersome due to large size and heavy weight. Vehicular access to the field site and relatively short driving distance was therefore considered a soft criterion. At the same time, vehicular access is mostly available near people and infrastructure, where UAV flying would not be possible due to the drone operating rules of Norway, without special permission. The default safety distance to populated areas is defined as 150 m. Additional no-fly zones exist around airports and objects relevant to national security.

A high readiness and flexibility in deciding on which field sites to use was therefore paramount. For practical and logistical reasons, this also meant that the study area would need to be close to where the equipment and people were, i.e. within a few hours drive from Oslo.

In addition to all the mentioned factors, the chosen field sites would need to have a snow depth high enough to actually be measurable by the ATLAS instrument and a significant forest canopy structure. Previous research on the vertical accuracy of ICESat-2 measurements indicated that at least 0.5 m, but preferably more than 1.0 m of snow depth would be enough to properly answer the research question.

2.2.1 The actual site selection process

During the winter season, the process to assess field site suitability ended up as a constant monitoring workflow:

1. Check if ICESat-2 ground tracks were planned within reasonable driving distance in the next 14 days, when the long-term weather forecast could indicate suitable conditions
2. Identify potential sites along the tracks with a suitable snow depth and canopy structure, while also being accessible with a vehicle
3. Confirm that the sites have 4G network coverage
4. From 3 days in advance, monitor the (now more accurate) weather forecast to ensure suitable conditions
5. Make practical preparations for field work (charge batteries, make reservations for UAV and vehicle, allocate time for personnel, etc.)

6. Depending on the weather forecast:

- If no cloud cover was forecast at time of overpass: Acquire data with UAV as close to the overpass as practically possible
- If low cloud cover was forecast, remain prepared to go immediately after the overpass in case the clouds did not actually appear.

In the event that a site was deemed unsuitable merely due to the forecast cloud cover, the Cloud Mask data product from EUMETSAT was used at the time of overpass to examine the actual cloud cover. If the forecast was wrong — and no clouds appeared — it could still be possible to visit the site within the following days, as long as all the other conditions were met. Most importantly the temperatures and precipitation would need to be low enough to maintain the same snow depth.

2.2.2 UAV data acquisition

To ensure evenly distributed points inside the study area of each site, the *Terrain Follow* function of the UAV platform was used. This relies on a DTM covering the study area, and allows the UAV to maintain the same distance from the ground during the entire flight. The flying height was set to 100 m. An additional, perpendicular flight was also conducted at each site, which in addition to doubling the points allows different angles through the forest canopy and therefore a better distribution of returns from the terrain.

When the UAV data was acquired, physical measurements of the snow depths were conducted. The measurements were made using a snow probe, and the measurements were registered using a phone camera with geotagging capabilities. These measurements were intended to be used as an additional check of the UAV snow depths.

2.2.3 ICESat-2 data acquisition

ICESat-2 data is made available at the National Snow and Ice Data Center (NSIDC) website 30-45 days after satellite observation. This delay is caused by the processing time needed to acquire precise solutions for the satellite position, in turn eliminating inaccuracies in the photon data. *Quick Look* datasets are made available within the first 3 days, which are less precise, but provide an indication of whether data was successfully acquired or not.

The *Quick Look* data was used as confirmation that the UAV data acquisition had yielded useful control data. If the quick look data had shown data gaps, or low quality data, at the chosen field site, the UAV data would not serve a purpose, and more field trips would have to be made. When the final standard data is released from the NSIDC, the Quick Look data files are removed.

2.3 Field sites

Over a period of three months, between January and April of 2023, a total of four field sites were eventually used to acquire data, as seen in table 2.5. The table also shows that the largest time difference between UAV and ICESat-2 acquisition was approximately 51 hours, at the site in Drammen. During this period, the area saw cold temperatures of approximately -15°C . The eastern track in the Vikerfjell site had a simultaneous acquisition of both UAV and ICESat-2 data.

The same field sites were revisited during the snow-off season, in October 2023. The purpose was to acquire high-resolution snow-off terrain models, which could be used to have more precise estimates of the actual snow depths and thus serve as control data. All of the field sites lie to the west of Oslo (fig. 2.6). The four sites contained varying slope and canopy cover, and had median snow depths above 0.5 m table 2.6.

Table 2.5: The field sites with their geodetic coordinates and times of ICESat-2 and UAV data acquisitions. The NMA DTM is based on ALS surveys (NDH), and the time periods for these surveys are also added. (All times are UTC.)

Field site	Location	ICESat-2	UAV (Snow-on)	UAV (Snow-off)	NMA NDH collection
Drammen	59.7882° N	2023-01-21	2023-01-23	2023-10-03	Jun 2017
	9.9843° E	12:12	15:05	15:40	May-Jul 2022
Hof	59.5191° N	2023-01-21	2023-01-23	2023-10-03	May-Jun 2016
	10.0352° E	12:12	11:08	10:55	Apr-Aug 2021
Jevnaker	60.2416° N	2023-04-19	2023-04-19	2023-10-03	May-Jun 2007
	10.4914° E	20:24	17:58	18:25	May-Jun 2016
Vikerfjell E	60.5021° N	2023-02-19	2023-02-19	2023-10-02	Oct 2008
	9.9773° E	10:48	11:30	12:20	Oct 2016
Vikerfjell W	60.5021° N	2023-02-20	2023-02-19	2023-10-02	Oct 2008
	9.9726° E	23:12	11:30	12:20	Oct 2016

Table 2.6: Descriptive statistics of the field sites, and dates of the NMA NDH data collection.

	Median snow depth	Canopy cover	Median slope	Median canopy height
Drammen	0.59 m	66 %	17 °	10.96 m
Hof	0.49 m	35 %	13 °	9.72 m
Jevnaker	0.45 m	60 %	14 °	10.03 m
Vikerfjell	1.01 m	49 %	10 °	8.02 m

2.3. Field sites

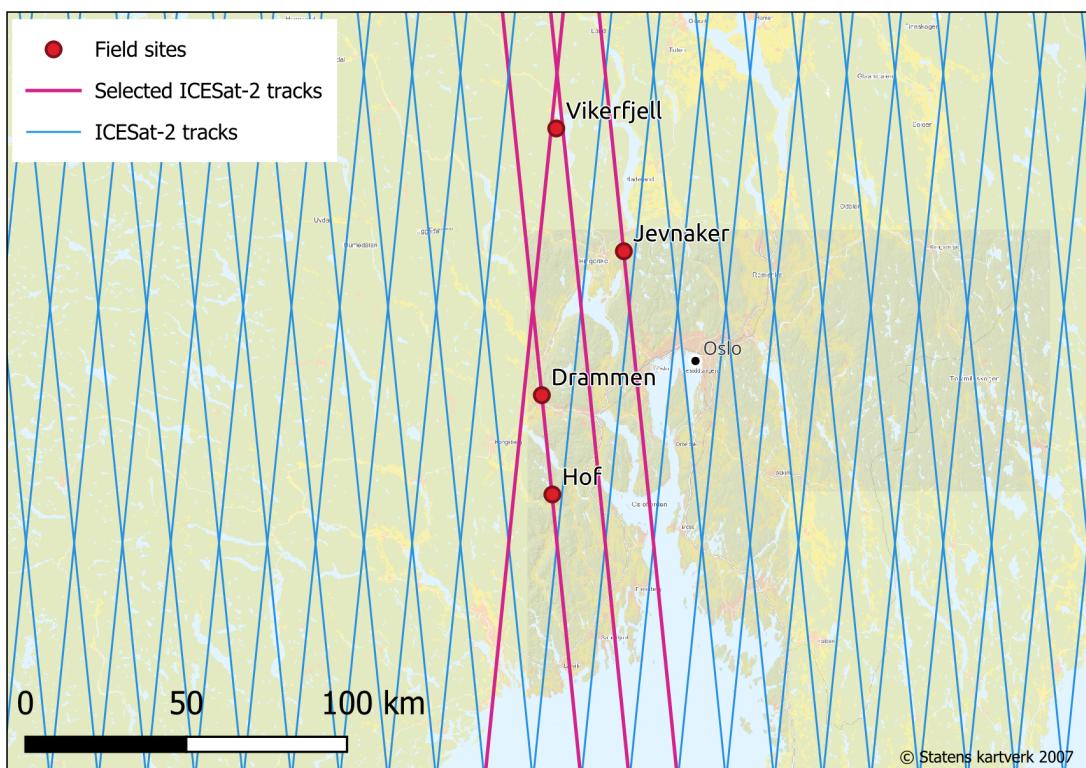


Figure 2.6: Overview map of Southern Norway, with each field site represented by red dots. The purple lines show the ICESat-2 ground tracks used in the thesis, while the blue lines show the other tracks in a single cycle.

2.3.1 Drammen

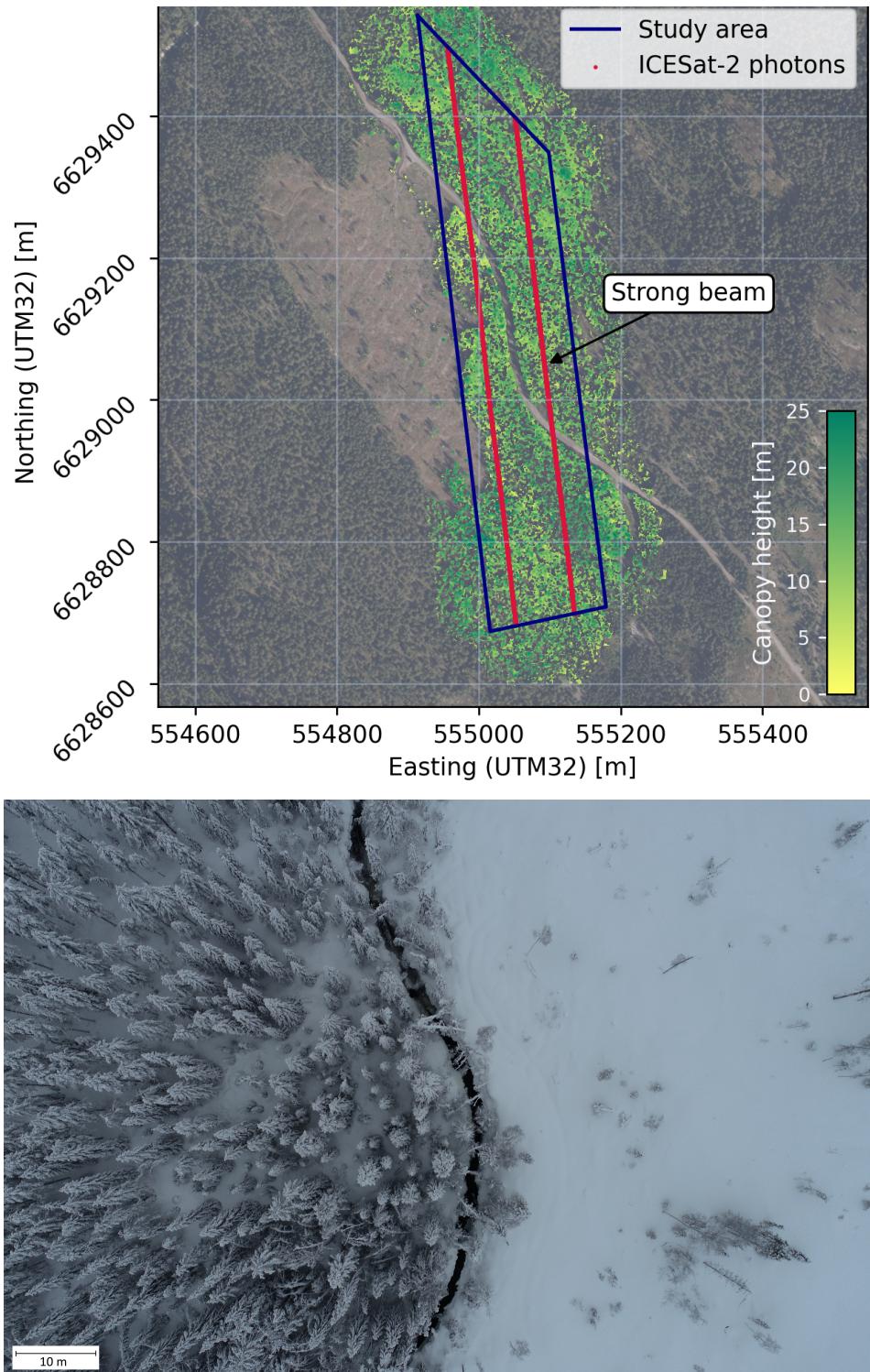


Figure 2.7: Upper: An overview map of the Drammen field site. The boundaries of the study area is shown with a dark blue line, while the ICESat-2 photons are shown in red. The green raster overlay shows the canopy height model, from the UAV data. Lower: Sample image from the UAV flight. The canopy structure **is** covered in snow.

2.3.2 Hof

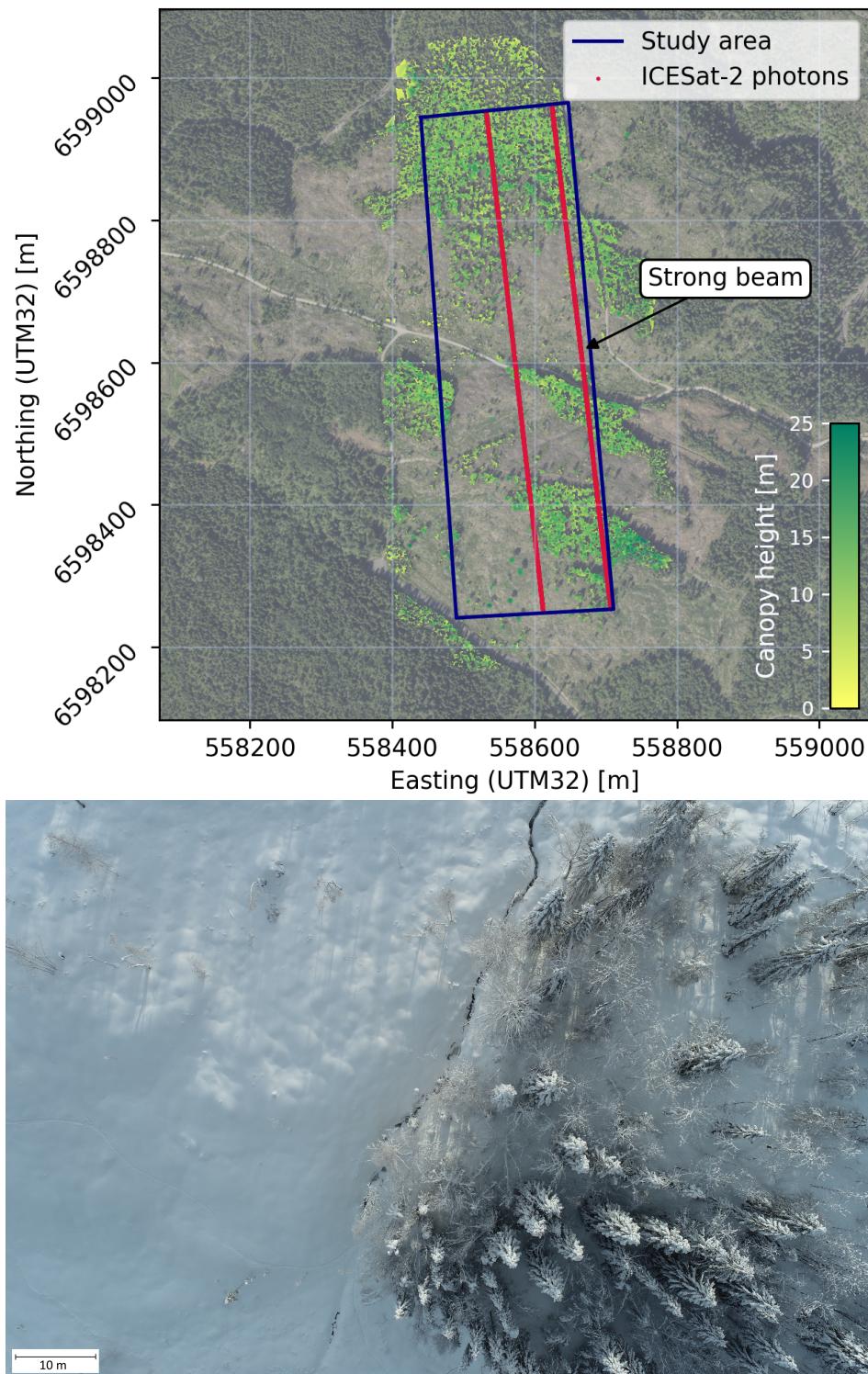


Figure 2.8: Upper: An overview map of the Hof field site. The boundaries of the study area is shown with a dark blue line, while the ICESat-2 photons are shown in red. The green raster overlay shows the canopy height model, from the UAV data. Lower: Sample image from the UAV flight. The canopy structure is covered in snow.

2.3.3 Jevnaker

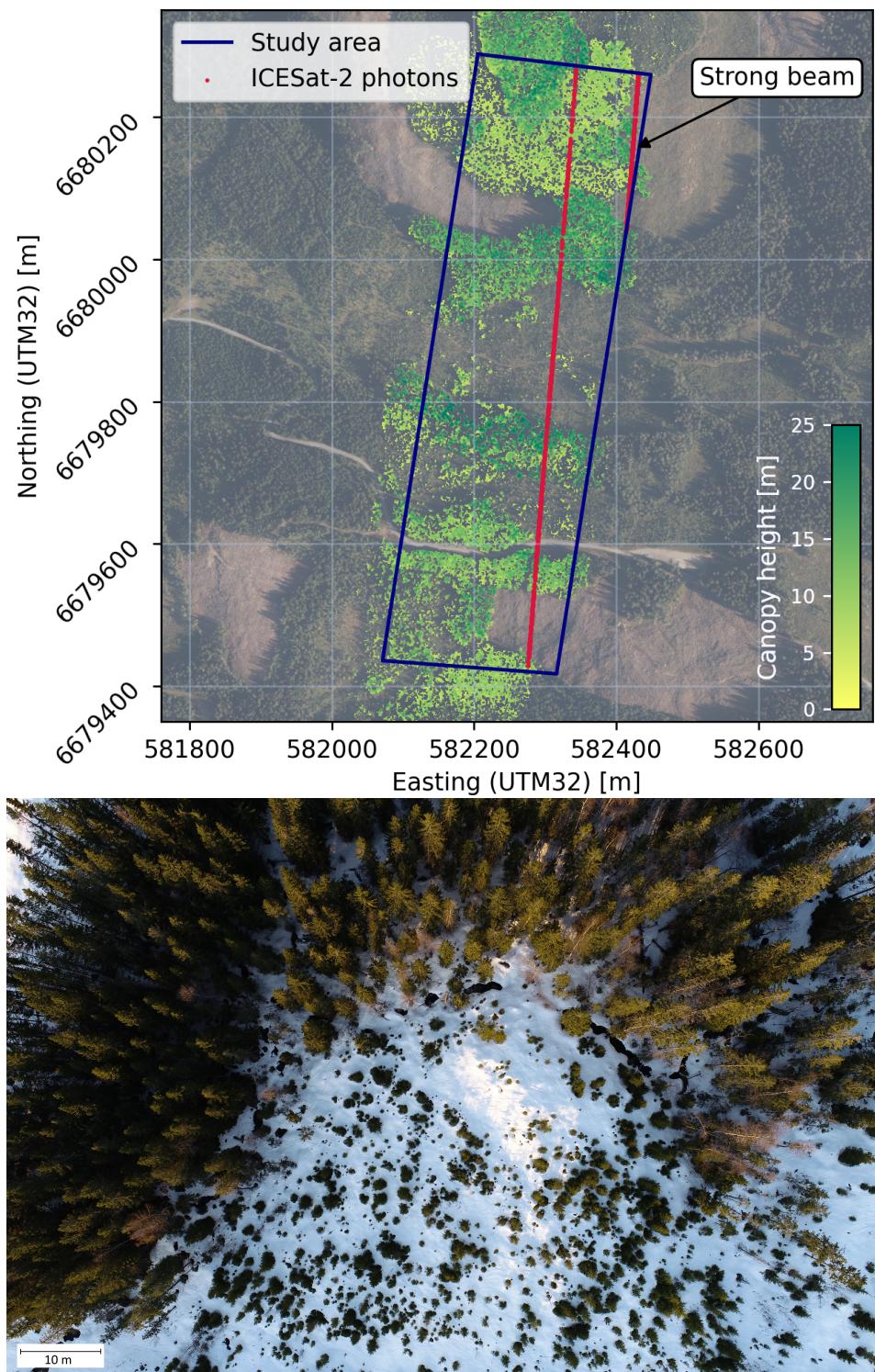


Figure 2.9: Upper: An overview map of the Jevnaker field site. The boundaries of the study area is shown with a dark blue line, while the ICESat-2 photons are shown in red. The green raster overlay shows the canopy height model, from the UAV data. Lower: Sample image from the UAV flight. The canopy structure **is not** covered in snow.

