



Norges miljø- og
biovitenskapelige
universitet

Masteroppgave 2022 30 stp
Fakultet for realfag og teknologi

Deteksjon av bergsprekker ved hjelp av lidar: En studie av intensitet

Detection of Rock Fractures Using Lidar: A Study
of Intensity

Bjørnar Nevland Knudsen
Geomatikk

Forord

Med denne oppgaven setter jeg punktum for mitt masterstudium i Geomatikk ved Norges miljø- og biovitenskapelige universitet (NMBU). Oppgaven tilsvarer 30 studiepoeng og er skrevet ved Fakultet for realfag og teknologi.

Jeg ønsker først og fremst å rette en stor takk til Ivar Maalen-Johansen for å alltid stille opp og være tilgjengelig for veiledning når det har vært nødvendig. Videre vil jeg også takke gjengen som var med til Rekefjord i oktober 2021 for interessant og lærerikt feltarbeid. Ikke minst en takk til gode venner på rom TF1-212 for å gjøre masterhverdagen og våren 2022 til en trivelig og minnerik vår. En stor takk også til mor for korrekturlesing og støtte. Jeg vil også takk gutta i 25B for å ha bidratt til en upåklagelig bosituasjon. Til slutt vil jeg også takke resterende venner og ansatte ved NMBU som har bidratt til fire uforglemmelige år her i Ås.

Bjørnar N. Knudsen

Bjørnar Nevland Knudsen
Ås, 15. mai 2022

Sammendrag

Manuell deteksjon av bergsprekker er tid- og arbeidskrevende jobb, spesielt ved kartlegging av større arealer som for eksempel steinbrudd. Som en del av et prosjekt hvor earthresQuesenteret ved NMBU er med, skal et dagbrudd ved Rekefjord i Sokndal kartlegges for å se om det egner seg til bruk som deponi av ikke-farlig avfall. En del av denne prosessen innebærer kartlegging av sprekker og sprekkesystemer som forekommer i bruddet. Kunnskap om sprekkesystemene som opptrer i bruddet er viktig for å forstå geologien i området, og for å kunne se om det er en mulig link mellom vann i bruddet og grunnvannet i området rundt. Dette er viktig informasjon for å kunne forhindre en potensiell konterminering av grunnvannet.

Målet med denne studien var å sammenligne RGB-bilder av sprekker i bruddet med intensitetsdata fra laserskanner over samme sprekker, for å finne en bedre metode for deteksjon av bergsprekker. Til dette arbeidet ble det benyttet Canny kantdeteksjon i et forsøk på å skille sprekke fra den omkringliggende berggrunnen og gjøre dem lettere å detektere.

Intensitetsdata fra laserskanner er et komplekst fagområde med flere variabler. I denne oppgaven ble det tatt høyde for noen av disse i et forsøk på å normalisere intensitetsverdiene til signalet som returneres til laserskanneren under skanning. Variablene det er tatt høyde for i denne studien var skanneavstand og innfallsvinkel, som er de variablene som har størst påvirkning på intensitetsverdiene. Ved å korrigere for disse var meningen å kunne få et resultat som vil være det samme uavhengig av skanneavstand og vinkel til overflaten i bruddet.

Undersøkelsen viser at Canny kantdeteksjon sliter med å gi en fullstendig deteksjon av bergsprekker, uavhengig om det er RGB eller intensitetsdata dataene farges etter. En manuell kartlegging i felten vil være nødvendig for å subsidiere mangelfulle sprekke data og få et mer fullstendig bilde av sprekkenettverket.

Abstract

Manually detecting rock fractures is time consuming and laborous work, in particular when mapping larger areas such as quarries. As part of a project led by the earthresQue center at NMBU, a quarry at Rekefjord in Sokndal is being mapped to see if it is suitable for use as a landfill site of non-hazardous waste. Part of this process involves mapping the network of fractures that occur in the quarry. Knowledge of the network of fractures in the quarry is important to understand the geology of the area, and to be able to see if there is a possible link between runoff water in the quarry and the groundwater in the bedrock below. This is important information in order to prevent a potential groundwater contamination.

The aim of this study was to compare RGB images of fractures in the quarry with intensity data from laser scanners over the same fractures, to find a better method for detecting rock fractures. Canny edge detection was used in this work in an attempt to make a clear separation between the fractures and the surrounding bedrock. By doing this, the fractures would hopefully become easier to detect.

Intensity data from laser scanners is a complex field of knowledge with several variables. Some of these variables have been taken into account in this study, in an attempt to normalize the intensity values of the signal returned to the laser scanner during scanning. The variables taken into account in this study are scan distance and incidence angle. These are the variables which have the greatest effect on the intensity values. By correcting these, the idea was that we would then get the same results regardless of the scanning position.

The study shows that Canny edge detection struggles to provide complete detection of all rock fractures within the quarry, regardless of whether it's the RGB or the intensity data the Canny operator is using. A manual mapping in the field would be necessary to subsidize the fracture data and get a more complete model of the fracture network within the quarry.

Innhold

Forord	2
Sammendrag	3
Abstract	4
1 Introduksjon	9
1.1 Bakgrunn og motivasjon	9
1.2 Tema og problemstilling	9
1.3 Studiemråde	10
2 Teori	12
2.1 Lidar (Light Detection and Ranging)	12
2.1.1 Intensitet	12
2.2 RGB-farger	14
2.3 Hardware og Software	14
2.3.1 FARO Focus 3D X130	14
2.3.2 FARO SCENE	15
2.3.3 CloudCompare	15
2.3.4 Quick Terrain Modeler (QTM)	16
2.3.5 Python	16
2.4 Kantdeteksjon	17
3 Metode	20
3.1 Datafangst	20
3.2 Preprosessering og prosessering	21
3.3 Korreksjon av intensitet	22
3.4 Kantdeteksjon	25
4 Resultater	26
4.1 Ulike terskelverdier for Punktsky 1 med korrigert intensitet	27
4.2 Ulike terskelverdier for Punktsky 2 med korrigert intensitet	29
4.3 Canny kantdeteksjon av RGB-bilder, punktsky og mesh	31
4.4 Canny kantdeteksjon av ukorrigerede intensitetsverdier, punktsky og mesh	33
4.5 Canny kantdeteksjon av korrigerede intensitetsverdier, punktsky og mesh	36
5 Diskusjon	39
5.1 Sammenligning av resultater	39
5.2 Maskinl�ring	42
6 Konklusjon	43
6.1 Videre studier	43

Figurer

1.1	Rekefjord med Vallnesbruddet markert i rødt	10
1.2	Tverrsnitt (Vest/øst) av Vallnesbruddet i <i>Quick Terrain Modeler (QTM)</i> . . .	11
2.1	Prinsippet bak lidar	12
2.2	Targets brukt for sammenkobling av flere skann	15
2.3	Visualisering av Vallnes i QTM	16
2.4	Masker for beregning av derivert i X-retning (venstre) og Y-retning (høyre) .	17
2.5	Bildene i midten og til høyre viser ulik Sobel kantdeteksjon av originalbildet til venstre (henholdsvis i X-retning, og i XY-retning)	18
2.6	Canny kantdeteksjon stegvis	19
3.1	Skanneposisjoner i Vallnesbruddet, markert med rød prikk	20
3.2	Veileder Ivar kontrollerer at laserskanningen går uten problemer	21
3.3	Ferdigprosessert punktsky av posisjon 1 (se figur 3.1). Rød firkant markerer et av områdene som ble klippet ut og brukt for videre testing	22
3.4	Et av utsnittene fra punktskyen	22
3.5	Innfallsvinkel	24
3.6	Helningsgrad på flaten	24
4.1	Punktsky 1, fargelagt med korrigerte intensitetsverdier	27
4.2	Nedre og øvre terskelverdi: 100	27
4.3	Nedre og øvre terskelverdi: 100 og 150	27
4.4	Nedre og øvre terskelverdi: 150	28
4.5	Nedre og øvre terskelverdi: 150 og 200	28
4.6	Punktsky 2, fargelagt med korrigerte intensitetsverdier	29
4.7	Nedre og øvre terskelverdi: 100	29
4.8	Nedre og øvre terskelverdi: 100 og 150	29
4.9	Nedre og øvre terskelverdi: 150	30
4.10	Nedre og øvre terskelverdi: 150 og 200	30
4.11	Punktsky 1, RGB-farger	31
4.12	Nedre og øvre terskelverdi: 100	31
4.13	Mesh av Punktsky 1, RGB-farger	31
4.14	Nedre og øvre terskelverdi: 100	31
4.15	Punktsky 2, RGB-farger	32
4.16	Nedre og øvre terskelverdi: 50 og 100	32
4.17	Mesh av Punktsky 2, RGB-farger	32
4.18	Nedre og øvre terskelverdi: 100	32
4.19	Punktsky 1, råintensitet	33
4.20	Nedre og øvre terskelverdi: 100	33
4.21	Mesh av Punktsky 1, råintensitet	33
4.22	Nedre og øvre terskelverdi: 100	33
4.23	Punktsky 2, råintensitet	34

4.24	Nedre og øvre terskelverdi: 100	34
4.25	Mesh av Punktsky 1, råintensitet	34
4.26	Nedre og øvre terskelverdi: 150	34
4.27	Punktsky 1, korrigert intensitet	36
4.28	Nedre og øvre terskelverdi: 100	36
4.29	Mesh av Punktsky 1, korrigert intensitet	36
4.30	Nedre og øvre terskelverdi: 100 og 150	36
4.31	Punktsky 2, korrigert intensitet	37
4.32	Nedre og øvre terskelverdi: 100 og 150	37
4.33	Mesh av Punktsky 2, korrigert intensitet	37
4.34	Nedre og øvre terskelverdi: 200 og 200	37
5.1	Utsnitt fra punktskyen til posisjon 1 på 1,8m x 1,8m. Den rød ringen viser hvor det var tilsig vann.	41

Tabeller

1	Spesifikasjoner for FARO Focus 3D X130	14
---	--	----

Introduksjon

1.1 Bakgrunn og motivasjon

Denne masteroppgaven er skrevet som et bidrag til Senter for forskningsdrevet innovasjon, EarthresQue, ved NMBU. EarthresQue har som mål å utvikle teknologi og systemer for bærekraftig håndtering og behandling av avfall og overskuddsmasser [1].

EarthresQue har satt sammen en PhD-stilling innenfor hydrogeokjemi. Denne omhandler karakterisering og modellering av gjennomstrømming av væsker i bergsprekker for å kunne bestemme naturlige geologiske barrierer til nye avfallsdeponi [2]. Dette er barrierer som vil kunne forhindre konterminering av grunnvannet rundt deponiet som følge av avrenning av forurensende fluider fra deponiet via sprekker i berggrunnen. Denne stipendiatstilling ble tildelt Bilal Tariq.

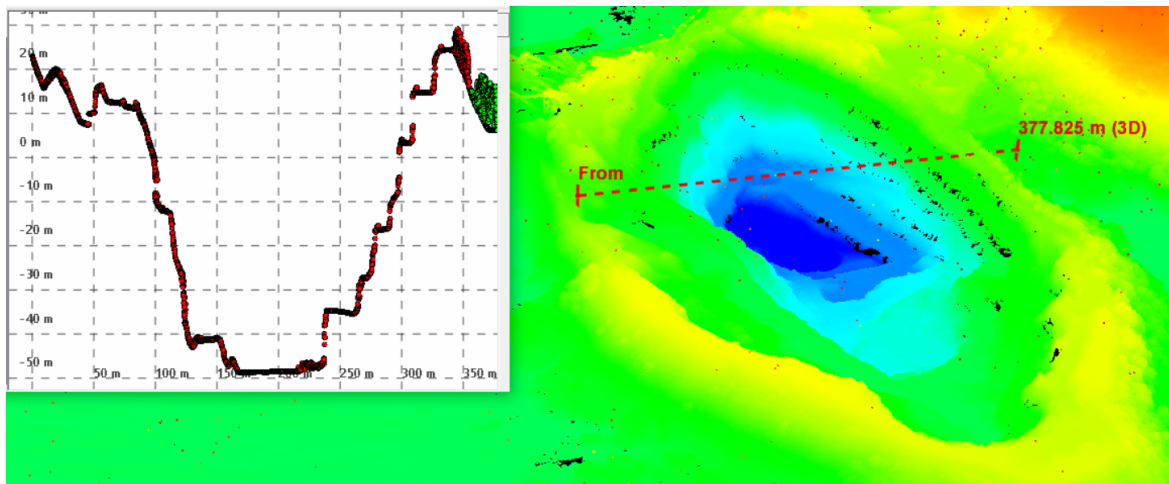
For å få pålitelige resultater fra gjennomstrømningsmodelleringen er det viktig å ha en best mulig oversikt over nettverket av sprekker fra berggrunnen i området som er tiltenkt deponi. Å kartlegge disse sprekkesystemene har i tillegg til å være en arbeids- og tidskrevende prosess, også vært potensielt farlig da bergveggene i områder som egner seg som deponi ofte er spesielt utsatt for steinsprang. Dette fordi graving og sprengning i det bruddet vil kunne gjøre berggrunnen i området ustabil.

