Norges miljø- og biovitenskapelige universitet

Masteroppgave 2020 **30 studiepoeng**
Fakultet for realfag og teknologi (Realtek)

Vanntette 3D-objekter ved hjelp av fotogrammetri

Watertight solids by photogrammetry

Marte Brekken
Master (sivilingeniør) - Geomatikk

FORORD:

Med denne oppgaven markeres avslutningen på min masterutdanning innen geomatikk ved Norges miljø - og biovitenskapelig universitet (NMBU). Denne oppgaven er skrevet våren 2020 i samarbeid med Plan - og bygningsetaten (PBE) i Oslo kommune.

Først ønsker jeg å takke den hyggelige gjengen ved Plan - og temakartenheten i Geodata avdelingen i PBE i Oslo, for god hjelp og at jeg fikk sitte der. Dere har vært veldig inkluderende og oppmuntrende underveis, og hjulpet med veiledning. Det har vært hyggelig å vite at vi alltid har vært velkommen til gode lunsjsamtaler og kakefredager. I tillegg ønsker jeg og gi en stor takk til bi-veiledere fra PBE, Mario GIl Sanchez for gode råde og innspill underveis, Webjørn Finsland for tilgang til maskinpark, og software og Even Erichsen og Trym Teigen for gode råde og bildematerialet som ble helt essensielt for å kunne gjennomføre denne oppgaven.

En stor takk må også rettes til min veileder Ivar Maalen-Johansen som har gitt god veiledning når det har trengtes. Det må også rettes en takk for samarbeidet med Vegard Bjørnberg i oppstarten av oppgaven, vi hadde ikke klart oss uten det samarbeidet i starten. Tilslutt må det rettes en takk til venner og familie som har vært støttende gjennom denne prosessen.

Norges miljø- og biovitenskapelige universitet,
Ås, 2. juni 2020

Marte Brekken

SAMMENDRAG:

Per i dag bruker PBE i Oslo kommune FKB data til å rekonstruere bygginger i byen. Ulempen ved en slik rekonstruksjon er at takskjegg blir borte. Dette fordi programmet som PBE bruker tar inn taklinjer fra FKB dataen og lager rette linjer fra takket og ned til bakkenivå. PBE ønsker med det å se på en annen løsning for rekonstruksjon av bygninger med fotogrammetri. De sitter på en del materialet. PBE har en del flybilder fra forskjellige år, skanninger med bil og fly som de ønsker å utnytte. I forbindelse med denne oppgaven ble det også tatt en del bakkebilder for supplering til eksisterende materialet.

PBE satser på visuell kommunikasjon som betyr at PBE ønsker å formidle byutviklingen på en opplysende, og mer forståelig måte ved bruk av digitale og tekniske løsninger. Med det menes at en borger kan se byplaner i 3D og ikke bare i et 2D kart.

Min oppgave har til hensikt å støtte studien ved å kartlegge mulighetene for å lage 3D modeller ved fotogrammetri. Der jeg ser på ulike programmer som kan være aktuelle for formålet. Samt si noe om forventet nøyaktighet og kvalitet på visualisering av modell for de aktuelle programmene.

For å finne ut hvilke programmer som er aktuelle for denne oppgaven og i hvor stor grad de er automatiserte, har jeg valgt å se på lisensbelagt programmer. PBE ønsker å kunne bruke datasett de allerede har liggende og lage vanntette solids. De ønsket å se på en annen måte å lage solids på enn med dagens metode.

PBE ønsker at programmene lager en 3D modell, der den kan skaleres den til riktig størrelse, og georefereres. De tre programmene jeg endte opp med å sammenligne, det var Agisoft Metashape, Reality Capture og Pix4d. Det blir også undersøkt hvor god kvalitet og nøyaktighet hver modell får med bruk av ulikt bildematerialet i hvert program. Dette for å se på forskjeller mellom programmer, med forskjellige datasett.

Datasett som blir benyttet er data som har blitt fotografert av Terratec i 2018 og laser i 2019, samt en lasersky fra ruter som ble tatt i 2017. I tillegg har det blitt fotografert bakkebilder i 2020 med speilreflekskamera. Så all materialet er tatt fra bakke, fly, eller bil. Oppgaven har kun konsentrert seg om å modellere bygninger ikke vegetasjon rundt byggene.

I oppgaven er det to testområder, et kvartal og et homogent bygg med mye vinduer. Dette for å se hva de ulike programmene taklet av type overflate. Det blir gjort en rekke analyser med ulike typer datasett for de to områdene. Fokuset ligger på fotogrammetri, siden ikke alle programmene taklet laserskyer.

Ser på ulike alternative metoder en kunne valgt, og drøfter resultatene ut fra testkriterier som ble satt opp. Konkluderer med at det er mulig å kombinere fly- og bakkebilder til felles punktsky som gir bedre kvalitet og nøyaktighet for en 3D modell enn med kun flybilder. Sjekker dette opp mot kravspesifikasjoner til PBE. Det kommer også godt fram at utvikling av programmene ikke er helt ferdig. Enkelte funksjonaliteter som skalering og georefering funker dårlig i enkelte av programmene. Det blir brukt støtte program for å løses disse funksjonalitetene.

ABSTRACT

Today, PBE in Oslo Municipality uses FKB data to reconstruct buildings in the city. The disadvantage of such a reconstruction is that the eaves are gone. Because the program, PBE uses takes roof lines from the FKB data and creates straight lines from the roof down to the ground level. PBE wants to look at another solution for the reconstruction of buildings with photogrammetry. They sit on many datasets. PBE has different scans of cars and planes. In connection with this thesis, some ground pictures were also taken to supplement the existing datasets.

PBE focuses on visual communication, that means that PBE wants to communicate urban development in an enlightening, and more understandable, way of using digital and technical solutions. This means that a people can see city plans in 3D and not just in a 2D map.
My task is to support the study by mapping the possibilities of making 3D models by photogrammetry. Where I look at various programs that maybe relevant for the purpose. And say something about the expected accuracy and quality of model visualization for the relevant programs.

To find out which programs are relevant to this task and to what extent they are automated, I have chosen to look at licensed programs. PBE wants to use data sets they already have and create solids. They wanted to look at a different way of making solids relative to today's method.

PBE wants the programs to create a 3D model where scaled to the correct size and georeferenced. The three programs I ended up comparing were Agisoft Metashape, Reality Capture and Pix4d. It also examines how good quality and accuracy each model gets with different imagery. Givs differences result for each program.

Datasets used are data that have been photographed by Terratec in 2018 and laser in 2019 and 2017. In addition, ground images this spring, 2020 with SLR cameras. So all the material is taken from ground, plane, or car. The task has only concentrated on modeling buildings not vegetation.

In this task there are two test areas, a quarter and a homogeneous building with lots of windows. To see what different surface ecah software tackled. A number of analyzes are done with different types of data sets for the two areas. The focus is on photogrammetry, since not all programs can deal with point clouds.

Looking at various alternative methods we can chosen, and discussing the results based on test criteria. Concludes that it is possible to combine fly and ground images into common point clouds that provide better quality and accuracy for a 3D model than with fly images only. Checking this against PBE's requirements specifications. It is not complete development each programs. For some function such as scaling and georeferencing, support programs must be used in order to do so.

INNHOLDSFORTEGNELSE:

FORORD: II
SAMMENDRAG: III
ABSTRACT V
INNHOLDSFORTEGNELSE: VII
FIGURLISTE: X
TABELLISTE: XII
FORKORTELSER: XIII
1. INTRODUKSJON: 1
1.1 Bakgrunn: 1
1.2 Problemstilling: 3
1.3 Forutsatte kunnskaper: 4
2. 3D REKONSTRKSJON: 5
2.1 Deteksjon av sammenbindingspunkt: 5
2.2 Structure from Motion: 8
2.3 Dense image matching: 10
2.3.1 Patced based metods: 11
2.3.2 Semi global matching: 14
2.4 Bundle adjustment metoden: 15
2.5 Faktorer som påvirker en 3D rekonstruksjon: 15
2.6 FKB - Felles kartdatabase: 18
3 BESKRIVELSE AV SOFTWARE: 19
3.1 Agisoft Metashape: 19
3.1.1 Align image: 20
3.1.2 Build dense cloud: 23
3.1.3 Build mesh: 24
3.2 Reality Capture: 26
3.2.1 Alignment 26
3.2.2 Reconstrusjon: 29
3.3 Pix4d Mapper: 31
3.4 Støtte software: 33
3.4.1 ArcGIS PRO: 33

3.4.2 Autodesk ReCap 33
3.4.3 FME - Feature Manipulation Engine: 34
3.4.4 NT 3Dbygger: 34
3.5 Andre software: 34
4. METODE: 36
4.1 Prosjekter: 36
4.1.1 Tilgjengelig data: 39
4.1.2 Egen datainnsamling: 39
4.2 Arbeidsgang: 40
4.2.1 Agisoft Metashape: 40
4.2.2 Reality Capture: 44
4.2.3 Pix4d: 47
4.3 Oslo kommunes spesifikasjoner/krav: 48
4.4 Solids/ måling: 48
4.5 Analyse av solids: 49
5. RESULTAT OG DISKUSJON: 50
5.1 Grad av automatisering 50
5.1.1 Agisoft Metashape: 50
5.1.2 Reality Capture: 51
5.1.3 Pix4d Mapper: 53
5.2 Ivaretakelse av geometri og tekstur: 53
5.2.1 Agisoft Metashape: 53
5.2.2 Reality Capture: 56
5.2.3 Pix4d Mapper 58
5.3 Vanntett solids: 61
5.3.1 Agisoft Metashape: 61
5.3.2 Reality Capture: 63
5.3.3 Pix4d Mapper 65
5.4 Sammenslåing av datasett: 65
5.4.1 Agisoft Metashape: 65
5.4.2 Reality Capture: 67
5.4.3 Pix4d Mapper 69
5.4.4 Solids med ulikt datasett: 72
5.5 Hvilken nøyaktighet kan oppnås: 75
5.5.1 Resultat: 75

5.5.1 Agisoft Metashape: 75
5.5.2 Reality Capture: 76
5.5.3 Pix4d Mapper: 76
5.5.4 Feilkilder: 77
5.5.5 Del konklusjon: 77
5.6 Andre data og bruksområder: 79
5.6.1 Sammenligning av Feature detection og matchingsalgoritmer: 79
5.6.2 Agisoft Metashape: 79
5.6.3 Reality Capture: 80
5.6.4 Pix4d: 81
5.7 Oppsummering av resultatet: 82
6.KONKLUSJON: 85
7.VEIEN VIDERE: 87
REFERANSELISTE: 88
VEDLEGG: 90

FIGURLISTE:

Figur 1 Eksempel på Feature detection 6
Figur 2 Sammenbindingspunkter er de som generer en punktsky 8
Figur 3 Et eksempel på at punktene ikke er helt riktig i forhold til hverandre. 9
Figur 4 visualisering av celler og patcher. [Furukawa et al..] 12
Figur 5 Ekspansjon, [Furukawa et al.] 13
Figur 6 Punktsky fra sporveien med 20 000 per kvadratmeter. 17
Figur 7 Punktsky fra terratec med 10 per kvadratmeter. 17
Figur 8 Bakkepunktsky fra Agisoft Metashape. 21
Figur 9 Parameter endring i Align Prosessen. 23
Figur 10 Parameter endring i Build Dense Cloud prosessen 24
Figur 11 ut fra denne oversikten kan mn lese antall bilder som er tatt ing og antall bilder som er klart å bli brukt. Antall merker som er satt inn. Hvor mange punkt de forskjellige punktskyene har og hvor mange tette punktskyer som er lagt inn. Dense cloud som ikke står i kursiv er den som blir brukt til å danne mesh og antall punkt i parantes. I meshen står det antall polygoner som er laget i meshen. 25
Figur 12 Parameterinnstilling i RC 27
Figur 13 Parameterinstillinger for Align prosessen. 28
Figur 14 Punktsky eksempel fra testområde 2 29
Figur 15 Parameterinstillinger i reconstruction. 30
Figur 16 Eksempel på modell fra terrassemodellen. 32
Figur 17 Sammensetning av punktskyer. 33
Figur 18 Øverst til venstre er kvartalet og nederst til høyre er nabobygget. De røde linjene er bygningslinje og grønne flatene er bygningsflater. 36
Figur 19 Testområde 1 fargebeskrivelse; rosa = testområde, gul =kvartalsoner, Orange= sporveienlasersky trase 37
Figur 20 Dekning av kvartalet fra sporveien. 20 000 punkt per kvm 38
Figur 21 Testområde 2 38
Figur 22 Punktsky fra sporveiene. Den ble klipt til så den bare dekket kvartalet. 39
Figur 23 Parameterinnstillinger for Align prosessen. 41
Figur 24 Parameterinnsillinger for Build Dense Cloud prosessen. 42
Figur 25 Kontrollpunkt for kvartalet 42
Figur 26 Parameterinnstilling for Build Mesh prosessen. 43
Figur 27 Bilde av generate report 44
Figur 28 Parameterinnstillinger for Alignment prosessen. 45
Figur 29 Parameterinnstillinger for Reconstruction prosessen. 46
Figur 30 Parameterinnstillinger for Simplify funksjon. 47
Figur 31 Eksempler på vanntett solid. 49
Figur 32 Vegetasjons utfordringer. 54
Figur 33 Hele bakgården med fly bilder, kan se her at teksturen ikke er punktskyen som meshen blir laget av er vanntett. 55
Figur 34 Modell fra bakke - og flybilder, testområde 1. 55
Figur 35 Modell fra bakke - og flybilder med teksturering. 56
Figur 36 testområde 2, vi ser tydelig hva som vises på vinduet og kan se det sitter en person der og hva han har på pulten. Vinduene med noe i er lettere å lage en solids på. Det er ser bedre ut med tekstur på. 56
Figur 37 Bilde av modell med tekstur, kvartal i fly perspektiv. 57
Figur 38 Testområde 1, Kvartalet. Ser at det er liggende bilder, dårlig til å takle flyfoto. 58
Figur 39 Testområde 1, med flybilder 59
Figur 40 Testområde 1, solid, kun bakkebilder. 60

Figur 41 Testområde 2, bakkebilder 60
Figur 42 Testområde 2, flybilder..... 61
Figur 43 Testområde 1, flybilder, Agisoft Metashape 62
Figur 44 Testområde 1, kombinasjon med bakke - og flybilder. 62
Figur 45 Testområde 2, flybilder, Reality Capture. 63
Figur 46 Testområde 1, flybilder..... 63
Figur 47 Testområde 2, kombinasjon av bakke - og flybilder 64
Figur 48 Testområde 1, kombinasjon av bakke - og flybilder 64
Figur 49 Testområde 2, kombinasjon av bakke - og flybilder. veggen mot PBE 66
Figur 50 Testområde 1, fra gata 66
Figur 51 Testområde 1 fra bakgård, Det er kun veggen som er tatt bakkebilder på ikke det i bakgården. 67
Figur 52 Testområde 2, viser kontrollpunkter..... 68
Figur 53 Testområde 2, Hele bygget med bakke - og flybilder. 69
Figur 54 Testområde 1, veggen ut mot gaten med kombinasjon av bakke - og flybilder..... 69
Figur 55 Testområde 1, bakkebilder, teksturering. 70
Figur 56 Testområde 1, kun flybilder modell. 71
Figur 57 RSMD (meter) fra Agisoft Metashape modellene..... 78
Figur 58 RSMD (meter) fra Reality Capture modellene..... 78
Figur 59 Dense cloud for testområde 1, kvartalet, 292 121 777 punkt. 80
Figur 60 Dense cloud testområde 2, nabobygget, 40 740 909 punkt. 80
Figur 61 Testområde 1,kvartal, 78 652 punkt. Figuren blir bedre når det blir lagt på struktur. 81

TABELLISTE:

Tabell 1 Kravspesifikasjoner 48
Tabell 2 flybilder, oppsummering 72
Tabell 3 Bakkebilder, oppsummering 73
Tabell 4 kombinasjon av datasett, oppsummering 74
Tabell 5 Oppsummering ved bruk av lasersky. 74
Tabell 6 Viser beregnet RMSD for alle målingene. 75
Tabell 7 Oppsummering av resultatet 82
Tabell 8 Forklaring på symboler 83

FORKORTELSER:

PBE – Plan og bygningsetaten i Oslo Kommune.

GSD – «Ground sample distance», hva avstand fra seter i et piksel til senter i det neste tilsvarer i avstand i bakken.

SGM – Semi Global matching, se kap. 2.3.2

NDVI - «Normalized difference vegetation index»

DSM – overflate modell

NMBU – Norges miljø- og biovitenskapelig universitet

MP- Megapiksler

RSMD - «Root mean square deviation»

FME - Feature Manipulation Engine**,** kap. 3.4.3

1. INTRODUKSJON:

1.1 Bakgrunn:

Bakgrunn for denne oppgaven er at Plan- og bygningsetaten ønsker å se på muligheten til å bruke eksisterende datasett til å lage 3D modeller/3D solids. Hvilke programmer som kan utnytte dataen best mulig til å få solids for bruk til 3D printing eller i en VR opplevelse.

PBE satser på visuell kommunikasjon som betyr at PBE ønsker å formidle byutviklingen på en opplysende, og folkelig måte ved bruk av digitale og tekniske løsninger. Med det menes at en borger kan se det visuelt for seg, i stede for i et kart.

PBE hadde i 2019 ute et anbud på et prosjekt som skulle sikre videre utvikling av kommunen i denne sammenheng. De ville arbeide fram bymodeller, områdemodeller og saksmodeller for å kunne se Oslos utvikling visuelt. Verktøyene og metodene som ble brukt skulle brukes i arbeidet for å tilpasse og skalerer det aktuelle planarbeidet, for egne planer men også for innsendte planer og hvordan det fungerte i samspill med hverandre. [Oslo kommune, 2019]

PBE ønsket å utrede ulike produkter med både bruk av data fra PBE og ny innhentet data. Ble i fjor lagt ut et anbud som baserer seg på bruk av eksisterende dat og innhenting av data og kombinasjon av dette. I alt var anbudet delt inn i 4 deler. Det er:

1. <u>3D-modell basert på PBEs skrå - og vertikalfoto:</u>
   At det kunne lages en 3D – mesh med teksturering som er generert med fotogrammetri/ bildematching. Kvaliteten på 3D meshen ønskes så god datagrunnlaget tillater. En slik modell kan brukes til 3D – visualisering av PBEs byplanlegging.