2.3.4 Vikerfjell

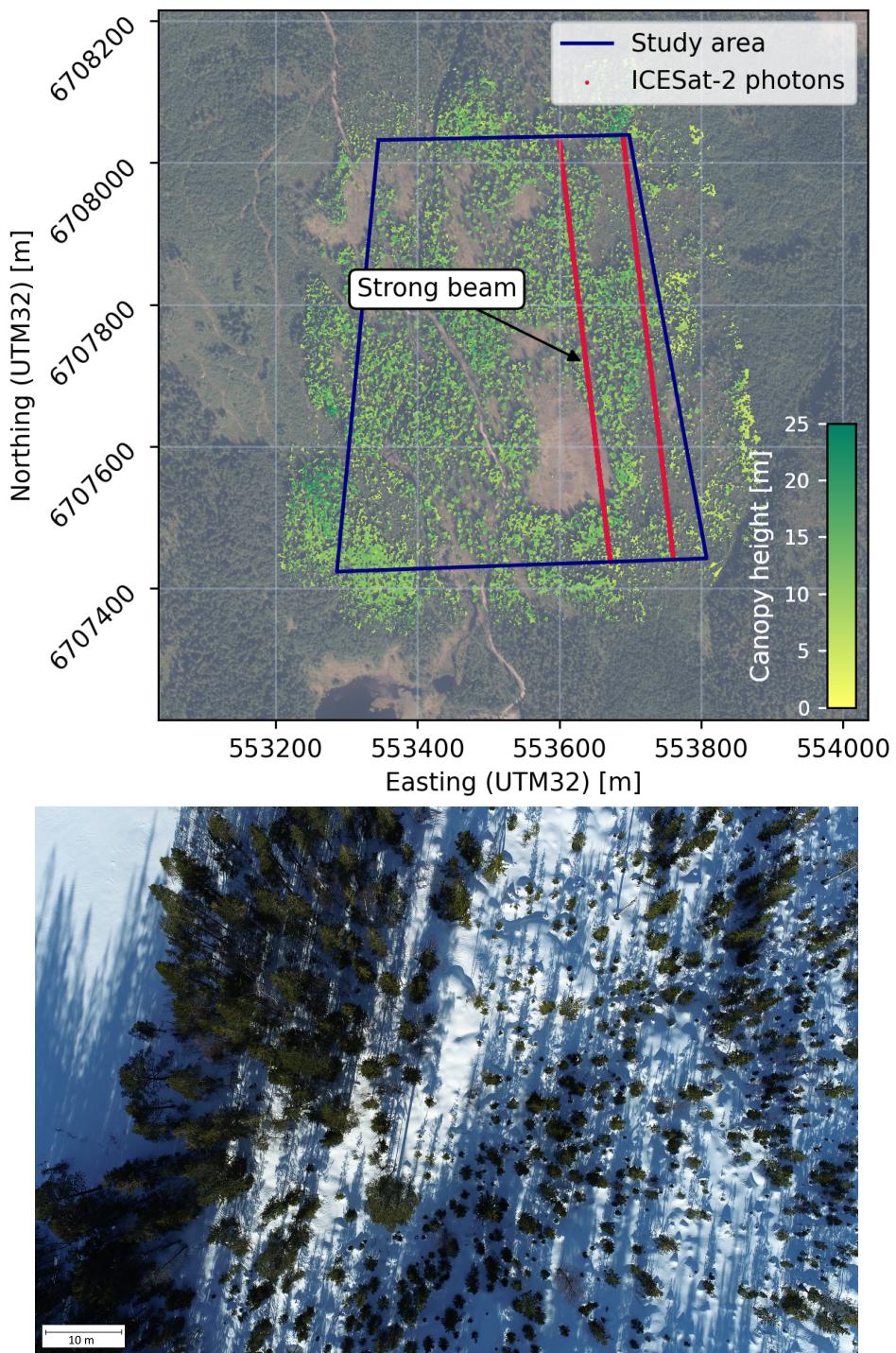


Figure 2.10: Upper: An overview map of the Vikerfjell field site, which had intersecting ICESat-2 tracks. The eastern is shown here. The boundaries of the study area is shown with a dark blue line, while the ICESat-2 photons are shown in red. The green raster overlay shows the canopy height model, from the UAV data. Sample image from the UAV flight. The canopy structure **is not** covered in snow.

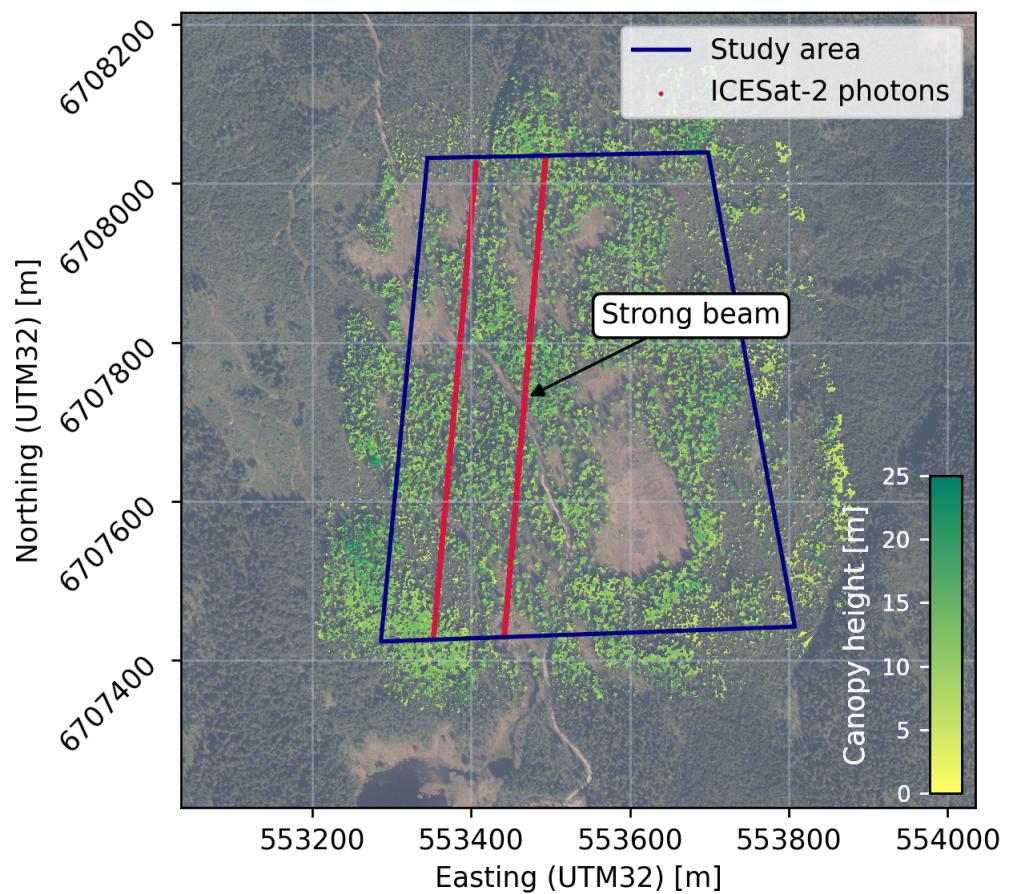


Figure 2.11: An overview map of the Vikerfjell field site, which had intersecting ICESat-2 tracks. The western is shown here. The boundaries of the study area is shown with a dark blue line, while the ICESat-2 photons are shown in red. The green raster overlay shows the canopy height model, from the UAV data.

2.4 Data

The primary data used in this thesis was the data acquired by the ATLAS instrument and the UAV-borne LiDAR sensor. Although both originated from LiDAR sensors, they contain different levels of information. The ATLAS instrument is a photon counting sensor, which essentially means that the data output is just individual returned photons and their two-way travel time. The Zenmuse L1 is a discrete return LiDAR sensor, with every point containing up to three returns. Each registered return also contains information on the reflectivity of the surface it illuminated. The return number information is used to classify ground points.

In addition to the ICESat-2 and UAV data, official elevation models from the Norwegian Mapping Authority (NMA) was used as additional control data. This data included Digital Terrain Model (DTM) and a geoid model. The purpose of using this data was to confirm whether the UAV acquired "ground truth" provided an reasonably accurate representation of the ground, and serve as an indicator of whether the results could be scalable. The NMA DTMs were also necessary to enable use of the UAV terrain following function, which ensures an even distribution of LiDAR points throughout the field site.

2.4.1 ICESat-2 products

The available ICESat-2 mission generates a wide range of products from the photon returns, although the main purpose of the mission is to measure polar sea ice and land ice. A total of 23 different product indicators are listed at the *Data Products* section of the ICESat-2 mission site. This thesis is based on the *Global Geolocated Photon Data* (ATL03) and *Land Water Vegetation Elevation* (ATL08) products. This applies both to the data downloaded directly from the NSIDC and from *SlideRule*. Both products are available in the *Hierarchichal Data Format (HDF)*.

Global Geolocated Photon Data — ATL03

As the name suggests, this product contains the spatio-temporal location of each photon returned to the ATLAS instrument. The photons are classified by signal vs. background, as well as by surface type (land ice, sea ice, land, ocean), including all geophysical corrections (NASA 2024). The photon products are provided in granules, each spanning several minutes of data collection.

Land Water Vegetation Elevation — ATL08

The ATL08 product is more specialised, and contains ground height and canopy surface. If the data permits it, this product also includes canopy height, canopy cover percentage, surface slope and roughness, and apparent reflectance (NASA 2024).

2.4.2 UAV-borne LiDAR point clouds

The result of the UAV data acquisition is a LAS/LAZ point cloud file (fig. 2.12). The points represent individual returns from the laser beam, their three-dimensional coordinates and the intensity of the return.

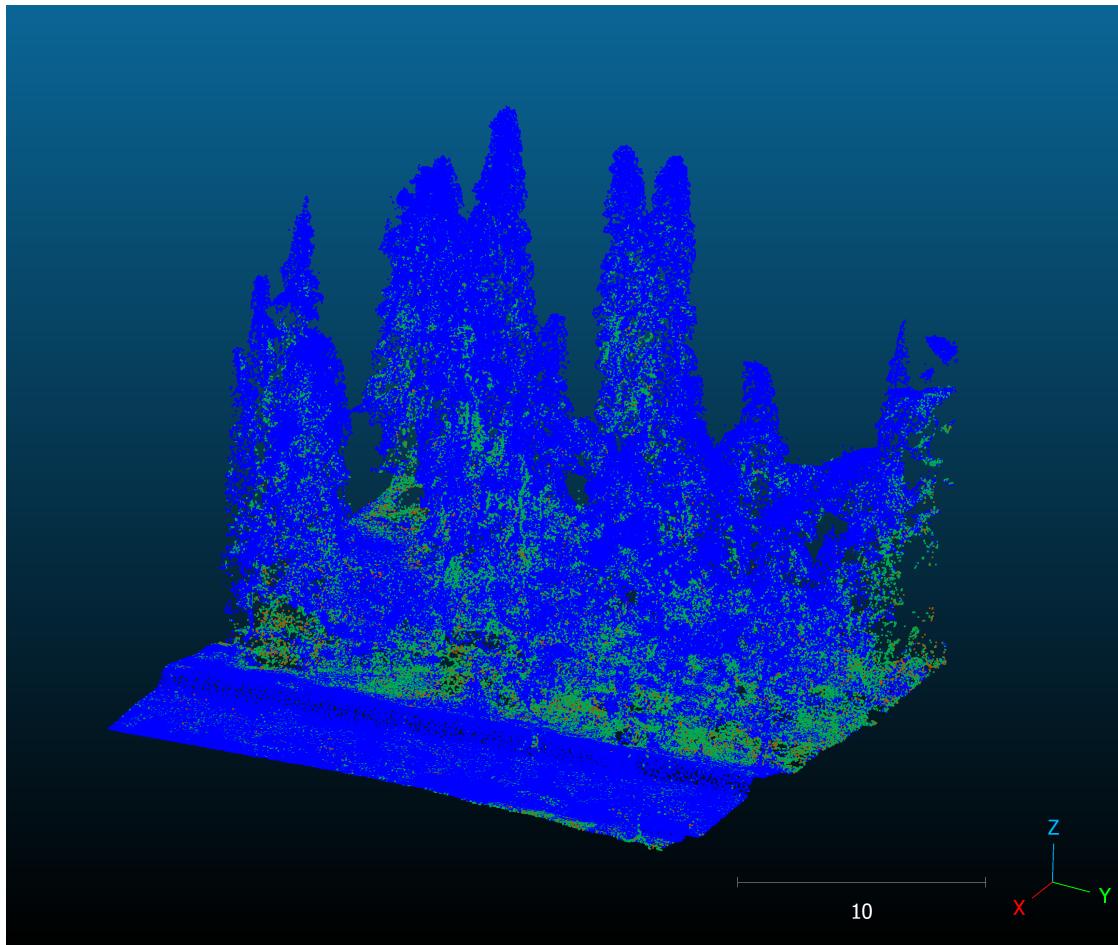


Figure 2.12: Sample point cloud from one of the field sites, showing canopy structure rising vertically, and half of a road passing through. Blue points are first returns. Green are second and red third returns.

2.4.3 Norwegian Mapping Authority (NMA) elevation data

Digital Terrain Model

The data used to calculate snow depths against the ICESat-2 photon heights was a DTM from the NMA, with a 1 m spatial resolution.

Geoid model

The UAV and ICESat-2 provide heights in a vertical datum which differs from the one used by NMA DTM, which is a local datum. These differences are caused by the undulations of the geoid, and therefore a geoid model was used to remove these differences. This also improves the use of the terrain following function on the UAV, which used the NMA DTM as input.

Chapter 3

Data processing

3.1 Software

The software used in this thesis is described in this section. All processing, except the initial post-processing of the UAV point clouds, was handled with open source programming libraries in either Python or R languages. The only exception is the initial post-processing of the UAV point clouds, which had to be done in the DJI Terra software due to the sensor outputs proprietary format. They were then saved to the LAS file format, which is open source and easy to work with (ASPRS 2019).

The actual processing steps, and parameters used, is further described in section 3.2.

3.1.1 DJI Terra

DJI Terra is a licensed software which is part of the same system as the M300 RTK and Zenmuse L1 LiDAR instrument. This software takes the raw point clouds as input — in addition to the IMU data, RTK base station, RTK main antenna, RTK, and radar and IMU calibration files — and converts them into a georeferenced LAS point cloud file. In addition, the software can also perform ground point classification, which was applied during this process. (Ideally, R code would have been used for this classification, because it enables the use of more suitable classification algorithms, but the output from the attempts to use *lidR* produced errors further along, in the rasterisation.)

3.1.2 R

R is a programming language which is particularly suitable for statistical computing and data visualisation, and is available on many platforms (R Core Team 2023). R was also available, along with necessary libraries, on the Department of Geosciences' processing server, which significantly reduced processing time of the computationally heavy LiDAR point clouds.

lidR

The R library *lidR* is a powerful alternative for working with point clouds in the LAS/LAZ format, which makes it possible to use parallel processing to create canopy height models (CHM) and digital elevation models (DEM) (Roussel et al. 2020).

terra

The *terra* library (Hijmans 2024) is useful for working with spatial raster files, like DEMs. Saving the DEM and CHM products to file was done with *terra*.

3.1.3 Python

Most of the processing in this thesis is written in Python. The language is easy to learn, and uses a vast amount of packages, for almost any use case. With a simple, English-like syntax and cross-platform basis it provides a great starting point for working with data science and visualisation (Van Rossum and Drake 2009). Several libraries are used in this thesis:

pandas

pandas (The pandas development team 2020) is the most widely-used library for data analysis in Python. It handles a wide range of tabular data formats, as well as time series and other structured data sets. Data is structured in DataFrames, which in turn consist of Series. *pandas* contains a multitude of classes and functions to analyse big datasets in a fast and flexible way. Pandas also makes it easy to export tables in LaTeX format, which are used in the writing of this thesis.

GeoPandas

GeoPandas is based on Pandas, but adds support for geospatial data and operations (Bossche et al. 2023). Examples include adding buffers and selecting data based on location, which is done in this thesis. GeoPandas uses GeoDataFrames as its basic data structure, which essentially is a Pandas DataFrame with an additional GeoSeries containing geometries (points, lines, polygons, etc.).

xDEM

Although in its early stages of development, xDEM is helpful when working with DEM. In this thesis it is mainly used to extract elevation data from DEMs at the photon centre points, but it also provides functions for co-registration of DEMs with a method developed by Nuth and A. Käab (2011).

SlideRule

Photons with the *Yet Another Photon Classifier* (YAPC) algorithm applied are downloaded from a web service API, through this *SlideRule* (Swinski et al. 2024) python client. YAPC (Sutterley 2022) is the basis of one of the filtering methods evaluated in this thesis.

rasterstats

The rasterstats library (Perry 2024) makes it possible to extract raster information in several ways, including inside a polygon (zonal statistics). The ICESat-2 photons are defined with centre point geometry — while the photons have a surface footprint diameter of approximately 13 meters — and this library was helpful in getting DEM and CHM statistics inside the entire footprint.

3.2 Processing

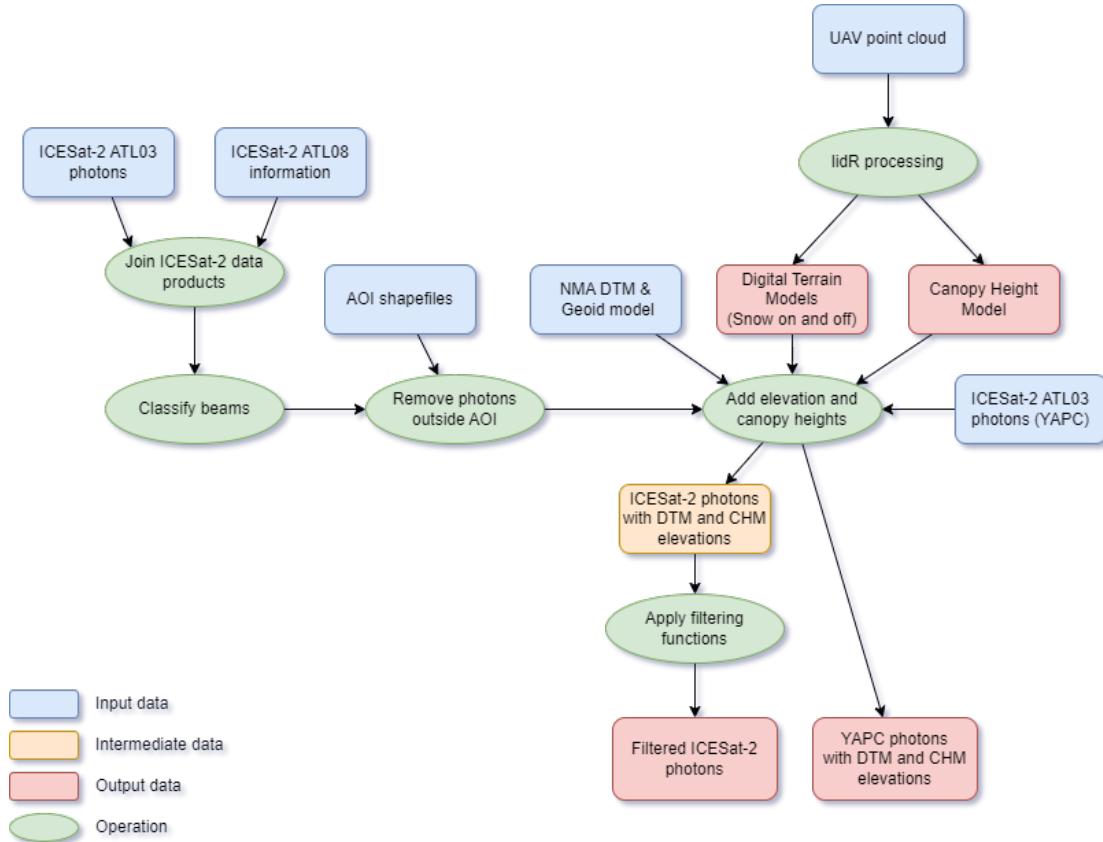


Figure 3.1: Flow chart describing the main workflow of the data processing.