I de senere år er det blitt forsket på om bruk av laserdata fra TLS (bakkebasert laserskanner) vil kunne brukes til geologisk kartlegging (som innebærer blant annet sprekker og sprekkesystemer) [3, 4]. Dette er forskning som har gitt tilfredstillende resultater, og det har vist seg at laserskannere har funksjonaliteten som er nødvendig for å produsere punktskyer med god nok punkttetthet til å gjøre geologisk kartlegging.

1.2 Tema og problemstilling

Mitt bidrag til Bilals oppgave var å benytte fjernmåling for å se etter en bedre metode for sprekkekartlegging. Valget falt på bruk av laserskanning og intensitetsdata, og så kombinere dette med en kantdeteksjonsalgoritme. Dette for å potensielt synliggjøre sprekkenes fra laserskanningen mer. Videre ville disse dataene gjøre deteksjonen og kartleggingen lettere og sikrere enn manuell kartlegging i felten. Problemstillinger og forsknings spørsmål var dermed satt:

- Vil intensitetsdata fra laserskanner kunne benyttes sammen med Canny kantdeteksjon for en bedre metode til kartlegging av bergsprekker?
- Vil korreksjon av intensitetsdataene kunne utvide funksjonaliteten til laserskanning med hensyn på sprekkekartlegging?

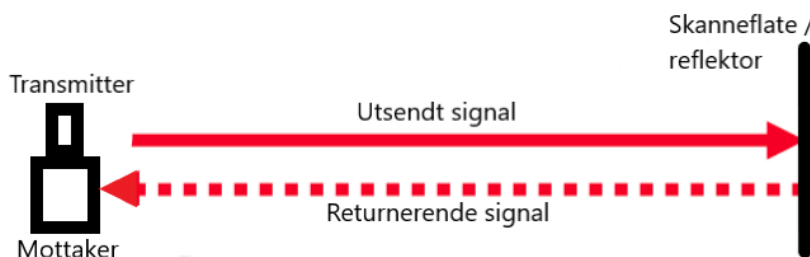


Figur 1.2: Tverrsnitt (Vest/øst) av Vallnesbruddet i *Quick Terrain Modeler (QTM)*

Teori

2.1 Lidar (Light Detection and Ranging)

Lidar er en metode innen fjernmåling for å portrettere romlige overflater som 3D-modeller. Her sendes lys ut fra en transmitter og reflekteres fra overflater tilbake til mottakeren. Tiden lyset bruker på å returnere til mottakeren, sammen med retningen til mottakeren, brukes til å lage en tre-dimensjonell punktsky. Lys blir hverken påvirket eller ødelagt av andre lyskilder, dermed kan laserskanning utføres uavhengig av lysforhold. For at lidar skal kunne gi optimale resultat er det viktig at det ikke er objekter i lufta, som vil kunne reflektere lyset før det treffer en fast flate. Lidar fungerer altså dårlig ved nedbør. Både regn og snø vil altså kunne skape store mengder støy i skannet.



Figur 2.1: Prinsippet bak lidar

I en terrestrisk (bakkebasert) laserskanner brukes ofte nær-infrarødt lys. Dette er delvis fordi detektoren som sitter på sensoren i en laserskanner bruker silikon som har best optisk respons på lys med bølgelengder i det nær-infrarøde delen av det elektromagnetiske spekteret [7]. En annen grunn er at det finnes få andre kilder til nær-infrarødt lys som potensielt vil kunne påvirke mottakeren og skape støy i skannet [8].

Laserskanneren som brukes i denne oppgaven er en fasebasert terrestrisk laserskanner. Fasebasert betyr det sendes ut et kontinuerlig signal fra skanneren, med en kjent bølgelengde. Faseforskjellen mellom utsendt og returnert signal settes så inn i formelen under for å finne avstanden mellom skanneren og overflaten signalet returnerer fra [9].

$$\rho = \frac{\Delta\phi\lambda_m}{4\pi n} \quad (1)$$

2.1.1 Intensitet

Signalet som returnerer til skanneren vil også ha en målbar styrke, denne blir kalt intensitet. Denne styrken er avhengig av refleksiviteten til overflaten signalet returnerer fra [10]. En

overflate med lav refleksivitet vil returnere et signal med lav intensitet, mens en overflate med høy refleksivitet vil returnere et signal med høy intensitet. I tillegg vil en mengde andre faktorer også spille inn på intensitetsverdien til retursignalet. Noen av disse er avhengige av laserskanneren som brukes: bølgelengden på signalet, emisjon av signalet, strålens åpningsvinkel og den indre kalibreringen til systemet [11]. Ellers vil eksterne faktorer også kunne påvirke intensiteten. Disse innebærer: Avstanden og innfallsvinkelen til overflaten som skannes, sammensetningen til- og fuktighet i overflaten. I tillegg vil også atmosfæriske forhold kunne påvirke laserstrålens bane i mindre grad. På grunn av alle disse variablene er det vanskelig å bruke intensitetsmålinger direkte til klassifisering av overflater. Det er mer vanlig å se på kontraster og forskjeller i intensitetsmålinger for å skjelne mellom ulike overflater.

Det er gjort forsøk på å korrigere for de påvirkende faktorene til intensitetsverdier. Spesielt for avstand og innfallsvinkel til overflaten, da disse fører til de største variasjonene i intensitet av de eksterne faktorene [12]. Et av de største problemene knyttet til korreksjonene av intensitet er formen på overflaten som skannes og spredningsmønsteret til signalet fra skanneren. Tidligere forsøk på korreksjoner har tatt utgangspunkt i en perfekt spredningsreflektor (kalt *Lambertian reflektor*) [13], noe som ofte ikke er tilfellet ved skanning av bergvegger, hvor det ofte er ujevnheter i overflaten. I 2015 kom en forskergruppe fra Universitetet i Lausanne med et forslag til en korreksjon av intensitetsverdier for ujevne flater [11], som ga gode resultater. Dette forslaget er brukt i denne oppgaven for å korrigere intensitetsverdiene fra skannene i Rekefjord. Korreksjonen er som følger:

$$I_{corr} = \frac{I_{raw}R^2}{\cos \alpha(A + B \sin \alpha \tan \alpha)} \quad (2)$$

I_{corr} er korrigert intensitet, I_{raw} er råintensitet, R er skanneavstanden og α er innfallsvinkelen. A og B er forklart ved:

$$A = 1 - 0.5 \frac{\sigma_{slope}^2}{\sigma_{slope}^2 + 2} \quad (3)$$

$$B = 0.45 \frac{\sigma_{slope}^2}{\sigma_{slope}^2 + 0.09} \quad (4)$$

σ_{slope} er standardavviket til helningsgraden på flatene i området som skannes. Denne formelen tar utgangspunkt i reflektansmodellen Oren-Nayar [14], som er en generalisering av Lamberts reflektansmodell. Oren-Nayar-modellen tar høyde for ujevnheter i overflaten, som ikke er tilfellet i Lamberts reflektansmodell.

2.2 RGB-farger

Fargemodellen som brukes i vanlige digitale og analoge kamera kalles RGB, og er lik fargene vi ser med øynene. Denne baserer seg på en blanding av fargene rød, grønn og blå, av ulike gradering i tre kanaler [15]. Det er vanlig å drapere punktskyer med bilder tatt i RGB-farger for å gjøre skannet visuelt lettere å orientere seg i.

2.3 Hardware og Software

2.3.1 FARO Focus 3D X130

Laserskanneren som ble brukt til datainnsamling er en FARO Focus 3D X130. Dette er en terrestrisk laserskanner som egner seg best til avstander opp mot 130 m [16]. Den er forholdsvis liten (24x20x10 cm) og lett 5.2 kg, og var derfor lett å frakte med seg mellom de ulike skanneposisjonene og oppstillingene. I tillegg har skanneren også elektronisk libelle, slik at jeg effektivt kunne sette den opp flere ganger uten å bruke lang tid på vatring. Laseren klassifiseres som en *Klasse-1 laser*, noe som betyr at effekten til laseren er såpass lav at den ikke er skadelig for mennesker [17]. Den har en målehastighet på opptil 976 000 punkter per sekund, men kan også justeres ned dersom det ikke er nødvendig med så høy oppløsning og nøyaktighet i punktskyen. Avstandsfeilen til FARO Focus 3D x130 er oppgitt å være ± 2 mm. For å koble sammen flere skann over samme område brukes “targets” (se figur 2.2). Disse settes opp eller plasseres på flater som er synlige i flere skann. De vil fanges opp av skanneren og gjenkjennes som “targets” i registreringen som blir gjort i FARO SCENE i etterprosesseringen av skannene. FARO Focus 3D X130 har også et integrert kamera som tar bilder i RGB-bilder, disse brukes til å fargelegge skannene og gjøre dem mer visuelle. For flere spesifikasjoner se Tabell 1.

Spesifikasjoner	FARO Focus 3D X130
Batteritid	4.5 timer
Bølgelende på laser	1550 nm
Strålestørrelse ved exit	2.25 mm
Stråledivergens	0.19 mrad (0.011°)
Vertikal FOV	300°
Horisontal FOV	360°

Tabell 1: Spesifikasjoner for FARO Focus 3D X130



Figur 2.2: Targets brukt for sammenkobling av flere skann

2.3.2 FARO SCENE

FARO SCENE er softwaren til skanneren som ble brukt til datainnsamling. Dette er software som kan brukes til både prosessering, bearbeiding og analyse av punktskyer og andre 3D-modeller. FARO SCENE tilbyr et vidt spekter av ulike filtre for fjerning av støy og andre uønskede punkter i punktskyen. Det fungerer også bra til automatisk registrering og sammenkobling av punktskyer ved hjelp av flate og kuleformede targets. FARO SCENE opererer med eget et format kalt Faro Laser Scan (.fls), men støtter også eksportering til flere formater [18].

2.3.3 CloudCompare

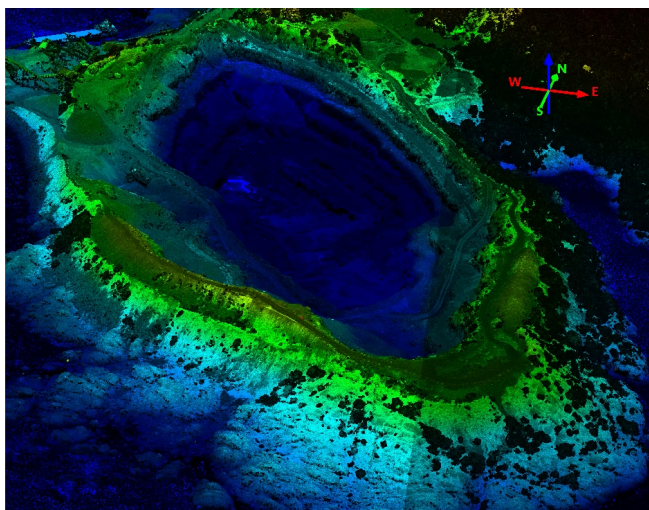
CloudCompare (CC) er et gratis open-source software for bearbeiding, behandling og analyse av punktskyer og andre 3D-modeller [19]. Programmet har flere nyttige funksjoner for målinger og uthenting av informasjon fra punktskyen. Etersom CloudCompare er open-source er det også mye god dokumentasjon tilgjengelig på internett. Dette gjør at programmet er forholdsvis lett å lære seg på egenhånd.

CC støtter mange forskjellige filformater, både til innlesing og eksport av data. Dette er veldig nyttig innen laserskanning og 3D-modellering da det her benyttes et vidt spekter av ulike filformater, som ikke alltid korresponderer like godt. Dette gjør at CloudCompare fungerer godt som et bindeledd mellom ulike typer software. CloudCompare støtter blant annet ASCII text-formater som CSV. Dette var ble brukt i oppgaven for å kunne lese inn punktskyer i Python for korrigerings av intensitetsverdier. En av funksjonene i CloudCompare er muligheten til å beregne normaler fra overflater. Denne var nyttig for å kunne beregne innfallsvinkler.

2.3.4 Quick Terrain Modeler (QTM)

QTM er et software utviklet av Applied Imagery for visualisering, redigering og analyse av punktskyer [20]. QTM er et bra verktøy for å legge sammen punktskyer og overflatemodeller, for videre å kunne gjøre geometriske beregninger i disse. Programmet har også egne algoritmer for klassifisering og ekstrahering av objekter i punktskyen basert på klassifisering. Denne funksjonen gjør det også mulig å rydde punktskyen for uønskede punkter som for eksempel støy.

I oppgaven ble QTM brukt til å få et overblikk over Vallnesbruddet. Her ble det hentet data fra **Høydedata** [21]. Det ble også gjort geometriske bergeneringer i punktskyen for å finne høydeforskjeller i bruddet.



Figur 2.3: Visualisering av Vallnes i QTM

2.3.5 Python

Python er et open-source programmeringsspråk. Det ligger mye dokumentasjon på koding i Python ute på internett og det er derfor lett å lære seg. Det er også mulig å bruke et stort antall tredjeparts moduler som er tilgjengelige i *Python Package Index (PyPI)* [22]. Dette innebærer både moduler utviklet av Python selv, i tillegg til moduler som er utviklet av brukerne av Python. Dette gjør Python til et veldig nyttig verktøy for alt som kan forenkles med koding.

Modulen *Pandas* i Python er et datamanipuleringsverktøy. I denne modulen lese data inn i *Dataframes*. Dette er et rammeverk for videre behandling av store mengder data. Denne modulen gjorde det mulig å enkelt redigere og korrigere intensitetsverdier i dataene som ble brukt i oppgaven min.