   Nøyaktigheten her må være bedre enn 20 cm i grunnriss og høyde altså en GSD = 5 og 3D-mesh må være vanntett slik at de kan printes. [Oslo Kommune, 2019]

2. <u>3D-modell basert på PBEs FKB SOSI data:</u>
   Det vil si å bygge modeller ut fra FKB data, det er det som blir brukt i dag i NT 3D bygger. Men PBE ønsket å kunne bruke skråbilder til teksturering av modellene som blir laget i NT3D bygger. Det å kunne skille terreng og bygninger fra hverandre. Og

legge på riktig teksturering på terreng og bygninger. I tillegg ønskes det da at bygninger kan hentes med bygningsnummer og settes inn i modellen. Denne modellen blir da brukt i 3D visualisering som omgivelser og 3D analyse av vind, skygge, overvann. [Oslo kommune, 2019]

3. <u>3D-modell basert på helt ny datafangst:</u>
   PBE er interessert i en veldig detaljert modell basert på bildematching, med en så høy oppløsning som mulig i teksturen. I denne delen var det nødvendig med ny datafangst for å kunne få en så detaljert modell. Det ønskes at det er muligheter for en kombinasjon av flere datasett som er hentet fra både terrestrisk og flybårne sensorer. Det kan skape en bedre nøyaktighet og kvalitet i modellen. PBE ønsker med dette er å kunne stå på gatenivå i VR og å ha en så god oppløsning i teksturen at det er leselig det som står på teksturen. Modellen kan også brukes til 3D printing. [Oslo kommune, 2019]

4. <u>3D-modell basert på helt ny datafangst, med oppdeling:</u>
   Dette er en videre føring av punkt 3. At terreng og bygninger blir seperarte og kunne skille bygninger ut fra bygningsnummer. Slik at det i en modell er mulig å separere bygninger ut fra bygningsnummer. Formålet med denne delen er at 3D modellen skal brukes i 3D visualisering av omgivelser og 3D analyse. [Oslo kommune, 2019]

Disse delene sier noe om PBEs ønske slik at de kan utvikle seg og var anbudet i 2019. Dette var et veldig stort prosjekt og krevde mye tid. Slik at ingen firmaer klarte å gi et forslag eller gjennomføre hele prosjektet. Det kom inn løsning på punkt 1 og 2 men ikke på punkt 3 og 4 pga. inn henting av data som tar ekstra tid.

Det ble dat uttrykt et ønske om at det ble testet med innhentet data. Dette for å se om nøyaktigheten og kvaliteten ble bedre. Det ble i den sammenheng valgt ut tre programmer for videre testing av anbudet fra 2019. Punktene som spesifikt blir sett på i denne oppgaven er punkt 1 og 3. Dette for å kunne se på forskjeller i kvalitet og nøyaktighet mellom eksiterende data og kombinasjon av eksiterende og ny data. I sammenheng med dette ønskes det å se på forskjellige måter og lage vanntette solids på. Noe som går inn i punkt 1 og 3.

Oppgaven bærer preg av at det var Corona epidemi når den ble gjennomført og skrevet. I form av at det ble hjemmekontor, mindre tilgang på programvare og vanskelig å innhente ny data. Det ble gjort det beste ut av situasjonen med bruk av data som ble hentet inn før Corona brøt ut og eksiterende data fra PBE.

1.2 Problemstilling:

Formålet med denne oppgaven er å undersøke kombinasjon av ulike datasett som representerer samme områder, til konstruksjon av vanntette 3D-mesh med ulik programvare.

<u>Problemstillingene som skal besvares er gitt som følgende:</u>

Hovedproblemstilling:

- Hvor godt egnet er fotogrammetri til å lage vanntette 3D- objekter for en bymodell?

For å kunne svare på hovedproblemstillingen har jeg underproblemstillinger som skal belyse på hvordan hovedproblemstillingen oppfylles:

- Hvilken eksisterende software er i stand til å gjøre det?
- Hvilket bildemateriale trengs for å lage det?
    - Hvilke faktorer påvirker gjennomføringen av en rekonstruksjon ved bruk av bilde.
- Hvordan er de ulike programmene tilrettelagt for å gjøre målinger og hvilken nøyaktighet gir disse?

1.3 Forutsatte kunnskaper:

Forutsetter at de grunnleggende teoriene bak begrepet fotogrammetri er kjent for leseren. Jeg vil ikke gå inn på teorien som omhandler dette. Med dette mener jeg f.eks.:

- Perspektivisk samband
- Sentralprojeksjon
- Kolinearitetsprinsippet
- Fram og tilbakeskjæring
- Indre - og ytre orientering
- Grunnleggende bildematching
- Flybåren laserskanning
- Triangulated Inrregular network (TIN)

Teori og algoritmer for TIN, teksturering vil ikke bli omhandlet i vesentlig grad da dette ikke er vektlagt i oppgaven. Oppgaven legger vekt på tetting av solids.

2. 3D REKONSTRKSJON:

I prosessen med å lage en 3D- rekonkonstruksjon er det mange steg man skal igjennom. Basert på litteratur og studier av programmene som er brukt har jeg kommet fra til følgende liste:

1. Innlesning av data (bilder, punktsky)
2. Deteksjon av sammenbindings punkt for bilder
3. Matching av bilder.
4. Orientering av bildene med spredt punktsky.
5. Orientere punktsky med kjent bakkepunkt.
6. Generering av felles og tett punktsky.
7. Ta en rekonstruksjon av felles punktskyen
8. Legge på tekstur.

Siden faget er i stadig utvikling, har jeg valgt å gå i dybden på enkelte deler av prosessen der den er dårlig forklart i skolelitteraturen og som er veldig sentral for oppgaven. Dette skyldes at vi stadig er i utvikling av metode og algoritmer for programmene samt at det blir stilt krav til ytelse og hastighet. De generelle prinsippene innen fotogrammetri gjelder enda. Har valgt å gå i dybden på de emnene som har vært mest i utvikling:

Hovedemne som blir beskrevet grundigere.

- Deteksjon av sammenbindingspunkter
- Structure from Motion
- Bildematching/ generering av felles punktsky som fører til rekonstruksjon av objekter

2.1 Deteksjon av sammenbindingspunkt:

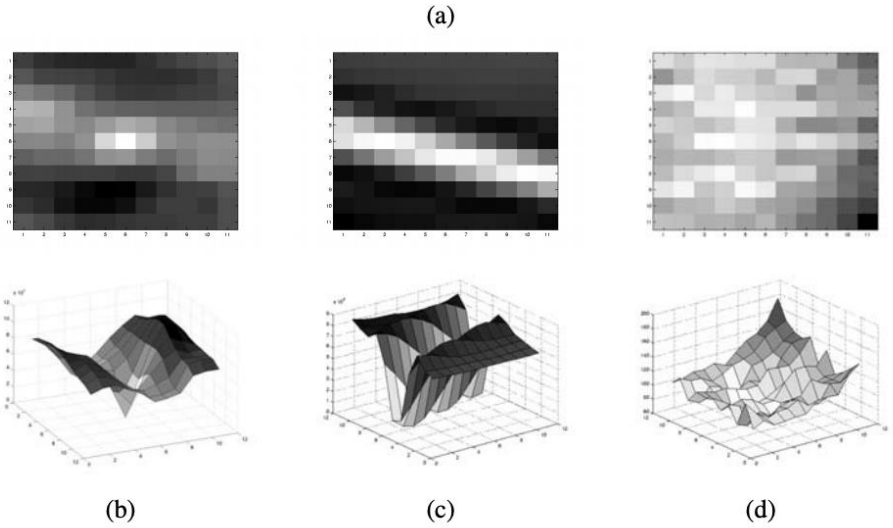
En viktig del i 3D – modellering er Feature detection. Objektene i bilder brukes til både orientering av bilder og sette sammen bildene til en mosaikk. Dette er et resultat av Feature detection som er et sett med sammenbindingspunkt. Et annet ord for feature detection er keypoint.

Det første man ser etter i et bilde er objekter som er skiller seg ut. For eksempel fjelltopper, hushjørne, vinduer eller dører, altså punkter med kontrastforskjeller rundt punkter er viktig. Det er viktig å vite at dette er objekter softwaren skal kunne finne automatisk med algoritme og ikke manuelt av et menneske. Objekter blir da funnet av en matematisk algoritme ved automatisk gjenkjenning. Samtidig som algoritmen gjennomfører en matching av bilder. Med det må objektene være lett å finne i flere bilder. Slike objekter kalles gjerne keypoint features. [Szeliski, 2010]

Det er finnes to metode å finne keypoints på. Den ene finner punkter i den ene bilde og bruker algoritmene for korrelasjon og minste kvadraters metode til å plassere punktene nøyaktig i forhold til hverandre. Metoden er best å bruke når vinkel mellom bildene er liten. Den andre metoden finner keypoints i begge bildene også matches disse sammen i forhold til hverandre. Metoden er da mest brukt når det er store forskjeller i bildene, til for eksempel stitching i panoramabilde eller objekt gjenkjenning. [Szeliski, 2010]

R. Szeliski beskriver punktbestemmelse i prosess på 4 steg:

1. Feature detection: Hvert bilde blir gjennomsøk etter punkter som kan være lett å finne i andre bilder. I søket vurderer man endring i gråtoner verdier i søkevindu. Er endringen i gradient stor, blir det letter å finne punktet i et annet bilde. Har man et bilde med mange homogene overflater vil dette være vanskelig å finne fellespunkter med andre bilder. [Szeliski, 2010]

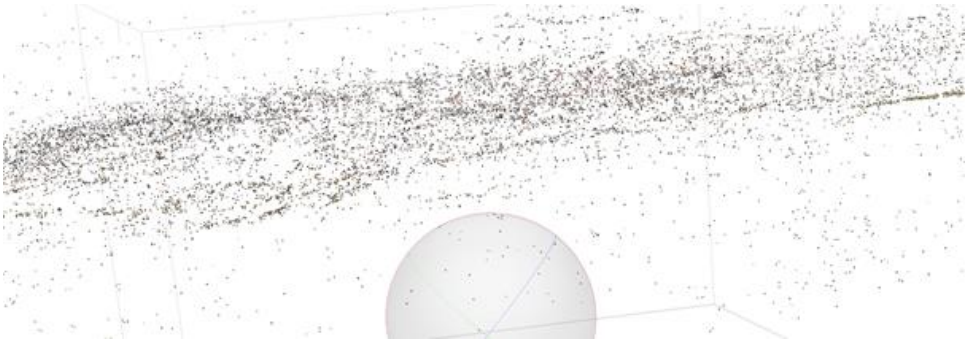


Figur 1 Eksempel på Feature detection

2. Feature description: Det vil være litt forskjell i keypoints i bildepar. Dette kommer fra orientering og skalerings forskjeller i keypoints. Noe som kommer av at de har gjennomgått en Affin transformasjon. Med det vil det være nødvending å utrykke parametere og endret innhold i søkevinduet slik at de passer med andre punkter. Så området rundt hvert punkt blir konverter til kompakte og stabile benevnelser som kan matches senere med andre keypoints. [Szeliski, 2010]
   Det finnes flere description metoder, som ikke blir beskrevet her. Men må ha med Scale invariant feature transform (SIFT) siden denne metoden er veldig lik key feature detection i Pacted based Methods som blir beskrevet senere.
   Dette er en algoritme som er beskrevet av David G.Lowe. Etter at punktene er funnet, blir det gjennomført en filtrering der dårlige keypoints blir fjernet. Deretter blir effekten av rotasjon og skalering fjernet før beskrivelsen blir lagres. Lokale gradienter beregnes i områder rundt punktet. Der lengden av gradientene blir lagt til i et histogram, som utgjør en del av beskrivelsen. Punktet blir beskrevet av nabopunktenes gradienter i forhold til plassering, orientering og skalering. All informasjon blir lagret i en database som videre brukes som en referanseliste. Referanselisten blir brukt til å se etter lignede punkt som blir funnet i nye bilder som gjennomfører samme prosess. Det blir også nyttet korrelasjonsverdier med en viss terskelverdi. [Lowe, 2004].
3. Feature matching: Etter at man har funnet keypoints i 2 eller flere bilder, startes en matching. Det velges en metode, deretter en algoritme. Metoden velges uavhengig av resultat. Forskjellen dreier seg om hvor mange punkter som antas vil bli matchet mellom bildene. I metoden stiching vil det være relativt mange matcher, men hvis man skal finnet et objekt i et stort bilde, vil det være få punkter som matcher. Algoritmene vil jeg ikke gå inn på. [Szeliski, 2010]
   Etter at matchingmetoden er valgt, er det viktig å velge en effektiv metode for søking. Den enkleste metoden her er å sammenligne hvert eneste punkt med andre punkter. Det gjør at antall søk blir kvadratisk i forhold til antall punkter. Det finnes flere algoritmer og datastrukturer for å gjøre søkingen enklere og mer effektiv. Hensikten med de algoritmene er å begrense søketiden etter punkter så mye som mulig. [Szeliski, 2010]
4. Feature tracking: det er som et alternativ til steg 3. Forskjellen fra punktet over er at det blir søkt på et lite område rundt hver keypoint. Dette fører til en raskere prosess og passer ypperlig i forbindelse med videobilder. [Szeliski, 2010]

Feature detection bruker også enn annen teknikk som kalles kantdeteksjon. Der kantene, for eksempel, taket på bygninger kan brukes til matching på grunn av deres orientering og lokale profil. Dette kan brukes til å finne objektets avgrensninger. Altså der linjer stopper, stopper objektet, altså hustak. [Luhmann et al., 2019].

Når sammenbindingspunkter er funnet i bildene, matches disse mot hverandre. Det er på den måten man finner igjen samme punkter i andre bilder og har da et utgangspunkt før orientering av bildene. Det blir på et vis en orientering av bildene i forhold til hverandre.



Figur 2 Sammenbindingspunkter er de som generer en punktsky

2.2 Structure from Motion:

Structure from motion brukes som et uttrykk for orientering og selvkalibreringsporsessen innen “computer vision” - bransjen. “Structur” er en betegnelse på 3D egenskapene til et 3D punkt, mens “motion” er egenskapene til kameraposisjonene [Szeliski, 2010]. I min oppgave så er punktposisjon ukjent og ytre orienteringen finnes i orienteringsfilen til hvert bilde, men i denne oppgaven blir disse funnet i prosessen. Modellen blir georeferert etter den er laget ferdig.

Utgangspunkter for orientering av bildene, er sammenbindingspunktene som ble beskrevet i kapittel 2.1. Altså i en punktsky så er det punktene som blir generert, som vi kan brukes til å orientere modellen riktig. Det kan oppstå feil i punktene slik at vi får støy, ut fra refleksjon, belysning slik at punktene i det ene bilde matcher med feil punkt i det andre bilde fordi verdiene er like, se figur 3. Disse punktene vil få for store residualer under prosessen og vil bli filtrert bort.

Figur 3 Et eksempel på at punktene ikke er helt riktig i forhold til hverandre.

Algoritmene som benyttes her er vinkelen mellom bildene mye mindre enn det som har vært vanlig i de sammenhengen jeg har erfaring fra. Har erfaring fra mer manuelle metoder. Samtidig at det er en god overlapp. Små vinkler vil gi en dårligere geometri om man prøver å orienterer bildene på en tradisjonell måte. Dette gjelder ikke her, siden det blir brukt god overlapp og algoritmer som finner veldig mange sammenbindingspunkter og godt spredt ut i bildene. I tillegg matches det bilder med forskjellige vinkler mellom hverandre.

Det å finne orienteringsparametere på sensor, ytre og indre orientering er en viktig del av fotogrammetrien. Indre orientering er noe som gjennomføres i et testfelt eller ved selvkalibrering. De programmene som blir brukt, har egne moduler for selvkalibering og benytter de parameteren som kommer ut av det, videre i prosessene. Mens i vanlig flyfotogrammetri brukte de tidligere kjentpunkt på bakken for beregning av ytre orienteringsparametere. Siden det med tiden har blitt innført GPS og IMU har man ikke i like stor grad bruk for kjentpunkter, men brukes som kontroll.

Ser vi på nærfotogrammetri er det større variasjon i kvalitet på utstyr som blir brukt. Kamerasystemene kan være dyre med liten deformasjon eller billige med stor deformasjon.

Noen prosjekter gjennomføres med GPS/IMU, det ligger i telefonen i dag, men blir speilreflekskamera brukt er det ikke alltid de har GPS. Kvaliteten er på nesten samme nivå som flyfotogrammetri. Utviklingen de siste årene har skyt i været. Men orienteringsprosessene er veldig viktig i nærfotogrammetri, samt god overlapp.

Sentralprojeksjon er utgangspunkter for orientering av sensoren.

2.3 Dense image matching:

Bildematching brukes i flere stadier i en 3D – rekonstruksjon. I starten brukes bildematching for å finne bildets ytre og kameraenes ytre orienteringsparametere. Resultatet av første bildematching er en punktsky. Det er i denne fasen deteksjon av sammenbindingspunkter er viktig. Deretter generes en tettere punktsky med bildematching og her kommer begrepet dens image matching fram.

Ved matching finner man punkt som er felles i minst to bilder. Dette er noe som høres enkelt ut, men tvert om. Det er flere faktorer som spiller inn, for ikke å få geometrisk feil i modellen. Kameraets indre orientering er hovedfaktorene, men hellinger og dybdeforskjeller kommer i tillegg. I tillegg kan det komme radiometriske feil som kommer av ulik belysning og refleksjoner. Disse faktorene har stor innvirkning på geometrien i modellen. Støyen fra sensor og sensorens GSD vil også påvirke radiometriske og geometriske feil i modellen. [Luhmann et al., 2019]

Det finnes flere metoder for bildematching, men de mest brukte er arealbasert og kantbasert matching. I arealbasert matching sammenligner et areal i et referansebilde med samme areal i andre bilder. Arealet inneholder gråtone verdier og sammenligner informasjon fra bildene for å finne like arealer. Her brukes det to algoritmer, den første baserer seg på Minste kvadraters metode og den andre på korreksjoner.

Mens i kantbasert matching søker man gjennom bilder og finner kanter som er like og passer sammen fra de ulike bildene. Den er veldig lik teorien til deteksjon av sammenbindingspunkter som ble beskrevet tidligere og med det ikke blir beskrevet igjen.

I de programmene som blir testet orienteres bilder uten å legge inn informasjon om kamera og det gjør det mulig å lage en modell ved bildematchingsteknikker, samt laste inn punktskyer fra laserscann og sett de sammen med bildepunktskyer. Punktskyene får med det en tetthet som beskriver overflaten med uregelmessigheter. Med det må punktskyen tilpasses slik at tettheten er god nok der det er nødvendig og liten der det er flater. Teknikken som brukes er:

- Patch based methoed.
- Multi image matching technique, også kalt Semi global matching.

Videre nå vil disse teknikkene beskrevet. De blir brukt i Agisoft Metashap, og ligner veldig på metoden i Reality Capture. Reality Capture holder kortene tett, med å ikke si hvilken metode som blir brukt, men etter å studert programmet er det felles trekk til disse teknikkene i Reality Capture. Mens Pix4d bruker Bundle adjustment metode og SIFT.