3.2.1 UAV data

In order to create the ground truth elevation and canopy height models, the UAV data had to be reworked from point clouds into suitable raster formats for later use with the ICESat-2 and Norwegian Mapping Authority (NMA) data.

Flight line alignment

The data collected from the UAV for this thesis was initially in a proprietary raw data format. To process and georeference it, the first step involved using the DJI Terra software for processing of the flight data and point clouds. The final output was a single LAS file, which was further converted to the LAZ format for a higher compression level of approximately 90 %. The conversion process utilised LASzip, which is an open-source tool developed specifically for this purpose (Isenburg 2017).

DTM and CHM generation

Point clouds are useful for discrete data, with information about each LiDAR return, but further processing relied on continuous data in the form of raster files. To generate the rasters, interpolation methods were applied to the point clouds. The point clouds for all

locations were then used to generate Digital Terrain Model (DTM) and Canopy Height Model (CHM). Point clouds are very useful for storing information about the LiDAR laser returns, but the discrete nature of the data makes it necessary to create continuous datasets, like rasters, to enable data interpolation at a later stage. The comparable datasets from NMA — like the geoid model and DTMs — are also in a GeoTIFF raster format.

The *R* library *lidR* was very useful to achieve this. It enables parallel-processing on several of its built-in calculation algorithms, and could therefore work through each file in relatively short time. Still, even when using one of the university’s high-performance servers the processing of each file took approximately one hour. The lidR script first takes an input LAS/LAZ file, and each point’s x, y and z values — in addition to classification, number of returns, and return number — was selected for further processing. The script then returns the LAS file information. Example output is shown below, which describes a point cloud from the Drammen site:

```
class      : LAS (v1.2 format 3)
memory     : 10.7 Gb
extent     : 554808.1, 555267.5, 6628592, 6629570 (xmin,xmax,ymin,ymax)
coord. ref. : WGS 84 / UTM zone 32N
area       : 299760 m2
points     : 318.3 million points
density    : 1061.85 points/m2
density    : 765.59 pulses/m2
```

The next step involves applying a rasterisation function on the data, which generates the DTM. This function allows the use of different algorithms to create DTMs, but the default Triangulated irregular network (TIN) method was applied here. As the name suggests, this method creates a triangulated network of the points classified as ground points, which makes it possible to get the height of any arbitrary point inside the network, using a bivariate function. The spatial resolution of the DTMs was set to 0.5 m, as a compromise between maintaining some benefits of high resolution input and requiring large computational resources to process further.

The point clouds were also used to create the Canopy Height Model (CHM), which were integral for answering the research questions of the thesis. The process is similar, but with a few important differences. Firstly, because some trees had been cut between the two data acquisitions (with and without snow), only the point clouds with snow cover could be used to create useful CHMs. Otherwise the models would not contain the same canopy structure as the ICESat-2 data. Second, three steps had to be completed before creating the output model:

1. The heights need to be normalised. In short, this process subtracts the ground height from each point, meaning that height values are now above-ground, instead of the elevation.
2. The next step was to rasterise the canopy. This is similar to when rasterising the terrain, but a higher resolution of 0.2 m was used, in addition to another algorithm, more suited for the large height variability of a canopy model compared to a ground model.
3. Lastly, before exporting the canopy model, a lower height threshold of 2 m was applied. Any structure above that height should be counted as forest canopy, which the thesis is focusing on. (No man-made structures were present in the study sites.)

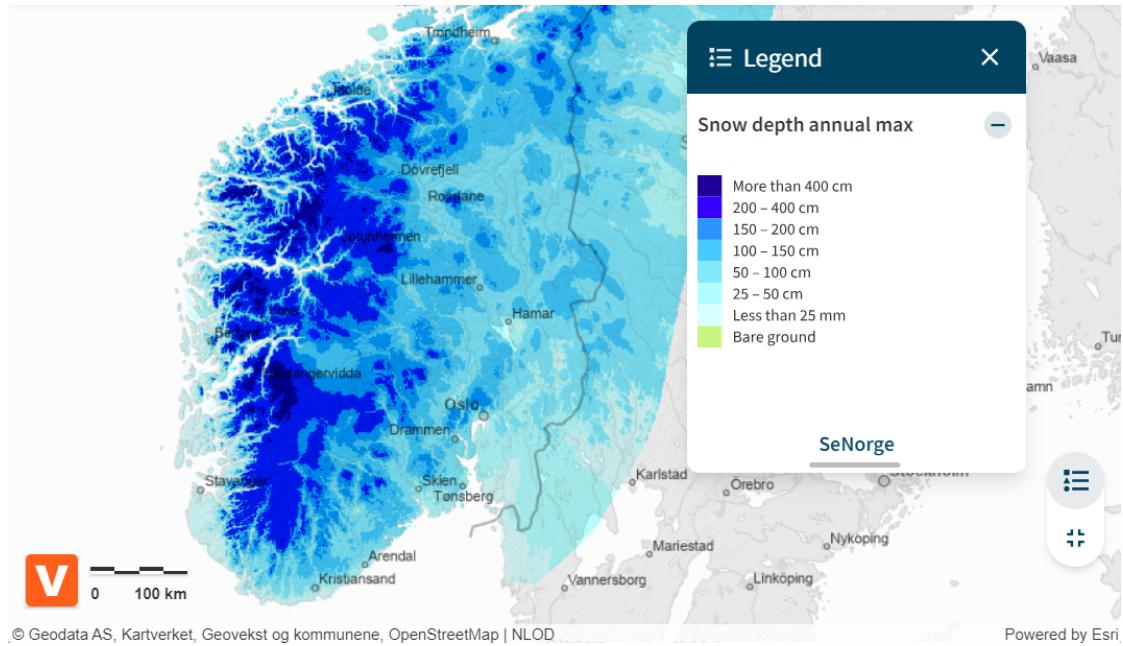


Figure 3.2: Example DTM output from the lidR processing, showing the Drammen field site from a top-down angle.

The CHM was finally exported. All files, both DTM and CHM, were stored in the *GeoTIFF* format, which is a widely-used Open Geospatial Consortium standard format (OGC 2019).

The R processing script was applied to each point cloud file using a simple GNU bash script (see Appendix A), which also creates a log file with all console output and timings.

3.2.2 Data structuring and metrics extraction

The data was now ready for the main processing part. The data sources are in several different formats, making it beneficial to restructure all relevant information into one dataset. *Pandas* DataFrames are easy to work with, and the Pandas library makes it easy to import from — and export to — a large variety of file formats. The entire processing code is available in Appendix B

The first step involved defining the regions of interest, and the corresponding time periods for data collection. An example is provided below, showing how the Hof site was defined:

```
'hof': {
  'time': ['2023-01-21',
            '2023-01-22'],
  'onTS': '20230123',
  'offTS': '20231003',
  'bbox': [10.03298,
          59.51847,
          10.03757,
          59.52495]
},
```

The data structure is that of a regular Python dictionary, where the key is the name of the field site. Within that, the 'time' key contains the date range where ICESat-2 had acquired data there. 'onTS' and 'offTS' are the dates when the UAV data was acquired. 'bbox' contains a list of geodetic coordinates defining a bounding box around the field site. All further processing either inserted or updated the contents of this dictionary.

The next step was to download the data from the NSIDC. The SNOWDEPTH project has developed code specifically for this purpose (Treichler and Mazzolini 2024), which takes the bounding box and collection dates from the dictionary as input. The function downloads the granules within the specified spatio-temporal range from the NSIDC as HDF files, and this was done for both the ATL03 and ATL08 products. The Yet Another Photon Classifier (YAPC) dataset was downloaded separately — via the SlideRule library's functions — because it is provided through their API. The function used to download SlideRule YAPC data with its parameters is attached in Appendix B, line 150.

The SNOWDEPTH code also contains functionality to extract the relevant data fields from the ATL03 and ATL08 HDF files (Treichler and Mazzolini 2024) and structure them as Pandas DataFrames, which were used after all the requested granules were downloaded. These DataFrames served as the basis for the further restructuring and processing. ATL03 files were loaded first, before data from the corresponding ATL08 files were added to the DataFrames.

Simple restructuring and interpretation followed:

- The timestamps of each photon were in an ATLAS epoch format (T. A. Neumann et al. 2023), which has a different origin than both the GPS and UNIX timestamps — while Pandas interprets timestamps from the UNIX origin (1 Jan 1970)
- Counting the photons in each beam to identify which beam was the strong beam and weak beam, respectively
- Remove photons with low signal confidence, by using a threshold on the values provided within the ATL03 data
- Remove photons outside the UAV flight area
- Add elevation information from the geoid model, DEMs and CHMs, by spatial interpolation

3.2.3 Canopy height model

A function made to apply information from the canopy height model to each photon footprint. Implements the *zonal_stats* function from the rasterstats library (Perry 2024), but also adds calculation of canopy cover. The canopy cover is calculated by defining a horizontal buffer around each photon centre point — corresponding to the photon footprint — and counting each cell in the CHM raster within the buffer. The canopy cover is the ratio of cells containing a non-null value, to the total cell count within the buffer. It returns 1 if all cells contain tree canopy, and 0 if there is no canopy present.

The code for this function is attached in Appendix B, line 54.

3.2.4 Photon filtering methods

The following subsection will describe the three photon filtering methods used to remove the ICESat-2 photons that likely do not represent the surface. The *threshold validation*

3.2. Processing

and *Point grouping* methods are developed for this thesis, while the *YAPC* method uses the algorithm with the same name. The point grouping and YAPC methods are similar in principle, but YAPC is a far more complex algorithm.

Nearest neighbour

The principle of the *nearest neighbor* filtering method is that if there are enough photons within a defined distance of each other, they are more likely to represent a surface than not. The function lets the user define a lower threshold for the number of photon returns necessary within a buffer, and the spatial extent of that buffer. This filtering is applied to the input photons with an inner function, before a rolling median is applied to the remaining photons. As with the *threshold validation* method, the values used in this function was found by trial and error until the results were optimal on one site, before testing the same values to all sites. The code for this function is attached in Appendix B, line 88.

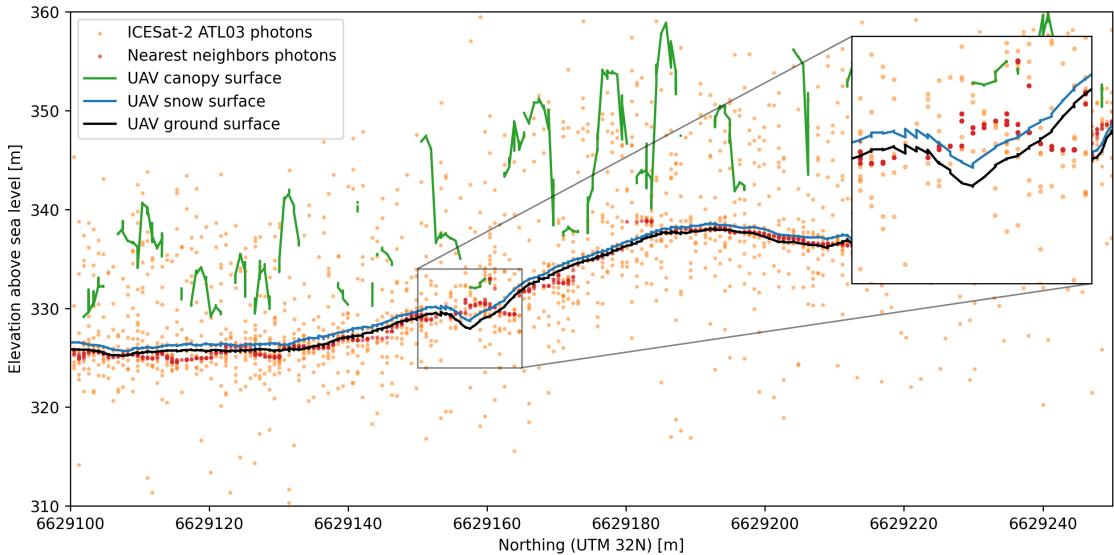


Figure 3.3: The effect of the nearest neighbour filtering method is shown in this plot. The ICESat-2 ATL03 photons (orange) show a large vertical spread, while the filtered photons (red) appear much closer to the UAV measured snow surface. The UAV canopy surface (green) is also displayed.

Threshold validation

The *threshold validation* filtering method was developed while working on this thesis. It takes an array of snow depth values as input, which represent the differences between photon heights and the terrain height in the same horizontal plane, i.e. snow depths. As some of these photon heights do not represent the snow surface — but rather forest canopies, stray photons, etc. — the purpose of this method is to filter unreasonable snow depths for that dataset.

This is achieved by an iterative function which starts with a subset of the 30 % lowest values in the dataset, disregarding negative values. For each iteration, an additional 5 % values are added, until that 5 % increase corresponds to a an increase of the maximum value that exceeds a pre-defined limit (See fig. 3.4).

This limit was set to a default value of 0.15 m in the function, which was found by simple trial and error on the ICESat-2 snow depths from one of the study sites. Several limits were tested until the median snow depth corresponded closely to that of the UAV snow depths. After testing it on the other sites, this margin turned out to work equally well there, regardless of the differences in terrain slope or canopy structure.

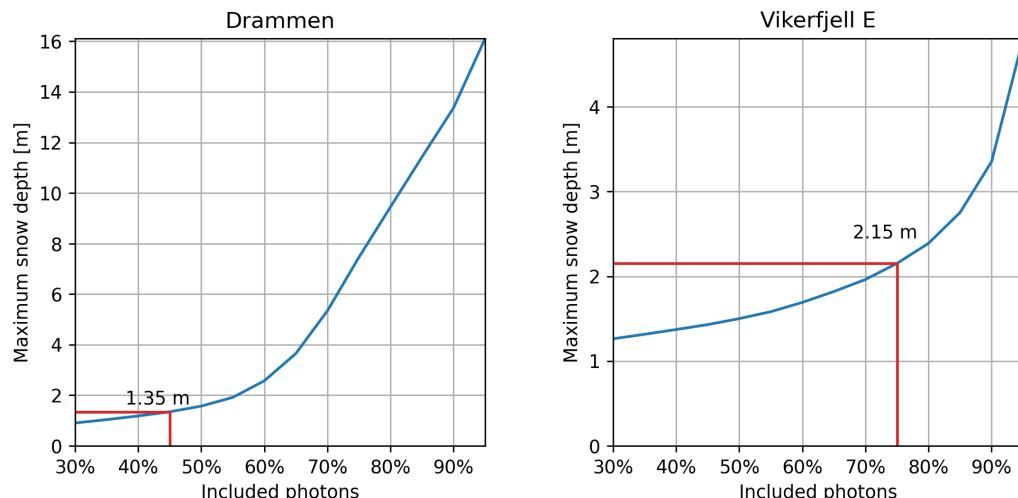


Figure 3.4: A visualisation of how the maximum snow depth in a subset increases when more photons are included, when applying the *threshold validation* filtering method to the photons from the Drammen and Vikerfjell East field sites. The red line shows where the slope increases beyond the defined limit (0.15 m / 5 % photons), and what the maximum snow depth in the subset is at that point. Snow depths below this threshold are considered valid by the function.

The iteration continues until this limit is exceeded, and the maximum snow depth value in the subset at that iteration is set as the validation threshold for the entire input dataset. The function returns a new array, containing the boolean value *True* where the input value was below the threshold, and *False* if they are above it. The original indices of the input array are kept, making it easy to concatenate the returned data into the original GeoDataFrame.

A downside to this method is that it requires the input values to be normalised before input, meaning that some form of "truth" dataset must be available in addition to the ICESat-2 data. In this thesis, the 1 m resolution DTM from NMA was used.

The function code is attached in Appendix B, line 124

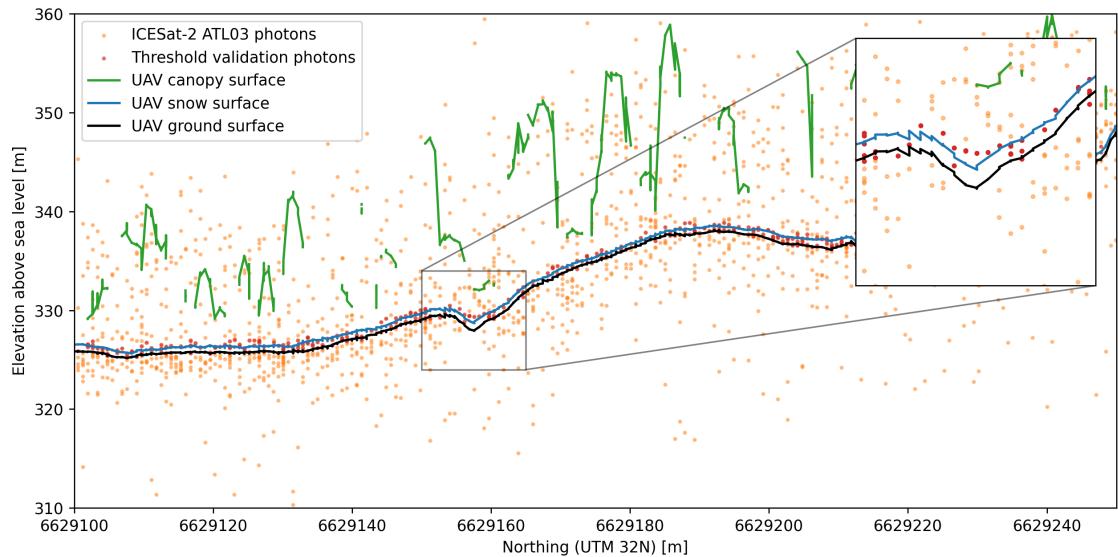


Figure 3.5: The effect of the threshold validation filtering method on a sample subset from the Drammen field site. The ICESat-2 ATL03 photons (orange) show a large vertical spread, while the filtered photons (red) appear much closer to the UAV measured snow surface. The UAV canopy surface (green) is also displayed.