2.4 Kantdeteksjon

Kantdeteksjon er en bildeprosesseringssteknikk som går ut på å detektere kanter i et bilde for å bedre kunne forstå den underliggende strukturen [23]. Dette gjøres ved å se etter brå endringer av fargeintensiteten i bildet. I Python finnes det moduler som kan benyttes til kantdeteksjon. En av disse er OpenCV, denne inneholder algoritmene Sobel kantdeteksjon og Canny kantdeteksjon som begge er populære kantdeteksjonsalgoritmer:

- **Sobel** - Sobel kantdeteksjon ser etter brå endringer i pikselintensitetsverdier. Disse blir synlige ved å derivere intensitetsfunksjonen, hvor man da vil få et toppunkt der det skjer en slik endring. Dersom disse toppene er over en forhåndsbestemt verdi vil de bli definert som en kant. Den deriverte til intensitetsfunksjonen finnes ved å bruke en 3x3 maske [24]. Det brukes to forskjellige masker for å estimere den deriverte. En i X-retning og en i Y-retning i bildet:

$$\begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \quad \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Figur 2.4: Masker for beregning av derivert i X-retning (venstre) og Y-retning (høyre)

Sobel kantdetektoren “konvolerer” (går puljevis) gjennom bildet med disse maskene. Ved å bruke masken for X-retningen vil vi få et sobel bilde som hvor kantene er forsterket i X-retning, og motsatt hvis vi bruker masken for Y-retningen. Dette blir gjort på følgende vis:

Sett G_x og G_y til å være intensitetsgradienten for X-retningen og Y-retningen. La A og B indikere henholdsvis maskerene for X-retning og Y-retning:

$$G_x = A * I \tag{5}$$

$$G_y = B * I \tag{6}$$

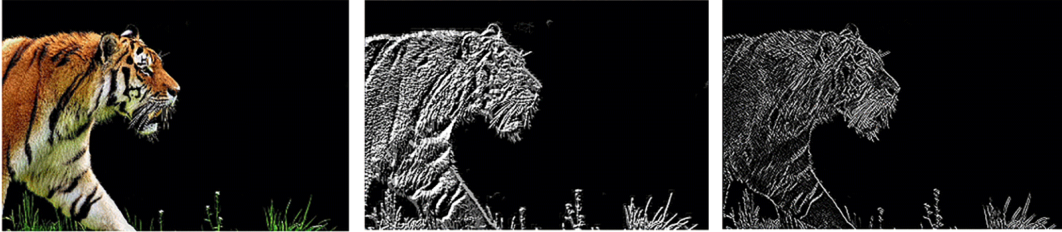
I representerer bildet, og $*$ er konvolusjon mellom bildet og maske.

Den sammenstilte gradientens styrke G beregnes så:

$$G = \sqrt{G_x^2 * G_y^2} \tag{7}$$

Orienteringen til den sammenstilte gradienten beregnes ved:

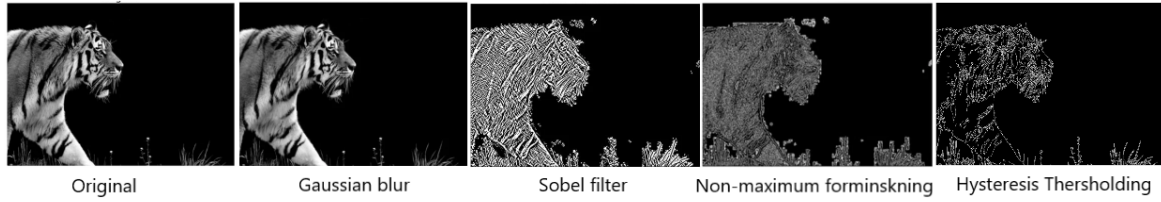
$$\Theta = \arctan(G_y/G_x) \quad (8)$$



Figur 2.5: Bildene i midten og til høyre viser ulike Sobel kantdeteksjoner av originalbildet til venstre (henholdsvis i X-retning, og i XY-retning)

- **Canny** - Canny kantdeteksjonsalgoritmen opererer som en 4-steps prosess. Disse stegene kan defineres som:
 1. **Støyreduksjonsfilter** - Et filter tilføres bildet for å fjerne unødvendige detaljer som gjerne opptrer i originalbilder. Dette er for å hindre deteksjon av uønskede kanter. Et slikt filter kalles “Gaussian blur filter” og bruker en “Gaussian funksjon” ([25]) for å jevne ut de uønskede kantene.
 2. **Utregning av intensitetsgradient** - Det gjøres på samme måte som i et Sobel filter. Her utregnes både gradientens styrke (7) og orientering (8).
 3. **Demping av feilaktige kanter** - Som et tredje ledd i Canny kantdeteksjon gjennomføres en “non-maximum” forminskning av de gjenværende kantene i bildet for å fjerne uønskede piksler. Dette gjøres ved å sammenligne gradientens retning til de omkringliggende pikslene. Pikseldier settes til null dersom gradientstyrken er lik eller mindre enn de omkringliggende gradientstyrkene til de omkringliggende pikslene. Dersom gradientstyrken er større, vil den bli stående.
 4. **Hysteresis Thresholding** - Den siste delen av Canny kantdeteksjon går ut på å sammenligne gradienstyrkene med to ulike terskelverdier:
 - Dersom gradientstyrken mellom to piksler er større enn den største terskelverdien vil pikselen defineres som en sterk kant og inkluderes i det endelige bildet.
 - Dersom gradientstyrken mellom to piksler er mindre enn den minste terskelverdien vil ikke pikselen defineres som en kant i det endelige bildet.

- Dersom gradientstyrken mellom to piksler er mellom terskelverdiene vil den defineres som en svak kant. Disse vil inkluderes i det endelige bildet dersom nabopikslen(e) er en sterk kant.



Figur 2.6: Canny kantdeteksjon stegvis

(~ 10 m). Som et siste steg av kontrollen før skanningen ble det kontrollert at det integrerte kameraet til laserskanneren var aktivert for bildetakning etter hvert fullførte skann.

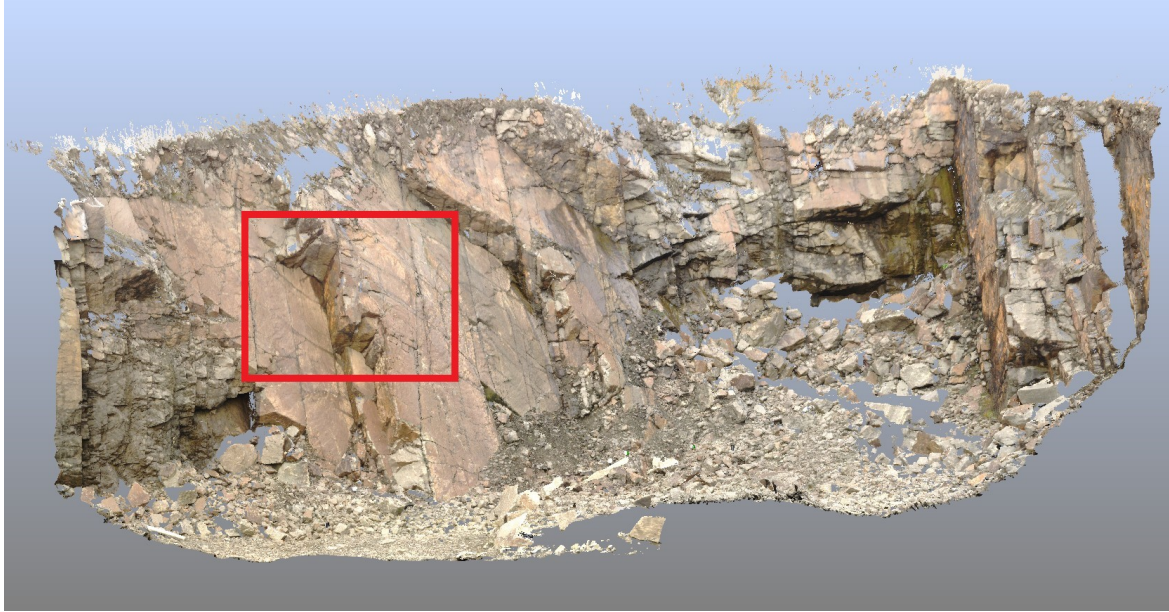
Skannene ble utført kronologisk, med biltransport mellom de ulike posisjonene for å minimere eksponeringstid i bruddet. Siden det var drift i bruddet mens vi drev med laserskanning var det viktig at vi ikke kom i veien for dette.



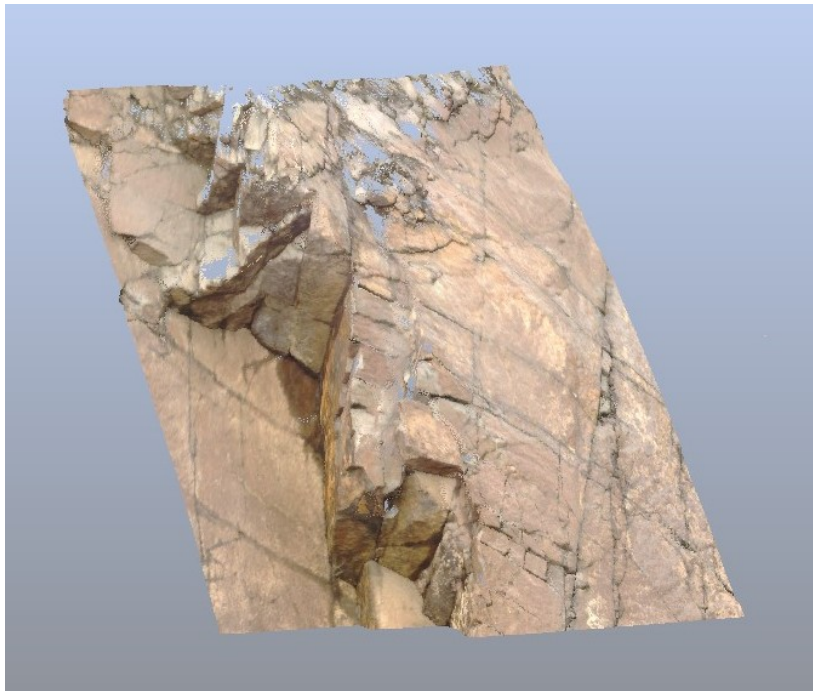
Figur 3.2: Veileder Ivar kontrollerer at laserskanningen går uten problemer

3.2 Preprosessering og prosessering

Rådata fra laserskanningen ble hentet inn i FARO SCENE for behandling. I SCENE ble data først prosessert. Dette er en prosess som innebærer fargelegging av skann basert på RGB-bildene som tas med det integrerte kameraet, påføring av ulike filtre etter behov (f.eks for fjerning av punkter utenfor en bestemt terskelavstand) og algoritmer for gjenkjenning av targets. Det ble ikke påført noe filter på skannene fra datainnsamlingen, men samtlige av utsatte targets ble gjenkjent. Videre ble det gjennomført registrering i SCENE. Her ble skannene fra de ulike oppstillingene i hver skanneposisjon koblet sammen, dette resulterte i tre endelige punktskyer. For videre behandling ble det bestemt å velge ut to små områder i punktskyen til posisjon 1 (se figur 3.1). Det ene med punktreturer fra flere oppstillinger, mens det andre med punktreturer fra kun en oppstilling. Dette var for å se om punktreturer fra flere skannere ville kunne skape uregelmessigheter i intensitetsverdiene. Utsnittene ble gjort i deler av punktskyen med sprekker som var representative for resten av punktskyen (se figur 3.3). Dette ble gjort for å korte ned databehandlingstiden i senere prosesser.



Figur 3.3: Ferdigprosessert punktsky av posisjon 1 (se figur 3.1). Rød firkant markerer et av områdene som ble klippet ut og brukt for videre testing



Figur 3.4: Et av utsnittene fra punktskyen

3.3 Korreksjon av intensitet

For å normalisere intensitetsverdiene i punktskyen, og potensielt få et bedre resultat i kant-deteksjonensdelen ble det som nevnt i 2.1.1 brukt korreksjonsformelen:

$$I_{corr} = \frac{I_{raw} R^2}{\cos \alpha (A + B \sin \alpha \tan \alpha)} \quad (9)$$

I_{corr} er korrigert intensitet, I_{raw} er råintensitet, R er skanneavstanden og α er innfallsvinkelen. A og B er forklart ved:

$$A = 1 - 0.5 \frac{\sigma_{slope}^2}{\sigma_{slope}^2 + 2} \quad (10)$$

$$B = 0.45 \frac{\sigma_{slope}^2}{\sigma_{slope}^2 + 0.09} \quad (11)$$

Hvor σ_{slope} er standardavviket av helningsgraden til flatene i veggen.