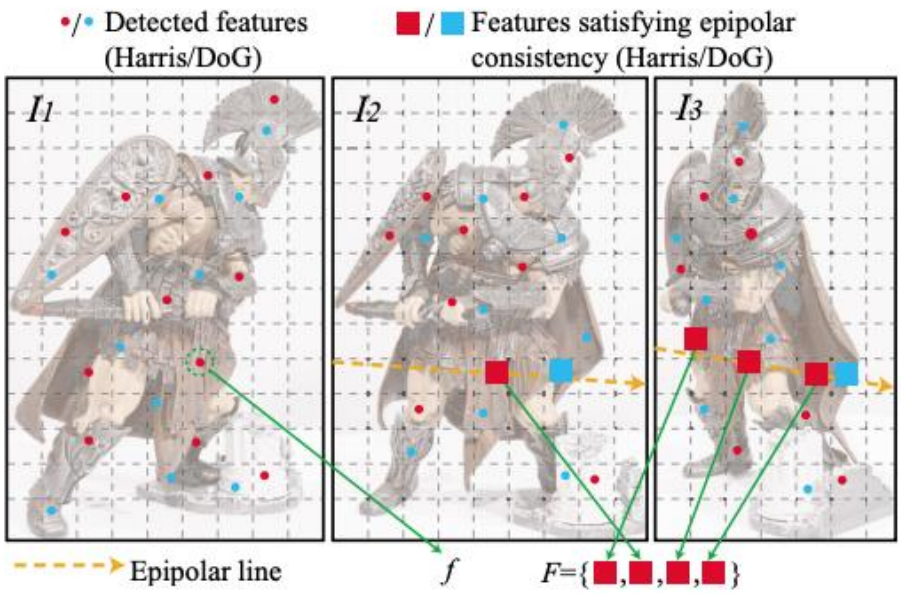
2.3.1 Patced based metods:

Agisoft Metashape bruker patced based metoden. Metoden starter med key feature detction som har tilnærmet fremgangsmåte som SIFT. Det blir så gjennomført en matching av keypoints, der et vellykket resultat generer en patch rundt keypoints slik at keypoint blir senter av patchen. Algoritmen som gjør dette er Harris og Difference of Gaussian operator. Harrisdetektoer kalles ofte hjørnedetektor. Den benytter seg av gradientene rundt en feature for å beskrive den. [Furukawa et al. 2008].

Utgangspunktet er at man har en feature fra referansebildet som skal finnes igjen i et annet bilde, finner man lignende features som ligger maksimalt to piksler fra epipolarlinjen og med det gjennomfører en triangulering av 3D punktene som knytter de til hver av disse. Når algoritmene har funnet en kombinasjon som tilfredsstiller den gitte terskelverdien gjøres punktet om til senterverdien i en patch. For å sikre seg og ikke få en falsk match, benyttes synlighets parametere. Altså en sannsynlighet for at punktet som er matchet, faktisk er synlig i begge bildene og som bidrar til å genere punktet. Det benyttes en tresholdvinkel (terskel) mellom patchens enhetsnormalvektor og retningen fra patchets senterpunkt og til kamera som skal matches mot terskelverdien. Vinkelen er satt til pi/3 og det gjør at vinkler med mindre enn denne verdien vil godkjennes. Kort sakt etableres en patch der det er en framtredene variasjon i teksturen, altså vi kan skille mellom fargeintensitet og at det er match mellom flere bilder der terskelverdien tilfredsstilles.

En patch er en rektangulær 3D flate som på et vis tangerer punkt som beskrives og man kan se fra alle kamera som matcher til patchen. Patchen velges til å være 7x5 pikselstørrelse. Det er for å kunne sample fargeverdier som ligger inne i patchen, og for å kunne finne verdien til senterpunktet så nøyaktig som mulig. Enhetsnormalvektoren for patchen peker mot referansekamera som er bestemt av algoritmen. Så i hvert bilde vil det bli mange patcher som plasseres utover flater i bildene og patcher gjenspeile pikselintensiteter i framtredende områder i bildene.

Hvert bilde som går igjennom prosessen får et kvadratisk rutenett med celler av størrelser n x n - piksler. Rutenettet vil orienteres etter bilde. Målet her er å generere minst en patch i alle celler for alle bilder. Patchene som er synlig for bilde lagres i rutenettet og de blir tatt med videre i prosessen for å gjøre de neste stegene mindre tidkrevende.



Figur 4 visualisering av celler og patcher. [Furukawa et al..]

Patcher som matcher hverandre blir tatt med videre og blir uttrykket som punkt i modellen. Resultatet av flere slike patcher er en spredt punktsky. Det er ikke sannsynlig at det blir funnet en patch per celle så prosessen må kjøres igjennom flere ganger, itereres. Samtidig kan det oppstå falske match tiltross for bruk av synlighetsparametere. Skyen som oppstår av patcher er grunnlaget for de neste stegene ekspansjon og filtrering. Filtrering av patchene fjerner falske punkter. [Furukawa et al. 2008].

Ekspansjonen har i oppgave å generer pather fra naboceller til de eksisterende patchene eller innenfor flere celler. Hvis celler mellom to bilder har dybdeforskjeller som ikke gjør det mulig å finne felles punkt, vil det gjøres en ekspansjon av en ny patch. Da går man over til nabocellen. Hvis ekspasjonen generer en patch som er lik en eksisterende patch i nabocellen, vil ikke den bli tatt med. Men hvis ekspansjonen ikke generer en lik patch vil den bli tatt med videre. Det er slik man kan gjengi små detaljer i objektets overflate. Algoritmen bruker et dybdekart for å kunne ekspandere patchene riktigst mulig i modellen. Det vil si hvis en overflate som er projisert har store dybdeforskjeller i forhold til nabocellen, blir den ikke ekspandert til nabocellen. Dette er fornuftig ut fra vurdering av faktorer som bl.a. lysforhold kan gjøre at en flate gjengis forskjellig eller feil ved dybdeforskjeller i forhold til virkeligheten. Disse funksjonene gjør at man vil kunne gjengi homogene flater med en høy presisjon. [Furukawa et al. 2008].

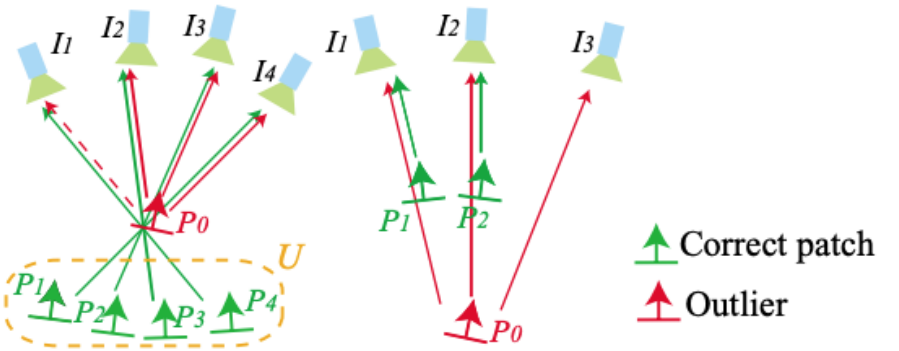


Figure 5. Outliers lying outside (left) or inside (right) the correct surface. Arrows are drawn between the patches p_i and the images I_j in $S(p_i)$, while solid arrows correspond to the case where $I_j \in T(p_i)$. U denotes a set of patches occluded by an outlier. See text for details.

Figur 5 Ekspansjon, [Furukawa et al.]

Det siste steget er filtrering. Filtrering fjerner patcher som blir matchet uten å være korrekte og fjerne uteliggere, altså outliers. Dette fører til en sikrer bestemmelse av punkt ved at geometrien blir ivaretatt. Algoritmen er delt opp i tre filtre. Der to av delene går inn på synlighetsinformasjon, der det første filtrerer bort patcher som ligger i samme celle men er av forskjellige typer og er synlig i få bilder. Den andre fjerner patcher som er generert i få antall bilder enn algoritmen har satt krav på. Tredje filtre detekterer og fjerner patcher som har mindre nabopatcher enn terskelverdien som er satt. Dette gjør at filtreringen fjerner det som

oppfattes som f.eks. enslig punkter som henger i lufta (støy) eller svart punkt på en hvitvegg osv. Altså punkter som tydelig ikke hører til der de har kommet. [Furukawa et al. 2008].

Ekspansjon og filtrering må itereres flere ganger slik at det dannes flest mulig patcher som utgjør den mest riktig 3D punktskyen uten for mye støy. Dette gjør at programmet har mulighet til å gjenskape små detaljer uten særlig støy. Programmene som blir brukt har filtreringsprosessene i ulike stadier av prosesseringene. [Furukawa et al. 2008].

2.3.2 Semi global matching:

Semi Global Matching (SGM) er en algoritme som tar hensyn til kostandene og begrensningene ved matching samt løser ytelses problemet som kommer ved bruk av global image matching. SGM baserer seg på piksel mot piksel matching, der gjensidig informasjon vurderes. SGM bruker teknikken Mutual Information, dette for å minimaliserer kostnadene ved matchingen. Mutual Information ble i utgangspunktet brukt for å måle likheter mellom bilder. Her var målet å finne en transformasjon mellom 2 bilder som fant gjensidig informasjon mellom bildene. Fordelene med denne algoritmen er at den håndterer intensitetsforskjeller mellom bilder. [Kim et al. 2005] [Gehrke et al.2008], [Hirschmuller, 2005]

SGM gjennomfører en form for segmentering av bildene der entropi altså variasjon i teksturer eller pikselverdier har stor betydning. Det deles inn i segmenter/felt som inneholder deler av entropien. Dette er for å få opp hastigheten ved at i et bildepar kunne kjøre en matching mellom segmenter med en viss likhet i entropi. Med det kan vi si at det blir brukt en slags vekting eller en koeffisient som beskriver en grad av entropi. Det er vanskelig å si nøyaktig hvor kantene til segmentene er. Med det benytter man seg av overlapp slik at kantsonene blir med gjennom flere segments delte matchingen. [Kim et al. 2005] , [Hirschmuller ,2005]

Algoritmen benytter disparity maps, noe som betyr at vi ser på hvor forskjellen i bildet mellom to matchede bilder kommer fram etter en piksel mot piksel matching. Bildet benyttes av algoritmen i en pyramideprosessering der man starter med f.eks. 10:1 i oppløsning og jobber seg til full oppløsning. I 1:1 oppløsning gjennomfører man da en piksel per piksel matching.

2.4 Bundle adjustment metoden:

Er også kalt strålebuntutjevning som er kjent fra flyfoto, hvor begrepene indre og ytre orientering er brukt. Ut fra store norske leksikon: *"For å kunne bestemme geometriske egenskaper for et objekt ved målinger i bilder av objektet, er det nødvendig å kjenne den indre geometrien (form og dimensjon) i opptakskameraet. Dette gjøres ved en kamerakalibering. Da er det mulig å rekonstruerer det stråleknippet som dannet bildet i eksponeringsøyeblikket; bildet indre orientering. Betegnelsen ytre orientering for opptaket angir opptakskameraets plassering i rommet, gitt ved posisjon og fotograferingsretningen for kameraet i romlig koordinatsystem".* [Dick, 2020]

2.5 Faktorer som påvirker en 3D rekonstruksjon:

For å få en god 3D konstruksjon er det flere faktorer som spiller inn, for å få et bra resultat. For Oslo kommune er det viktig å være klar over disse faktorene før 3D modellering. Reality Capture og Agisoft Metashape har beskrevet faktorer som påvirker nøyaktighet i modellen. Dette er faktorer som også påvirker andre programmer også: [Reality capture support, 2019], [Photomodler]

<u>Bildets oppløsning</u>: Jo bedre oppløsning på et bilde er jo større sjanse er det for å få en bedre modell og for å oppnå en nøyaktig plassering av modellen og objekter i modellen. I alt får en bedre nøyaktighet. Om det er dårlig oppløsning på bilde blir det vanskeligere å finne sammenbindingspunkter mellom bilder og man vil få en modell med en lav oppløsning og små detaljer utelates.

<u>Skanningens oppløsning:</u> Det er litt som ved bilder jo dårligere oppløsning man har per kvadrat meter i punktskyen jo dårlige oppløsning får man på modellen. Ved en skanning er det desto viktigere å ha en så god oppløsning som mulig, altså så mange punkt per kvadratmeter for å få en så godt resultat som mulig. Dette for å slippe hull/ manglene informasjon over et område.

<u>Kamerakalibering:</u> Deformasjon i punktsky og i bilde vil gjøre det vanskelig for en orienteringsalgoritme å få riktig plassering i en punktsky fra skanning og en punktsky fra bildematching skal matches samtidig som punkter, kan føre til unøyaktighet i plassering i en modell.

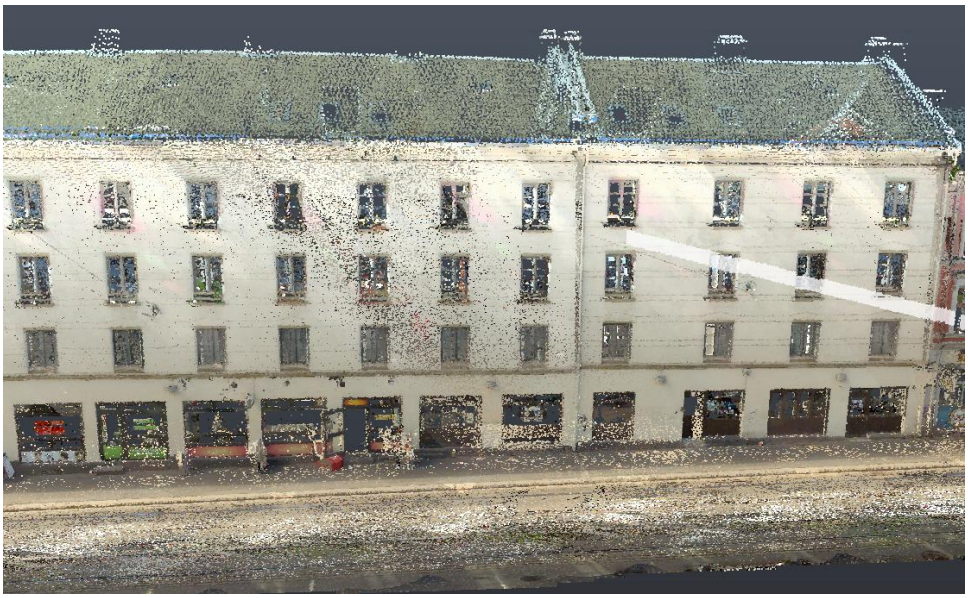
Vinkel mellom bilder: Ved en stor vinkel mellom bilder vil man får en dårlig bestemmelse av hvert punkt ved en matching. Dette fordi en stor vinkel vil gi få sammenbindingspunkter mellom bildene som fører til en dårlig orientering av bilder. I tillegg er dette noe som fører til en ikke tett punktsky ved generering. Har man motsatt altså for liten vinkel vil det føre til dårlig geometri for bestemmelse av punktets posisjon. Dette påvirker selve orienteringsporsessen negativt.

Bildets orienteringsnøyaktighet: Hvis orienteringen av bildene ikke er riktig vil det føre til en dårlig nøyaktighet i modellen. Det er derfor viktig med mange nøyaktige sammenbindingspunkt mellom bildene. I tillegg er det viktig at punktene er spredte utover hele bilde slik at store deler av bildet er dekket. Det er noe som hjelper på orienteringen. For man kan oppleve for få punkter eller deformasjon av modellen som en indiktor på en dårlig relativ orientering. Det er rett og slett fordi punkter plasseres feil i forhold til hverandre og med det fileteres bort.

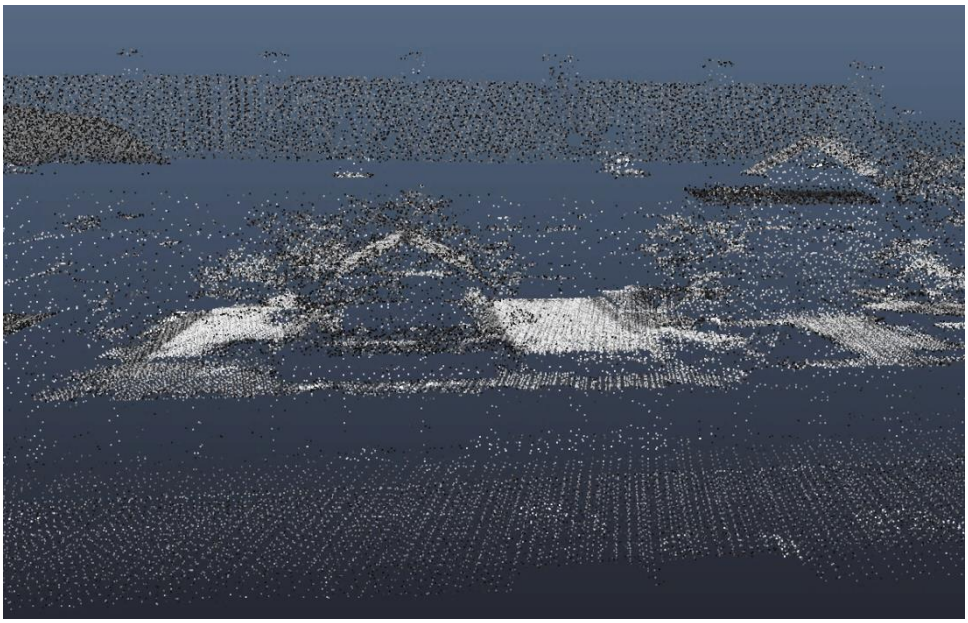
Skannets orienteringsnøyaktighet: Hvis orienteringen av punktskyen fra skanning ikke er riktig vil det fører til en dårligere modell. Det er derfor viktig her med flere kontroll punkt og bruk av kjentmerker siden da er det lett å kontrollere at skannet ligger på rett sted. Dette er med på å sette flere skann sammen også. Har man for få kjentpunkt vil deler av skannet vri seg i forhold riktig plassering.

Antall bilder i beregning og punktskyer: Jo flere bilder et punkt er markert i, jo bedre blir nøyaktighet for posisjon til det punktet. Det minste antall bilder man kan ha er 2, men gir dårlig nøyaktighet så det her viktig med god overlapp slik at man har flere bilder som kan bestemme punktet og gi en bedre nøyaktighet.

I forhold til antall skann har ikke det så mye å si, det er viktigere med punkt tetthet per kvadratmeter. Men det er alltid fordel med flere skann for det kan være få punkter som ikke dekker alt, så ved å sette sammen med flere skann så øker tettheten i den sammensatte punktskyen. Det fører til en bedre punktsky med flere punktskyer fra skann, det gir en bedre nøyaktighet i modellen.



Figur 6 Punktsky fra sporveien med 20 000 per kvadratmeter.



Figur 7 Punktsky fra terratec med 10 per kvadratmeter.

Med en variasjon i tekstur fra bildene er det tilleggs faktor som må tas med. Hvis man har en svært homogen tekstur i bildene vil det være vanskelig å finne sammenbindingspunkter i det området.

Det er viktig å skille mellom nøyaktighet i plassering av punktene og kvalitet på 3D modell. Det som er den viktigste egenskapen man ønsker med en 3D modell er full dekning, dvs. At vi får et vanntett mesh slik at den kan printes. For at det skal oppnås, må man ha en tett punktsky, som krever liten vinkel mellom bildene. Dette oppnås raskere ved å bruke en kombinasjon av punktsky fra laserskanning og en punktsky generert fra bildematching.

Med bakgrunn i egne erfaringer vil de øvrige faktorene identifisere, de vil gi en peke pinn på hvilke faktorer som må tas med i betraktning ved en fotografering eller skanning, før en 3D – modellering. Det første er at punkter som ikke ligger i flere bilder eller ikke er med i skann på et objekt ikke vil bli med i modellen. Samtidig er det utrolig viktig med overlapp mellom bilder som blir tatt, dette for å få en god orientering, og relativt til hverandre. Videre er det viktig å tenke at det skal være mulig å knytte bilder som er avbilder utenfor det mest sentrale i de to først bilder. Derfor er god overlapp mellom alle bildene viktig, helts 80 %.