Yet Another Photon Classifier

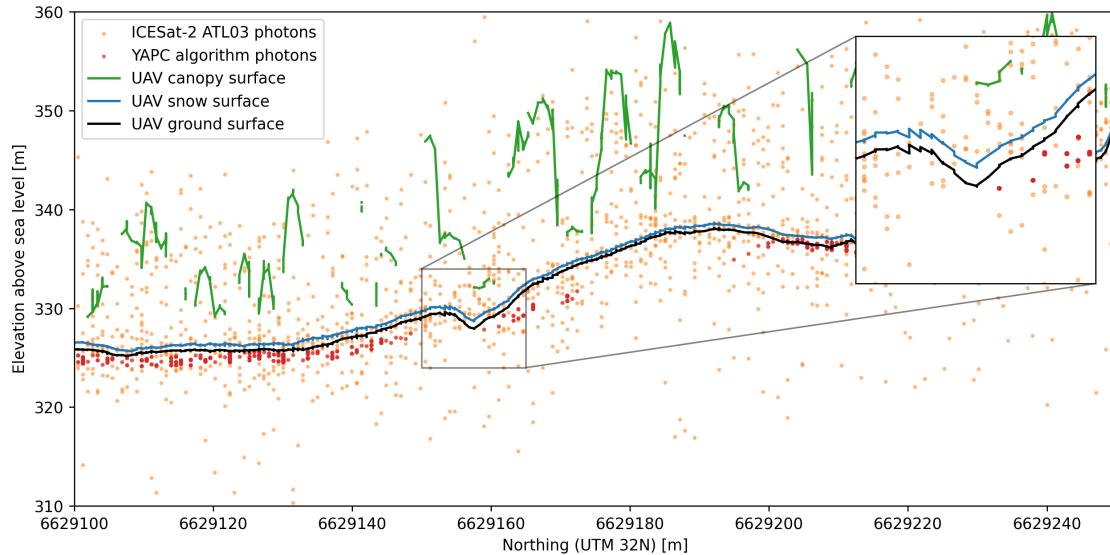


Figure 3.6: The effect of the filtering by applying the YAPC algorithm on a sample subset from the Hof site. The ICESat-2 ATL03 photons (orange) show a large vertical spread, while the filtered photons (red) appear much closer to the UAV measured snow surface. The UAV canopy surface (green) is also displayed.

This method uses the Yet Another Photon Classifier (YAPC) algorithm, which is developed by Jeff Lee at NASA’s Goddard Space Flight Center (Sutterley 2023). It is a pure implementation of already-existing functionality, provided by the Python *SlideRule* library, described in section 3.1.3.

The algorithm calculates a score for each photon, which is based on the number of nearest neighbours and a three-dimensional buffer. This is similar to what is done in the point grouping method above, but the calculation is more complex. It also allows the user to set a new threshold without recalculating anything, just by allowing a lower or higher score in further processing.

3.2.5 Snow depth calculations

Snow depths are calculated by simply subtracting the ground surface height at a location from the measured snow surface height. For the control data, the UAV snow-on and snow-off DEMs were used, while the NMA DTMs were subtracted from the ICESat-2 photon heights.

3.2.6 Accuracy metrics

The accuracy of the photon classification and ICESat-2 snow depth estimates are assessed by comparing the ICESat-2 heights to the UAV heights. The metrics used to analyse the accuracy include Bias, Mean Absolute Error (MAE) and Root Mean Square Error

(RMSE):

$$Bias = \frac{\sum \Delta H}{n} \quad (3.1)$$

$$RMSE = \sqrt{\frac{\Delta H^2}{n}} \quad (3.2)$$

$$MAE = \frac{\sum |\Delta H|}{n} \quad (3.3)$$

$$Relative\ Bias = \frac{Bias}{Mean(H_{SD})} \cdot 100 \quad (3.4)$$

$$Relative\ RMSE = \frac{RMSE}{Mean(H_{SD})} \cdot 100 \quad (3.5)$$

Where ΔH is the difference between the ICESat-2 photon height and the UAV surface height in the same coordinates in the horizontal plane. H_{SD} is the difference between the snow-on and snow-off surfaces from the UAV data, which is the closest we can get to a "ground truth". n is the number of ATLAS photons present within the area of interest.

Bias

A systematic, vertical skew of values in either positive or negative direction is referred to as bias. When comparing several different field sites — each with their own characteristics — the bias can function as an indicator of whether errors stem from the system, method or subject. If it possible to determine what circumstances (i.e. sensor settings, measuring environment, method, terrain features) are linked to a certain change in bias, this bias can be adjusted for in further analyses.

Mean Absolute Error

The mean absolute error (MAE) is the mean of the errors of the ICESat-2 photons, compared to the UAV snow surface, disregarding the direction of the error.

Root Mean Square Error

The Root Mean Square Error (RMSE) describes the difference between the ICESat-2 and UAV measured snow surfaces, by calculating the standard deviation of the residuals. Low values indicates a good fit.

The relative RMSE is calculated as a percentage, showing the RMSE relative to the mean snow depth as measured by the UAV, which in turn is the difference between the snow-on and snow-off surfaces from the UAV.

3.2.7 Canopy cover vs. bias correlation

To assess whether the canopy cover in each field site affects the snow depth measurement errors (bias), a Pearson correlation coefficient r is calculated between these two variables. This coefficient can be used to indicate whether there is a linear correlation between two quantitative variables (Moore, McCabe and Craig 2012).

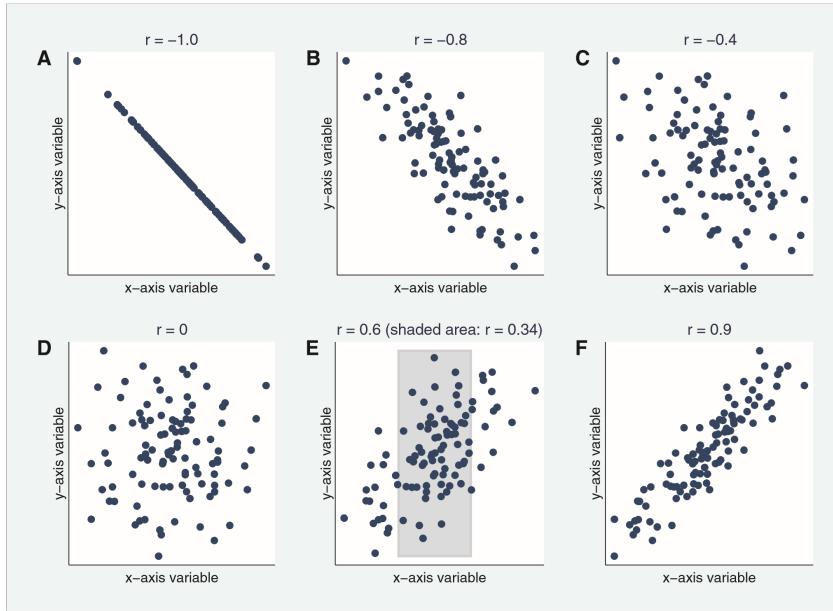


Figure 3.7: Scatter plots showing different distributions of points along both axes, and their corresponding Pearson correlation coefficients (r). The scatter resembles a straight line when r approaches either $|1|$. In sub-figure E, the effect of the chosen range is visible, with an r of 0.34 for the data within the shaded box, and 0.60 when all the data is counted. (From Schober, Boer and Schwarte (2018))

The Pearson correlation coefficient can be expressed as the mean of the products of the standard scores of the two variables x and y , with n observations:

$$r_{xy} = \frac{1}{(n-1)} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{s_x} \right) \left(\frac{y_i - \bar{y}}{s_y} \right) \quad (3.6)$$

The returned values $-1 \leq r \leq 1$ are indicative of either a positive or negative linear correlation. A high $|r|$ indicates a strong correlation, while values closer to 0 indicate little to no correlation (fig. 3.7). It is important to note that correlation does not equal causation, meaning that a high correlation value should not be interpreted definitively as a causal relationship between the two variables. Conversely, a low r can be interpreted as a lack of both correlation and causation, as long as certain conditions are met (Moore, McCabe and Craig 2012): Both variables must be quantitative and their relationship linear. An examination of the scatter plot, and not just the returned r value, can provide insight into whether there could be other correlations between the data.

Chapter 4

Results

The processing results in a set of statistics and metrics, aimed at answering the research questions of the thesis. In addition to tables of statistics, plots will be shown in this chapter to describe the different results.

The first research question addresses the ability of the ATLAS instrument to measure snow surface heights in forested terrain. The results aimed at answering this question will be shown in the first section. The second section goes deeper into the performance of the filtering methods.

A master table with all the error statistics is attached as Appendix C

4.1 Snow depth estimates

The snow depth figures (figs. 4.1 to 4.5) show the resulting snow surfaces from the filtering methods, in addition to the original ICESat-2 photons and UAV canopy surfaces. The general impression from these figures is that the *nearest neighbor* method produces a spotty result, leaving large gaps along the entire transect (fig. 4.2). The *Yet Another Photon Classifier (YAPC)* filtering yields continuous surfaces along the transect, but often underestimates the snow surface, and even lies lower than the snow-off terrain. The snow surface from the *threshold validation* method consistently lies above the terrain, closer to the UAV snow depths. It also appears to be less affected by vegetation (fig. 4.1) than the other two methods. Steep terrain appears to introduce larger errors in all methods. Flat terrain with no measured forest canopy yields a surface with relatively high accuracy regardless of method (fig. 4.5).

4.1.1 Drammen

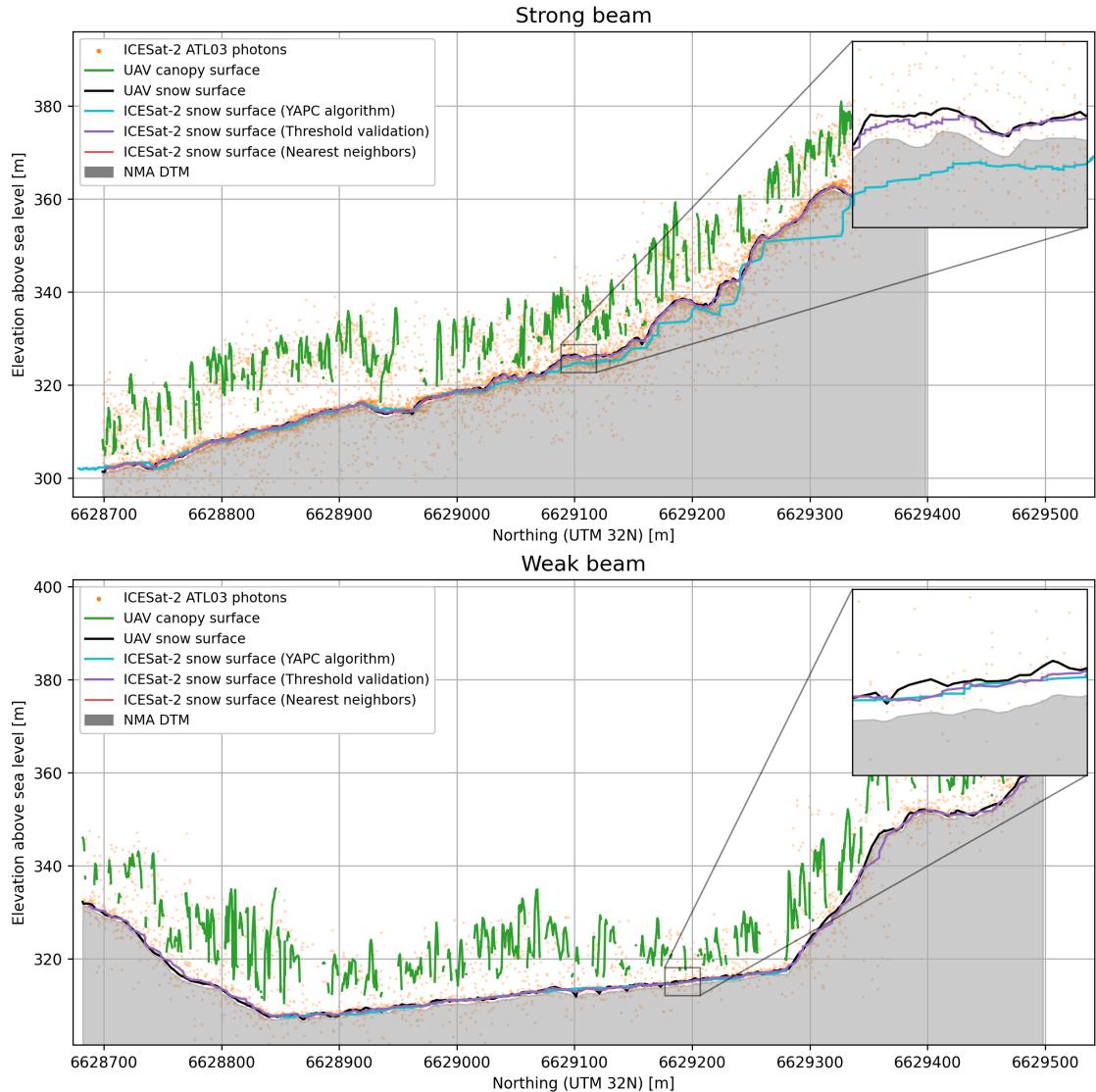


Figure 4.1: Snow depths as measured by the UAV and the two beams of ICESat-2 at the Drammen site. ICESat-2 photons have been filtered with three different methods. The DTM from NMA is shown in solid grey.

4.1.2 Hof

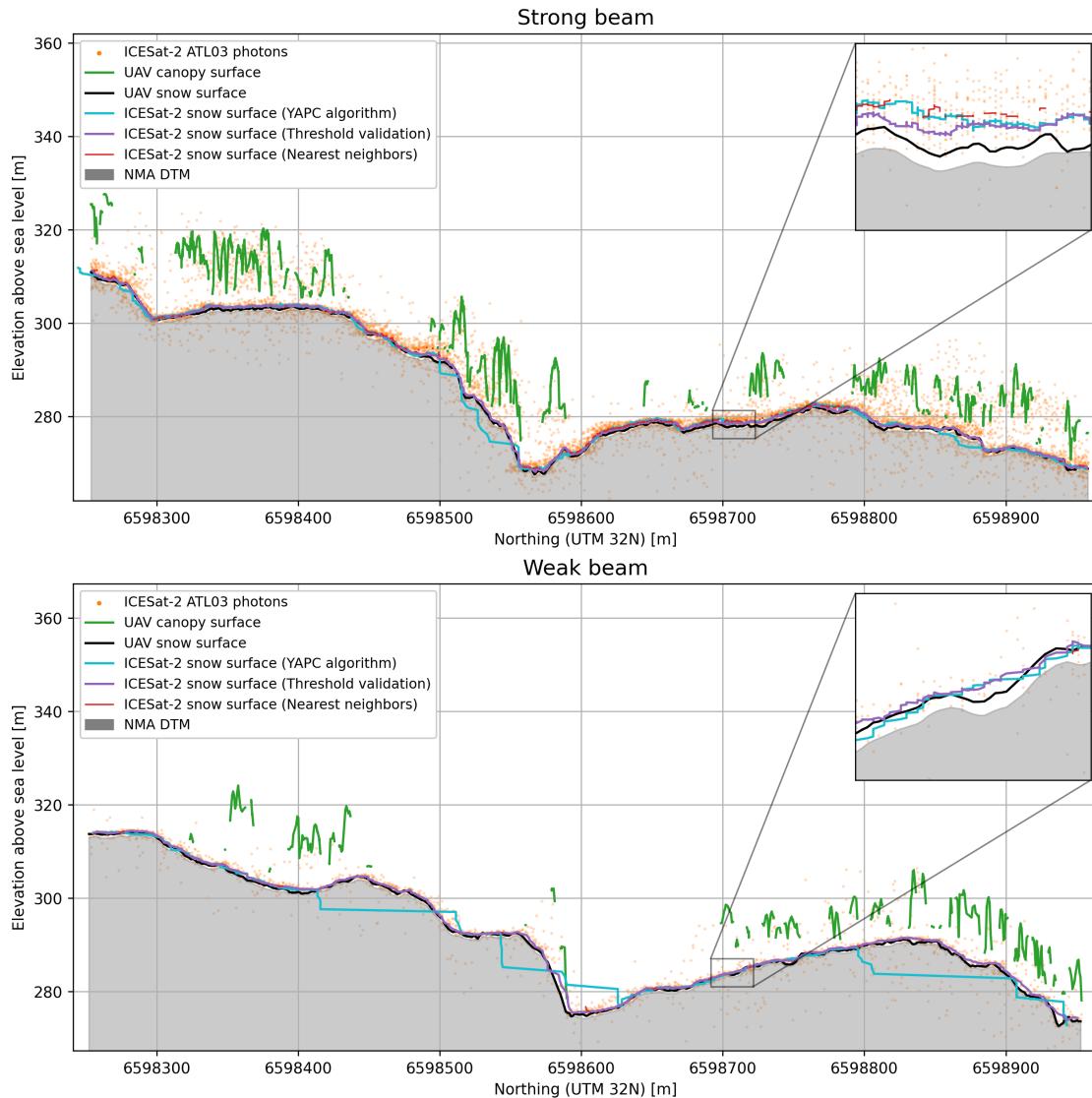


Figure 4.2: Snow depths as measured by the UAV and the two beams of ICESat-2 at the Hof site. ICESat-2 photons have been filtered with three different methods. The DTM from NMA is shown in solid grey.

4.1.3 Jevnaker

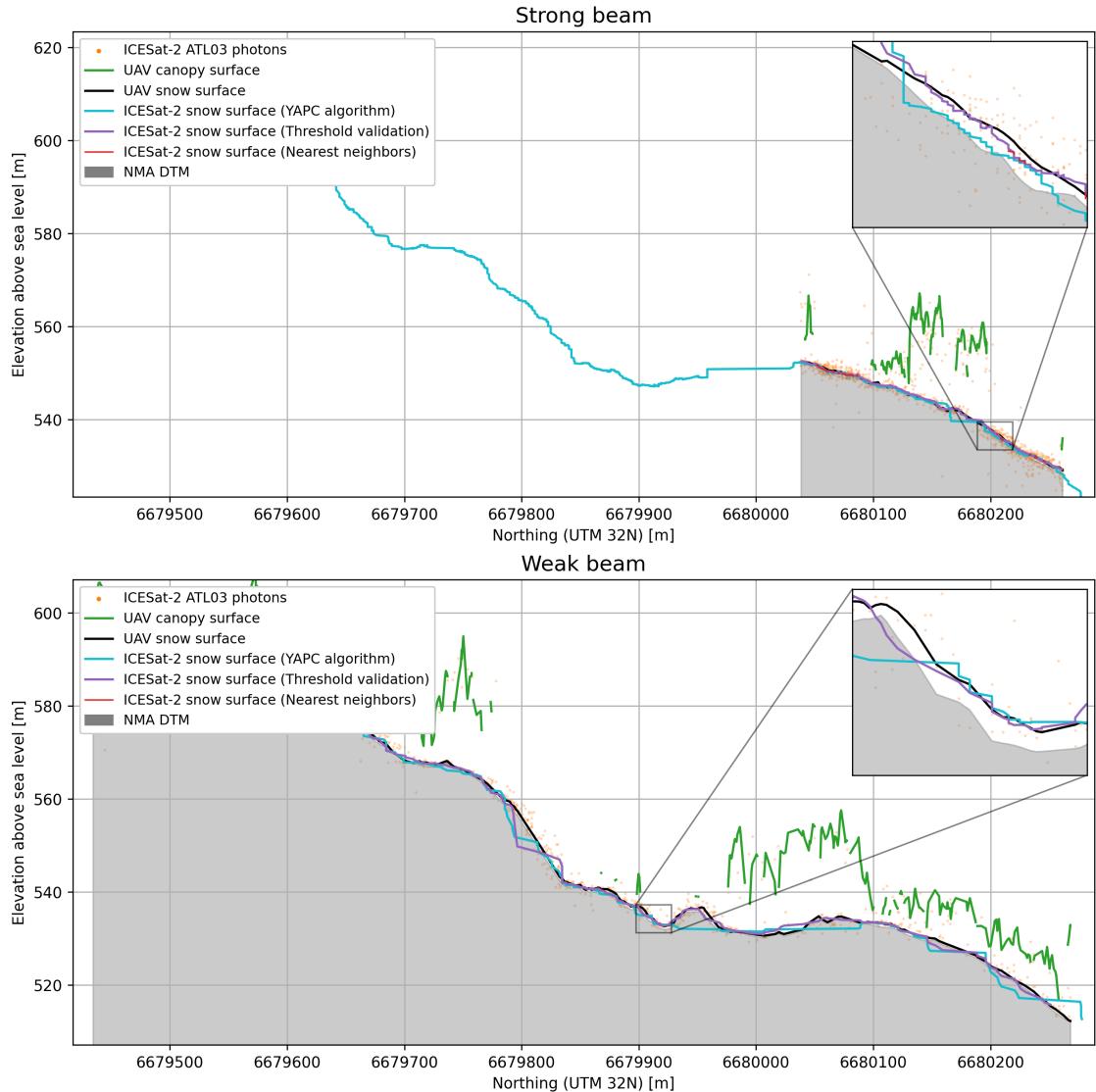


Figure 4.3: Snow depths as measured by the UAV and the two beams of ICESat-2 at the Jevnaker site. ICESat-2 photons have been filtered with three different methods. Note the strong beam being about 1/4 the length of the weak beam, due to low spatial overlap with the UAV data. The DTM from NMA is shown in solid grey.