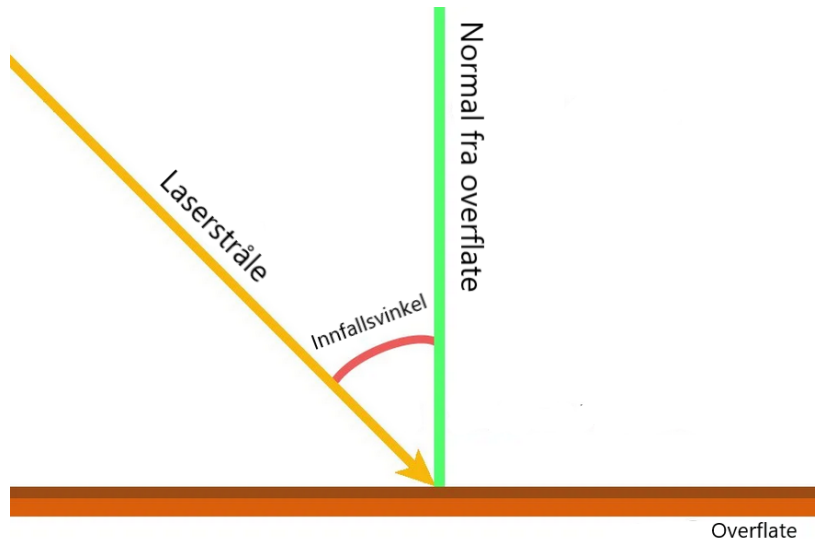
For å kunne korrigere intensitetsverdien for hvert enkelt retursignal i punktskyen måtte variablene i formelen over også finnes for hvert enkelt retursignal. I CloudCompare ble punktskyene for testområdene hentet inn. Funksjonen for utregning av normaler brukt, disse var nødvendige for videre utregning av innfallsvinkler (α). Det ble brukt 2D triangulering til beregning av lokal overflatemodell ettersom denne har vist seg å gi best resultat på skarpe kanter [26]. Normalene ble gitt i form av retningsvektorer og lagt inn i som kolonner i filen. Med koordinater, råintensiteter (I_{raw}), og normaler for hvert punkt i punktskyen ble den eksportert som CSV-fil for videre behandling i Python.

I Python ble dataene hentet inn som en Pandas Dataframe. Utregning av skanneavstand (R) ble gjort ved å regne ut avstanden mellom skannerne og hvert enkelt punkt basert på koordinatene til punktet og skanneren. Dette kunne gjøres fordi punktskyen ikke var georeferert og brukte derfor sitt egne lokale XYZ-koordinatsystem. Innfallsvinkelen for hvert punkt ble beregnet ved å finne vinkelen mellom normalen til hvert punkt, og vektoren mellom skanneren og hvert punkt (se figur 3.5).

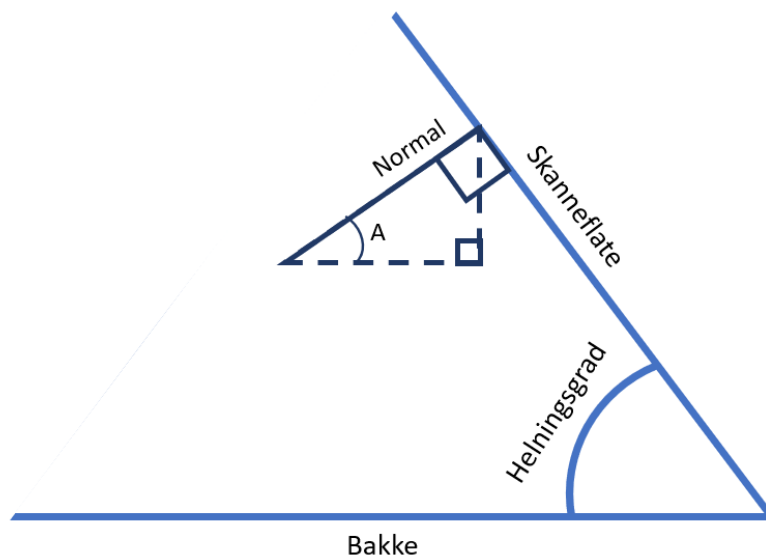
Helningsgraden til flatene ble funnet ved å betrakte koordinatene til normalen i hvert punkt som en rettvinklet trekant (se figur 3.6). Normalvektoren ble dekomponert til XY-vektor og Z-vektor. Vinkel A (se figur 3.6) ble funnet ved enkel trigonometri, tangensinvers av lengden av Z-vektoren over lengden av XY-vektoren. Videre ble helningsgraden funnet ved formelen:

$$Helningsgrad = 90^\circ - \angle A \quad (12)$$

Med helningsgraden til hvert punkt i punktskyen ble også standardavviket enkelt beregnet og brukt videre i formlene 10 og 11.



Figur 3.5: Innfallsvinkel



Figur 3.6: Helningsgrad på flaten

Med alle variablene på plass kunne intensitetsverdiene beregnes med formel (9). Dataene ble så hentet inn i CloudCompare igjen og fargelagt basert på den korrigerste intensiteten.

Samtlige av beregningene i forbindelse med korrigerst av intensitetsverdiene ble gjort i Pandas Dataframe. Dette fordi Pandas Dataframe er effektiv til håndtering av store mengder data, og selv om kun to små utsnitt fra den totale punktskyen fra posisjon 1 ble brukt til databearbeiding, var det fortsatt snakk om 1.7 millioner punkter.

3.4 Kantdeteksjon

I CloudCompare ble det lagret bilder fra begge utsnittene med farger basert på både ukorrigert- og korrigert intensitet, samt RGB-farger. Her ble det brukt png-format, da dette er et format uten kvalitetstap. Med dette unngås det en eventuell situasjon hvor viktig informasjon i bildet forsvinner som følge av komprimering. Dette kunne vært tilfellet ved bruk av andre formater som for eksempel jpeg, som bruker en komprimeringsmetode med kvalitetstap. Det ble interpolert en mesh til begge utsnittene ved hjelp av triangulering med *k nearest neighbor* [27] via funksjonen “mesh” i CloudCompare. Denne ble tatt bilder av med alle tre fargekombinasjonene. Dette for å ta høyde for at ”hull” (steder uten tilstrekkelig punkttetthet) kunne gi store utslag på kantdeteksjonen.

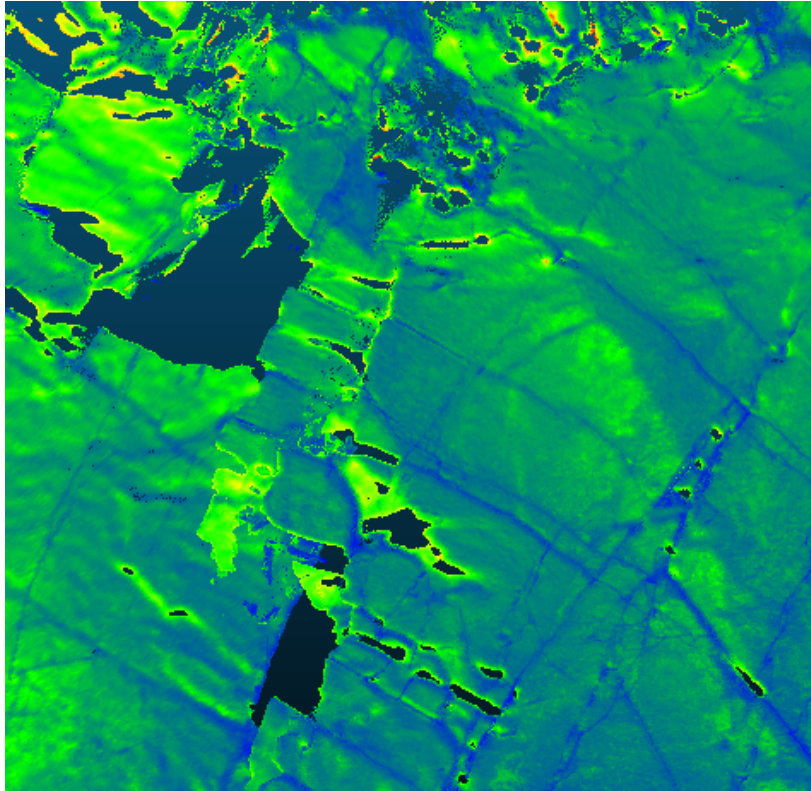
Samtlige av bildene lagret fra CloudCompare ble kjørt gjennom Canny kantdeteksjon i Python. Ulike terskelverdier ble systematisk testet for å finne høyest konsentrasjon av sprekker, samtidig som konsentrasjonen av annen støy i bildet ikke ble for høy. Etter prøving og feiling ble det konkludert med at terskelverdier på intervaller mellom 50 og 200, med hopp på 50 av gangen, ga best resultat. Det ble gjort forsøk på alle bildene, hvor de best resultatene med tilhørende terskelverdier ble lagret og sammenlignet.

Resultater

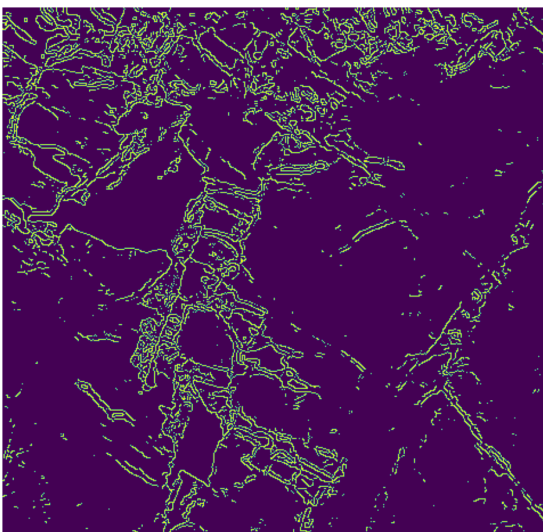
For å gjøre resultatene mer lettelserlige er utsnittet fra punktskyen som inneholder returer fra flere oppstillinger med laserskanneren kalt Punktsky 1, mens det utsnittet fra punktskyen gjort med returer fra kun en oppstilling er kalt Punktsky 2. Samtlige av forsøkene gjort med terskelverdier mellom 50 og 200 på bilder av punktskyen med de korrigerede intensitetsverdier er vist i delkapitlene 4.1 og 4.2. Resultatene inkluderer Canny kantdeteksjon utført på både punktsky og mesh fra Punktsky 1 og Punktsky 2, fargelagt basert på RGB-farger (delkapittel 4.3), råintensitetsverdier (delkapittel 4.4) og korrigerede intensitetsverdier (delkapittel 4.5). Råintensitetsresultatene er inkludert for å kunne se om det er noen klare forskjeller i forhold til korrigerede intensitetsverdier og for å se om de kan ha noen nytte på tross av de vesentlige variasjonene som kan forekomme. De vil også være nyttige for å kunne se forskjeller i punktskyer med returer fra kun en oppstilling, kontra flere.

Strukturen i resultatene er satt opp slik at bildet det er utført kantdeteksjon på er plassert til venstre, mens resultatet av kantdeteksjonen er plassert til høyre. Det er ikke inkludert noen fargeskala i intensitetsbildene siden det ikke er fargen i seg selv som er vesentlig for kantdeteksjonen. Her er det kontrasten mellom fargene og med det gradienten mellom nærliggende piksler i bildene som vil kunne gi utslag i kantdeteksjonen. Bergvegger inneholder mange ujevnheter som nødvendigvis ikke er sprekker. Dette kan være ruhet og forvitring i overflaten, tilvekst av vegetasjon som for eksempel lav og mose, fuktighet eller variasjoner i mineralsammensetning. Alle disse kan spille inn og skape støy i punktskyen og med det også påvirke meshen. Disse er det forsøkt tatt høyde for med omfattende testing gjort med terskelverdiene brukt i kantdeteksjonen. Størrelsen på bildene fra Punktsky 1 er 6,9m x 6,9m, og størrelsen på bildene fra Punktsky 2 er 3,8m x 3,8m.