2.6 FKB - Felles kartdatabase:

FKB er et begrep som brukes om de mest detaljerte grunnkartdataene. Grunnkartdataene beskriver generelle objekter i det fysiske landskapet. De deler i data i forskjellige lag som data over vann, bygninger, eiendommer, tak linjer, veier, steder, høyder, arealer, ledninger og mye mer i hvert sine lag slik at det er lett å hente ut det man ønsker å se på. [kartverket, 2014]

3 BESKRIVELSE AV SOFTWARE:

Det ble benyttet internett til å lete opp eksisterende Software som kunne dekke behovet samt at jeg fikk en del tips om aktuelle software via PBE og NMBU.

Det endte opp med Reality Capture, Agisoft Metashape, Pix4D og en del andre støtteprogrammer for å kunne jobbe i disse programmene.

3.1 Agisoft Metashape:

Agisoft Metashape er et fotogrammetriprogram som er den nye versjonen av Agisoft Photoscann. Det blir brukt mange bilder som overlapper hverandre og med det mulig å generere 2D og 3D kart med høy presisjon og oppløsning. Dette programmet blir brukt til å vise byområder i 3D og få en nær virkelighetsfølelse av verden.

Med Agisoft Metashape kan man få ut mange ulike type datasett:

- 3D punktsky: det er en sky som består av tusenvis av punkter der hvert punkt har en geografisk posisjon og punktskyen blir generert fra overlappende bilder og hvert punkt inneholder koordinater. Punktskyen er målte data altså ingen interpolering, derfor egner punktsky seg svært godt til oppmåling og analyse. Punktskyen blir visualisert med RGB eller klassifiserte fargekoder.
- 3D modell: er en modell som generer en fotorealistisk tekstur. 3D modellen tar utgangspunkt i punktskyen og bruker interpolasjon til å fylle tomrommene mellom punktene. teksturen blir hentet fra de opprinnelige bildene med det får vi en 3D modell som kan brukes videre i VR
- Digital høydemodell: denne modellen gir oss høydeverdier på hver piksel i kartet. høydemodellen får vi fra punktskyen og den kan eksporteres i en rekke ulike formater som kan senere importeres i GIS.
- Orotomosaik: er kart som blir laget av overlappende bilder som blir satt sammen til et georeferert kart uten distorsjoner:
- Volumberegning: Agisoft Metashape har også innebygd funksjon slik at man kan beregne avstand, areal, og volum i 2D/3D modellen.
- Multispektrale data: Agisoft Metashape støtter multispektrale og termiske sensor data for generering av vegetasjon indeks (NDVI) og termiske kart.

[Agisoft, 2020].

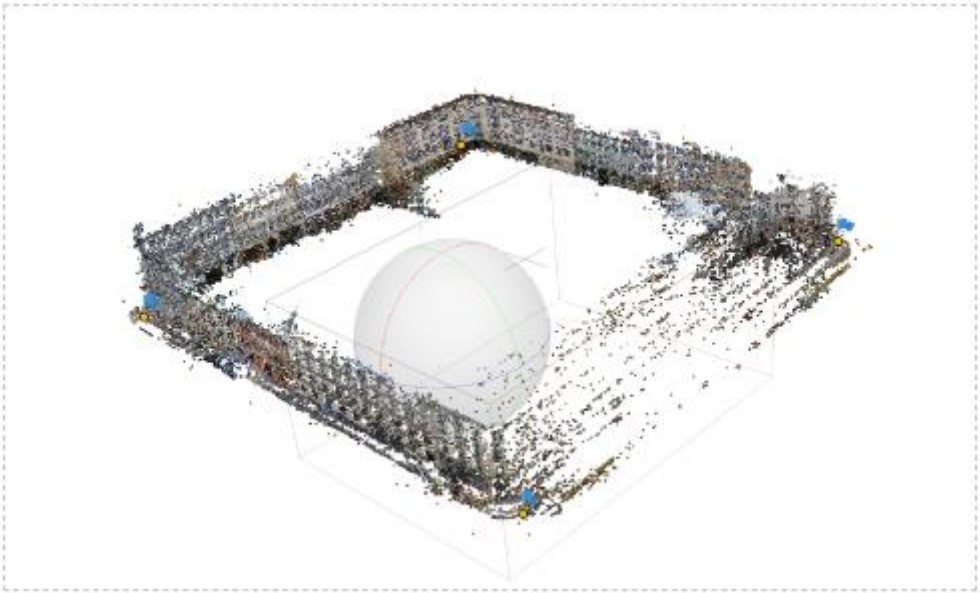
Siden Agisoft Metashape kan ta inn mange ulike datasett og lage mange ulike typer datasett er det veldig interessant for oppgaven. Agisoft Metashape er nye versjonen av Agisoft Photoscann. Metashape bruker algoritmen SFM (kap 2.2) for orientering og selvkaliberingsprosessen som lager 3D punkt i punktsky, ut fra deteksjon av sammenbindingspunkter (kap 2.1). Ved bildematching blir patced based metods brukt til å danne en punktsky og videre blir punktskyen triangulert til en mesh.

Agisoft Metashape professional Edition er lisens belagt program. Det betyr at du trenger lisens for å kunne bruke. De har en prøve lisens på Standard Edition versionen, men der får man ikke tilgang til alle funksjoner, som blant annet å eksportere fil. Eksportering av fil er noe som har blitt avgjørende i denne oppgaven. Det ble med det sendt en forespørsel til Agisoft og lurt på om å kunne få en prøve lisens på for å teste ut programmet og kunne få eksporterte filer.

Velger å beskrive de delene av Agisoft Metashape som er mest aktuelle for problemstillingen og testkriteriene.

3.1.1 Align image:

Her finner metashape kameras orientering og posisjon, ut fra SFM (kap 2.2) for hvert bilde og bygger med det en punktsky modell. Det blir brukt algoritmer for feature detection også for å finne sammenbindingspunkt (kap. 2.1). I denne prosessen kan man justere parametere som gjør at resultat blir bedre.



Figur 8 Bakkepunktsky fra Agisoft Metashape.

Beskriver hva parameteren gjør:

Nøyaktighet: ved å sette høy nøyaktig oppnås nøyaktig estimering av kameraposisjon. Så ved å sette lavere nøyaktighet ved prosessering vil det føre til en grovere estimering av kameraposisjonen. Så har vi et originalbilde vil det bli skalert med faktor 4 om det settes ned nøyaktigheten og med 16 ved enda lavere nøyaktighet. Så ved å sette nøyaktigheten til høy, oppskaleres bilde med faktor 4. Dette pga. at tie punktet er estimert på grunnlag av feature punkt i referansebildet, med det kan det gi mening å oppskalere referansebilde for å finne nøyaktigheten til et tie punkt. Ved å ha på høyes nøyaktighet anbefales kun når man har skarpe bilder og har et forskningsmål, pga. Det er veldig tidkrevende.

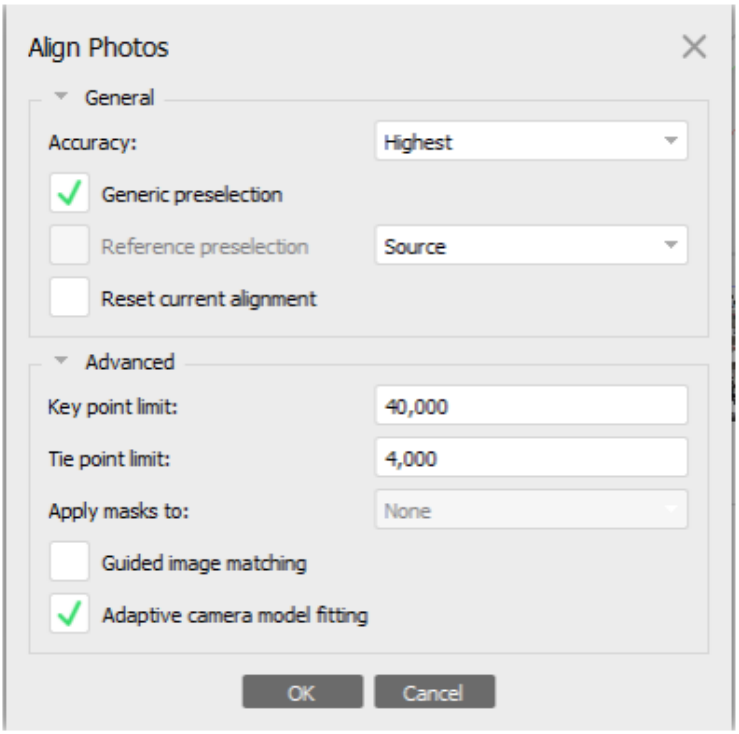
Pair preselection: alignment prosessen for et stort datasett tar tid. Tiden som brukes blir i hovedsak brukt på matche de detekterte feature på tvers av bildene. Om man forhåndsvelger bildepar kan hastigheten på prosessen gå betydelig opp. Ved generisk forhåndsvalg vil bildepar bli matchet med bilder på en nivå lavere. Bildeparene velges på bakgrunn av kameraplassering. Ved skrå bilder er det da nødvendig med høyden til kamera over bakken i tillegg til roll, patch og yaw data.

Key point limit: det indikerer øvre grense for funksjonspunkter i hvert bilde som ønskes skal være med i prosesseringen. Ved å sette null her så tolker Metashape det som at den kan finne så mange punkter som mulig, men fører til færre pålitelige punkter.

Tie point limit: Det er et tall som indikerer øvre grense for matchet punkt for hvert bilde. Setter man null vil innen sammenbindings punkter brukt. Tie punkt gjør det mulig å optimalisere ytelsen for oppgaven og påvirker ikke kvaliteten videre i modellen er anbefalt å sette til 4000. Dette for å slippe å gå glipp av sammenbindingspunkt som fører til dårligere modell, om den blir satt for lavt eller for høyt.

Sette på maske: Her har vi to tilfeller som kan velges blant:
Om masken knyttes til key punkt, blir området som har hatt maske tidligere ekskludert fra prosedyren for deteksjon av sammenbindingspunkter. Mens ved tie punkt menes at tie punkt er ekskludert fra alignment prosessen. Dette innebærer at området som har blitt "masks" finner key point for resten av bildet som gir det samme området og de blir da også ekskludert fra alignment prosessen. Et tie punkt er 3D punkt som anslås til å finnes i matchede bilder.

Adeptiv camera model fitting: ved å huke av denne inkluderes automatiske valg av kameraparameter i align prosessen som er basert på pålitelighetsestimering. F.eks. der datasettet har høy geometri, ved en bygning der det har blitt tatt bilder av alle sidene og i forskjellige nivåer, hjelper det å justere parameterne ved kamerajusteringen. Men ved et datasett med svak geometri, hjelper parameteren med å forhindre avvik mellom parametere, f.eks. radielle forvrenging. Blir ikke denne parameteren huket av vil Metashape sette parametere for brennvidde, hovedpunktposisjon, tre radielle forvrenginger og to tangensiell forvrenginger.



Figur 9 Parameter endring i Align Prosessen.

3.1.2 Build dense cloud:

Metashape generer en tettere punktsky modell ut fra tie punkt. Det er basert på de estimerte kamera posisjonene som er beregnet fra dybdeinformasjon for hvert kamera. I denne tette punktskyen er det mulig å klassifisere.

Beskrivelse av parametere:

Kvalitet: sier noe om hvor nøyaktig og detaljert geometri som ønskes, jo høyre ønsket kvalitet på modell som ønskes, jo lengre prosesserings tid. Kvalitetsparameteren ligner veldig på nøyaktighetsparameteren i align prosessen. Forskjellen er at høy kvalitet betyr behandling av originalbilder og ikke nedskalering av bilder som det betyr i align prosessen.

Dybde filtrerings modus: Det blir laget et dybdekart for hvert bilde ved rekonstruksjon i Metashape. Ved dårlige bilder (støy eller dårlig fokus), kan det komme uteligger. Ligger det inne algoritmer som kan fjerne uteliggere. Her er det tre valg:

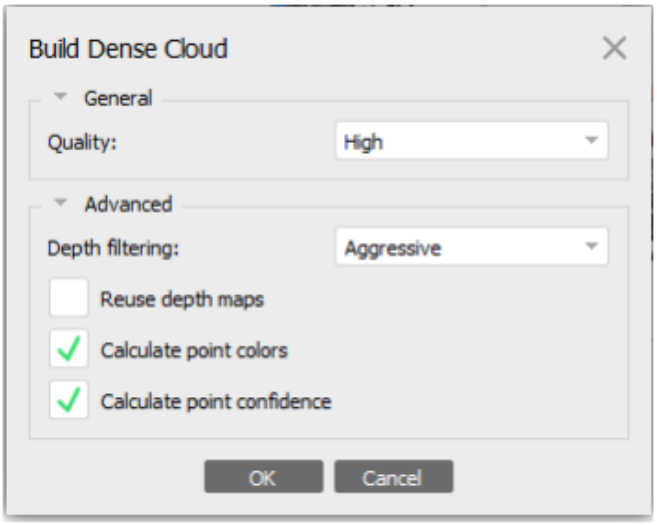
1. Mild: hvis man ønsker små detaljer på objektet som rekonstrueres, så brukes denne dybdefiltreingen. Denne funksjon tar ikke bort uteligger. Denne parameteren kan være nyttig ved bruk av bilder tatt fra fly.

2. Aggressiv: vis objektet du skal rekonstruerer ikke inneholder små detaljer som er viktig kan aggressiv dybdefiltreringnmodus brukes for å sortere ut de fleste utliggerne. Dette er også en god parameter for bruk på flybilder, om det ønskes klare linjer i takstruktur f.eks.

3. Den siste er moderat filtrering og den gir oss et resultat mellom mild og aggressive.

Kalkulerer fargepunkt: kan markes hvis man ikke skal ha punktfarge. I min oppgave er det uvesentlig siden jeg skal lage vanntett solids.

Dybdekart: denne parameteren gjør at det går å gjenbruke dybdekart som er laget tidligere.



Figur 10 Parameter endring i Build Dense Cloud prosessen

3.1.3 Build mesh:

Det er i dette stadiet det blir laget en polygon modell av den tette punktskyen. Det skjer ved en triangulering. Her som i de andre stadiene kan det velges parametere som gjør resultatet bedre:

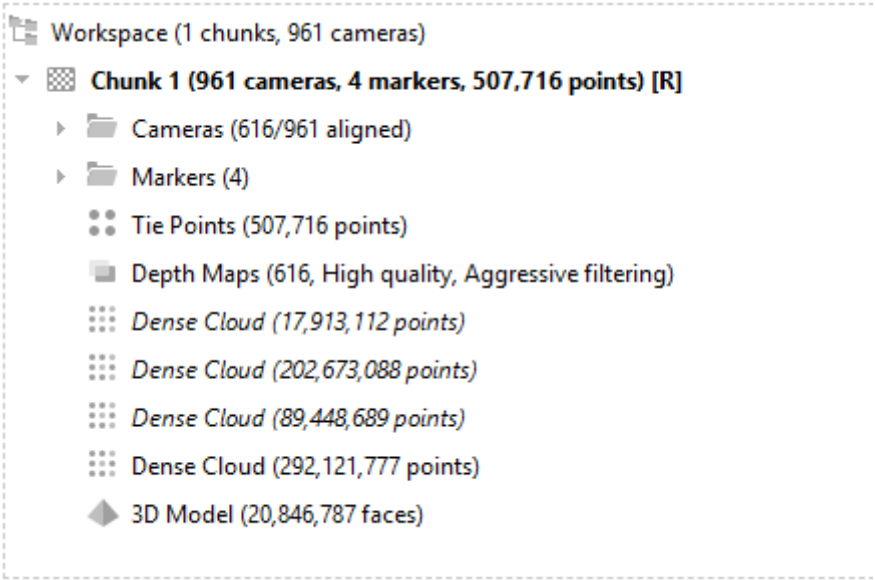
Overflate: Det er en funksjon som gjør at modellen blir lukket uansett modell, men tar lengre tid. Egnet for å lage overflatemodeller der man ønsker en terrengmodell.

Kildedata: det er funksjon som forteller oss hvor tett punktsky som ønskes og brukes. Dybdekart gjør det mulig å bruke all informasjon fra originalbildene mer effektivt og gjør det da mindre ressurs krevende.

Kvalitet: her spesifiseres ønsket kvalitet på rekonstruksjonen i dybdekartene som forutsatt at det er valgt som kilde. Dette fører til bedre geometri på modellen. Det ligner nøyaktighetsparameterne i align prosessen. Forskjellen er at her kan det velges Ultra high kvalitet ved behandling av originalbildene.

Face count: angir antall polygoner i endelig modell. Ved å velge høy, medium, lav som tilsier antall polygoner og hvor detaljert modellen blir. Det blir beregnet ut fra tettheten på den genererte punktskyen. Og forholdet er 1/5, 1/15 og 1/45. Det går å definere antall polygoner selv også ved egendefinering. Da er viktig å huske på at for lite polygoner gjør at modellen blir veldig grov som fører til runde kanter på f.eks. bygninger.

Interpolering: deaktivere vi denne fører det til nøyaktigere rekonstruksjon, siden bare området som tilsvarer tette punktsky rekonstrueres. Det kan da føre til hull som fylles i etter kant. Med et aktiv interpolasjon blir det interpolert innen en sirkel med en vis radius, det gjør at hull kan bli tett innenfor radiusen, men ikke utenfor. Det siste alternativet her er extrapolated modus som generer modell med ekstrapolert geometri. Store områder med ekstra geometri kan da generes. [Agisoft, 2020].



Figur 11 ut fra denne oversikten kan mn lese antall bilder som er tatt ing og antall bilder som er klart å bli brukt. Antall merker som er satt inn. Hvor mange punkt de forskjellige punktskyene har og hvor mange tette punktskyer som er lagt inn. Dense cloud som ikke står i kursiv er den som blir brukt til å danne mesh og antall punkt i parantes. I meshen står det antall polygoner som er laget i meshen.

3.2 Reality Capture:

Reality Capture er et nytt program på markedet (2016), men har blitt mer og mer tatt i bruk i 3D modellering. Det er et fotogrammetri program som kan lage 3D modeller utfra bakke- og flybilder eller fra laserskann eller en kombinasjon av dette. Dette er mest brukt til å lage 3D objekter, gaming, kartlegging, og VR.

Ut fra nettsiden til programmet kommer det fram at det går å modellere alt fra små objekter til store bygninger og kvartaler, med 360 grader dekning, dette er noe som er interessant for oppgaven. Det virker som programmet har kommet et skritt lenger enn mange andre 3D programmet.

Reality Capture inkluderer bildematching, automatisk kalibrering, beregning av polygonnett, fargelegging, teksturering, georeferering, DSM, korordinatsystemkonvertering, forenkling, skalering, filtrering, glatting, måling, inspeksjon og eksport/import.