4.1.4 Vikerfjell East

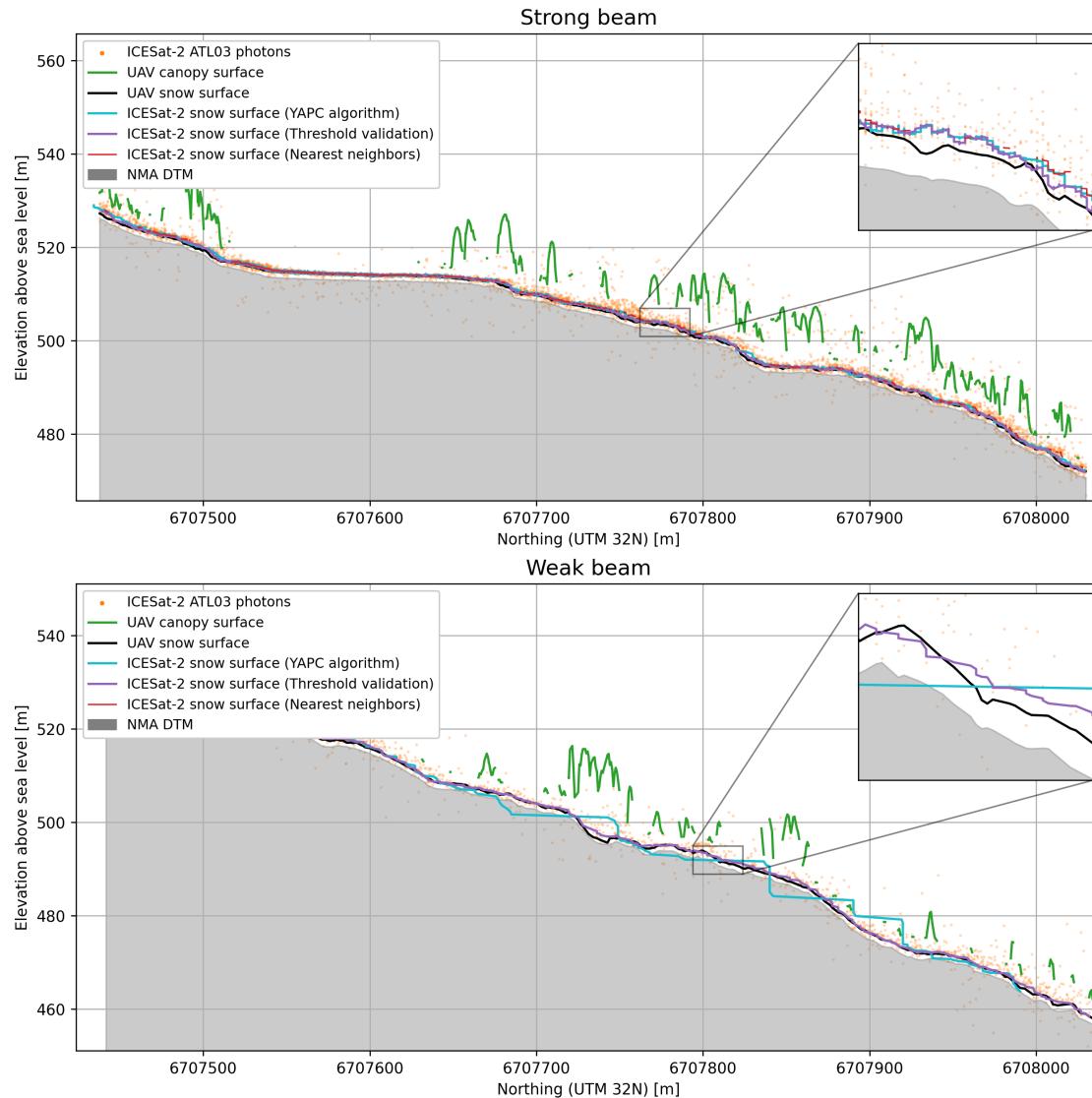


Figure 4.4: Snow depths as measured by the UAV and the two beams of ICESat-2 at the eastern part of the Vikerfjell site. ICESat-2 photons have been filtered with three different methods. The DTM from NMA is shown in solid grey.

4.1.5 Vikerfjell West

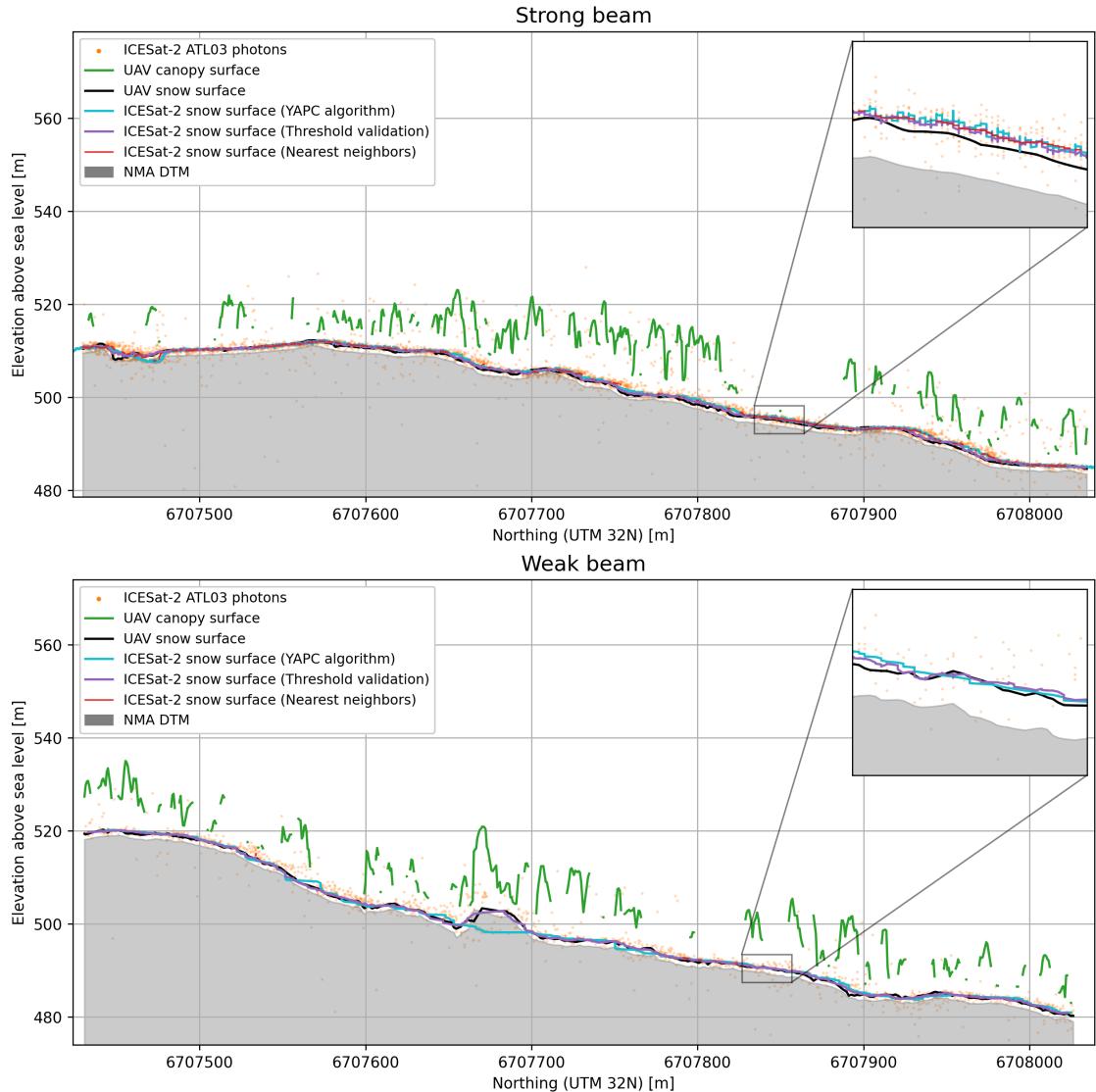


Figure 4.5: Snow depths as measured by the UAV and the two beams of ICESat-2 at the western part of the Vikerfjell site. ICESat-2 photons have been filtered with three different methods. The DTM from NMA is shown in solid grey.

4.2 Performance of filtering methods

This section will contain plots and tables describing how each filtering method performs. In fig. 4.6, the bias The overall results (table 4.1) show that the output from the *threshold validation* method consistently has the lowest RMSE (0.48 m), compared to the nearest neighbor (0.72 m) and YAPC (0.59 m). However, the bias is lowest for the YAPC algorithm at 0.08 m overall. *Threshold validation* shows a bias of 0.14 m, while the nearest neighbor method is at 0.22 m.

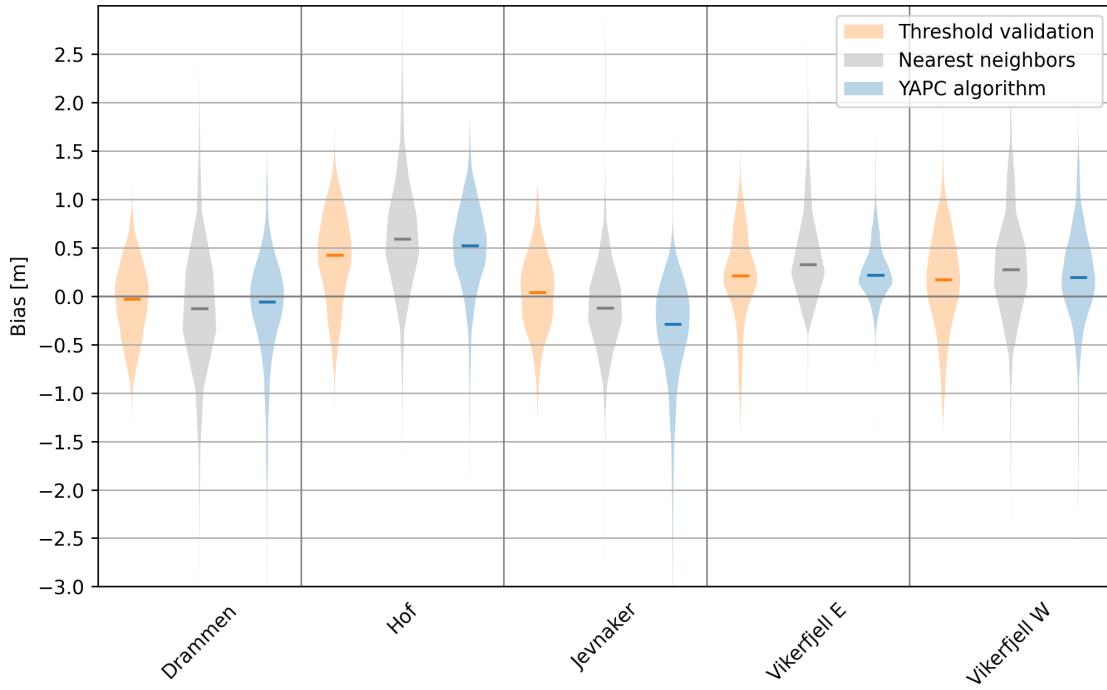


Figure 4.6: The distribution of the vertical measurement bias of the photons after each filtering method is applied, per field site. Median bias is indicated by the lines.

Chapter 4. Results

Table 4.1: Quality metrics per filtering method, with both beams, across all the field sites. r (canopy) is the correlation coefficient between measurement bias and canopy cover. Photons kept after filtering is also shown (not applicable for YAPC).

Method	Bias	MAE	RMSE	Rel. Bias	Rel. RMSE	r (canopy)	Photons
Unfiltered	0.97 m	2.09 m	3.91 m	169%	662%	0.18	100.0%
Threshold validation	0.14 m	0.39 m	0.48 m	22%	76%	0.09	54.3%
Nearest neighbors	0.22 m	0.30 m	0.72 m	30%	115%	0.07	60.9%
YAPC algorithm	0.08 m	0.44 m	0.59 m	19%	78%	0.00	N/A

Table 4.2: Quality metrics per filtering method, with only strong beams, across all the field sites. r (canopy) is the correlation coefficient between measurement bias and canopy cover. Photons kept after filtering is also shown (not applicable for YAPC).

Method	Bias	MAE	RMSE	Rel. Bias	Rel. RMSE	r (canopy)	Photons
Unfiltered	0.98 m	2.09 m	3.97 m	176%	702%	0.19	100.0%
Threshold validation	0.15 m	0.38 m	0.47 m	23%	77%	0.09	56.2%
Nearest neighbors	0.22 m	0.35 m	0.57 m	29%	92%	0.10	74.9%
YAPC algorithm	0.08 m	0.44 m	0.59 m	20%	77%	-0.01	N/A

Table 4.3: Quality metrics per filtering method, with only weak beams, across all the field sites. r (canopy) is the correlation coefficient between measurement bias and canopy cover. Photons kept after filtering is also shown (not applicable for YAPC).

Method	Bias	MAE	RMSE	Rel. Bias	Rel. RMSE	r (canopy)	Photons
Unfiltered	0.89 m	2.10 m	3.76 m	147%	614%	0.14	100.0%
Threshold validation	0.12 m	0.42 m	0.51 m	21%	78%	0.12	50.5%
Nearest neighbors	0.23 m	0.17 m	0.92 m	32%	136%	0.06	26.4%
YAPC algorithm	0.05 m	0.47 m	0.66 m	6%	100%	0.03	N/A

4.3 Impact of canopy cover on measurement bias

Overall, the results show little to no indication of linear correlation between canopy cover and snow depth measurement bias, regardless of the filtering method used.

As seen in section 4.3, the canopy cover distribution varies between the sites, meaning that the data points are distributed differently along the x-axis. But when combining that with the y-axis, which shows the measurement bias, there is no visible pattern. The Pearson correlation coefficients are all $|r| < 0.3$, indicating at best a low correlation. For some filtering methods they are as low as $|r| < 0.05$, which means no correlation. There are no visible patterns in the scatter plots either, which could hint at other types of correlation.

In table 4.4, the RMSEs have been calculated on photons with forest canopy within their footprint, and on photons from vegetation-free terrain, separately. These results show an overall increase in RMSEs of 0.21 m when the photon footprint contains forest canopy.

4.3. Impact of canopy cover on measurement bias

Table 4.4: Comparisons of the RMSE values on the snow depth measurements within the canopy and in vegetation-free terrain, on photons filtered with the threshold validation method.

Field site	RMSE with canopy	RMSE without canopy	Difference
Overall	0.54 m	0.48 m	0.05 m
Drammen	0.37 m	0.41 m	-0.04 m
Hof	0.66 m	0.57 m	0.09 m
Jevnaker	0.41 m	0.38 m	0.03 m
Vikerfjell E	0.58 m	0.44 m	0.14 m
Vikerfjell W	0.55 m	0.48 m	0.07 m

4.3.1 Per field site

Table 4.5: Quality metrics of the filtering methods, for each individual site. Photons kept after filtering is also shown (not applicable for YAPC).

Area	Method	Bias	MAE	RMSE	Rel. Bias	Rel. RMSE	Photons
Drammen	Unfiltered	1.77 m	3.91 m	6.38 m	300%	1,081%	100.0%
	Threshold validation	-0.05 m	0.32 m	0.39 m	-8%	66%	33.3%
	Nearest neighbors	-0.12 m	0.23 m	0.68 m	-21%	115%	48.8%
	YAPC algorithm	-0.15 m	0.41 m	0.64 m	-26%	108%	N/A
Hof	Unfiltered	1.23 m	2.34 m	4.09 m	246%	820%	100.0%
	Threshold validation	0.39 m	0.51 m	0.60 m	78%	121%	49.9%
	Nearest neighbors	0.62 m	0.47 m	0.80 m	125%	161%	66.3%
	YAPC algorithm	0.48 m	0.58 m	0.68 m	100%	141%	N/A
Jevnaker	Unfiltered	0.89 m	1.79 m	3.88 m	200%	872%	100.0%
	Threshold validation	0.02 m	0.31 m	0.39 m	5%	87%	45.3%
	Nearest neighbors	-0.12 m	0.13 m	0.65 m	-27%	147%	38.7%
	YAPC algorithm	-0.38 m	0.49 m	0.71 m	-23%	43%	N/A
Vikerfjell E	Unfiltered	0.64 m	1.19 m	2.31 m	65%	235%	100.0%
	Threshold validation	0.20 m	0.40 m	0.50 m	21%	50%	70.1%
	Nearest neighbors	0.43 m	0.34 m	0.82 m	44%	84%	74.7%
	YAPC algorithm	0.25 m	0.30 m	0.39 m	26%	39%	N/A
Vikerfjell W	Unfiltered	0.32 m	1.24 m	2.87 m	33%	299%	100.0%
	Threshold validation	0.16 m	0.41 m	0.51 m	16%	53%	72.7%
	Nearest neighbors	0.30 m	0.35 m	0.66 m	31%	69%	75.7%
	YAPC algorithm	0.19 m	0.41 m	0.56 m	19%	58%	N/A

Drammen

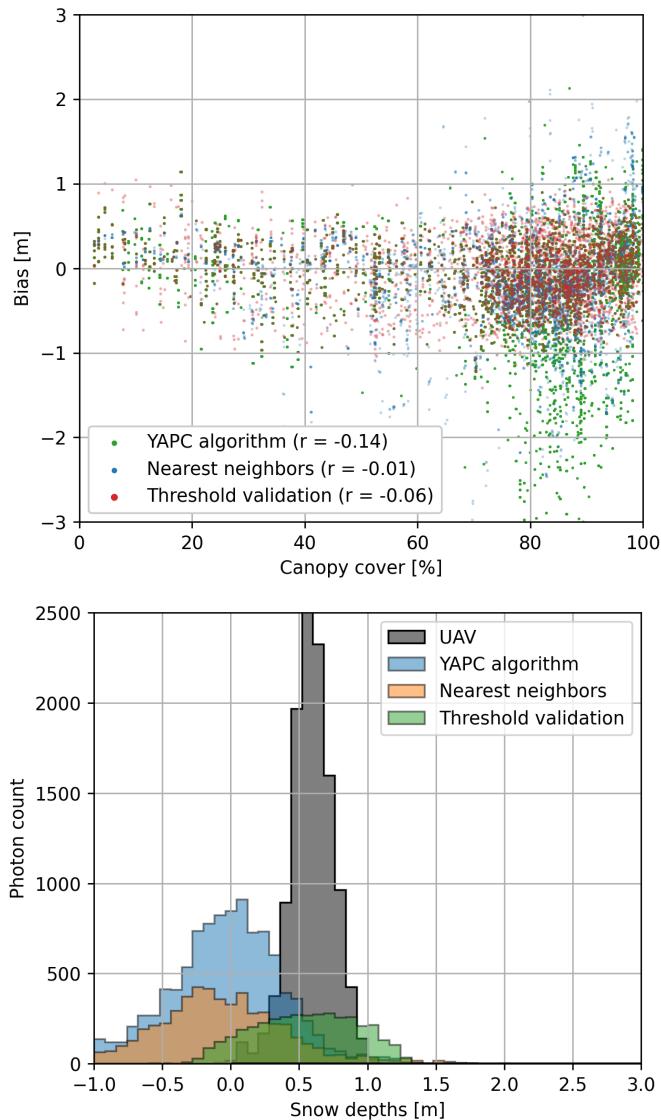


Figure 4.7: Upper: Scatter plot with measurement bias from all the filtering methods used, shown in individual colours, versus the canopy cover inside the photon footprint. A clear linear correlation would be present if the points aligned to a 45° line ($|x| = |y|$). No such pattern is visible for any of the methods. Lower: The distribution of snow depths in the field site, as estimated after each filtering method has been applied. The snow depths from the UAV data is shown in grey.