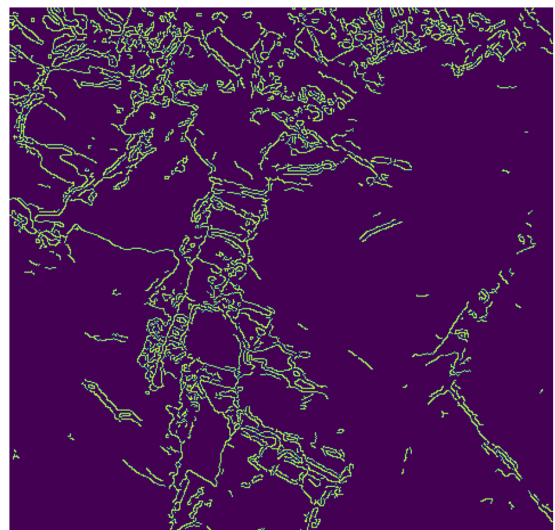
4.1 Ulike terskelverdier for Punktsky 1 med korrigeret intensitet



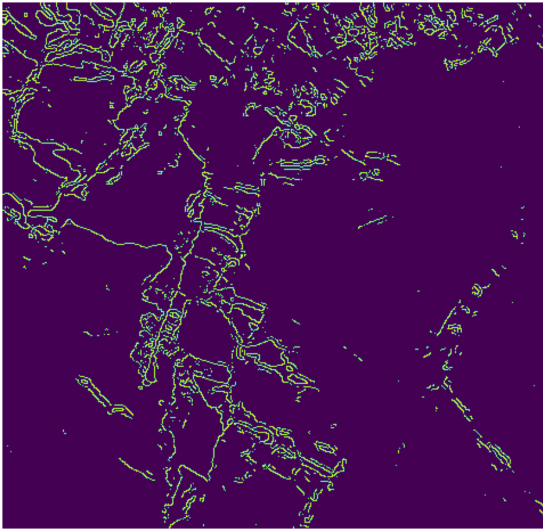
Figur 4.1: Punktsky 1, fargelagt med korrigerete intensitetsverdier



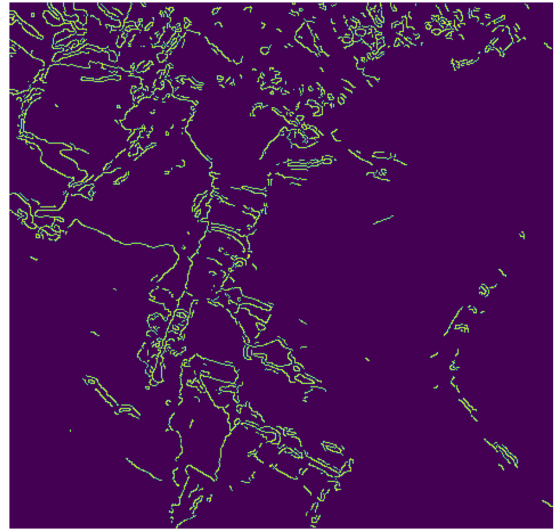
Figur 4.2: Nedre og øvre terskelverdi: 100



Figur 4.3: Nedre og øvre terskelverdi: 100 og 150



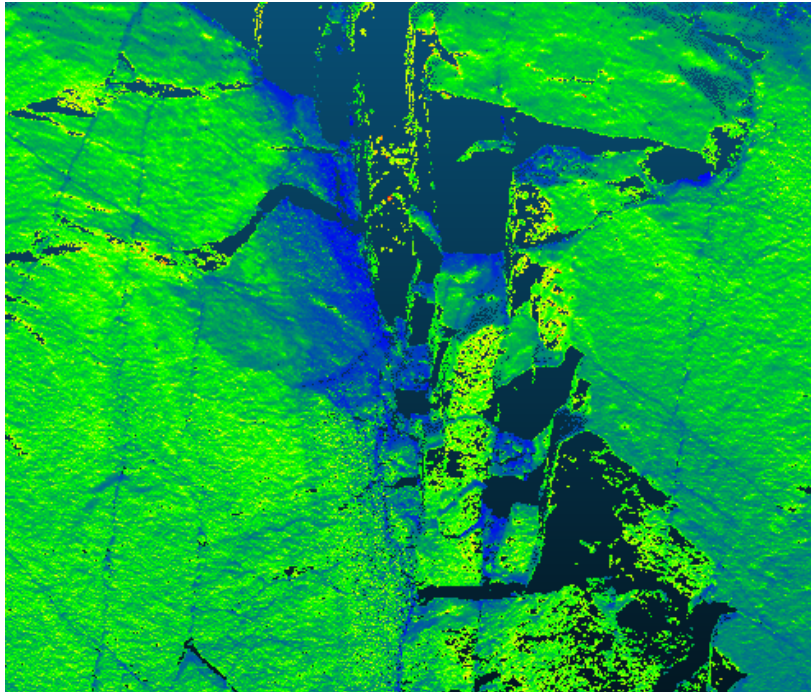
Figur 4.4: Nedre og øvre terskelverdi: 150



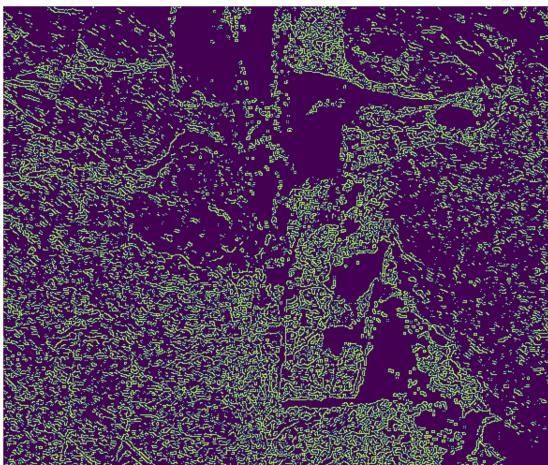
Figur 4.5: Nedre og øvre terskelverdi: 150 og 200

Testing av ulike terskelverdier på Punktsky 1, fargelagt med korrigerede intensitetsverdier viste små, men essensielle forskjeller mellom terskelverdiene. Generelt er deteksjonen av sprekker vanskelig med kun kantdeteksjonsbildet tilgjengelig. Terskelverdier under 50 resulterte i en overestimering av kanter og skapte såpass mye støy at disse ble utelatt. På motsatt side ga terskelverdier over 200 såpass stor underestimering av sprekker at også disse ble utelatt. I enkelte tilfeller ga det å unngå “svake kanter” best resultat, altså ved å sette nedre og øvre terskelverdi til lik hverandre (se figur 4.2 og 4.4). For Punktsky 1, med farger basert på de korrigerede intensitetsverdiene, ga kantdeteksjon med både nedre og øvre terskelverdi lik 100 (se figur 4.2) det beste resultatet. Her ble sprekker som ikke ble vist med høyere terskelverdi vist, mens de med lavere terskelverdi forsvant i støy.

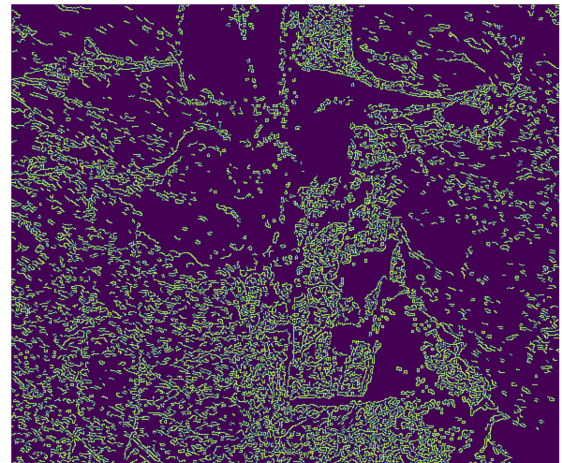
4.2 Ulike terskelverdier for Punktsky 2 med korrigert intensitet



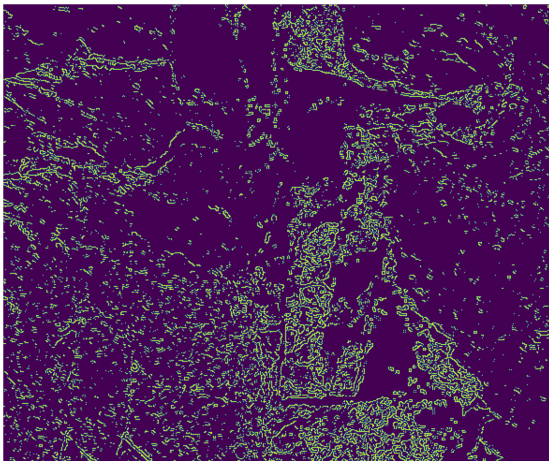
Figur 4.6: Punktsky 2, fargelagt med korrigerede intensitetsverdier



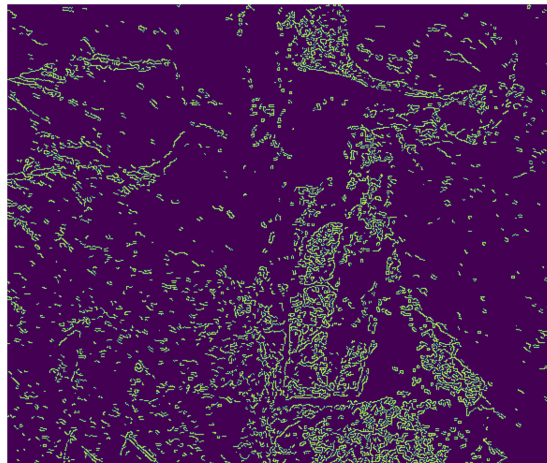
Figur 4.7: Nedre og øvre terskelverdi: 100



Figur 4.8: Nedre og øvre terskelverdi: 100 og 150



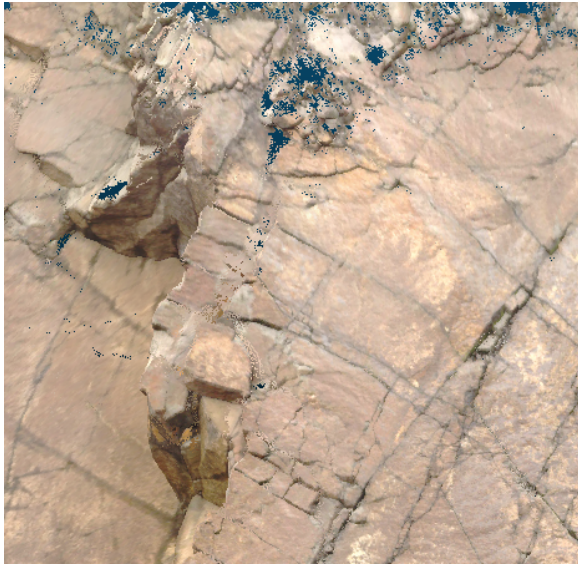
Figur 4.9: Nedre og øvre terskelverdi: 150



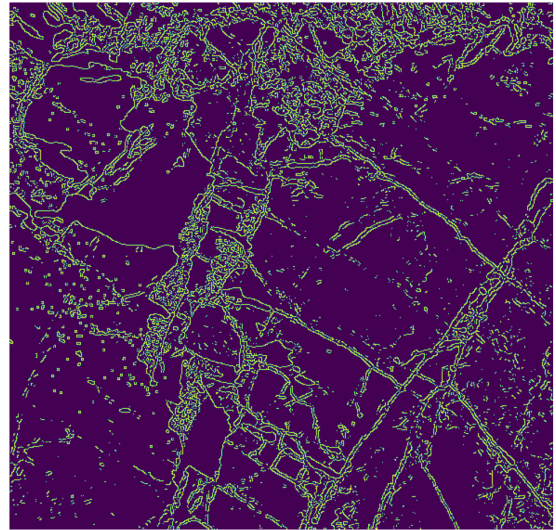
Figur 4.10: Nedre og øvre terskelverdi: 150 og 200

For kantdeteksjon av Punktsky 2 (figur 4.6) så vi en mye større forekomst av støy. Denne var krevende å fjerne uten å miste selv store og markante sprekker. Dette kommer trolig av punktettheten i punktskyen, som minker betraktelig ved bruk av kun én oppstilling. Også i Punktsky 2, med fargelegging basert på korrigert intensitet, ble det best resultat av kantdeteksjonen med nedre terskelverdi lik 100 (se figur 4.8), men her med øvre terskelverdi 150. Valget ble, på tross av den store mengden støy, bestemt ut ifra et par vertikale sprekker nede i venstre hjørne. Disse var tydeligere med terskelverdiene 100 og 150. Ved nøye studie vil man også kunne se et mønster i støyen, noe som gjøre sprekkenene relativt tydelige i det som kan virke veldig diffust ved første øyekast.

4.3 Canny kantdeteksjon av RGB-bilder, punktsky og mesh



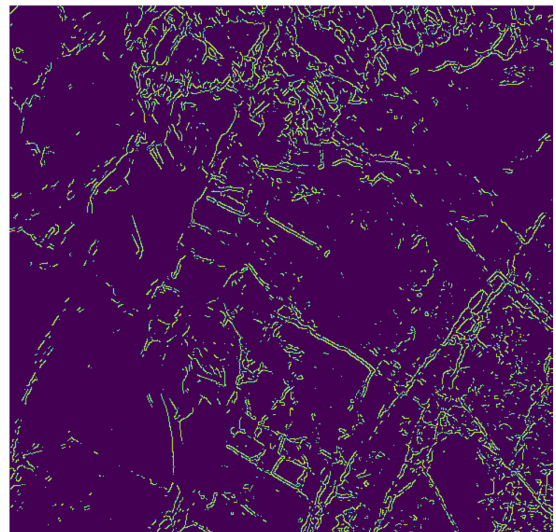
Figur 4.11: Punktsky 1, RGB-farger



Figur 4.12: Nedre og øvre terskelverdi: 100



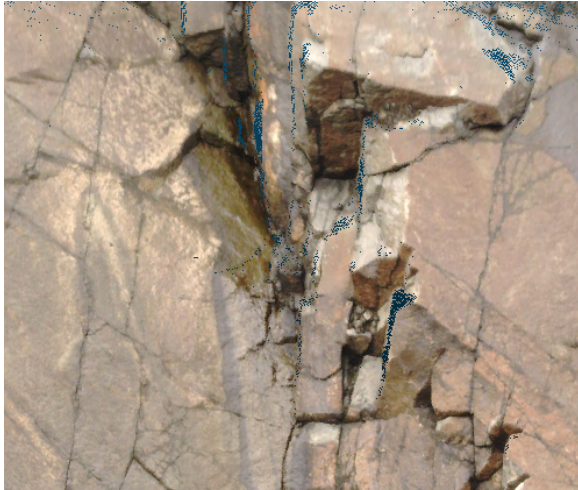
Figur 4.13: Mesh av Punktsky 1, RGB-farger



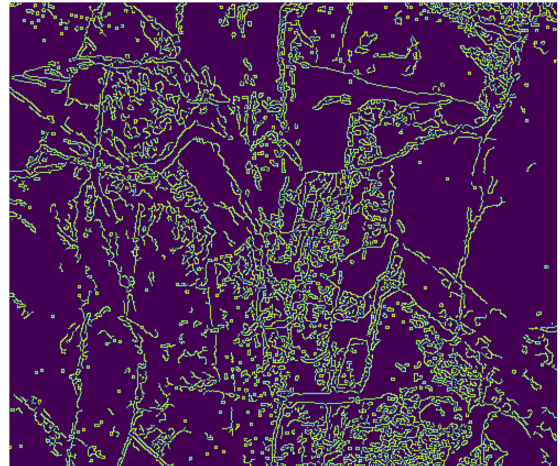
Figur 4.14: Nedre og øvre terskelverdi: 100

Utført Canny kantdeteksjon på Punktsky 1 og 2, fargelagt med RGB-farger, ga klare kanter. For både punktsky og mesh av Punktsky 1 ga terskelverdi 100 for både nedre og øvre grense de beste resultatene (se figur 4.12 og 4.14). Her ga meshen lagt over punktskyen en klar minking i støy og feilmarkerte kanter. Dette viser at eventuelle hull i punktskyen gir store utslag på kantdeteksjonen og kan bety at bruk av mesh egner seg bedre til kantdeteksjon. Motsatt ble det observert at punktskyen har mer sammenhengende kanter i resultatbildet. Felles for begge er at øvre del av bildene ga store utslag. Dette er et veldig oppsprukket område med en

betydelig mengde hull i punktskyen. Selv om både mesh og punktskyen har forholdsvis gode resultater, er begge et godt stykke unna å markere alle sprekke som er synlige i RGB-bildet.