Programmet orienterer bilder i forhold til hvor de er tatt, f.eks. bilder fra fly orienteres sammen, bakkebilder sammen, laseskyer og blir lagt i hver sin komponent som brukeren senere manuelt må slå sammen ved hjelp av kontrollpunkter. Kontrollpunkter er punkter som er felles i alle komponentene. Sammenslåing av komponenter fører til en mer detaljert punktsky. Det blir estimert kamera og laserskanningsposisjoner. [Reality Capture, 2015]

Velger å beskrive de delene i Reality Capture som er mest aktuell for problemstillingen og testkriteriene.

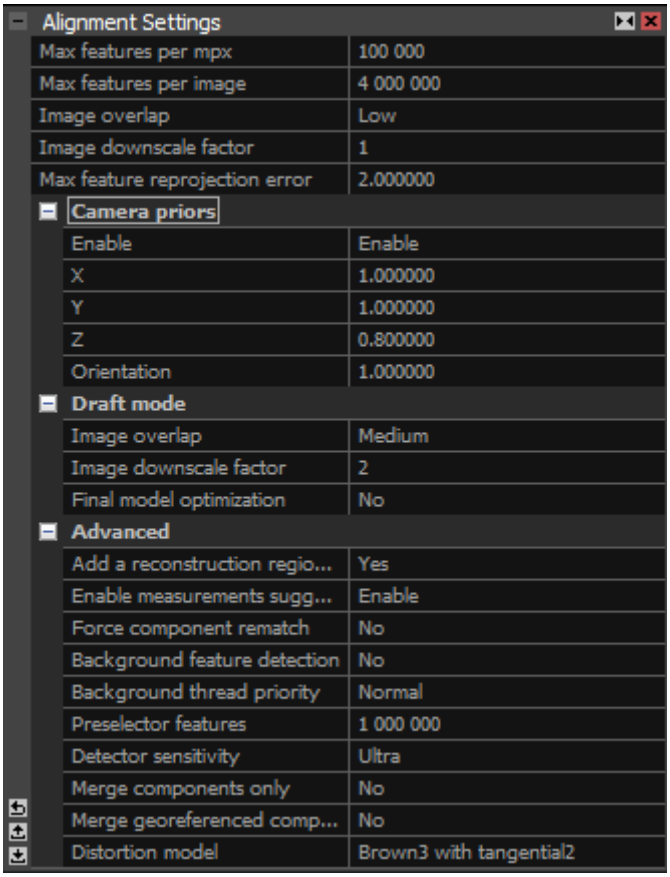
3.2.1 Alignment

Helt i oppstarten av programmet trekkes de datasetta man ønsker å ha med inn. Så kjøres en align images der programmet automatisk lager en punktsky. Bildene som brukes i denne prosessen må ha god overlapp, ingen områder uten dekning og en liten vinkel mellom bildene. I tillegg bør overflater i bilde ikke være glatte eller homogene. Jo større variasjon som finnes i bildene jo lettere er det å få en god bildematching.

Reality Capture gjennomfører den align images automatisk på følgende måte:

- Registrerer bilder og kamera parametere

- Deteksjon av sammenbindingspunkt
- Bildematching
- Orientering av bilder, SFM



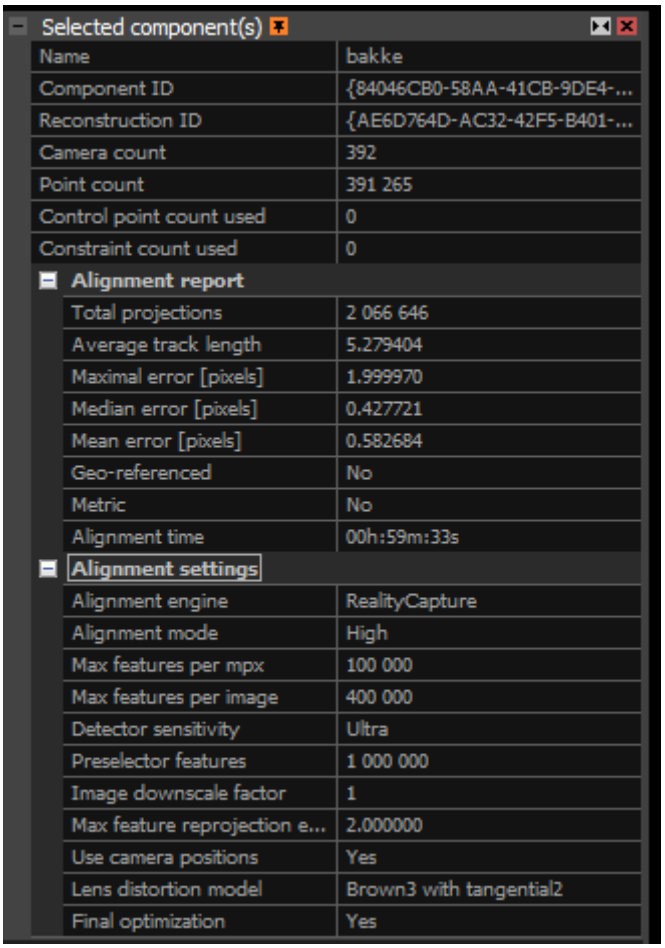
Alignment Settings	
Max features per mpx	100 000
Max features per image	4 000 000
Image overlap	Low
Image downscale factor	1
Max feature reprojection error	2.000000
Camera priors	
Enable	Enable
X	1.000000
Y	1.000000
Z	0.800000
Orientation	1.000000
Draft mode	
Image overlap	Medium
Image downscale factor	2
Final model optimization	No
Advanced	
Add a reconstruction regio...	Yes
Enable measurements sugg...	Enable
Force component rematch	No
Background feature detection	No
Background thread priority	Normal
Preselector features	1 000 000
Detector sensitivity	Ultra
Merge components only	No
Merge georeferenced comp...	No
Distortion model	Brown3 with tangential2

Figur 12 Parameterinnstilling i RC

Før denne prosessen har man en valg enn kan ta, innstillinger på hvor mange punkt per bilde og per megapiksel, overlapp, kamera posisjon, sette sammen komponenter og at kun de bildene som er georefererte blir med i prosessen. Det siste kriteriet om georeferering på forhånd ved import av bilder ble ikke brukt i denne oppgaven. Av den grunn at mange av datasettene ikke var georeferert, pga. Tatt med speilreflekskamera. De funksjonene som ble brukt mest var bilde overlapp, max features per image og max features per mpx. Dette er to funksjoner som sier noe om maks antall sammenbindingspunkt som kan finnes i bildet. Disse innstillingene har noe å gjøre med hvor mange keypoints som programmet kan finne i bildet og vil si noe om tettheten i bildet.

Etter dette så starter en parvis bildematching av bildene med bakgrunn på innstillingene som ble satt. Alle sammenbindingspunktene ble benyttet for å skaffe en så tett punktsky som

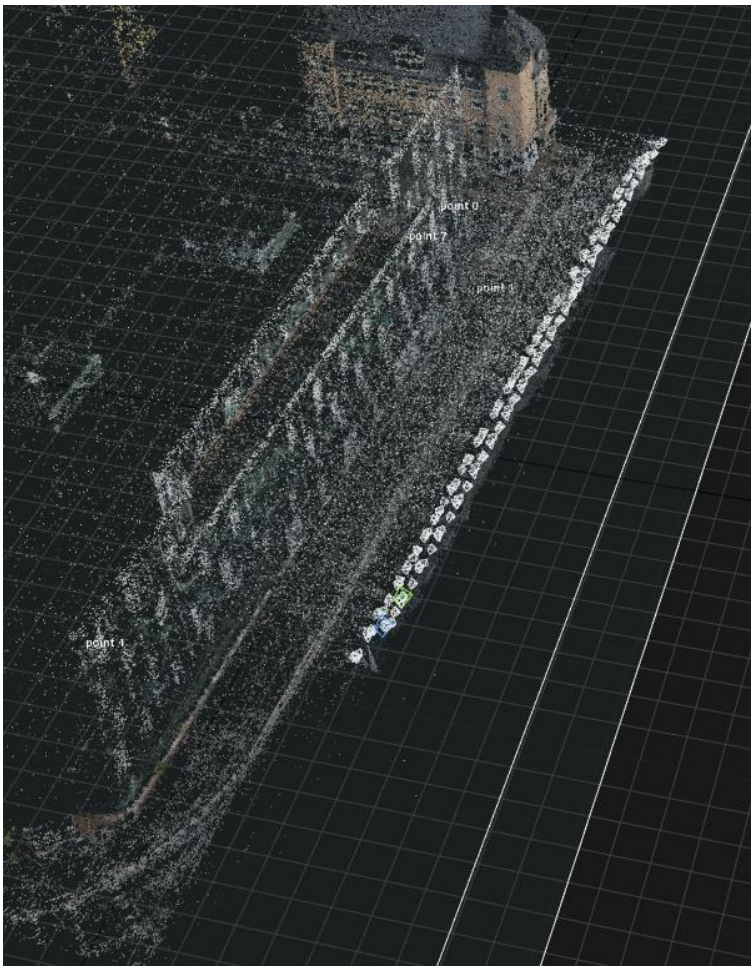
mulig. Tilslutt ble bildene orientert i forhold til hverandre med bakgrunn i de matchede punktene som ble funnet. Antall matche punkt finnes i slected component.



Selected component(s)	
Name	bakke
Component ID	{84046CB0-58AA-41CB-9DE4-...
Reconstruction ID	{AE6D764D-AC32-42F5-B401-...
Camera count	392
Point count	391 265
Control point count used	0
Constraint count used	0
Alignment report	
Total projections	2 066 646
Average track length	5.279404
Maximal error [pixels]	1.999970
Median error [pixels]	0.427721
Mean error [pixels]	0.582684
Geo-referenced	No
Metric	No
Alignment time	00h:59m:33s
Alignment settings	
Alignment engine	RealityCapture
Alignment mode	High
Max features per mpx	100 000
Max features per image	400 000
Detector sensitivity	Ultra
Preselector features	1 000 000
Image downscale factor	1
Max feature reprojection e...	2.000000
Use camera positions	Yes
Lens distortion model	Brown3 with tangential2
Final optimization	Yes

Figur 13 Parameterinstillinger for Align prosessen.

Prosessen varierer i hastighet utfra hvilke innstillinger som settes i align settings. Resultatet ble en punktsky. Etter denne prosessen er det mulig å sette sammen flere komponenter som ikke ble matchet sammen, ved å sette kontrollpunkter. Kontrollpunkter er punkt som man kan finne i alle komponentene slik at de kan brukes som referansepunkt. Det er i dette stadiet man georeferer punktskyen også skjer. Da settes det inn koordinater på bakkepunkt i punktskyen. Så kjøres prosessen på nytt. Både kontrollpunkt og bakkepunkt settes inn manuelt. Det er mulig å legge inn en tekstfil med koordinater på bakkepunkt men må manuelt plassere de i modellen.



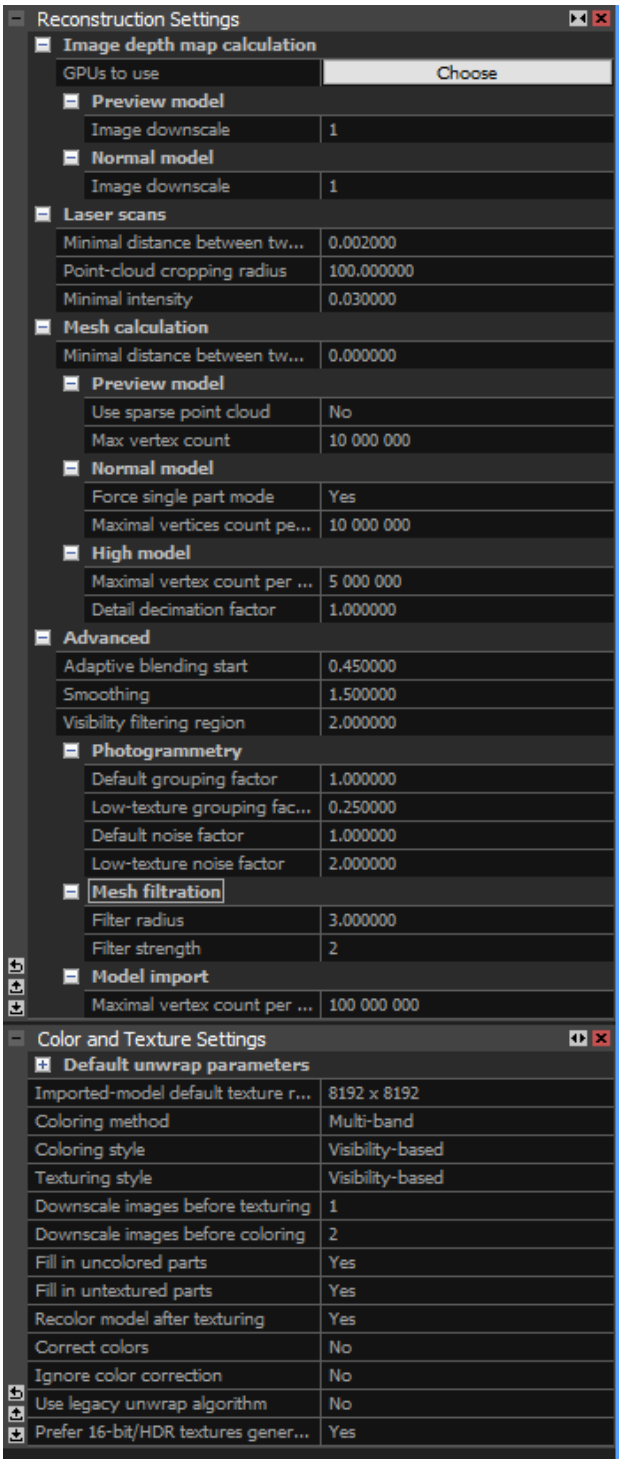
Figur 14 Punktsky eksempel fra testområde 2

Det er ikke mulig å finne hvilke algoritme som ligger til grunn her og er vi på forumet til Reality Capture sier de at det er konfrensielt. Det gjør at det er vanskelig å si hvilken algoritme som blir brukt her, men kan gå ut fra at det blir brukt SFM (kap 2.1) i orientering av bilder og en av dense image metodene (kap. 2.3.) til bildematching. Det er ikke noe vi kan si med sikkerhet.

3.2.2 Reconstrusjon:

Neste prosess etter å ha laget en tett punktsky er og bygge en solids. Oppgaven går ut på å genere et vanntett solids. Under denne prosessen blir det generert en tettere punktsky, solids og TIN. Det er også her tesaurering av objektet skjer.

I dette stadiet er det også en del parametere som kan justeres, i forhold til glatthet på ferdig modell, filtrering, radius til neste punkt som skal være med i trianguleringen. Med det får en Bruker svært mange valg muligheter og innstillinger.



Figur 15 Parameterinstillinger i reconstruction.

Etter modellen er ferdig rekonstruert er det mulig å se på ulike verktøy for å gjøre den enda litt bedre:

Texture: er at solid som har blitt laget blir satt på tekstur ut fra bilder og farge til punkt. Det kan velges ulik teksturering, og samlings typer per tekstur.

Simplify tools*:* her kan man redusere antall polygoner på modellen. Slik at man får en enklere og indre modell i den forstand med større polygoner som er satt sammen med hverandre.

Close Holse: tetter eventuelle hull som er i modellen.
Filtrering selection: her kan man fileterer bort uteliggere. Som ikke ønskes, men dette må gjøres manuelt. Bruk lasso og markere de punktene som ikke ønskes og trykk filter selection.
Smoothing tools: det er en funksjon som glatter ut modellen slik at den blir glatt og får runde overganger i skarpe kanter. Her kan det velges mellom ulike typer. Det kan også velges mellom ulike stiler, om man vil se på alle punkt eller bare punkt som klassifiseres som bakke eller bare bygninger. Det kan bestemmes antall iterasjoner som skal skje.

3.3 Pix4d Mapper:

Er et profesjonelt fotogrammetriprogram laget spesielt for dronekartlegging. Her blir det også brukt bilder til å genere 2D og 3D kart med høy presisjon og oppløsning ved hjelp av overlappende bilder. Her som i Agisoft Metashape får man ut ulike datasett i ulikt format. Det som er felles med Agisoft Metashape og pix4D mapper er at formater som 3D punksky, 3D modell, digital høydemodell, orthomosaic og volumberegninger får man ut i begge programmer. I tillegg i pix4d mapper har man termografi og det det støter termiske sensor data og generer kart der hver piksel inneholder en temperaturverdi. [pix4dmapper, 2020].

Det er veldig automatisk program, der du velger det utvalget av bilder som ønskes og prosesseres og hvilken funksjon man ønsker. Ønskes en 3D modell, DTM, DSM, osv. Ellers er resten automatisk. Programmet er bygget på sammen algoritmer som Agisoft metashape, SFM for orientering av bilder og Bundle adjustment metoden: (kap. 2.2 og 2.4) i bildematching.

De deler inn prosesseringen inn i tre deler Initial prosessering der det blir brukt features detection for å finne keypoints i bilder, keypoints blir brukt til å finne bilder med samme keypoints. Det skjer en kamerakalibering og orientering av bilder og kan her legge inn georeferering i form av EXIF –fil. Det blir laget automatis Tie punkt.

Neste del i prosessering er punktsky og mesh. Her blir tie punktene laget til dense cloud og videre til en mesh ved bruk av Bundle adjustment metoden. Videre ender man med en Mesh ut fra triangulering og tilslutt legges det på tekstur.

I det siste steget som ikke det blir godt så mye inn på siden vi ønsker å stoppe etter laget mesh, blir det laget DSM, reflektans kart, index kart og orthomosaik. Det blir laget i den automatiske prosesseringen, men blir som sakt ikke sett så mye på siden det ikke går inn under problemstillingen. [pix4d mapper, 2020].

En utfordring med Pix4D var hvor stor filstørrelse var lov å ta inn. Programmet tar ikke inn mer enn 55 megapiksles bilder om man ikke har lisensutgave av programmet. Dette skaper problemer når datasett inneholder bilder som har flere megapiksler. I lisens utgaven er det mulig å ta inn bilder over 55 Megapiksler, dette ble testet.

Figur 16 Eksempel på modell fra terrassemodellen.

3.4 Støtte software:

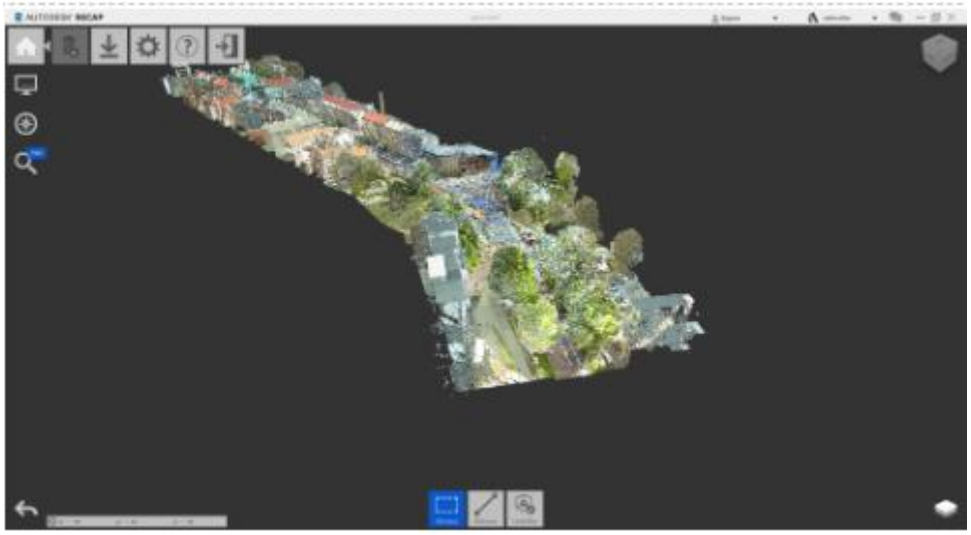
Det har blitt brukt noen programmer for å finne koordinater til georeferering, og sjekke punktskyer og modeller. Så det er programmer som blir brukt til sjekking av modellen.