4.3. Impact of canopy cover on measurement bias

Hof

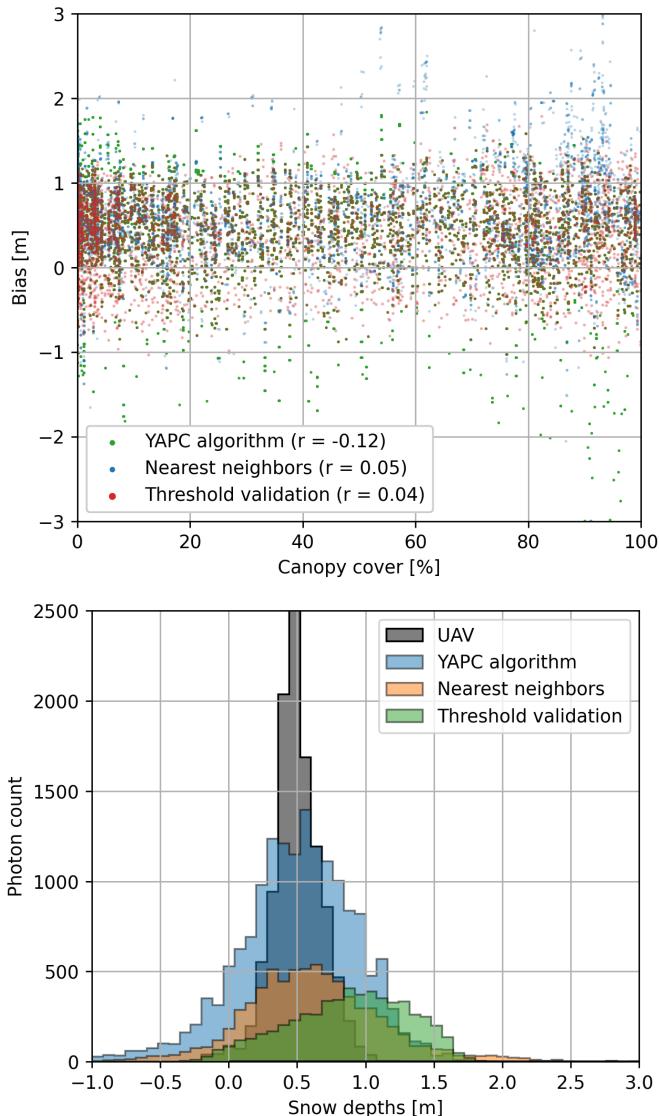


Figure 4.8: Upper: Scatter plot with measurement bias from all the filtering methods used, shown in individual colours, versus the canopy cover inside the photon footprint. A clear linear correlation would be present if the points aligned to a 45° line ($|x| = |y|$). No such pattern is visible for any of the methods. Lower: The distribution of snow depths in the field site, as estimated after each filtering method has been applied. The snow depths from the UAV data is shown in grey.

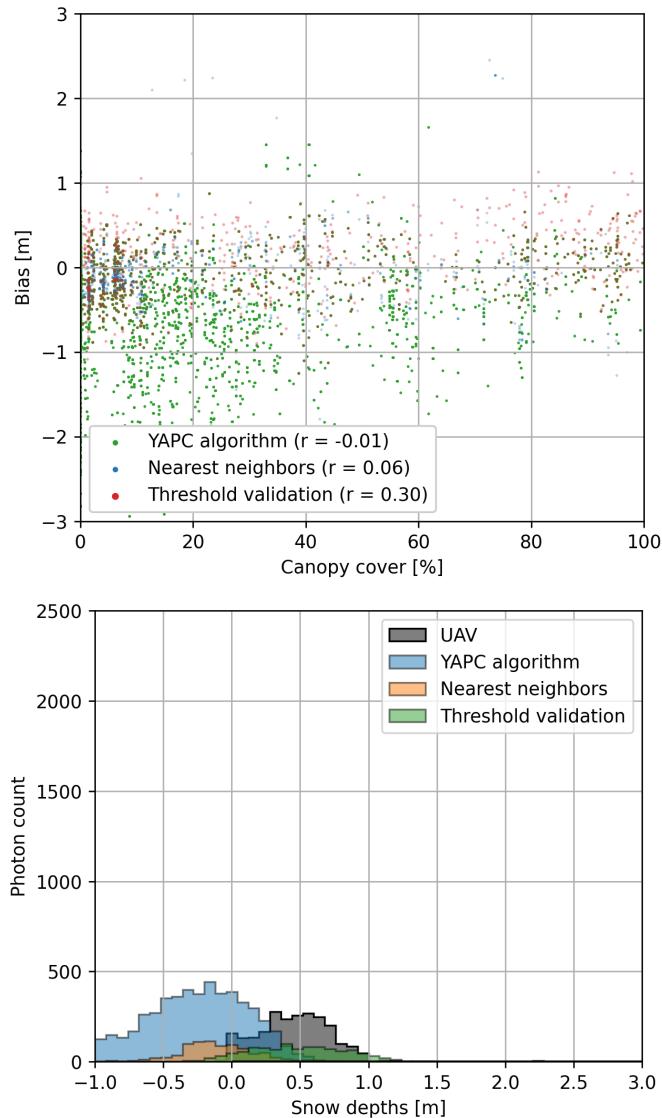
Jevnaker

Figure 4.9: Upper: Scatter plot with measurement bias from all the filtering methods used, shown in individual colours, versus the canopy cover inside the photon footprint. A clear linear correlation would be present if the points aligned to a 45° line ($|x| = |y|$). No such pattern is visible for any of the methods. Lower: The distribution of snow depths in the field site, as estimated after each filtering method has been applied. The snow depths from the UAV data is shown in grey.

4.3. Impact of canopy cover on measurement bias

Vikerfjell East

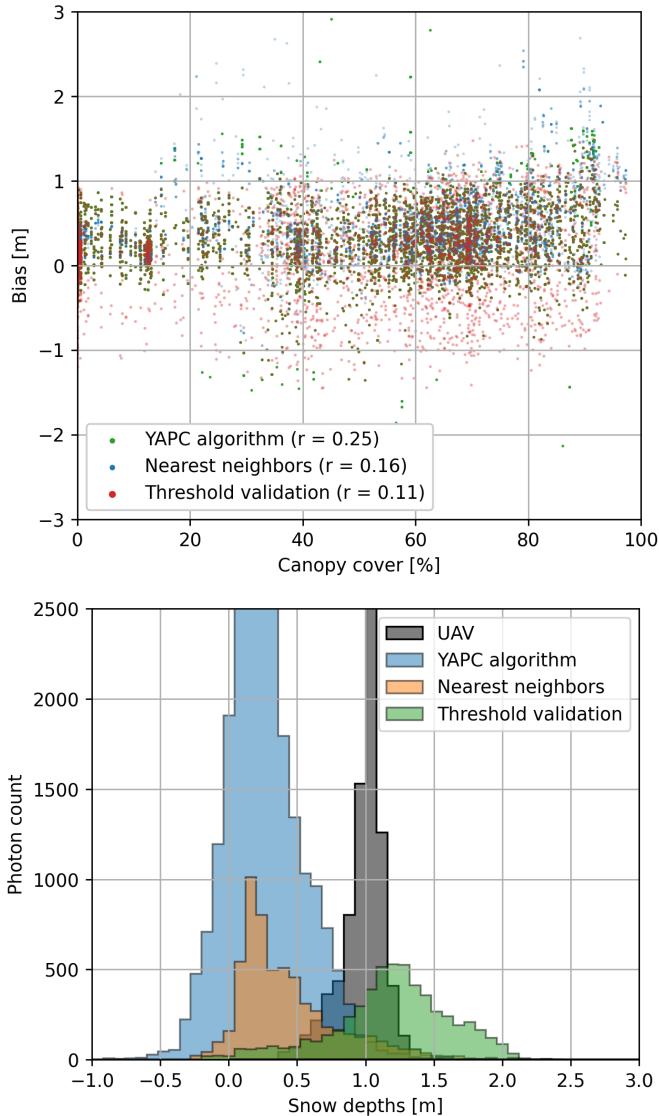


Figure 4.10: Upper: Scatter plot with measurement bias from all the filtering methods used, shown in individual colours, versus the canopy cover inside the photon footprint. A clear linear correlation would be present if the points aligned to a 45° line ($|x| = |y|$). No such pattern is visible for any of the methods. Lower: The distribution of snow depths in the field site, as estimated after each filtering method has been applied. The snow depths from the UAV data is shown in grey.

Vikerfjell West

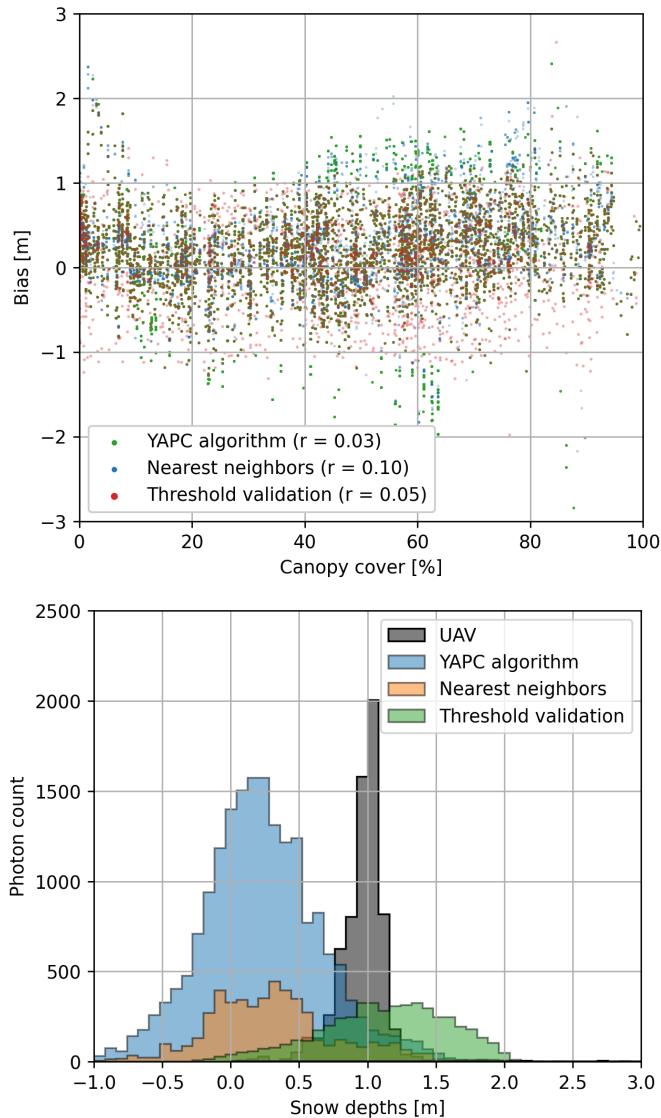


Figure 4.11: Upper: Scatter plot with measurement bias from all the filtering methods used, shown in individual colours, versus the canopy cover inside the photon footprint. A clear linear correlation would be present if the points aligned to a 45° line ($|x| = |y|$). No such pattern is visible for any of the methods. Lower: The distribution of snow depths in the field site, as estimated after each filtering method has been applied. The snow depths from the UAV data is shown in grey.

Chapter 5

Discussion

This chapter will further examine the results, data and workflow of the thesis project. The first section will look at how the snow depth estimation performs in forested terrain. The second section will discuss the output of the filtering methods, while the third section will examine the influence of the chosen data and methods.

The results will in each section be viewed in relation to previous research is necessary to examine differences and similarities, to help identify methodological weaknesses, and can at the same time aid in discovering new research questions. The last section of the discussion chapter will therefore explore possible uses for the output and illuminate questions yet to be answered.

5.1 Snow depth estimation in forested terrain

As previously stated in section 1.3.1, little research has been done to explore snow depth estimation in forested terrain using satellite laser altimetry. A few studies have used the ATL08 and/or ATL06 data products from ICESat-2 to measure snow depths (Hu et al. 2021; Enderlin et al. 2022) on glaciers and unforested, low-sloped terrain, while Treichler and Andreas Kääb (2017) has used the older ICESat satellite to estimate snow depths, also in unforested terrain. Enderlin et al. (2022) found that the ATL08 and ATL06 data products were suitable for watershed-scale snow depth estimates in regions with low slope and thick snow packs. Hu et al. (2021) ends their conclusion by stating that further investigation is needed to assess whether snow depths can be estimated accurately in forested terrain. Their study indicated an accuracy at sub-decimetre level, with RMSEs as low as 4.20 cm.

A recent study by Deschamps-Berger et al. (2023) found that a precision of approximately 0.5 m can be achieved in unforested terrain with slopes $< 10^\circ$, when differencing with a snow-off DTM from ALS or satellite photogrammetry. In dense tree cover, the snow depths were affected negatively if comparing to a satellite photogrammetry DEM, but had little impact when comparing to an ALS DTM.

These results are comparable to what was found in this thesis, with the *threshold validation* method applied to ATL03 data providing an RMSE of 0.48 m. The low differences in RMSE (table 4.4) strengthens the argument that an accuracy of 0.5 m is possible for snow depth measurements even in dense forest.

5.2 Filtering methods

5.2.1 Nearest neighbor

The *nearest neighbor* method was the first attempt at developing a method to filter out photons which do not represent the snow-covered terrain surface. The principles are very rudimentary, and because it relies on a lower threshold number of photons within a set buffer it is vulnerable to two opposing problems: Too few points nearby which represent the snow surface, i.e. within dense forest canopies, leads to these photons being disregarded entirely; Or, when there are dense clusters of photons which do not represent the snow surface, e.g. tree crowns, are kept. Setting the threshold for necessary nearby photons in either positive or negative direction will increase one of these problems. The threshold would also have to be different for each beam in a pair, as the strong beam has four times the photon density of the weak beam. These effect are visible in tables 4.2 and 4.3, where the strong beam has retained 56.2 % of its original photons, while the weak beam only retained 26.4 %. The snow depth plots in figs. 4.1 to 4.5 also illustrate this problem, with large gaps in the snow surface resulting from this method.

5.2.2 Threshold validation

Threshold validation was the second attempt at developing a filtering method. Although this was the most effective filtering method, yielding the lowest RMSE values of all the methods table 4.1, it relies on a DTM as additional input. The DTM in this case was of 1 m resolution and based on ALS. DTMs of this quality is not available everywhere, and it is highly likely that the output of this method would be significantly reduced in dense forests. This case is also specifically mentioned in the research of Deschamps-Berger et al. (2023), which found increased bias when using satellite photogrammetric snow-off DEMs.

The major benefit of this method, compared to the other two, is that the resulting snow depths automatically disregards any extreme values, making it suitable for larger scale. A precondition for this use case is that the thresholds are recalculated for sub samples of the data, because of its reliance on the normal distribution of snow depths: When a dataset covers large enough areas, snow depths that are within the normal in one part of the area can equal unreasonable snow depths in another part, which should be filtered there.

The *threshold validation* method worked well in forested terrain, and also in dense forests. Even though the median slope of all of the field sites was in the range of 10–17°, the RMSEs are within a decimetre of what was achieved in low-density forests by Deschamps-Berger et al. (2023) (table 4.4). The difference between forested and unforested terrain was overall at just 5 cm. The snowdepth plots (figs. 4.1 to 4.5) also show that this method provides a snow surface which fits the UAV snow DEM nicely.

5.2.3 Yet Another Photon Classifier (YAPC)

YAPC (Sutterley 2023) was the only method that was not developed for this thesis. The *YAPC* filtered photons were the direct output from an API, which also means that the data is not directly comparable to the other ATL03 data. *YAPC* has the major benefit of not relying on any additional data, as is necessary with the *threshold validation* method. Even still, it provided a snow surface with an RMSE of 0.59 across all sites, with both beams. This is just 0.11 m higher than the *threshold validation* method. The fact that it

does not rely on additional data, such as DEMs, makes this method far easier to apply on a larger scale. Although several attempts were made to alter the parameters for the classification algorithm, it is likely that it is possible to improve the outputs further, and also vary the parameters depending on slope, forest density and forest type.

5.3 Workflow review

In this section the overall processing workflow will be discussed, along with the software used. The processing workflow relies on two different programming languages — R and Python — in addition to a proprietary software, DJI Terra. DJI Terra is part of the UAV system, together with the Zenmuse L1 LiDAR instrument and the Matrice M300 RTK UAV. The data from the Zenmuse L1 instrument can not be used without DJI Terra, and as such there were no real alternatives for the initial pre-processing of the UAV point clouds.

Using just one programming language was considered, to simplify the workflow, and Python would then have been preferred due to it being more familiar. However, the *lidR* library in the *R* language is very powerful for working with computationally heavy data such as point clouds, and lacks an equal alternative in the *Python* domain. There were also attempts to use *lidR* for the ground point classification, but all attempts failed due to errors which was not explored further, instead prioritising other parts of the workflow. The point classification was therefore done in DJI Terra. This could affect the DTMs, as the *lidR* enables the use of algorithms which are more suited for the sloped, forested terrain of this thesis. Further work would include renewed attempts at creating the DTMs and CHMs with *lidR*.

An additional step which was eventually skipped was the spatial co-registration of the DTMs, CHMs and ICESat-2 transects. Functionality exists in the *xdem* Python library to achieve co-registration of either two rasters (e.g. Nuth and A. Käab (2011)) or a raster with point data, such as the ICESat-2 photons. Unfortunately, attempts at using this functionality also yielded errors which were not overcome in the dedicated timespan. Co-registration is mentioned by both Deschamps-Berger et al. (2023) and Enderlin et al. (2022) as possible contributors to improved accuracy in the results, and getting this to work would plausibly improve the accuracy of the results here as well. Some of the measurement bias could possibly be explained by the lack of co-registration, where the terrain aspect is not aligned perfectly to the ground track.

Yet Another Photon Classifier (YAPC) is also not just available through the *SlideRule* API but also as a Python library with functionality to perform calculations directly on ICESat-2 granules (Sutterley 2022). Applying these to the same *GeoDataFrame* as the other two methods would make it easier to compare the results of the methods, as it would ensure the same total number of photons, and not having to do some of the calculations (e.g. DTM and CHM interpolation) twice. Some attempts were made to do this at an initial stage, but also resulted in errors. The processing would likely be more efficient if this library could be used, and require less ($\approx 50\%$).

5.4 Control Data and Field Work

The control data used in this thesis, acquired from the UAV flights, was very important. The UAV data acquisition had to be timed with the ICESat-2 ground tracks, and this worked relatively well. As can be seen in table 2.5, the largest time delta between ICESat-2 and UAV data acquisition was on the Drammen site, at approximately 51

hours. The Hof site had a time delta of approximately 47 hours. These two sites were used in a period of cold temperatures, which helped keep the snow pack stable. The resulting errors of the snow depths from all the sites leave no indication that the snow pack had changed in these two sites, compared to the other ones.