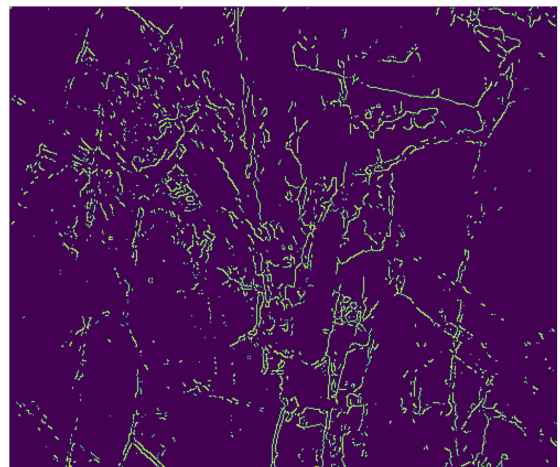
Figur 4.15: Punktsky 2, RGB-farger



Figur 4.16: Nedre og øvre terskelverdi: 50 og 100



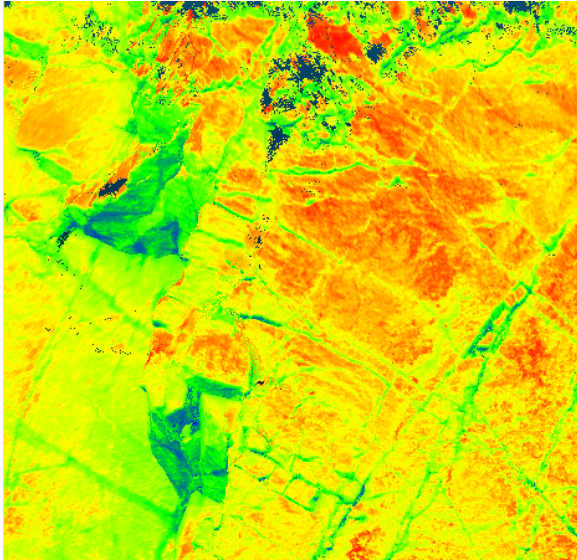
Figur 4.17: Mesh av Punktsky 2, RGB-farger



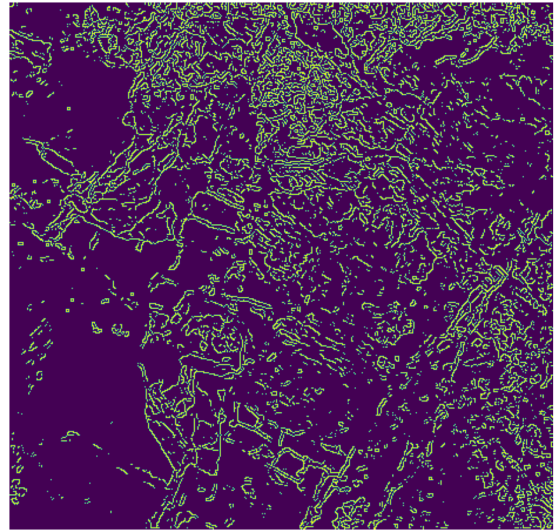
Figur 4.18: Nedre og øvre terskelverdi: 100

For Punktsky 2, fargelagt med RGB-farger, så vi en gjentakende trend med store støyforekomster som var vanskelig å fjerne uten å gi betydelig utslag på sprekkeforekomsten. Her fungerte draperingen av en mesh over punktskyen som et filter og fjerner store mengder av støyen som er mellom kantene i den sprekkesonen som beveger seg skrått mot venstre oppover i bildet. Dette uten å ha noe særlig negativ innvirkning på sprekke i sonen. Også sprekke langs den ytre del av bildet ble godt gjengitt i deteksjonen. Terskelverdiene som ga beste resultat fortsatte å være rundt 100 (se figur 4.3).

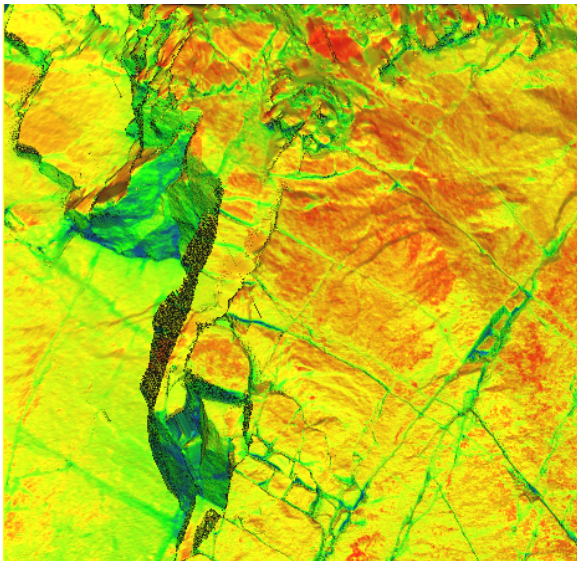
4.4 Canny kantdeteksjon av ukorrigerede intensitetsverdier, punktsky og mesh



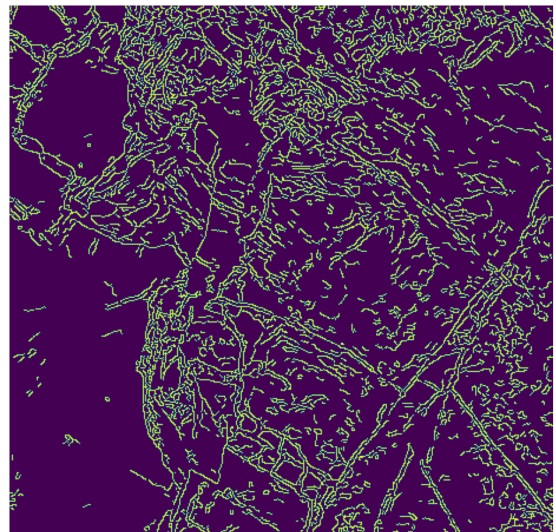
Figur 4.19: Punktsky 1, råintensitet



Figur 4.20: Nedre og øvre terskelverdi: 100



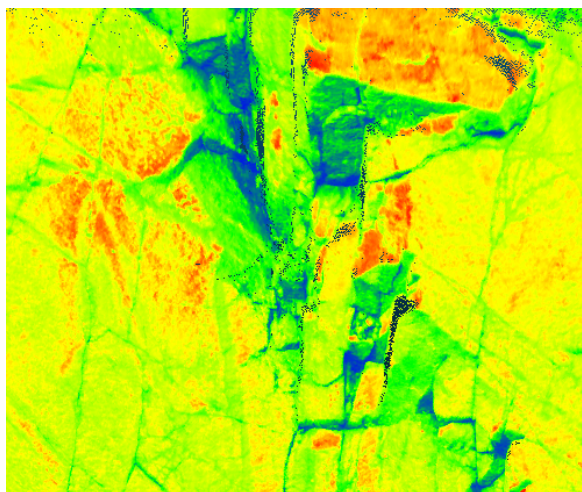
Figur 4.21: Mesh av Punktsky 1, råintensitet



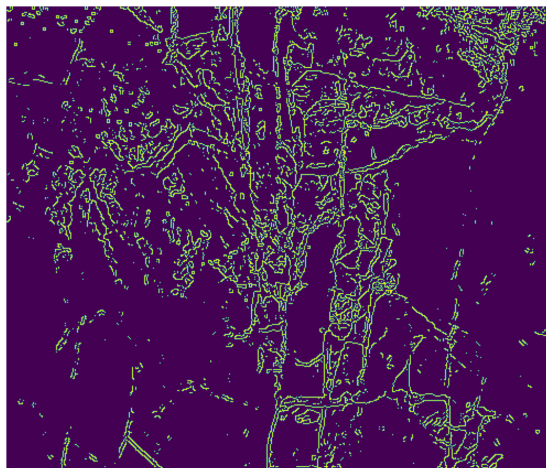
Figur 4.22: Nedre og øvre terskelverdi: 100

Påføring av Canny kantdeteksjon på Punktsky 1, med terskelverdi 100 for nedre og øvre grense, både for punktsky og mesh (figur 4.20 og 4.22), ga også støyfulle resultat. Meshen ga forholdsvis godt markerte kanter, mens punktskyen ga et rotete resultat hvor det var vanskelig å skille mellom sprekker og støy. De største og mest markante sprekkeene var mer fremtredende og mulige å kartlegge i punktskyen, mens de mindre og mer tettsittende ble glidende sammen til en røre. I meshen kom også disse mindre sprekkeene tydeligere frem, og det var mulig å skille mellom hva som var støy og sprekker. Dette fungerte selv i områder

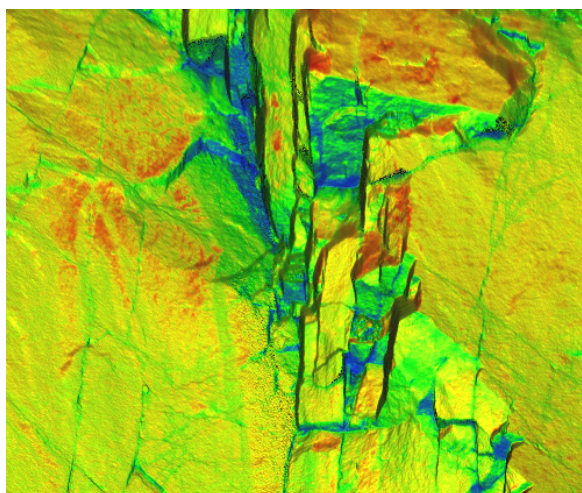
med små tettsittende sprekker da disse var mer sammenhengende i meshen, mens støyen var mer usammenhengende uten noe tydelig mønster. Vi så også at området til venstre i bildet har mindre støy (dette kommer tydelig frem i figur 4.21). Dette var lengre borte fra skanneren da blotningen har ett innhakk her. Vi ser her nytten av å korrigere intensitetsverdiene da det trolig vil gi et mer normalisert bilde av overflaten, uavhengig av avstand til skanneren.