3.4.1 ArcGIS PRO:

ArcGIS PRO er et profesjonelt GIS program utarbeidet av Esri. Programmet brukes til utforsking, visualisering og analysering av data. Det kan lages 2D kart og 3D visualiseringer. i oppgaven brukes dette programmet til å finne koordinater til bakkepunkter i forhold til FKB data fra kommunen.

3.4.2 Autodesk ReCap

Autodesk Recap PRO er et programvare for virkelighetsregistrering og 3D skanning for å skape intelligente modeller. Programmet kan lage 3D modeller fra importere bilder og laserskanning. Ut fra ortofoto kan det bli produsert punktsky som kan brukes i CAD og BIM. I denne oppgaven er Autodesk Recap Pro blitt brukt til å se flere punktskyer i forhold til hverandre, hvor godt ulike punktskyer passer i forhold til hverandre. Det er også et nyttig verktøy i å flette sammen flere punktskyer og eksportere til ønsket format. I denne oppgaven ble Recap brukt til å sette sammen flere punktskyer til en stor punktsky.



Figur 17 Sammensetning av punktskyer.

3.4.3 FME - Feature Manipulation Engine:

FME er et program som hovedsakelig transformere et datasett fra ett format til et annet format. Det blir brukt en transformasjons operator, som man kan legge inn mange forskjellige transformasjon. I denne oppgaven ble transformasjon funksjonen "projecetor" brukt for å transformerer et datasett fra et koordinatsystem til et annet koordinatsystem, samt se om to punktskyer lå langt unna hverandre eller hva georeferert likt. Det kan også brukes til skalering av modell. [safe software, u.d.]

3.4.4 NT 3Dbygger:

Er Norsk Tilpassing 3Dbygger som utnytter FKB-data for å bygge bygninger sammen med terrengmodell og som gjør kartvektor til 3D-modeller. Dette gjør det mulig å genere landskapsmodeller, med det øker nytteverdien av detaljerte, og kostbar datafangst. Med NT 3D bygger kan vi bygge ekte 3D – modeller og utnytte datafangst vi har. NT 3D bygger er et geoprosesseringsverktøy i ArcGis som også inneholder separate verktøy for å rette feil i kildedata.

Prinsippet for NT 3D bygger er å bygge bygninger ved å bruke FKB – linjer og en TIN-modell. FKB linjene blir drapert ned på Tin-modellen, altså vertikale linjer fra FKB linjene og ned på TIN-modellen. Det generes i 4, 3D nivåer for å få det meste detaljert bygg. Jo, høyere 3D-nivå jo mer detaljer. Dette har noe med hvor mange linjer fra FKB linjene som klarer å generes i første omgang. Derfor må det genereres på det samme sette flere ganger på ulikt nivå. Dette er metoden Oslo kommune brukte før til å lage vanntette solids, og blir en sammenlignes metode. [geodata, u.d.]

3.5 Andre software:

Det ble sendt ut forespørsler til andre leverandører for programmer som virket interessant og disse ble testet ut i en korte perioder før de enten ble forkastet eller eventuelt satt på vent. Disse er:

- iWitnessPRO
- Building Reconstrction
- autodesk ReCap Photo
- 3dsMax

- Autodesk ReCap, brukt som transformasjon program mellom ulike filformater
- nframes
- Count Capture

I noen tilfeller var det kompleks installerings prosedyre, fikk de ikke til å fungere. Det gjaldt blant annet med Building Reconstrction. Fikk det til å fungere, men plutselig hang det seg og stoppet å funke. Autodesk ReCap ble brukt til å sette sammen punktskyer og eksportere de i formater som ble brukt i de programmene som ble brukt i oppgaven. Se nærmere beskrivelser av brukte software i kapittel 3.

I det neste kapittel beskrives metodene som er brukt for å gjøre det mulig å teste hvert enkelt program opp i mot testkriteriene.

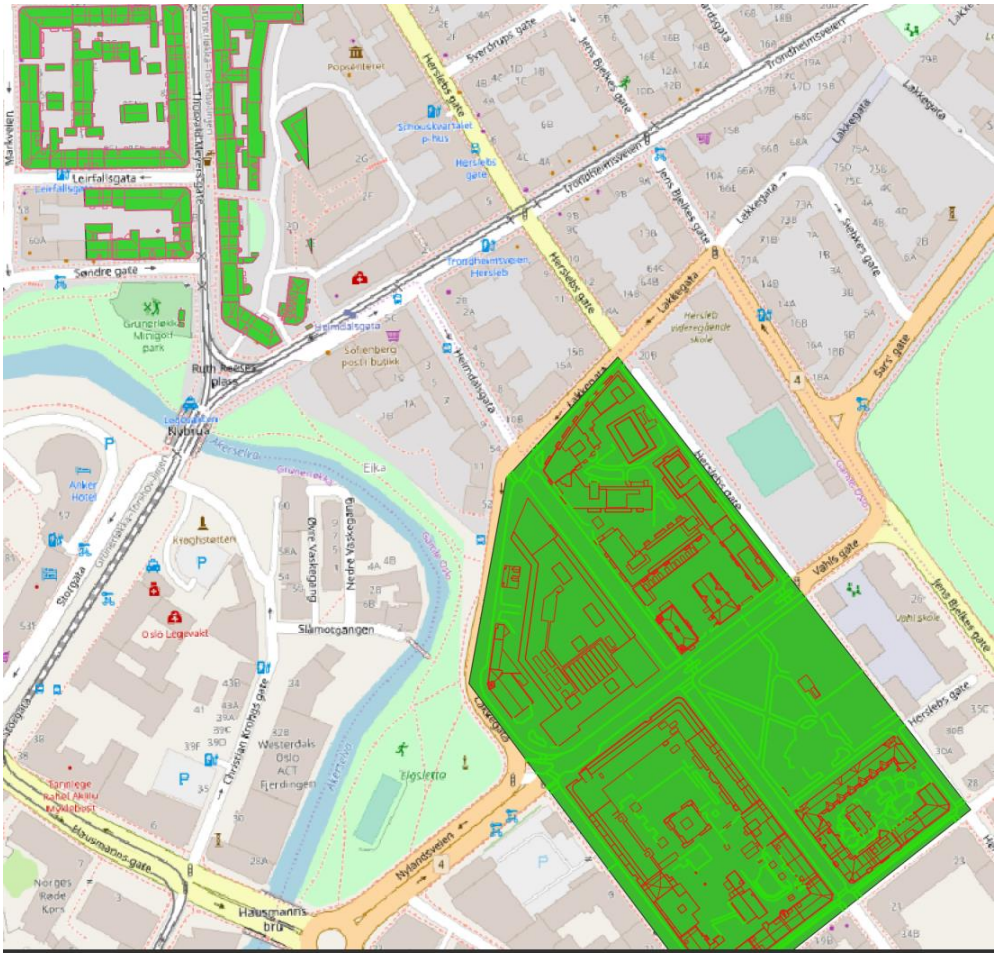
4. METODE:

Ut fra bakgrunn og problemstillingen min kan jeg trekke ut følgende testkriterier for de ulike programmene. De er som følger:

1. Grad av automatisering
2. Ivaretakelse av geometri og tekstur
3. Vanntett solids
4. Sammenslåing av datasett
5. Hvilken nøyaktighet er mulig å oppnå?

Vil også beskrive andre observasjoner og erfaringer som vil være av interesse for Oslo kommunes arbeid innen 3D- rekonstruksjon.

4.1 Prosjekter:



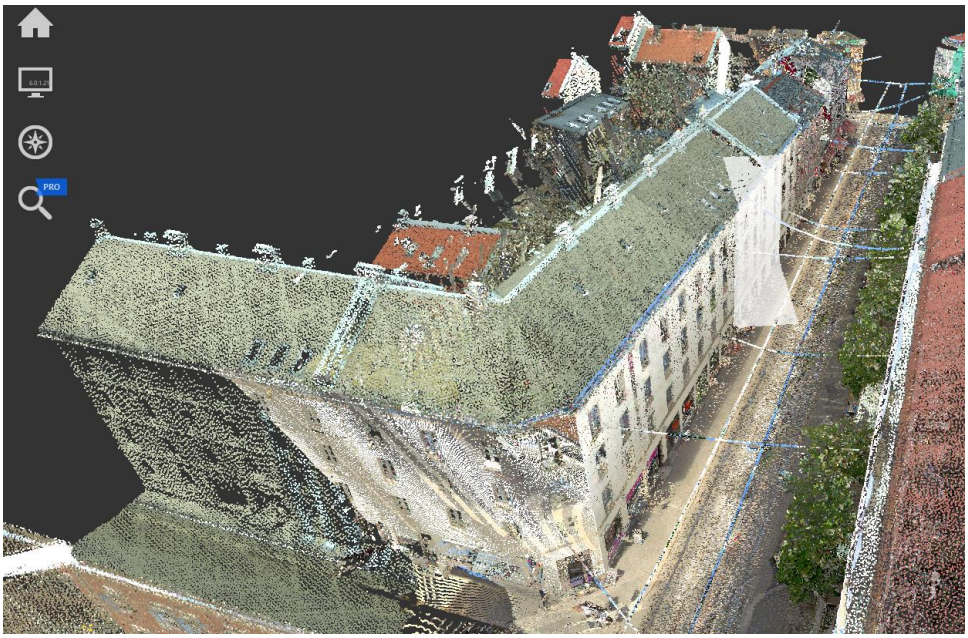
Figur 18 Øverst til venstre er kvartalet og nederst til høyre er nabobygget. De røde linjene er bygningslinje og grønne flatene er bygningsflater.

For å kunne teste ut og analyserer de tre programmene som er valgt å gå i dybden på. Definerer man to områder, henholdsvis et kvartal ved Thorvald Meyers gate, testområde 1 og "nabobygget" til PBE, testområde 2. Det ble plukket ut datasett og laget egne datasett for hvert prosjekt, som skulle være utgangspunktet for prosjektet. Det er datasett som er tatt med forskjellige kameraer, både bakke og fra fly. Det ble gjennomført en del modelleringer per prosjekt for å komme fram til et godt resultat. Prøvd ut forskjellige "settings"/parametere. Dette ga meg en bedre forståelse av programmene, hvilken settings ga de forskjellige resultatene og hvor mye støy gikk det å fjerne i de forskjellige stadiene.

Testområde 1: Kvartal i Thorvald Meyers gate

- 37 skråbilder fra Blom
- 961 Bakkebilder
- Canon EOS 60D med linse 10- 22 mm, låst på 22mm
- Sporveien punkttetthet, med en tetthet på 20 000 per kvm

Figur 19 Testområde 1 fargebeskrivelse; rosa = testområde, gul =kvartalsoner, Orange= sporveienlasersky trase



Figur 20 Dekning av kvartalet fra sporveien. 20 000 punkt per kvm

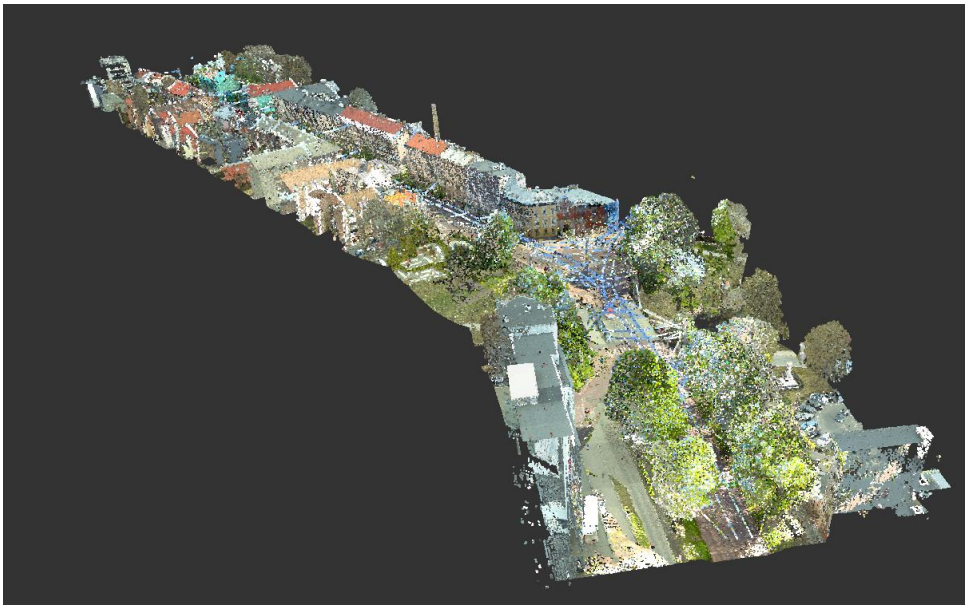
Testområde 2: Nabobygget til PBE:

- 50 skråbilder fra BLOM
- 234 Bakkebiler
- 196 terrasebilder fra PBE bygget
- Canon EOS 60D med linse 10-22 mm, låst på 22 mm
- Canon G16

Figur 21 Testområde 2

4.1.1 Tilgjengelig data:

Det blir benyttet datasett fra forskjellige leverandører, blant annet skann fra bil og fly og skråbilder. Sporveien har gjort en bilskanning av alle trikkeskinner i Oslo, det er med en punkttetthet på 20 000 kvm. Oslo kommune har fått skannet hele byen med fly i 2011 og 2019, disse punktskyene har 10 punkt per kvadratmeter. I 2018 tok Blom Norway AS skråbilder over hele byggesonen i Oslo som ble levert til Oslo kommune. Det ble brukte et Vexcel Ultracam Osprey Prime II. Flybildene har en oppløsning på 240 ppt (pixel per tomme). Det ble også laget eget datasett med bakkebiler.



Figur 22 Punktsky fra sporveiene. Den ble klipt til så den bare dekket kvartalet.

4.1.2 Egen datainnsamling:

Kom fram til at det ikke var nok tilgjengelig data, for hele kvartalet i testområde 1. Trengte jeg å samle inn egen data for å kunne få et så godt resultat som mulig. Dette pga. vegetasjon som var i bakgården og det ga ikke godt nok innsyn med flyfotografering ble med det gjennomført en bakkefotografering.

Det ble også fotografert en vegg av testområde 2 med samme kamera som ved testområde 1. Dette for å kunne sammenligne de ulike prosjektene siden testområde 2 er preget av homogeneflater og mye vinduer i forhold til testområde 1.

Det ble brukt et speilreflekskamera, Canon EOS 60D med en linse på 10-22 mm, den ble låst på 22 mm. Dette er for sikre dekning av områder der vegetasjonen var i veien for

flyfotograferingen. All flyfotografering skjer på sommeren og da kommer problemet med vegetasjon som er i full blomstring.

4.2 Arbeidsgang:

For å lage den ønskete solids etter spesifikkekrav (kap.4.5). Ble det som nevnt tidligere testet ut tre programmer. Ingen av programmene har lik arbeidsgang, de har ulike justering av parametere, se kapitel 3. Det som er felles i alle programmene at det ble sett på fly – og bakkebilder hver for seg, og senere i en kombinasjon. Dette gjelder både for testområde 1 og testområde 2. I tillegg ble det bruk lasersky i testområde 1. Kunne klart oss godt med laserskyen om den hadde dekket hele testområde 1 også bakgården. Siden den ikke gjorde det så måtte en bruke en kombinasjon med bakkebilder, flybilder og laserskyen.

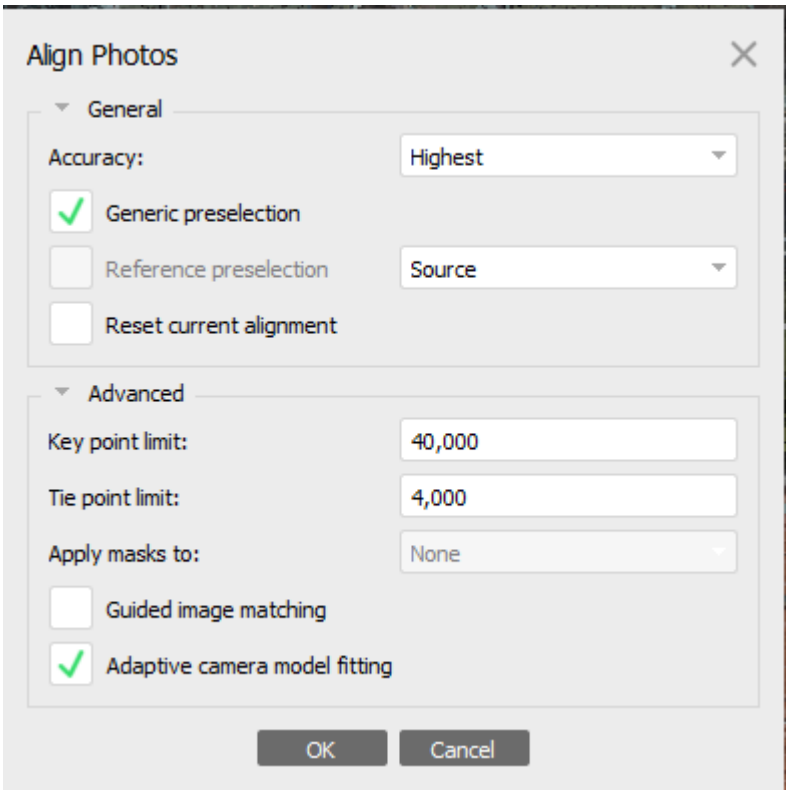
Det blir med det gått igjennom arbeidsgangen i de ulike programmene ut fra de ulike datasett kombinasjonen som er beskrevet over:

4.2.1 Agisoft Metashape:

Det første som blir gjort er å add Photos, noe som ble gjort i 3 omganger pga. det ble sett på flybilder, og bakke bilder for seg og til slutt laget en kombinasjon mellom bakke- og fly bilder. I tillegg ble det sett på laserpunktskyen fra sporveien, men siden den var mangelfull på baksiden av hus. Ble prosjektet avhengig av fly- og bakkebilder for å kunne få dekket rundt hele huset.

For å komme til en solids i Agisoft metashap må man igjennom tre steg etter at bildene er tatt inn i programmet:

1. Align photo, denne funksjonen er beskrevet i 3.1.1. Resultatet her er en tie punktsky som blir laget av algoritmene SFM (2.2) og feature detection (2.1). Det mulig å justere parametere før prosessen blir kjørt.



Figur 23 Parameterinnstillinger for Align prosessen.