The sites are all relatively uniform, in most aspects, as seen in table 2.6. The canopy cover varies from 35 % to 66 %; The forest types are all mostly evergreen conifers, with a small presence of deciduous trees, with a median canopy height between 8.02–10.96 m; The median slope varied between 10 °and 17 °; The median snow depths were in the range between 0.49–1.01 m, as measured by the UAV. This is beneficial for identifying and eliminating systematic bias, but also it makes the results less applicable on a larger scale. Acquiring data from additional forest types would have made the results even more robust, as it would be able to do the same analyses over different forest types. Southern Norway is relatively limited in terms of forest types, with most of it being boreal coniferous trees at the altitudes with enough snow depth.

However, acquiring the field site data with the chosen method is relatively resource demanding, and requires a high readiness and relative proximity to the field sites in the days before and after an ICESat-2 overpass. Another option would be to assess the quality of the NMA DTM as suitability for the terrain heights, and then use Airborne Laser Scanning (ALS) to cover a larger part of the ICESat-2 ground tracks during the snow-on season. ALS is more expensive than the UAV method, but would also provide control data at a significantly larger scale, in a single survey.

Except for the Vikerfjell sites, the NMA DTMs are all based on ALS surveys from the time period of April to August (table 2.5), a period where there is likely to be a significant presence of under-story vegetation. This can affect the LiDAR measurements, as the laser returns are both more likely to represent vegetation and harder to filter out as non-ground returns in the point classification. This can in turn introduce errors in the DTMs, of up to several decimetres (Simpson, Smith and Wooster 2017). The Uncrewed Aerial Vehicle (UAV) Digital Terrain Model (DTM)s are also affected by this problem, but while the Airborne Laser Scanning (ALS) surveys typically have a handful (2-5) of returns per square metre, the Zenmuse L1 data has several hundred. It also measures up to three returns per pulse, which can help separate the terrain from even short under-story vegetation. More modern sensors, like the Zenmuse L2, can register up to five returns per pulse, which would likely provide an even better representation of the terrain surface in the forest.

The canopy height threshold, which was applied to the Canopy Height Model (CHM), was set to 2 m. This is a suitable height to separate trees from shrubs and under-story vegetation, but one problem that was encountered was that there had been some logging activities in several sites, in the period between the snow-on and snow-off UAV acquisitions. The absence of trees meant the point distribution on the actual terrain surface would likely be higher, but it also meant that we could not generate the CHM from the snow-off surfaces. In other words, the 2 m threshold actually translated to 2 m plus the snow depth.

An in-depth comparison between the UAV LiDAR data and the NMA NDH LiDAR would have been an interesting, to identify any similarities between the resulting DTMs. If the DTMs correlate nicely, it would also make it possible to test the application of the thesis methods on a larger scale. This is still an option for future work, now that the data exists.

The snow depth in the field sites were also recorded with in-situ measurements, using a snow probe and registration by phone camera with geotagging via the internal

5.4. Control Data and Field Work

GNSS positioning. However, the positions were not accurate enough to provide more use than to confirm that the in-situ measurements were within the normal range of snow depths in the study area. There is also a possible error source related to what the probe hits. It could have been a tree stub, rock, or even ice, which would not necessarily be representative for the terrain around it. A more systematic approach to these measurements, using RTK or PPK positioning for the probe, would have made these measurements far more useful.

Chapter 5. Discussion

Chapter 6

Conclusion

The main goal of this thesis was to investigate whether the ATLAS instrument on ICESat-2 is able to measure snow surface heights with the necessary accuracy to reliably estimate snow depths. The results indicate that the ICESat-2 ATLAS instrument is capable of estimating snow depths in forested terrain, with an accuracy of approximately 0.5 m. The measurements are not significantly less accurate in forested than unforested terrain (< 0.05 m), when using LiDAR based Digital Terrain Model (DTM)s as the snow-off surface data. This is comparable to the findings of another study, conducted by Deschamps-Berger et al. (2023). However, to achieve more robust results, the filtering methods should be examined at a deeper level, and tested on a larger variation of forest types. It is highly likely that the accuracy can be improved further on the *threshold validation* and *YAPC* methods, while the simpler *nearest neighbor* method does not indicate particular usefulness.

Chapter 6. Conclusion

Bibliography

- Adams, E. E. et al. (Jan. 2011). ‘Local terrain-topography and thermal-properties influence on energy and mass balance of a snow cover’. In: *Annals of Glaciology* 52.58, pp. 169–175. ISSN: 0260-3055, 1727-5644. DOI: 10.3189/172756411797252257.
- ASPRS (9th July 2019). *LAS Specification 1.4 - R15*. The American Society for Photogrammetry & Remote Sensing.
- Bossche, Joris Van den et al. (June 2023). *geopandas/geopandas: v0.13.2*. Version v0.13.2. DOI: 10.5281/zenodo.8009629.
- Deschamps-Berger, César et al. (13th July 2023). ‘Evaluation of snow depth retrievals from ICESat-2 using airborne laser-scanning data’. In: *The Cryosphere* 17.7. Publisher: Copernicus GmbH, pp. 2779–2792. ISSN: 1994-0416. DOI: 10.5194/tc-17-2779-2023.
- Enderlin, Ellyn M. et al. (15th Dec. 2022). ‘Uncertainty of ICESat-2 ATL06- and ATL08-derived snow depths for glacierized and vegetated mountain regions’. In: *Remote Sensing of Environment* 283, p. 113307. ISSN: 0034-4257. DOI: 10.1016/j.rse.2022.113307.
- Faller, Nikolaus, Marco Weber and Infoterra GmbH (2007). ‘TerraSAR-X and TanDEM-X: Revolution in spaceborne radar’. In: *2007 IEEE International Geoscience and Remote Sensing Symposium*. 2007 IEEE International Geoscience and Remote Sensing Symposium. Barcelona, Spain: IEEE, pp. 4924–4928. ISBN: 978-1-4244-1211-2. DOI: 10.1109/IGARSS.2007.4423966.
- Feng, Tuo et al. (1st June 2023). ‘A systematic evaluation of multi-resolution ICESat-2 ATL08 terrain and canopy heights in boreal forests’. In: *Remote Sensing of Environment* 291, p. 113570. ISSN: 0034-4257. DOI: 10.1016/j.rse.2023.113570.
- Finger, David (2018). ‘The value of satellite retrieved snow cover images to assess water resources and the theoretical hydropower potential in ungauged mountain catchments’. In: *Jökull* 68, pp. 47–66.
- Forskningsrådet (2022). *SNOWDEPTH - Global snow depths from spaceborne remote sensing for permafrost, high-elevation precipitation, and climate reanalyses - Prosjektbanken*. Forskningsrådet - Prosjektbanken. URL: <https://prosjektbanken.forskningsradet.no/project/FORISS/325519> (visited on 08/06/2022).
- Frey, Julian et al. (June 2018). ‘UAV Photogrammetry of Forests as a Vulnerable Process. A Sensitivity Analysis for a Structure from Motion RGB-Image Pipeline’. In: *Remote Sensing* 10.6. Number: 6 Publisher: Multidisciplinary Digital Publishing Institute, p. 912. ISSN: 2072-4292. DOI: 10.3390/rs10060912.

Bibliography

- Hijmans, Robert J. (2024). *terra: Spatial Data Analysis*.
- Hu, Xiaojing et al. (25th May 2021). ‘Can the Depth of Seasonal Snow be Estimated From ICESat-2 Products: A Case Investigation in Altay, Northwest China’. In: *IEEE Geoscience and Remote Sensing Letters* 19. Conference Name: IEEE Geoscience and Remote Sensing Letters, pp. 1–5. ISSN: 1558-0571. DOI: [10.1109/LGRS.2021.3078805](https://doi.org/10.1109/LGRS.2021.3078805).
- Isenburg, Martin (30th Oct. 2017). *LASTools - efficient LiDAR processing software*. Version 171030, unlicensed.
- Jost, Georg et al. (15th Dec. 2007). ‘The influence of forest and topography on snow accumulation and melt at the watershed-scale’. In: *Journal of Hydrology* 347.1, pp. 101–115. ISSN: 0022-1694. DOI: [10.1016/j.jhydrol.2007.09.006](https://doi.org/10.1016/j.jhydrol.2007.09.006).
- King, Matt A. (Mar. 2009). ‘The GPS Contribution to the Error Budget of Surface Elevations Derived From Airborne LIDAR’. In: *IEEE Transactions on Geoscience and Remote Sensing* 47.3. Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 874–883. ISSN: 1558-0644. DOI: [10.1109/TGRS.2008.2005730](https://doi.org/10.1109/TGRS.2008.2005730).
- Komarov, Anton and Matthew Sturm (31st Dec. 2023). ‘Local variability of a taiga snow cover due to vegetation and microtopography’. In: *Arctic, Antarctic, and Alpine Research* 55.1. Publisher: Taylor & Francis _eprint: <https://doi.org/10.1080/15230430.2023.2170086>, p. 2170086. ISSN: 1523-0430. DOI: [10.1080/15230430.2023.2170086](https://doi.org/10.1080/15230430.2023.2170086).
- Liu, Z. et al. (30th Apr. 2018). ‘DECOMPOSITION TECHNIQUES FOR ICESA-T/GLAS FULL-WAVEFORM DATA’. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLII-3. Conference Name: ISPRS TC III Mid-term Symposium “Developments, Technologies and Applications in Remote Sensing” (Volume XLII-3) - 7–10 May 2018, Beijing, China Publisher: Copernicus GmbH, pp. 1179–1182. ISSN: 1682-1750. DOI: [10.5194/isprs-archives-XLII-3-1179-2018](https://doi.org/10.5194/isprs-archives-XLII-3-1179-2018).
- Mallet, Clément and Frédéric Bretar (1st Jan. 2009). ‘Full-waveform topographic lidar: State-of-the-art’. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 64.1, pp. 1–16. ISSN: 0924-2716. DOI: [10.1016/j.isprsjprs.2008.09.007](https://doi.org/10.1016/j.isprsjprs.2008.09.007).
- Moore, David S., George P. McCabe and Bruce A. Craig (2012). *Introduction to the practice of statistics*. 7. ed., internat. ed. New York: Freeman [u.a.] 101-105. ISBN: 978-1-4292-8664-0.
- Næsset, Erik et al. (1st Dec. 2004). ‘Laser scanning of forest resources: the nordic experience’. In: *Scandinavian Journal of Forest Research* 19.6. Publisher: Taylor & Francis _eprint: <https://doi.org/10.1080/02827580410019553>, pp. 482–499. ISSN: 0282-7581. DOI: [10.1080/02827580410019553](https://doi.org/10.1080/02827580410019553).
- NASA (28th June 2022). *NASA Approves Continuation of ICESat-2 After 3+ Years of Big Results - NASA*. URL: <https://www.nasa.gov/earth-and-climate/nasa-approves-continuation-of-icesat-2-after-3-years-of-big-results/> (visited on 12/05/2024).
- (2024). *Data Products / ICESat-2*. Data Products | ICESat-2. URL: <https://icesat-2.gsfc.nasa.gov/science/data-products> (visited on 11/04/2024).
- NASA’s Earth Observing System* (2024). URL: <https://eospso.gsfc.nasa.gov/> (visited on 09/04/2024).

- Neuenschwander, Amy et al. (15th Dec. 2020). ‘Validation of ICESat-2 terrain and canopy heights in boreal forests’. In: *REMOTE SENSING OF ENVIRONMENT* 251. ISSN: 0034-4257. DOI: 10.1016/j.rse.2020.112110.
- Neumann, T. A. et al. (2023). *ATLAS/ICESat-2 L2A Global Geolocated Photon Data, Version 6*. DOI: 10.5067/ATLAS/ATL03.006.
- Neumann, Thomas A. et al. (1st Nov. 2019). ‘The Ice, Cloud, and Land Elevation Satellite – 2 mission: A global geolocated photon product derived from the Advanced Topographic Laser Altimeter System’. In: *Remote Sensing of Environment* 233, p. 111325. ISSN: 0034-4257. DOI: 10.1016/j.rse.2019.111325.
- Nuth, C. and A. Kääb (29th Mar. 2011). ‘Co-registration and bias corrections of satellite elevation data sets for quantifying glacier thickness change’. In: *The Cryosphere* 5.1. Publisher: Copernicus GmbH, pp. 271–290. ISSN: 1994-0416. DOI: 10.5194/tc-5-271-2011.
- OGC (14th Sept. 2019). *GeoTIFF Standard*. URL: <https://docs.ogc.org/is/19-008r4/19-008r4.html> (visited on 14/04/2024).
- Pearse, Grant D. et al. (1st Aug. 2018). ‘Comparison of high-density LiDAR and satellite photogrammetry for forest inventory’. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 142, pp. 257–267. ISSN: 0924-2716. DOI: 10.1016/j.isprsjprs.2018.06.006.
- Perry, Matthew (26th Mar. 2024). *perrygeo/python-rasterstats*. original-date: 2013-09-18T06:44:29Z.
- Pomeroy, JW and Eric Brun (2001). ‘Physical properties of snow’. In: *Snow ecology: An interdisciplinary examination of snow-covered ecosystems* 45. Publisher: Citeseer, p. 118.
- Pronk, Maarten, Marieke Eleveld and Hugo Ledoux (6th July 2023). ‘Assessing vertical accuracy and spatial coverage of ICESat-2 and GEDI spaceborne lidar for creating global terrain models’. In: Publisher: EarthArXiv.
- R Core Team (2023). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.
- Rahlf, Johannes et al. (1st Dec. 2014). ‘Comparison of four types of 3D data for timber volume estimation’. In: *Remote Sensing of Environment* 155, pp. 325–333. ISSN: 0034-4257. DOI: 10.1016/j.rse.2014.08.036.
- Roussel, Jean-Romain et al. (15th Dec. 2020). ‘lidR: An R package for analysis of Airborne Laser Scanning (ALS) data’. In: *Remote Sensing of Environment* 251, p. 112061. ISSN: 0034-4257. DOI: 10.1016/j.rse.2020.112061.
- Schober, Patrick, Christa Boer and Lothar A. Schwarte (May 2018). ‘Correlation Coefficients: Appropriate Use and Interpretation’. In: *ANESTHESIA AND ANALGESIA* 126.5. Place: TWO COMMERCE SQ, 2001 MARKET ST, PHILADELPHIA, PA 19103 USA Publisher: LIPPINCOTT WILLIAMS & WILKINS Type: Article, pp. 1763–1768. ISSN: 0003-2999. DOI: 10.1213/ANE.0000000000002864.
- Simpson, Jake E., Thomas E. L. Smith and Martin J. Wooster (Nov. 2017). ‘Assessment of Errors Caused by Forest Vegetation Structure in Airborne LiDAR-Derived DTMs’.

Bibliography

- In: *Remote Sensing* 9.11. Number: 11 Publisher: Multidisciplinary Digital Publishing Institute, p. 1101. ISSN: 2072-4292. DOI: [10.3390/rs9111101](https://doi.org/10.3390/rs9111101).
- Sutterley, Tyler (June 2022). *tsutterley/yapc: v0.0.0.8*. Version 0.0.0.8. DOI: [10.5281/zenodo.6717591](https://doi.org/10.5281/zenodo.6717591).
- (2023). *YAPC Overview — pyYAPC 0.0.0.8 documentation*. Rev. 87023d6e. URL: https://yapc.readthedocs.io/en/latest/getting_started/Overview.html#yapc-goals (visited on 17/04/2024).
- Swinski, J. P. et al. (Mar. 2024). *ICESat2-SlideRule/sliderule: v4.3.1*. Version v4.3.1. DOI: [10.5281/zenodo.10797886](https://doi.org/10.5281/zenodo.10797886).
- Szafarczyk, Anna and Cezary Toś (Jan. 2023). ‘The Use of Green Laser in LiDAR Bathymetry: State of the Art and Recent Advancements’. In: *Sensors* 23.1. Number: 1 Publisher: Multidisciplinary Digital Publishing Institute, p. 292. ISSN: 1424-8220. DOI: [10.3390/s23010292](https://doi.org/10.3390/s23010292).
- The pandas development team (Feb. 2020). *pandas-dev/pandas: Pandas*. Version latest. DOI: [10.5281/zenodo.3509134](https://doi.org/10.5281/zenodo.3509134).
- Treichler, Désirée and Andreas Kääb (15th Mar. 2017). ‘Snow depth from ICESat laser altimetry - A test study in southern Norway’. In: *REMOTE SENSING OF ENVIRONMENT* 191. Web of Science ID: WOS:000397360500029, pp. 389–401. ISSN: 0034-4257, 1879-0704. DOI: [10.1016/j.rse.2017.01.022](https://doi.org/10.1016/j.rse.2017.01.022).
- Treichler, Désirée and Marco Mazzolini (Apr. 2024). *ATL03_snowdepths*. Version 1.0.0.
- Van Rossum, Guido and Fred L. Drake (2009). *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace. ISBN: 1-4414-1269-7.
- Wilson, John P. (15th Jan. 2012). ‘Digital terrain modeling’. In: *Geomorphology*. Geospatial Technologies and Geomorphological Mapping Proceedings of the 41st Annual Binghamton Geomorphology Symposium 137.1, pp. 107–121. ISSN: 0169-555X. DOI: [10.1016/j.geomorph.2011.03.012](https://doi.org/10.1016/j.geomorph.2011.03.012).