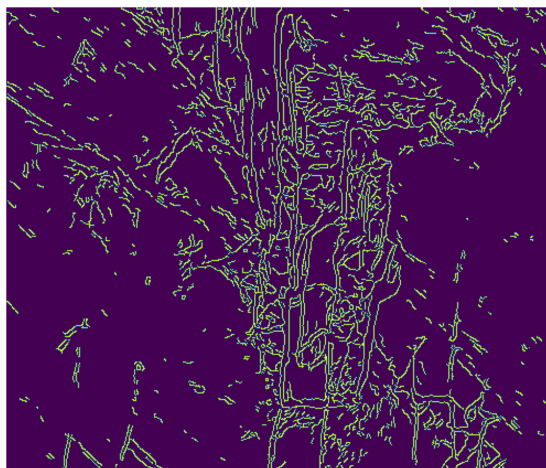
Figur 4.23: Punktsky 2, råintensitet



Figur 4.24: Nedre og øvre terskelverdi: 100



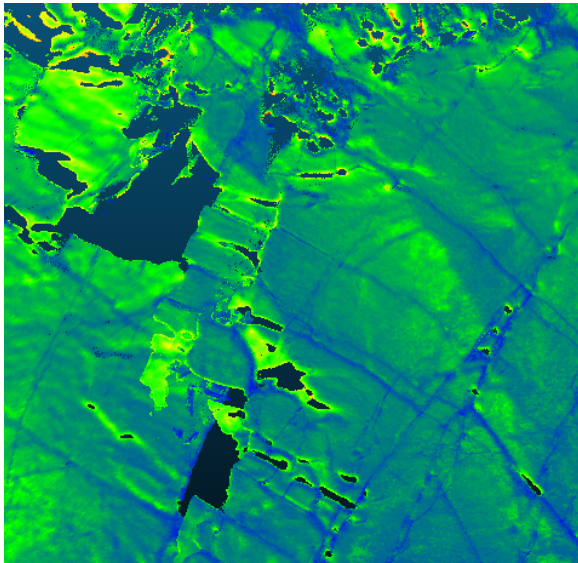
Figur 4.25: Mesh av Punktsky 1, råintensitet



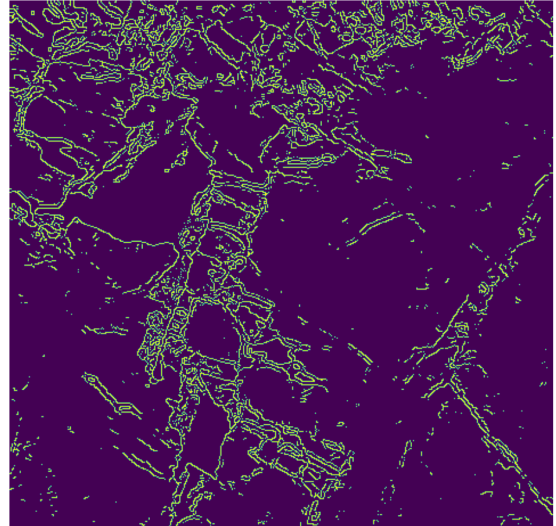
Figur 4.26: Nedre og øvre terskelverdi: 150

For resultatene av kantdeteksjonen på råintensitetsverdiene i Punktsky 2 så vi trenden med beste terskelverdi på rundt 100 fortsatte. Ellers var det vanskelig å finne sammenheng i sprekkenes forekomst utenfor den store sprekkesonen midt i bildet (dette gjaldt både figur 4.24 og 4.26). Inne i sprekkesonen ble det observert distinkte sprekker som skilte seg fra den omkringliggende støyen, spesielt i meshen (4.26). Her så vi at sprekker som førte til relativt store endringer i koordinater ga tydelige utslag i råintensitetsverdier. Dette var sprekker som ikke bare opptrådte som “streker” i berget. Dette ga igjen sterke kanter i kantdeteksjonen.

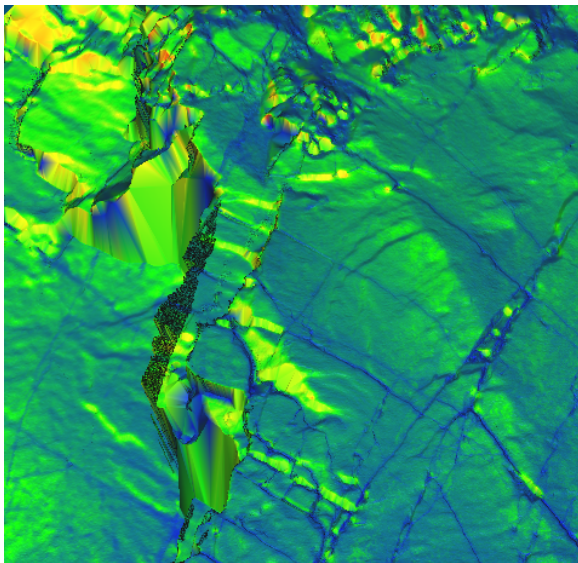
4.5 Canny kantdeteksjon av korrigerede intensitetsverdier, punktsky og mesh



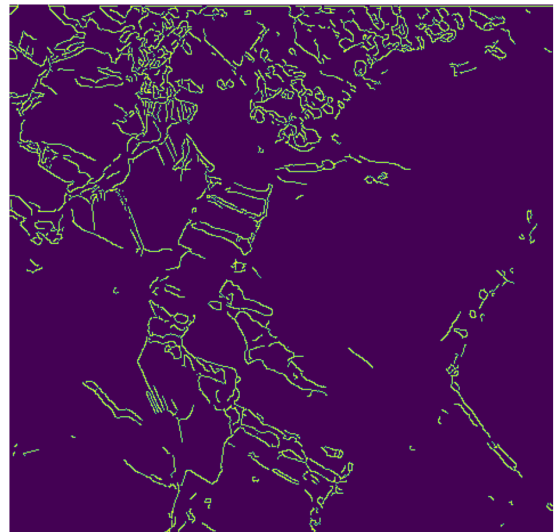
Figur 4.27: Punktsky 1, korrigert intensitet



Figur 4.28: Nedre og øvre terskelverdi: 100



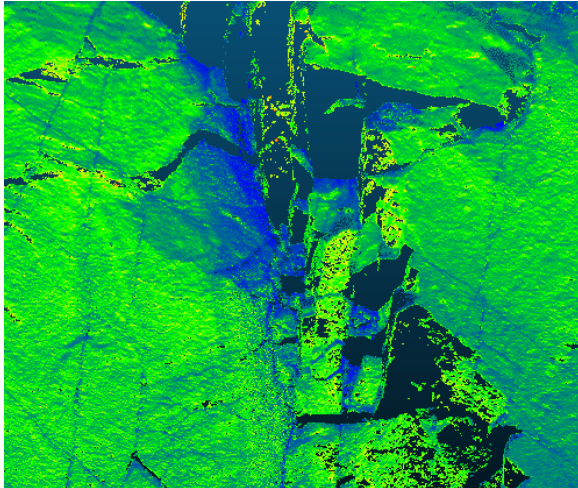
Figur 4.29: Mesh av Punktsky 1, korrigert intensitet



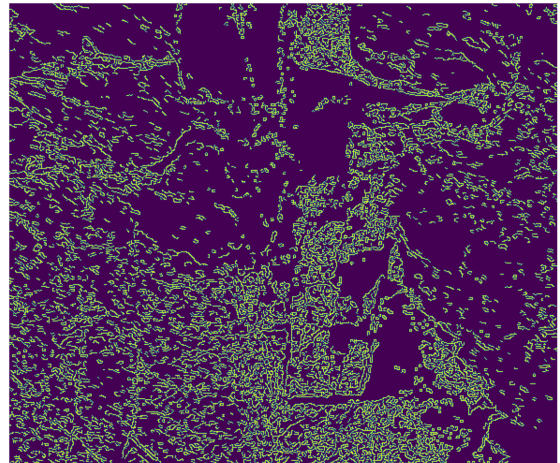
Figur 4.30: Nedre og øvre terskelverdi: 100 og 150

Korreksjon av intensitetsverdiene i Punktsky 1 og 2 viste en klar innskrenking i verdier i forhold til de ukorrigerede verdiene (se figur 4.27 og 4.29). Dette ble sett ut ifra fargene her som er mer sentrerte rundt ulike graderinger av grønn og blå. Dette hadde stor innvirkning på støyen og ga mye “renere” bilder. For Punktsky 1 så vi også effekten av å bruke mesh som filtrerer ut mye av støyforekomsten fra punktskyen, som vist i figur 4.30. Canny slet derimot med å detektere flere av de enkeltstående sprekke i bildet for både punktsky og mesh.

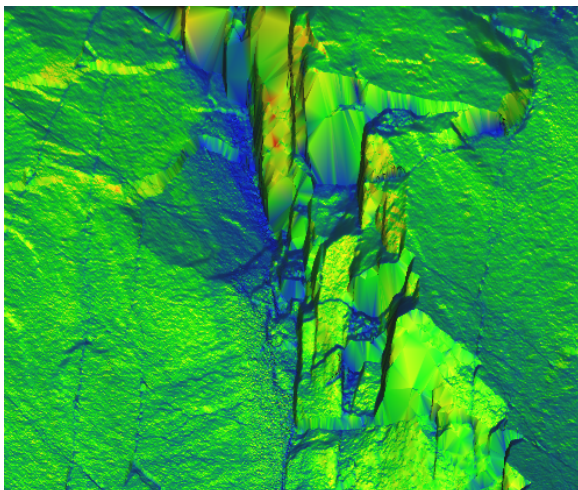
Dette på tross av at de var tydelig markerte i inputbildene. I punktskyen (figur 4.27) ble det observert store hull som ikke var tilstede i hverken RGB- eller råintensitetsbildene. Disse var resultater av komplikasjoner ved beregning av innfallsvinkler. De endte med verdier på over 90° , noe som er umulig siden de ikke ville blitt registrert som punkter da skanneren ikke ville sett disse flatene. Det ble derfor valgt å fjerne disse fra punktskyen.



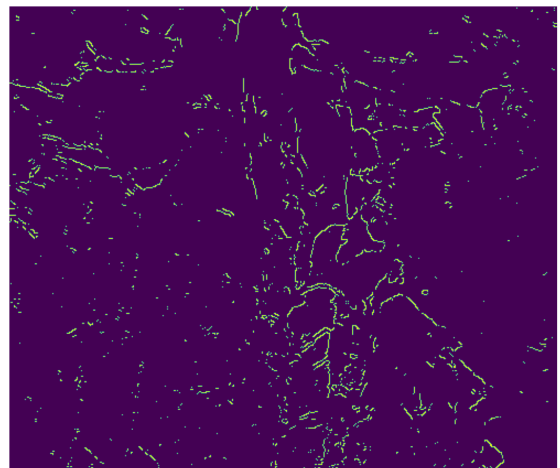
Figur 4.31: Punktsky 2, korrigert intensitet



Figur 4.32: Nedre og øvre terskelverdi: 100 og 150



Figur 4.33: Mesh av Punktsky 2, korrigert intensitet



Figur 4.34: Nedre og øvre terskelverdi: 200 og 200

For kantdeteksjon av de korrigerte intensitetsverdiene til Punktsky 2, så vi de samme problemene som tidligere med store støyforekomster (se figur 4.32 og 4.34), som trolig kommer av lav punkttetthet. Her ble det gjort forsøk på å fjerne dette ved å ta i bruk høye terskelverdier i kantdeteksjonen, men vi så konsekvensene av dette i meshen (figur 4.34), hvor kantene som vises opptrer diffust og uten noe klart mønster. Ved nøye studie av meshen kan man se antydning til sammenheng mellom de enkeltstående punktene som er utenfor senter av bil-

det. Motsatt av de ukorrigerede intensitetsdataene har ikke kantene langs sprekkesonen midt i bildet gitt like stort utslag. Generelt var det veldig vanskelig å hente ut noe informasjon fra punktskyen til Punktsky 2. De mest markerte kantene kan man se at forekommer på steder med hull i skyen, mens resten av bildet er stort sett dekket i støy.

Diskusjon

5.1 Sammenligning av resultater

Fra resultatene og kantdeteksjonen ser vi en generell trend at det er terskelverdier omkring 100 for både øvre og nedre grense som gir best resultat. Dette kan ikke settes som noen fasit, men ved videre forsøk vil det være mer tidseffektivt å jobbe seg ut ifra terskelverdier på 100 for deteksjon av kanter. Dette var tilfellet både på RGB-bildene og på intensitetsbildene, både for punktsky og mesh.

Ved sammenligning av alle kantdeteksjonsresultatene blir det synlig at benyttelse av mesh øker kvaliteten betraktelig, og sprekker blir lettere å skille fra omkringliggende støy. I tillegg fungerer også meshen bra for å minske det generelle støynivået i bildet. Det ser tilsynelatende ut som at korrigering av intensitetsverdiene også har en positiv effekt på fjerning av støy. Dette har trolig en sammenheng med at intervallet av intensitetsverdier er skrumpet inn under korreksjonen, noe som fører til mindre fargevariasjon i bildet. Det gir igjen glattere flater i bildet, og ujevnheter i veggen vil gi mindre utslag under kantdeteksjonen enn før korrigeringen. Utglattingen er ikke bare positivt da vi ser enkelte sprekker som detekteres i råintensitetsbildet, forsvinner i kantdeteksjonen av bildet for korrigerte intensitetsverdier. Trolig på grunn av gradienten mellom fargene i bildet, som blir betraktelig mindre som følge av intensitetskorreksjonen.

Skarpe kanter i fjellveggen ga veldig bra utslag på kantdeteksjonen for samtlige fargekombinasjoner, i både Punktsky 1 og Punktsky 2. Ved disse kantene skjer det også brå endring i vinkel på fjellveggen, noe som er en vesentlig kilde til hull i punktskyene da endringen i vinkel kan gi blindsoner for skanneren. Det kan derfor diskuteres om disse kantene er mer et resultat av hull i punktskyen enn de skarpe kantene i seg selv. Mer sannsynlig er det nok at det er en kombinasjon av disse som gjør at kantene fremtrer såpass tydelig i kantdeteksjonen. Her burde det vært gjennomført flere oppstillinger, med ulik vinkel til veggen for å kunne ta høyde for orienteringen til alle de ulike flatene. På den andre siden kan det vise seg å bli vanskelig å beregne innfallsvinkler ved mange ulike oppstillinger i én punktsky, hvor informasjon om hvilken oppstilling hvert punkt i punktskyen er en retur fra, er essensiell informasjon. For å kunne ta høyde for dette må hver retur merkes med et nummer som viser tilhørende oppstilling. Dette bør være nokså uproblematisk og enkelt kunne automatiseres ved videre forsøk.