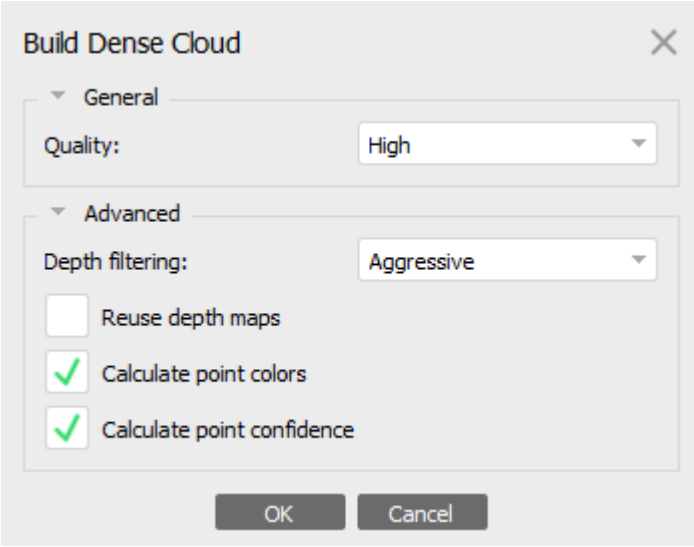
Disse parameterne ble valgt for å skape en tettere punktsky, som har en god nøyaktighet. Noe som fører til at estimering av kameraposisjon blir estimert godt og får med det nøyaktige tie punkt. Dette kunne gjøres siden oppløsningen på skråbildene var god og denne metoden framstiller solids på en god måte. Generic preselection ble huket av pga. Bildepar med lavere nøyaktighets nivå blir matchet. Dette velges på bakgrunn av kameraplassering.

På key point limit og tie point limit ble default verdiene brukt, siden det er indikasjon på øvre grense på matchet punkt og funksjonspunkt i bildet. Det ble prøvd høyere verdier, men det ga ikke noe bedre resultat. Prosesseringen tok med det lengre tid enn ved bruk av default verdiene. Tie punkt limit er med på å optimalisere ytelsen for oppgaven. Ved å justere på den risikeres det at enkelte sammenbindingspunkter ikke blir med.

Siden det ikke ble lagt inn noen parametere eller posisjon for kameraene så huket vi av adeptiv kamera model fitting. Den indikerer at programmet kan ta automatiske valg av kameraparametere, som baseres på pålitelighetsestimering. Her kommer bruk av SFM om orienterer og det at programmet finner parameteren i bildene ved

kameraestimering. Noe som justeres etter geometrien i bildene. Ved en svak geometri hjelper algoritmen til med å forhindre avvik, siden det ikke er lagt inn parametere.

2. Build Dense Cloud: I dette steget blir det bygget en tettere punktsky ut fra tie punktene.



Figur 24 Parameterinnsillinger for Build Dense Cloud prosessen.

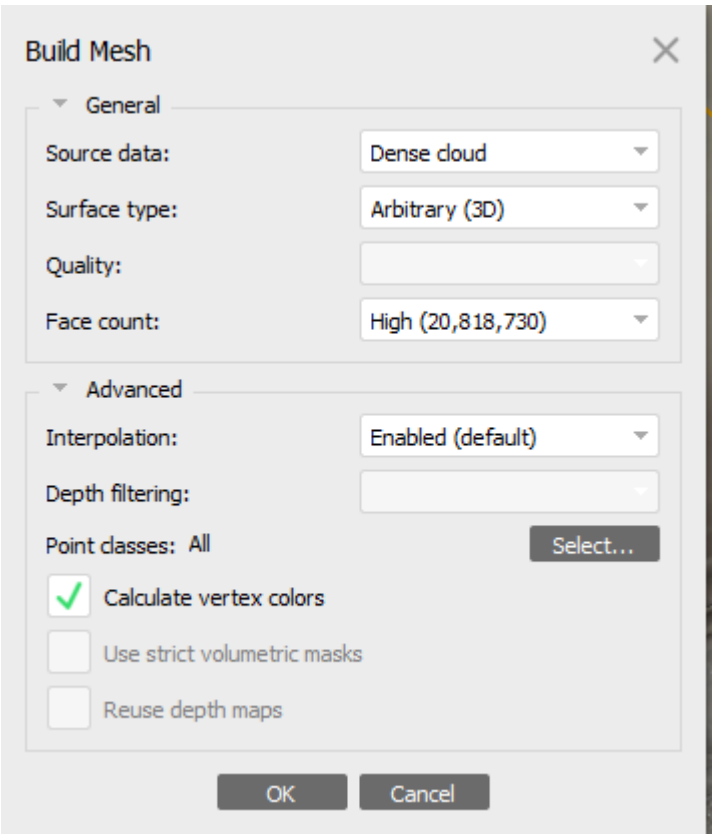
Generelt ønskes kvaliten så høy som mulig for å få en mest detaljert modellen. Det blir brukt en aggresiv depth filtering pga. denne funksjonen sorterer ut utligger og det vi konstruerer inneholder mye mer rene flater. Får klare linjer i takstrukturen på flybildene noe som er viktig for det er det eneste datasettet som dekker alle takene i prosjektet. Har valgt å kalkulere farge på punktene, dette for å kunne se tydelig objektet som skal rekonstrueres. Det er til hjelp, ved at bildene ikke er georeferert når de ble lagt inn. georeferer dense clouden ved å sett markes på kjentpunkt /kontrollpunkt. Koordinater ble tatt fra FKB – data. Endret fra lokalt til UTM sone 32N koordinatsystem. Med det ble punktskyen skalert slik at modellen som generes i neste steg er skalert og georeferert.

Markers	Easting (m)	Northing (m)	Altitude (m)	Accuracy (m)	Error (m)	Projections	Error (pix)
point 1	598252.940000	6643811.670000	12.630000	0.005000	0.251914	2	9.668
point 2	598360.170000	6643718.980000	10.830000	0.005000	0.240055	11	0.671
point 3	598260.240000	6643715.700000	11.620000	0.005000	0.355654	13	0.672
point 4	598353.210000	6643816.130000	12.290000	0.005000		0	0.000
Total Error							
Control points					0.287273		2.758
Check points							

Figur 25 Kontrollpunkt for kvartalet

Grunnen til at vi må georeferer allerede i dense clouden var for å kunne kombinere flere ulike datasett sammen. Laserpunktskyen fra sporveien er georeferert med at det er koordinater i hvert eneste punkt. Mens bildene ikke er georeferert. Så de ble lagt inn i et lokalt koordinatsystem og SFM algoritmen orientert bildene i forhold til hverandre. Grunnene er at det ikke var med orienteringsfil for alle bildene. Når alle dense clouden ble georeferert kunne man sette sammen alle dense cloudene til en felles dense cloud ved merge. For å få et vanntett solids ble mergete dense clouden brukt videre siden dette ble en tett punktsky og dekte bygningen.
PS: Det ble kun merget dense clouder når det ble laget en kombinasjon av datasett.

3. Build mesh: Her brukes den merge dense clouden til å danne en mesh ved triangulering.

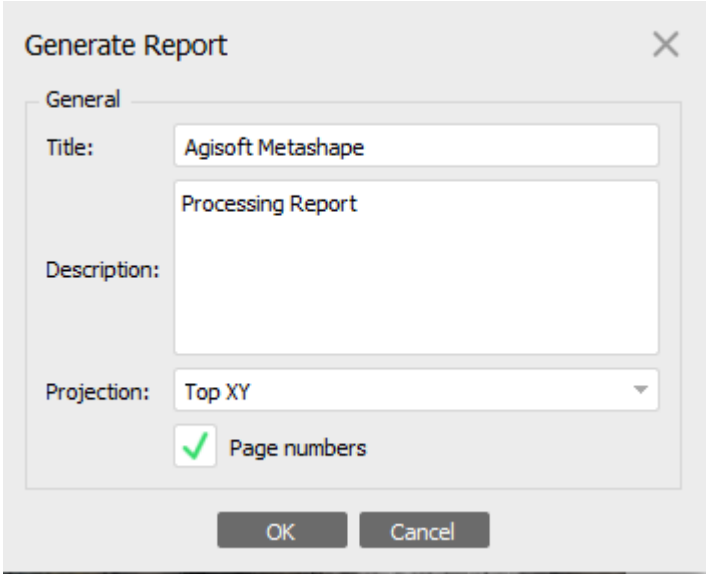


Figur 26 Parameterinnstilling for Build Mesh prosessen.

For å danne solids kan det velges mellom ulike kilder. I prosjektet ble det brukt dense clouden siden den ble georeferert i sist steg. Det kan også velges kvalitet her for å få en bedre geometri, settes til høy pga. den har samme funksjon som i align prosessen. Så kan det bestemmes face count som gir endelig antall polygoner i ferdig modell. Her ble det brukt høy altså forhold 1/5 for å forsikre seg at modellen ikke ble for grov. Den

siste parameterne som kan justeres er interpolering og den ble satt på deaktivert når dense clouden fra flybilder ble brukt. Denne funksjonen bygger mesh kun der det er punkter. Når den merge dense clouden ble brukt til å bygge mesh ble "enabled" brukt for å kunne bruke hele den mergete dense clouden ellers klarte algoritmen å trekke ut den beste dense clouden som hadde blitt merget.

4. Rapport med alle opplysninger og innstillinger som blir brukt finnes ved file -> export -> generate report:



Figur 27 Bilde av generate report

Valgte current view for å få den vinkelen jeg ville ha på modellen ved visning av modellen i rapporten.

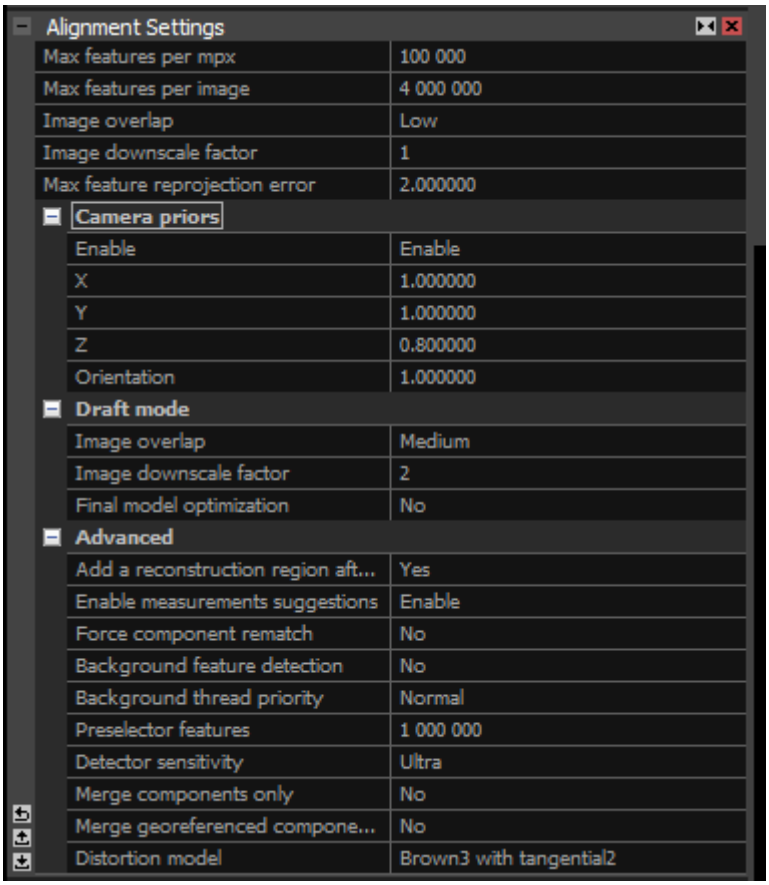
5. For å måle avstand brukes linjalen opp i verktøylinjen. Trykker der du vil starte og trekker mustasten bort til der du ønsker å slutte å måle.

4.2.2 Reality Capture:

Her starter med adde datasett. Denne prosessen ble gjort 3 ganger, bare flybilder, bare bakkebilder og siste med alle bilder inne. Reality Capture har som nevnt tidligere konfrensiell algoritme. Derfor blir det vanskelig å si noe konkret om hvordan funksjonen som blir brukt fungerte.

Prosessen for å rekonstruere en solid er delt opp i to deler i Reality Capture.

1. Alignment: Den roterer bilder i forhold til hverandre og finner sammenbindingspunkt i mellom bildene, og det skjer en bildematching. Det blir ikke lagt inn noen orienteringsdata på kamera til bildene. Så alt blir lagt i et lokalt koordinatsystem.



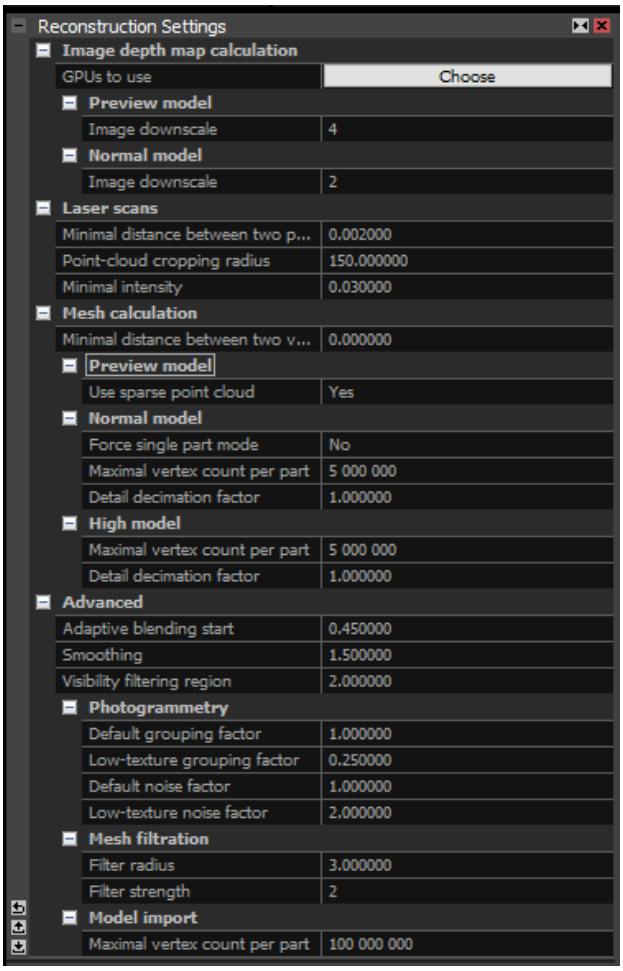
Alignment Settings	
Max features per mpx	100 000
Max features per image	4 000 000
Image overlap	Low
Image downscale factor	1
Max feature reprojection error	2.000000
Camera priors	
Enable	Enable
X	1.000000
Y	1.000000
Z	0.800000
Orientation	1.000000
Draft mode	
Image overlap	Medium
Image downscale factor	2
Final model optimization	No
Advanced	
Add a reconstruction region aft...	Yes
Enable measurements suggestions	Enable
Force component rematch	No
Background feature detection	No
Background thread priority	Normal
Preselector features	1 000 000
Detector sensitivity	Ultra
Merge components only	No
Merge georeferenced compone...	No
Distortion model	Brown3 with tangential2

Figur 28 Parameterinnstillinger for Alignment prosessen.

Parameterne som ble endret er “max feature per mpx” og “max feature per image” for de sier noe om hvor mange piksler og punkt per bilde. Det ble forhøyet for å få en tetter punktsky. Det ble også satt “image overlapp” til lav for å forsikre seg om at bilder med dårlig overlapp også ble med. Camera priores og draft mode ble ikke justert ble brukt defaultverdier fordi kameraet aldri står i ro på et fast sted. Satte “detector sensitivity” til ultra, litt for at da skal programmet være ekstra sensitivt for uteliggere. Og valgte og ikke merge komponenten av den enkle grunn av at programmet klarte ikke å matche bilder tatt med ulikt vinkler. Vinkelen ble for stor. Ser det godt på datasettene til testområde 2, der det er tatt fra bakkenivå, terrasse og fly. Det ble tydeligvis ikke god nok overlapp mellom terrasse og bakkenivå. De ble lagt i hver sin komponent.

For at det skal bli et vanntett bygg er vi avhengig av en felles komponent. Dette for å få ønsket resultat. Brukes bare flybilder vil modellen bli dårlig nær bakken se figur i kap.5. Siden bildene har dårlig med georeferering så blir komponentene laget i lokale koordinatsystem. For å sette komponentene sammen ble det satt inn kontrollpunkter. Det må settes inn minst 6 kontrollpunkt for at komponentene skal klare å merge sammen. Jeg valgt 9 kontrollpunkt for å slippe forvrenginger mellom komponentene. Komponentene klarer ikke matche, så ved store punktskyer er det lurt å ha flere en 6 kontrollpunkt for å slippe at ikke komponentene klarer å matche. Det blir i tillegg lagt inn bakkepunkter for å kunne skalere og georeferere modellen riktig.

2. Reconstrusjon: det ble prosessert rekonstruksjon av komponent med bare flybilder og den "merge" komponenten. Det her i dette steget at det skjer en trianguleringen og vi får en solid.



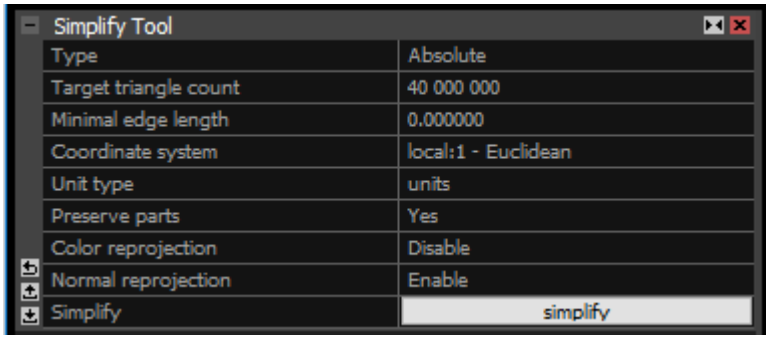
Reconstruction Settings	
Image depth map calculation	
GPUs to use	Choose
Preview model	
Image downscale	4
Normal model	
Image downscale	2
Laser scans	
Minimal distance between two p...	0.002000
Point-cloud cropping radius	150.000000
Minimal intensity	0.030000
Mesh calculation	
Minimal distance between two v...	0.000000
Preview model	
Use sparse point cloud	Yes
Normal model	
Force single part mode	No
Maximal vertex count per part	5 000 000
Detail decimation factor	1.000000
High model	
Maximal vertex count per part	5 000 000
Detail decimation factor	1.000000
Advanced	
Adaptive blending start	0.450000
Smoothing	1.500000
Visibility filtering region	2.000000
Photogrammetry	
Default grouping factor	1.000000
Low-texture grouping factor	0.250000
Default noise factor	1.000000
Low-texture noise factor	2.000000
Mesh filtration	
Filter radius	3.000000
Filter strength	2
Model import	
Maximal vertex count per part	100 000 000

Figur 29 Parameterinnstillinger for Reconstruction prosessen.

Her ble det utforsket med mye forskjellig parametere kombinasjoner men ente med default verdiene, de ga det beste resultat som ble funnet. Programmet er veldig godt utviklet ser det ut til, når det blir små forskjeller på modellene ved forandring av

parametere. Det er også et veldig nytt program på markedet noe som gjør at algoritmen er utviklet godt.