Appendix A

UAV LiDAR processing script

A.1 GNU bash script

```
1 #!/bin/bash
2
3 LOGDATE=$(date +'%Y%m%d_%H%M%S_')
4 {
5 module -q r master
6
7 # List of input LAS/LAZ files, separated by newline
8 inFileList=""
9
10 # output directory
11 outDir=""
12
13 COUNTER=0
14 TOTAL=$(echo $inFileList | wc -w)
15 for f in $inFileList
16 do
17     inFile=$(basename -- "$f")
18     outFile="$(echo ${inFile%.*} | cut -c 10-)"
19     let COUNTER++
20     START=$(date -u +%s)
21
22     echo "=====
23     "
24     echo " ($COUNTER/$TOTAL) $inFile"
25     echo "=====
26     "
27     echo " Started $(date -u -d @$START)"
28
29     echo -e "\n 1. Running las2dem.R:"
30     Rscript las2dem.R $f $outFile
31
32     echo -e "\n 2. Moving output files to directory:\n $outDir"
33     rsync --remove-source-files *.tif $outDir
34
35     TDELTAS=$((($date -u +%s) - $START))
36     TIMESTAMP=$(date -u -d @$TDELTAS +"%H %M %S")
37     echo -e "\n Finished $(date -u)\n      (Time spent: $TIMESTAMP)"
38     echo -e "=====
39 =====\n\n"
40 done
41
42 } > ${LOGDATE}log.txt
```

A.2 R code

```

1 library(lidR)
2 library(tools)
3
4 # Enable verbose mode and (try to) remove progress bars for logging
5 options(lidR.verbose=TRUE, lidR.progress=FALSE)
6
7 # lidR uses 50 % of CPU threads by default. Reduce to 25 %
8 set_lidr_threads(get_lidr_threads()/2)
9
10 # Fetch command arguments
11 args = commandArgs(trailingOnly=TRUE)
12
13 # Check if correct number of arguments is provided
14 if (length(args) != 2) {
15   stop("Please include input filename => 'las2dem.R <inFile> <outFile>'")
16 }
17
18 # Set variables from command arguments
19 inFile = args[1]
20 outFile = args[2]
21
22 # Start processing
23 cat("      a.  Open LAS file: ", file_path_sans_ext(basename(inFile)))
24 las <- readLAS(inFile, select="xyzrnc")
25
26 cat("\n      b.  LAS file information:\n")
27 print(las)
28
29 cat("\n      c.  Generating DTM\n")
30 dtm <- rasterize_terrain(las, res = .5, algorithm = tin())
31 terra::writeRaster(dtm, paste(outFile, "_DTM.tif", sep=""), filetype =
32   "GTiff", overwrite = TRUE)
33
34 if (grepl("SnowOn", inFile, fixed=TRUE)) {
35   cat("\n      d.  Generating canopy height model\n")
36   nrm <- normalize_height(las, algorithm = tin())
37   chm <- rasterize_canopy(nrm, res = .2, p2r(.2, na.fill = tin()))
38   chm <- terra::clamp(chm, 2, values=FALSE)      # Set cell values < 2 m
39   to NaN
40   terra::writeRaster(chm, paste(outFile, "_CHM.tif", sep=""), filetype
41   = "GTiff", overwrite = TRUE)
42 }
43
44 cat("\n      e.  las2dem.R terminating successfully!\n")

```

Appendix B

Python script

```
1 import atl03_functions as atl
2 from sliderule import icesat2
3 icesat2.init()
4
5 import numpy as np
6 import pandas as pd
7 import geopandas as gpd
8 import xdem
9
10 import pickle
11
12 granules = './granules/' # Where to store the granules from NSIDC
13 geoid = './dem/no_kv_HREF2018B_NN2000_EUREF89.tif' # Geoid model file
14 SKdtms = './dem/' # DTMs from NMA
15 uav_dtms = './uav/' # DTMs from UAV acquisition
16 aoi = gpd.read_file('./vector/aoi.shp') # AOI shapefiles
17
18
19 regions = {
20     'drammen': {
21         'time': ['2023-01-21', '2023-01-22'],
22         'onTS': '20230123',
23         'offTS': '20231003',
24         'bbox': [9.97863, 59.79217, 9.98314, 59.79998]
25     },
26     'hof': {
27         'time': ['2023-01-21', '2023-01-22'],
28         'onTS': '20230123',
29         'offTS': '20231003',
30         'bbox': [10.03298, 59.51847, 10.03757, 59.52495]
31     },
32     'jevnaker': {
33         'time': ['2023-04-19', '2023-04-20'],
34         'onTS': '20230419',
35         'offTS': '20231003',
36         'bbox': [10.48238, 60.24314, 10.48953, 60.25098]
37     },
38     'vikerfjell-e': {
39         'time': ['2023-02-19', '2023-02-20'],
40         'onTS': '20230219',
41         'offTS': '20231002',
42         'bbox': [9.97001, 60.49940, 9.97951, 60.50487]
43     },
44     'vikerfjell-w': {
```

Appendix B. Python script

```

45     'time': ['2023-02-20', '2023-02-21'],
46     'onTS': '20230219',
47     'offTS': '20231002',
48     'bbox': [9.97001, 60.49940, 9.97951, 60.50487]
49   }
50 }
51
52
53 # Define function for Canopy Height Model statistics
54 def chmStats(gdf, raster, diameter=13, stats=['mean'], density=False):
55   """
56   Usage: chmStats(gdf, raster, diameter, stats, density)
57   Use a canopy height model (CHM) raster to calculate a set of stats
58   for the area within a photon footprint diameter. Defaults to
59   mean canopy height. With density=True, the canopy density is
60   calculated
61   by also adding a 'count' and 'nan' column, which counts cells with
62   values
63   and NaNs respectively.
64
65   gdf:           geopandas.GeoDataFrame with the photons. Must contain a
66   geometry field
67   raster:        The CHM raster to use for calculations
68   diameter:      Photon footprint diameter [m]
69   stats:         Which statistics to calculate. Alternatives derive from
70   rasterstats.zonal_stats
71   density:       Whether to calculate canopy density. Will also add 'count'
72   ' and 'nan' to stats
73
74   Output:        The input gdf with new columns for stats
75   """
76   from rasterstats import zonal_stats
77
78   if density:
79     stats += ['count', 'nan']
80   data = gdf.geometry.apply(
81     lambda pt: zonal_stats(pt.buffer(diameter/2), raster, stats=stats,
82     all_touched=True)[0]
83   )
84   chdf = pd.DataFrame(list(data), index=data.index.tolist())
85   if density:
86     chdf['density'] = chdf['count'] / (chdf['count'] + chdf['nan'])
87   for col in chdf.columns:
88     chdf.rename({col: f'chm_{col}'}, axis=1, inplace=True)
89   gdf = gdf.merge(chdf, how='outer', left_index=True, right_index=True)
90   return gdf
91
92
93 # Function for nearest neighbor point grouping
94 def grpheights(gdf, hcol='h_ph', xybuff=1, zbuff=1, threshold=5, window
95 =20):
96   """
97   Usage: grpheights(gdf, hcol, xybuff, zbuff, threshold, window)
98   For each photon in the data column hcol, use a 3D buffer of
99   horizontal size
100  xybuff and vertical size zbuff, to scan for other photons. If number
101  of
102  photons surpass threshold, the original photon is assigned the mean
103  height
104  of the all photons in buffer. Otherwise np.nan is assigned. Finally,
105  a

```

```

95     a rolling median function is assigned to all photons.
96
97     gdf:           geopandas.GeoDataFrame containing photon heights
98     hcol:          The name of the photon height column
99     xybuff:        Horizontal buffer size
100    zbuff:          Vertical buffer size
101    threshold:    Photon count threshold
102    window:        Window to use for rolling median
103
104    Output:        geopandas.Series with new photon height values
105    ''
106
107    def group(row):
108        grp = gdf.loc[
109            ( gdf.within(row.geometry.buffer(xbuff)) ) &
110            ( abs(row.h_ph - gdf.h_ph) < zbuff )
111        ].h_ph
112        if len(grp) >= threshold:
113            return grp.mean()
114        else:
115            return np.nan
116
116    h_grp = gdf.apply(lambda row: group(row), axis=1)
117    h_grp = h_grp[h_grp.notnull()]      # Drop NA rows
118    h_grp = h_grp.rolling(window, closed='both', center=True).median()
119
120    return h_grp
121
122
123 # Threshold validation function
124 def eval_ph(values, margin=.15):
125     ''
126     Usage: eval_ph(values, margin)
127     Takes a set of dH values and accepted margin as input. Calculates a
128     suitable
129     percentile threshold to use for photon selection, and returns an
130     equal-shape
131     series with boolean True where photons are within threshold. ((
132     Preserves input
133     indices.)
134
135     values:      pandas.Series with dH values
136     margin:      Max acceptable dH increase per .05 quantile increase
137
138     Output:      pandas.Series with boolean True/False for valid/invalid
139     photons
140     ''
141     valid = values
142     x = np.arange(0.3, 1, .05)
143     y = values[values > 0].quantile(x)
144     y0 = y.iloc[0]
145     for threshold in y:
146         if threshold - y0 >= margin:
147             valid[(values > threshold) | (values < 0)] = False
148             return valid.astype('bool')
149     y0 = threshold
150     return values.astype('bool')
151
152
153 # Function to get Sliderule ATLO3 extents with YAPC scores
154 def getsliderule(coords, dates):
155     poly = [{`lon`: coords[0], `lat`: coords[1}],

```

Appendix B. Python script

```

152         {'lon': coords[2], 'lat': coords[1]},
153         {'lon': coords[2], 'lat': coords[3]},
154         {'lon': coords[0], 'lat': coords[3]},
155         {'lon': coords[0], 'lat': coords[1]}]
156     parms = {
157         'poly': poly,
158         't0': dates[0],
159         't1': dates[1],
160         'srt': 0,
161         'cnf': 0,
162         'pass_invalid': False,
163         #'quality_ph': 0,
164         'at108_class': 'at108_ground',
165         'yapc': {
166             'score': 0,
167             'knn': 0,
168             'min_ph': 5,
169             'win_h': 3,
170             'win_x': 5
171         },
172         'cnt': 3,
173         'len': 5,
174         'res': 5,
175         'ats': 1
176     }
177     sld = icesat2.atl03sp(parms, keep_id=False)
178     sld = sld.to_crs(crs)
179     sld['x'] = sld.geometry.x
180     sld['y'] = sld.geometry.y
181     return sld
182
183
184
185 # Download the data from NSIDC
186 for reg in regions:
187     name = reg.capitalize()
188     data = regions[reg]
189     coords = data['bbox']
190     dates = data['time']
191     atl.downloadATL('ATL03', coords, dates, uname, email, granules, reg)
192     atl.downloadATL('ATL08', coords, dates, uname, email, granules, reg)
193
194
195 # Processing of the photon datasets
196 for reg in regions:
197     name = reg.capitalize() if reg[-2] != '-' else reg[:-2].capitalize()
198     data = regions[reg]
199     coords = data['bbox']
200     dates = data['time']
201     poly = [{ 'lon': coords[0], 'lat': coords[1]},
202              { 'lon': coords[2], 'lat': coords[1]},
203              { 'lon': coords[2], 'lat': coords[3]},
204              { 'lon': coords[0], 'lat': coords[3]},
205              { 'lon': coords[0], 'lat': coords[1]}]
206
207     crs = atl.getUTMcrslation(np.mean([coords[1], coords[3]]), np.mean([
208         coords[0], coords[2]]))
209     gdf, ancillary = atl.loadATLdata('ATL03', granules, reg, crs)
210
211     # Load and merge ATL08 data
212     atl8 = atl.loadATLdata('ATL08', granules, reg, crs)

```

```

212 gdf = atl.addATL08class(gdf, atl8)
213
214 # Add a proper timestamp
215 td_gps = 315964800      # GPS vs UNIX epoch timedelta [s]
216 td_atlas = ancillary['atlas_sdp_gps_epoch'][0]      # ATLAS vs GPS
epoch timedelta [s]
217 gdf['datetime'] = pd.to_datetime(gdf.delta_time + td_gps + td_atlas,
unit='s')
218
219 # Classify beams
220 gdf['strongbeam'] = False
221 gdf['strongbeam'][gdf.pb == gdf.pb.value_counts().index[0]] = True
222
223 # Remove photons outside study area
224 mask = gdf.within(aoi.loc[aoi.index[aoi.Name == name][0], 'geometry'])
225 gdf = gdf.loc[mask]
226
227 # Filter based on signal conf
228 if sum(np.isnan(gdf.signal_conf_ph_landice)) > 0.9*len(gdf) or sum(
gdf.signal_conf_ph_landice) < -0.9*len(gdf):
    gdf = gdf[(gdf.signal_conf_ph > 1)]
else:
    gdf = gdf[(gdf.signal_conf_ph_landice > 1)]
229
230 # Interpolate data in gaps, if present
231 subset = (gdf.lon_ph >= coords[0]) & (gdf.lon_ph <= coords[2]) & (gdf
.lat_ph >= coords[1]) & (gdf.lat_ph <= coords[3])
232 gdf = atl.interpolateATL03_alongtrack(gdf, subset)
233 gdf = gdf.sort_values('delta_time')
234
235 # Read height from DEMs
236 geoid = xdem.DEM(geoid)
237 SKdtm = xdem.DEM(SKdtms + name + '_DTM1_UTM32.tif')
238 uavSnow = xdem.DEM(uav_dtms + name + '_SnowOn_DTM.tif')
239 uavBare = xdem.DEM(uav_dtms + name + '_SnowOff_DTM.tif')
240 chm = xdem.DEM(uav_dtms + name + '_SnowOn_CHM.tif')
241
242 # Write heights to GeoDataFrame
243 gdf['geoid'] = geoid.interp_points(np.array((gdf['lon_ph'].values,
gdf['lat_ph'].values)).T, input_latlon=True, order=1)
244 gdf['SKdtm'] = SKdtm.interp_points(np.array((gdf['x'].values, gdf['y']
.values)).T, input_latlon=False, order=1)
245 gdf['uavSnow'] = uavSnow.interp_points(np.array((gdf['x'].values, gdf
['y'].values)).T, input_latlon=False, order=1)
246 gdf['uavBare'] = uavBare.interp_points(np.array((gdf['x'].values, gdf
['y'].values)).T, input_latlon=False, order=1)
247
248 # Evaluate if photon heights are valid, based on threshold distance
from DTM
249 gdf['valid'] = eval_ph(gdf.h_ph - gdf.SKdtm - gdf.geoid)
250
251 # Apply grouped values to new column, based on 'h_ph'
252 gdf['h_grp'] = grpheights(gdf)
253
254 # Set unreasonable values to NaN
255 gdf.loc[gdf.uavBare < 0, 'uavBare'] = np.nan
256 gdf.loc[gdf.uavSnow < 0, 'uavSnow'] = np.nan
257 gdf.loc[gdf.SKdtm < 0, 'SKdtm'] = np.nan
258
259 # Drop rows without photon height

```

Appendix B. Python script

```
263 gdf = gdf[gdf.h_ph.notna()]
264
265 phfp = 13 # Footprint diameter
266
267 # Add canopy heights and calculate canopy statistics
268 gdf['ch'] = chm.interp_points(np.array((gdf['x'].values, gdf['y'].values)).T, input_latlon=False, order=1)
269 gdf = chmStats(gdf, uav_dtms + name + '_SnowOff_CHM.tif', diameter=phfp, density=True)
270
271 # Repeat some processing with the Sliderule data
272 yapc = getsliderule(coords, dates)
273 yapc['strongbeam'] = False
274 yapc['strongbeam'][yapc.pair == yapc.pair.value_counts().index[0]] = True
275 yapc['uavSnow'] = uavSnow.interp_points(np.array((yapc['x'].values, yapc['y'].values)).T, input_latlon=False, order=1)
276 yapc['uavBare'] = uavBare.interp_points(np.array((yapc['x'].values, yapc['y'].values)).T, input_latlon=False, order=1)
277 yapc['geoid'] = geoid.interp_points(np.array((yapc.geometry.to_crs(4326).x.values, yapc.geometry.to_crs(4326).y.values)).T, input_latlon=False, order=1)
278 yapc['ch'] = chm.interp_points(np.array((yapc['x'].values, yapc['y'].values)).T, input_latlon=False, order=1)
279 yapc['SKdtm'] = SKdtm.interp_points(np.array((yapc['x'].values, yapc['y'].values)).T, input_latlon=False, order=1)
280 yapc = chmStats(yapc, uav_dtms + name + '_SnowOff_CHM.tif', diameter=phfp, density=True)
281
282
283 # Store data in the regions dictionary
284 data['crs'] = crs
285 data['gdf'] = gdf
286 data['ancillary'] = ancillary
287 data['yapc'] = yapc
288
289 print(f'Found {len(gdf)} photons in {reg.capitalize()}')
290
291
292 # Export the entire dictionary as a pickle file in the script directory
293 with open('data.pkl', 'wb') as outfile:
294     pickle.dump(regions, outfile, protocol=pickle.HIGHEST_PROTOCOL)
```

Appendix C

Complete errors table

Table C.1: Complete errors data for all methods and both beams, in the Drammen and Hof field sites. (PART I)

Area	Beam	Method	Bias	MAE	RMSE	Rel. Bias	Rel. RMSE
Drammen	Both	Unfiltered	1.77 m	3.91 m	6.38 m	300%	1,081%
		Threshold validation	-0.05 m	0.32 m	0.39 m	-8%	66%
		Nearest neighbors	-0.12 m	0.23 m	0.68 m	-21%	115%
		YAPC algorithm	-0.15 m	0.41 m	0.64 m	-26%	108%
	Strong	Unfiltered	1.65 m	3.93 m	6.36 m	280%	1,079%
		Threshold validation	-0.05 m	0.31 m	0.39 m	-8%	65%
		Nearest neighbors	-0.13 m	0.27 m	0.71 m	-21%	120%
		YAPC algorithm	-0.16 m	0.42 m	0.65 m	-27%	110%
	Weak	Unfiltered	2.20 m	3.87 m	6.45 m	371%	1,087%
		Threshold validation	-0.06 m	0.32 m	0.40 m	-10%	67%
		Nearest neighbors	-0.09 m	0.08 m	0.56 m	-15%	94%
		YAPC algorithm	-0.08 m	0.35 m	0.52 m	-13%	85%
Hof	Both	Unfiltered	1.23 m	2.34 m	4.09 m	246%	820%
		Threshold validation	0.39 m	0.51 m	0.60 m	78%	121%
		Nearest neighbors	0.62 m	0.47 m	0.80 m	125%	161%
		YAPC algorithm	0.48 m	0.58 m	0.68 m	100%	141%
	Strong	Unfiltered	1.43 m	2.49 m	4.28 m	281%	836%
		Threshold validation	0.40 m	0.52 m	0.61 m	78%	120%
		Nearest neighbors	0.64 m	0.52 m	0.79 m	125%	155%
		YAPC algorithm	0.50 m	0.59 m	0.69 m	103%	141%
	Weak	Unfiltered	0.38 m	1.75 m	3.22 m	86%	722%
		Threshold validation	0.37 m	0.48 m	0.57 m	82%	129%
		Nearest neighbors	0.50 m	0.24 m	0.83 m	112%	187%
		YAPC algorithm	0.24 m	0.43 m	0.55 m	55%	130%

Appendix C. Complete errors table

Table C.2: Complete errors data for all methods and both beams, in the Jevnaker and Vikerfjell field sites. (PART II)

		Unfiltered	0.89 m	1.79 m	3.88 m	200%	872%
Both	Jevnaker	Threshold validation	0.02 m	0.31 m	0.39 m	5%	87%
		Nearest neighbors	-0.12 m	0.13 m	0.65 m	-27%	147%
		YAPC algorithm	-0.38 m	0.49 m	0.71 m	-23%	43%
		Unfiltered	0.84 m	1.77 m	4.09 m	217%	1,055%
Strong		Threshold validation	0.01 m	0.30 m	0.37 m	4%	97%
		Nearest neighbors	-0.12 m	0.16 m	0.27 m	-31%	69%
		YAPC algorithm	-0.40 m	0.50 m	0.71 m	-23%	40%
		Unfiltered	0.93 m	1.80 m	3.66 m	187%	735%
Weak		Threshold validation	0.04 m	0.32 m	0.40 m	7%	81%
		Nearest neighbors	-0.10 m	0.11 m	0.87 m	-19%	175%
		YAPC algorithm	-0.19 m	0.45 m	0.63 m	-38%	125%
		Unfiltered	0.64 m	1.19 m	2.31 m	65%	235%
Both	Vikerfjell E	Threshold validation	0.20 m	0.40 m	0.50 m	21%	50%
		Nearest neighbors	0.43 m	0.34 m	0.82 m	44%	84%
		YAPC algorithm	0.25 m	0.30 m	0.39 m	26%	39%
		Unfiltered	0.65 m	1.08 m	2.23 m	67%	231%
Strong		Threshold validation	0.22 m	0.38 m	0.47 m	23%	49%
		Nearest neighbors	0.42 m	0.37 m	0.53 m	44%	55%
		YAPC algorithm	0.26 m	0.29 m	0.38 m	26%	38%
		Unfiltered	0.60 m	1.57 m	2.59 m	58%	250%
Weak		Threshold validation	0.12 m	0.49 m	0.59 m	12%	57%
		Nearest neighbors	0.47 m	0.23 m	1.46 m	46%	141%
		YAPC algorithm	0.12 m	0.60 m	0.78 m	12%	80%
		Unfiltered	0.32 m	1.24 m	2.87 m	33%	299%
Both	Vikerfjell W	Threshold validation	0.16 m	0.41 m	0.51 m	16%	53%
		Nearest neighbors	0.30 m	0.35 m	0.66 m	31%	69%
		YAPC algorithm	0.19 m	0.41 m	0.56 m	19%	58%
		Unfiltered	0.31 m	1.17 m	2.88 m	33%	306%
Strong		Threshold validation	0.16 m	0.39 m	0.49 m	17%	52%
		Nearest neighbors	0.29 m	0.40 m	0.58 m	31%	61%
		YAPC algorithm	0.19 m	0.40 m	0.54 m	20%	56%
		Unfiltered	0.32 m	1.51 m	2.86 m	31%	278%
Weak		Threshold validation	0.13 m	0.46 m	0.57 m	13%	55%
		Nearest neighbors	0.38 m	0.18 m	0.88 m	36%	86%
		YAPC algorithm	0.15 m	0.50 m	0.82 m	15%	80%