Etter systematisk testing med Canny kantdeteksjon er det kommet frem et gjennomgående problem med at flere sprekker ikke detekteres. Det er ingen av forsøksbildene som gir 100% detekterte sprekker med Canny kantdeteksjon. Ut av alle forsøkene er det Punktsky 1, mesh fargelagt med RGB-farger som nevnt i delkapittel 4.3 som gir beste resultat. Her er de aller fleste sprekke detektert. Men områder med mange småsprekker og løsmasser skaper trøbbel for Canny og det blir vanskelig å skille sprekker fra støy. Dette løser den bedre på

forsøksbildet med korrigert intensitet (også her mesh av Punktsky 1) hvor det meste av støy er borte. Men her forsvinner også en del sprekker som gjør at RGB-bildet gir et bedre resultat enn den korrigerte intensiteten. Vi ser også at kantdeteksjon med Punktsky 1 gir bedre resultater enn Punktsky 2, uavhengig av fargelegging.

Et av målene med sprekke-deteksjonen i denne oppgaven var å kunne bruke den til å kartlegge sprekker som mulige passasjer for væske i Vallnesbruddet. Til forsøk med kantdeteksjon ble det hverken i Punktsky 1 eller Punktsky 2 benyttet områder fra bruddet hvor det var blitt observert utfelling av væske. For å få best mulige resultater ble det heller lagt vekt på å bruke områder i punktskyen hvor sprekken var fremtredende og visuelt lette å skille fra den omkringliggende berggrunnen. På tross av dette ga ikke kantdeteksjonen ønskelig resultat av antall detekterte sprekker. Basert på resultatene jeg fikk, er det rimelig å tro at å kjøre kantdeteksjon på delene av bruddet hvor det forekom gjennomstrømning av væske ikke ville gi noe bedre resultat på sprekke-deteksjonen. Som vi ser på figur 5.1, er den ene sprekken hvor det var blitt observert utfelling av vann lokalisert i et område med få “rene” sprekker. Selve sprekken hvor vannet siver ut er veldig tydelig, men den ligger i et område med mye oppsprukket stein, noe som vil kunne være en stor kilde til støy i en eventuell kantdeteksjonsalgoritme.



Figur 5.1: Utsnitt fra punktskyen til posisjon 1 på 1,8m x 1,8m. Den rød ringen viser hvor det var tilsig vann.

5.2 Maskinl ring

Et tilskudd til denne studien kunne v rt   ta i bruk maskinl ringbasert objekt-deteksjon. Bruk av ulike klassifiseringsmetoder kan gj re deteksjonen betydelig mer effektiv. Dette er blitt gjort i studier tidligere, blant annet “Applied Machine Learning Methods for Detecting Fractured Zones by Using Petrophysical Logs” [28] og “Machine-learning-based object detection in images for reservoir characterization: A case study of fracture detection in shales” [29] hvor sprekkesoner i oljerelaterte bergartstyper som karbonater og skifre har v rt i fokus. Begge disse studiene har gitt gode resultater med 90%+ suksess p  deteksjonene. Disse bergartstypene har ofte mer monotone overflater og fargekombinasjoner enn magmatiske bergarter, som noritten vi finner i Rekefjord. P  tross av dette vil trolig objektbasert klassifisering av sprekker ogs  kunne gi gode resultater p  bergarten i Rekefjord, da sprekkenes skiller seg tydelig ut fra den omkringliggende overflaten.

Konklusjon

Formålet med denne oppgaven var å sammenligne laserdata, mer konkret intensitetsdata fra laserskanning, med RGB-bilder. For å se om disse kunne brukes til en bedre metode for deteksjon av bergsprekker. Resultatene fra undersøkelsen indikerer at intensitetsdata ikke egner seg til sprekke-deteksjon. Selv om mange sprekker ble markert i med Canny kantdeteksjon var det også en vesentlig andel som ble unnlatt, både for intensitetsbildene og RGB-bildene. Dessuten ga RGB-bildene de beste resultatene. Det blir dermed vanskelig å tro at en vil kunne detektere flere sprekker ved bruk av intensitetsdata, enn ved å kun bruke de vanlige RGB-fargene som man ser med det blotte øye. Det var også et problem at småsprekker og løsmasser i veggen ga store mengder støy i kantdeteksjonen. I flere tilfeller var det vanskelig å skille sprekker av betydning, med andre uviktige objekter som også ga utslag i kantdeteksjonen. Korreksjon av intensitetsdataene ga heller ikke noen løsning på problemet med mangelfull deteksjonsevne. De korrigerte dataene ga riktig nok en glattere overflate som igjen ga mindre støy, men mange flere av de reelle sprekke-uteble fra deteksjonen. Generelt sett var deteksjonenevnen til Canny dårligere for de korrigerte intensitetsdataene, enn både råintensitetsdataene og RGB-bildene. Intensitetsdata kombinert med Canny kantdeteksjon egner seg derfor dårlig til deteksjon av bergsprekker. Korreksjon av disse verdiene vil heller ikke gi noen bedre metode eller utvidet funksjonalitet for bruk av laserskanning til deteksjon av bergsprekker.

Uten å ha fått noe særlig fotfeste som et nyttig verktøy innen kartlegging og klassifisering har intensitet lenge bare blitt sett på som et biprodukt ved laserskanning. Dette har i stor grad vært på grunn av de mange variablene som spiller inn på intensitetsverdiene. Ved korrigering av disse variasjonene tror jeg intensiteten til signalreturer fra laserskanner kan ha flere nytteområder. Til klassifisering og kartlegging av objekter og arealer som ellers er krevende å gjøre med RGB-farger, kan intensitetsverdier ha et bruksområde. Å komme dit er en lang vei, og bruk av Canny kantdeteksjon er trolig ikke svaret, men den kan sees på som et forsøk i en retning hvor det er uendelig mange muligheter.

6.1 Videre studier

I denne studien var intensitetsdataene til lidar i fokusområdet. Disse ble sett på i bildefor-
mat sammen med kantdeteksjon. Det ble ikke i noen stor grad benyttet den andre funksjonen til lidar, nemlig 3D-modellering. Ved videre undersøkelser kan det være interessant å benytte seg av 3D-modelleringen, og beregnede normaler til flatene som skannes for å fjerne eventuelle kanter som ikke er sprekker. Dette kan for eksempel være utstikkende kanter, noe som vil gi klare utslag i en kantdeteksjonsalgoritme som Canny. Ved å sammenligne normaler fra begge sider av kanten vil man kunne om det er en sprekke, eller bare en utstikkende kant. Dersom normalene går fra hverandre vil det være en utstikkende kant, og motsatt hvis normalene går mot hverandre.

Ved videre arbeid vil det også være interessant å vektorisere kantene. Med vektoriserte kanter kan man enkelt fjerne kanter over/under en viss lengde. Dette vil være nyttig innen sprekkkartlegging da man kan sette sprekker under en viss lengde som uinteressante for kartleggingen, avhengig etter hva man ser etter. Spesielt for studien i Rekefjord, hvor kobling mellom sprekker som vil kunne gi væskegjennomstrømning var i fokus. Her er det gitt at sprekker må være av en viss størrelse for å være aktuelle for studien.

Resultatene fra Canny deteksjon på bildene fra bergveggen var ikke tilfredsstillende nok til at metoden kan brukes til noen form for omfattende sprekkkartlegging. Dette var likevel et nyttig forsøk, og vi ser at flere sprekker trer tydelig frem. Med modifikasjoner av kantdeteksjonsalgoritmen Canny, og mulig utvikling av en ny algoritme direkte rettet mot deteksjon av sprekker i berg, vil det sannsynligvis kunne gi et bedre resultat. Det er blitt gjort forsøk med bruk av *Pulse Coupled Neural Network* [30] som alternativ til Canny kantdeteksjon for deteksjon av bergsprekker. Disse baserer seg kun på RGB-bilder fra bergsprekker, men resultatene viser god deteksjonsevne på selv komplekse sprekkesystemer.

Referanser

- [1] NMBU. *earthresQue*. URL: <https://www.nmbu.no/en/services/centers/earthresque>. (accessed: 01.02.2022).
- [2] NMBU. *Characterization of fractured rock and flow modelling to determine the natural geological barrier for new landfill sites*. URL: <https://www.nmbu.no/download/file/fid/54967>. (accessed: 05.04.2022).
- [3] J. Holmlund S. Ahlgren. «Using 3-D outcrop laserscan for fracture analysis». I: *Search and Discovery Article* 40099 (2003).
- [4] W. Lei C. Ting X. Ancheng og M. Liguang. «Automatic fracture detection based on Terrestrial Laser Scanning data: A new method and case study». I: *Computers and Geosciences* 9 (2017), s. 209–216. DOI: 10.1016/j.cageo.2017.04.003.
- [5] NOAH. *Om Rekefjord Stone*. URL: <https://rekefjordstoneas.no/om-rekefjordstone/>. (accessed: 09.04.2022).
- [6] Rekefjord Stone AS. «Driftskonsesjon Rekefjord Øst». I: (2016), s. 42.
- [7] Paul Mcmanamon. «Lidar Technologies and Systems». I: (2019).
- [8] P. R. Wolf C. D. Ghilani. *Elementary Surveying*. Pearson, 2011.
- [9] H. G. Maas G. Vosselman. *Airborne and Terrestrial Laser Scanning*. Whittles Publishing, 2010.
- [10] Geodetics. *Lidar Intensity: What is it and what are its applications*. URL: <https://geodetics.com/lidar-intensity-applications/>. (accessed: 20.04.2022).
- [11] Dario Carrea mfl. «Correction of terrestrial LiDAR intensity channel using Oren–Nayar reflectance model: An application to lithological differentiation». I: *ISPRS Journal of Photogrammetry and Remote Sensing* 113 (2016), s. 17–29.
- [12] Sanna Kaasalainen mfl. «Analysis of Incidence Angle and Distance Effects on Terrestrial Laser Scanner Intensity: Search for Correction Methods». I: *Remote Sensing* 3.10 (2011), s. 2207–2221. ISSN: 2072-4292. DOI: 10.3390/rs3102207. URL: <https://www.mdpi.com/2072-4292/3/10/2207>.
- [13] Marco Franceschi mfl. «Terrestrial laser scanner imaging reveals astronomical forcing in the Early Cretaceous of the Tethys realm». I: *Earth and Planetary Science Letters* 305.3 (2011), s. 359–370. ISSN: 0012-821X. DOI: <https://doi.org/10.1016/j.epsl.2011.03.017>. URL: <https://www.sciencedirect.com/science/article/pii/S0012821X11001610>.
- [14] Michael Oren og Shree K. Nayar. «Generalization of Lambert’s Reflectance Model». I: *SIGGRAPH '94* (1994). DOI: 10.1145/192161.192213. URL: <https://doi.org/10.1145/192161.192213>.
- [15] Nix Color Sensor. *What is RGB Color*. URL: <https://www.nixsensor.com/blog/what-is-rgb-color/>. (accessed: 21.04.2022).

- [16] FARO Technologies Inc. *FARO Laser Scanner Fococus 3D X130 February 2015*. 2015.
- [17] Direktoratet for strålevern og atomsikkerhet. *Laserklasser*. URL: <https://dsa.no/laser-og-lys/laserklasser#>. (accessed: 11.04.2022).
- [18] FARO Technologies Inc. *FARO SCENE manual*. 2021.
- [19] D Girardeau-Montaut. *CloudCompare user's manual for version 2.6.1*.
- [20] Applied Imagery. *Quick Terrain Modeler: Features*. URL: <https://appliedimagery.com/features/>. (accessed: 12.04.2022).
- [21] Statens Kartverk. *Høydedata*. URL: <https://hoydedata.no/LaserInnsyn/>.
- [22] Python. *Python, about*. URL: <https://www.python.org/about/>. (accessed: 13.04.2022).
- [23] OpenCV. *Edge Detection Using OpenCV*. URL: <https://learnopencv.com/edge-detection-using-opencv/>. (accessed: 13.04.2022).
- [24] Victor Powell. *Image Kernels*. URL: <https://setosa.io/ev/image-kernels/>. (accessed: 18.04.2022).
- [25] Yaokun Wu Siddhart Misra. «Machine learning assisted segmentation of scanning electron microscopy images of organic-rich shales with feature extraction and feature ranking». I: *Machine Learning for Subsurface Characterization* (2020).
- [26] CloudCompare. *Normals\Compute*. URL: <https://cloudcompare.org/doc/wiki/index.php?title=Normals%5CCompute>. (accessed: 25.04.2022).
- [27] IBM. *K-Nearest Neighbors*. URL: <https://www.ibm.com/topics/knn>. (accessed: 06.05.2022).
- [28] Haleh Azizi og Hassan Reza. «Applied Machine Learning Methods for Detecting Fractured Zones by Using Petrophysical Logs». I: *Intelligent Control and Automation* 12 (2021), s. 44–64. DOI: 10.4236/ica.2021.122003. URL: <https://doi.org/10.4236/ica.2021.122003>.
- [29] Xiao Tian og Hugh Daigle. «Machine-learning-based object detection in images for reservoir characterization: A case study of fracture detection in shales». I: *The Leading Edge* 37 (2018), s. 435–442. URL: <https://doi.org/10.1190/tle37060435.1>.
- [30] ChangTao He og WeiXing Wang. «A PCNN-Based Edge Detection Algorithm for Rock Fracture Images». I: *2010 Symposium on Photonics and Optoelectronics*. 2010, s. 1–4. DOI: 10.1109/SOP0.2010.5504347.



Norges miljø- og biovitenskapelige universitet
Noregs miljø- og biovitenskapelige universitet
Norwegian University of Life Sciences

Postboks 5003
NO-1432 Ås
Norway