Utfordringer med å skulle modellere store bygninger er at Reality Capture klarer kun å vise 40 MP og blir det triangulert over det antall polygoner vil man ikke kunne se hele modellen i programmet. Kun ved eksporter modell. Når man prosesser en rekonstruksjon så brukes høy detaljer funksjon for å kunne få den mest detaljerte modellen. Men det fører til at det blir over 40 MP for å løse at hele modellen skal vises kan man bruke funksjonen simplify tools og redigere antall polynomer til 40 MP i "target triangle count". Med denne funksjonen slåes de minste polygonene sammen med andre polygoner, på den måten reduseres antall polygoner.



Figur 30 Parameterinnstillinger for Simplify funksjon.

Ut fra problemstillingen så skal vi lage et vanntett solids og med det ikke sett så mye på teksturering siden PBE ønsker en renn og tett solids uten tekstur. Så er kun standard innstillingen testet, men er flere muligheter for teksturering som nevnt i kap. 3.2.2

3. måling i modell skjer ved at det settes et start og ende punkt for målingen. Så blir «define distanse» brukt for å finne avstanden mellom de to punktene.

4.2.3 Pix4d:

Dette er det programmet som ikke har noen parametere justeringer, så her legges det inn bildene man ønsker trykker neste også kommer man til valg av hvilken modell som ønskes. I denne oppgaven ønskes en 3d modell / solid. Den gir en høy kvalitet. Så ble det kjørt prosessering av default verdier. Det er ikke mulighet å justere på disse før prosesseringen. Programmet er veldig selv forklarende, med lite justerings muligheter. Det kommer ut rapport på hvor mange av bildene som er lagt inn som ble tatt med i selve rekonstruksjonen og kalibrerte parametere.

4.3 Oslo kommunes spesifikasjoner/krav:

Plan og bygningsetaten avdeling Geodata her laget en kravspesifikasjon for et prosjekt som de hadde 2019. I den nevner PBE nøyaktighetskrav for en modell og velger å bruke dette siden oppgaven min bygger på det prosjektet som ble gjennomført i høst. Disse kravene kommer fra Produktspesifikasjon for orotofoto [kartverket, 2020]. Nøyaktighetskrav på enkelt objekttype kan man se i hjelpetabell 1[kartverket, 2011], [Kartverket, 2018].

Det finnes ingen nasjonale eller internasjonale standarder til 3D mesh per i dag for geodata produksjon. Kartverket har krav for stedfestingsnøyaktighet for ulike produkttyper. Under her er det beskrevet hva nøyaktighetskrav til 3D ortofoto er. [kartverket, 2020].

Tabell 1 Kravspesifikasjoner

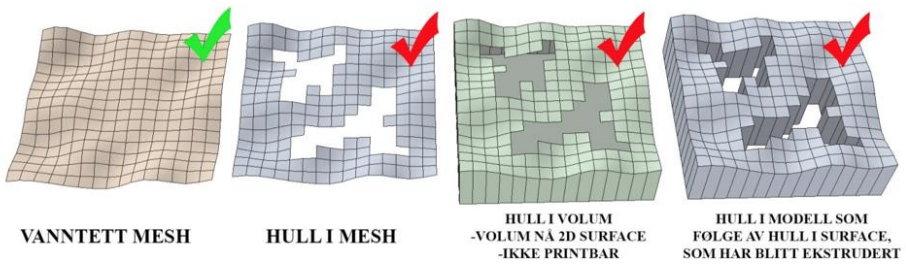
Spesifikasjoner:	Beskrivelse:
Nøyaktighet	Bedre enn 20 cm både i grunnriss og høyde (+/- 4*GSD) 3D – mesh/ surface må være vanntett (kap.4.5) slik at det kan 3D – printes.
Filformat:	Filene må være redigerbare og være i et filformat som kan brukes i andre programmer.
Kvalitet på tekstur	Vi har behov for så høy kvalitet i tekstruen som mulig i forhold til det datagrunnlaget tillater.

4.4 Solids/ måling:

Hvert enkelt program har individuelle fremgangsmåter og parametere for å kunne bygge solids (kap. 3 og 4). Det ble gjennomført flere ulike datasett og kombinasjon av ulike datasett for å øke kvaliteten på solids og tette eventuelle hull som skulle komme av dårlige datasett.

For å fange opp eventuelle målefeil som skyldes utydelige detaljer i modellen, ble det gjennomførte 6 målinger per detalj og midlet disse. I enkelte tilfelle ble det gjennomført mer. Det ble gjennomført målinger i alle tre programmene siden de har ulike måter å måle på, men samme måte for skalering er brukt i form av FKB data.

4.5 Analyse av solids:



Figur 31 Eksempler på vanntett solid.

Noe som er viktig for Oslo kommune er å kunne få en modell som kan gjøres om til solids, for printing med 3D printer. PBE ønsker å se på andre måter å lage en Solids på enn det de gjør i dag. I dag brukes NT 3D bygger (se kapittel 3.4.4) for å konstruere mulitipatche bygg/ solid fra grunndata. Det er viktig å se på hvordan de ulike programmene kan lage og eksportere Solids i ulike formater. Se på visuelle forskjeller mellom modellene og i forhold til dagens situasjon.

Grunndata som brukes til å konstruere multipatchen blir da en del av analysen for å se på avvik mellom grunndata og ferdig modell. Grunndataen blir fasit. Det blir tatt målinger i ferdig modell som f.eks. hustak, bredde på hus...osv. Her regnes det ut differansen mellom fasit og målt verdi på bygning. Med det beregnet et absolutt gjennomsnittlig avvik fra fasit. Til dette har det blitt benyttet *Root –mean square deviation* (RMSD). RMSD benyttes ofte for å beskrive avstandsforskjeller i et utvalg. På den måten er jeg i stand til å beskrive forventet usikkerhet i modell i form av centimeter, uavhengig av lengden på detaljene som måles. Dette vil være en verdi som forteller en bruker hva PBE kan forvente av modellen med bruk av disse programmene i forhold til bruk av NT 3D bygger.

Har sammenlignet mengden aktuelle sammenbindingspunkter som programmene finner og antall matcher mellom de ulike kombinasjonene av datasettene. Dette har en del å si for resultatet av modellen. Ser også på matchede bilder mellom ulike datasett og hvor mange punkt man får da. Det avhengig fra program til program hvordan det matches og hvor mange sammenbindingspunkter det blir.

5. RESULTAT OG DISKUSJON:

I kapittel 4 beskrev jeg metoden for oppgaven. Der vil hvert enkelt program bli vurdert opp mot testkriterier. Disse er:

1. Grad av automatisering
2. Ivaretakelse av geometri og tekstur
3. Vanntett solids
4. Sammenslåing av datasett
5. Hvilken nøyaktighet er mulig å oppnå

Resultatet fra testprogrammene, har mye å si for bruken videre av eksisterende og ny data.

Testing av programmene vil i stor grad basere seg på funksjonaliteten i hvert enkelt program for en god solids, vil teste ut hvordan det er å georeferere i programmet og blir modellen skalert. De to sist nevnte kan også gjøres etter at modellen eksportert i f.eks. FME.

5.1 Grad av automatisering

Grad av automatisering vil si hvor stor grad brukeren selv må bruke tid på å få programmet til å fungere og få ut fornuftig og ønsket resultat. Kommer litt innpå hvorfor de parameterne som ble satt i kapitel 4 ble satt.

Det ble også vurdert ytelsen på hvert program som en del av testen.

5.1.1 Agisoft Metashape:

Agisoft Metashape er i hovedsak et trinnvis automatisksystem der man kan velge å benytte deafaultverdiene som ligger inne eller justere verdiene. Her har det vist seg at det for mitt bruk bør ligge inne andre verdier enn deafault. Med trinnvis menes at man først setter igang algin photo som inneholder SFM og feature detection. Før neste steg er “build dense cloud” som er en fortetting som blir grunnlaget for å generer en mesh. En mesh er en triangulering av punktene. Når det generes mesh med tekstur kan man velge å bruke dense cloud eller dense

map. De er tette punktskyer begge to. Det ble i denne oppgaven brukt dense cloud siden i det ene tilfelle ble brukt en “merge” punktsky. Det kan også i meshen se på modellen både med og uten tekstur. Dette er bra siden PBE ønsket seg vanntett solids som betyr uten tekstur.

Ved å overstyre de mulige innstillingene i algin photo og dense cloud, kan man enkelt få et meget godt resultat ut av få, små og dårlige bilder. Justeringen i disse trinnene gjorde at vi fikk en tettere punktsky og med det en bedre mesh kvalitet messig.

For å kunne måle modellen benyttes måleverktøyet som ligger i programmet. Målefunksjonen har fungert ved at man med et klikk setter startpunkt for målingen og klikker en gang til i endepunktet. Dette gjøres direkte i modellen og fungerer til sammenligning med FKB data. Med andre ord en veldig enkel funksjon.

For å genere en tett punktsky med orientering av bilder med kun 37 flybilder opplevde jeg at prosesseringstiden var alt fra 15 min til 1 time. Ved generering av en tettere punktsky kan det opplevelse at programmet brukte noe lengre tid. Det ble testet med få og mange bilder. Noe som sier seg selv er at i prosessering med flere bilder enn flybilder tok det lengre tid, opptil 3 timer. Ved den merge dense clouden tok prosessering av mesh opptil 2 døgn, mens ved kun flybilder bare 3-4 timer. Det har noe å gjøre med at det er 4 ganger så mange punkt i den kombinerte dense clouden enn fra dense clouden fra flybilder.

For å oppsummere er Agisoft Metashape en trinnvis prosess som generer en tie punktsky så en tettere puntksky før helt tilslutt generer en mesh med og uten tekstur. Man kan gjøre innstillinger etter behov, men også velge å bruke deafultverdiene. Disse bør tilpasses en standard i forhold til bruksområde.

5.1.2 Reality Capture:

I Reality Capture er i hovedsak et trinnvis automatisksystem der man kan velge å benytte defaultverdiene som ligger inne. Det ble valgt å bytte ut for å få et bedre resultat med andre verdier se begrunnelse i 4.4.2. For å kunne få et enda bedre resultat. Det at det ble byttet ut slik at lav overlapping mellom bildene, gjorde at vi fikk en tettere punktsky. Noe som førte til bedre kvalitet på modellen. Det skilte store visuelle forskjeller med å sette parameterne slik det ble gjort enn og kun bruke defaultverdiene. Det ble også en tettere punktsky, slapp å bruke close hulse funksjonen. Siden punktskyen ble tett nok til at programmet greide å bygge TIN.

Ved bruk av defaultverdier blir modellene også dårligere på nært hold, blir mer skurrete. Kommer mer inn på det under neste testkriteriet

Med trinnvis menes at man først må sette igang en bildematching og feature detection, for å få generert en punktsky. Deretter har man en tett punktsky som man kan sette inn kontrollpunkter for å sette sammen flere komponenter. Får det kjøres en generering av punktsky på nytt for å få en felles punktsky. Dette er det som danner grunnlaget for en generering av TIN uten tekstur og resulterer i en solid. Siste steg er å generere solid.

For å kunne målemodellene som ble generert er det mulig å bruke "define distance" som er innebygd funksjon i programmet. Målefunksjonen fungerer slik at man må ha et fast punkt som man starter og slutter i. Altså må det opprettes punkt hvor linjen skal starte og slutte. Med det får man opp lengden mellom disse punktene i figuren. Det blir brukt som sammenligning med FKB data.

Det som er utfordrende med dette programmet, nevnt i kap. 3, er at leverandøren ikke vil oppgi hvilken algoritme som ligger i bunn i alle prosessene sine. Reality Capture sier de er konfidensielt.

For å generere en punktsky ble det brukt forskjellige typer bilder. Det ble brukt flybilder og bakkebilder. Prosesseringstiden varierte alt fra 10 min til et døgn, dette avhenger av antall bilder og type bilder. Ved orientering og finne sammenbindingspunkter tok alt fra 10 min til en time. Mens prosessering av modell fra generert punktsky brukte programmet alt fra en time til et døgn og kanskje flere døgn, det avhengig av hvilke parametere som ble brukt. Det som ble lagt merke til er at kun flybilder tok kortere tid enn bakkebildene og kombinasjon. Bakkebilder utgjør mange flere bilder enn flybilder og er flere overlapp enn ved flybildene som er større og dekker større området.

For å oppsummere så er Reality Capture en trinnvisprosess som først generere en tett punktsky, så en ny prosess for generering av en modell. Modellen blir ikke ferdig teksturert dette er en prosess som gjøres helt til slutt. Man kan gjøre innstillinger etter behov, men også velg å bruke defaultverdier om man ønsker å se en modell og ikke trenger å gå nært innpå. Disse bør tilpasses en standard i forhold til bruksområdet.

5.1.3 Pix4d Mapper:

I Pix4d kreves det svært lite for å lage en 3D modell. Programmet er selvforklarende og enkelt og bruke. Dette gjelder i stor grad alle funksjonene i programmet. Disse er enkel og bruke og krever lite forhåndskunnskaper. Etter modellering er ferdig kommer det ut rapport om antall punkt, tid, triangulering og alt som har skjedd under prosessering.

Det er ikke mulig å endre parametere før prosessen starten. Med dette ville en eventuell brukeren med noen erfaring klare å inkludere bilder som ikke går igjennom automatisk, samt utnytte dårlige bildesett i større grad. Det ble brukt defaltverdier, så raskt at trial lisensen til programmet ikke taklet flybilder. Dette pga. høy flyhøyde og for mange megapiksler i bildene. Ut fra det fikk jeg lånt meg en lisens for programmet til å ta inn flybildene.

Programmet klarer å gjennomføre prosessering veldig raskt alt fra 1 time til maks 4 timer avhenger av antall bilder. Men slik programmet fungerer er det mer avhengig av type bilder som blir tatt inn og netthasigheten i stedet for maskinvaren for en god kvalitet.

5.2 Ivaretakelse av geometri og tekstur:

Prosessene i programmene er ikke uten feil. Det er flere steg man skal igjennom for å få en 3D mesh og lukket solid. Underveis i prosessene kan algoritmene gi resultater som har feil. Dette kan være for få eller dårlig sammenbindingspunkter, eller feil i kombinasjon av datasett. Disse feilene blir med i videre prosesser og med det få en fordreid modell til slutt. Andre faktorer som påvirker dette er skrevet i kapittel 2.6. Det er snakk om kvaliteten på ferdig modell.

5.2.1 Agisoft Metashape:

Alle bygninger blir gjengitt med riktig former i den forstand at vi ikke har kontrollert vinkler, men rent visuelt.

Langs kanter som grenser mot friluft eller det ikke er noe mer punkter, hender det at man får med piksler som tilhører veggen som ligger bak, og vi får en strekking av veggen eller bakken. Disse er godt synlig i punktskyen og blir klipt vekk før det blir laget en modell. Ellers blir det ikke en god modell. Dette forekommer i ytterkant av modellen og det er lite punkt.

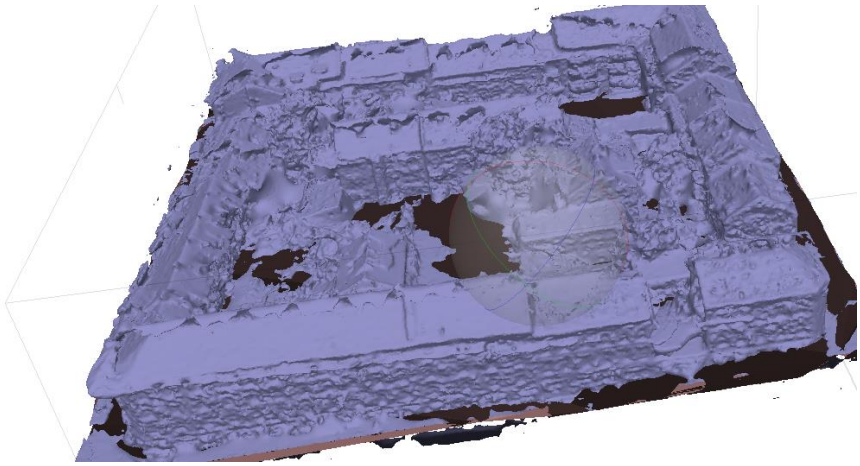
I Agisoft Metashape er som sagt en moduel build mesh som generer selve solidsen fra en punktsky. Modellen er enkel å bruke, kan velge mellom å vise solids eller 3D modell med tekstur. Modulen er enkel og grei, ikke mange valg (se kap 4.4.1 for beskrivelse av valg). Den endelig 3D solids avhengig av hvor mange tie punkt som blir funnet under bildematchingen og er avgjørende på dense cloud.

Figur 32 Vegetasjons utfordringer.

Figur 32 viser at softwaren sliter med generering av vegetasjon så enkelt av vegetasjon blir satt sammen med veggen, dette er spesielt i bakgården der det er vegetasjon nærme veggene. Dette fører til at geometrien på bygg ikke blir helt rett der det er vegetasjon nært vegger.

Vegetasjonsproblemet er noe som kan fikses ved at det blir fjernet vegetasjon enten ved å klassifisere punktskyen, men det er dat skummelt for noen av punktene blir ikke regnet som vegetasjon eller blir noe av veggen regnet som vegetasjon. Det hjelper også når det ble tatt bilder i bakgården. Dette gjør at vi klarer å skille bedre vegetasjon og vegg om de er veldig nærme hverandre. Det har nok noe å gjøre med at det blir dekt og får en tettere punktsky i

forhold til bare bruke skråbilder fra fly. Det blir en tettere punktsky og med det klarer algoritmene å skille mellom vegetasjon og bygg punkter. Noe den sliter med ved bruk av flybilder. Det blir skarpere kanter og med det ser tydeligere linjer.



Figur 33 Hele bakgården med fly bilder, kan se her at teksturen ikke er punktskyen som meshen blir laget av er vanntett.

Ser man på ferdig modell i forhold til orientering av bilder og funnet av sammenbindingspunkter i forhold til hverandre har programmet klart dette veldig godt ved bruk av SFM algoritmen. For orientering av kameraer har noe å si på resultat. Det hadde også hjulpet å lage inn kameraposisjoner noe som ikke ble gjort, siden det ikke fulgte med noen posisjoner for bildene.

Figur 34 Modell fra bakke - og flybilder, testområde 1.

For å generer teskstur på modellen kan man *build textur* ved å bruke bildene. I tillegg til at ved *build mesh* blir det laget på farge ut fra punktfargen og intensiteten ut fra pased based metode. Den ser på nærmeste naboen og setter samme farge verdi på de rundt, i *build dense cloud*. Dette er ikke noe som blir gått inn på siden det handler mest om å sette riktig farge på objektet noe som ikke er relevant til oppgaven siden den handler mer om å lage et vanntett solid uten teksturering.

Figur 35 Modell fra bakke - og flybilder med teksturering.

5.2.2 Reality Capture:

Opplever at Reality Capture er det programmet som lager den beste modellen rent visuelt, spesielt på nært hold. Dette er noe som er viktig for oppgaven siden kommunen ønsker at modellen skal printes ut og skal være på linje med NT 3D bygger resultatet. Modellene er svært detaljerte og har liten grad av fortegning. Programmet klarer å gjengi bygninger veldig bra, både ved kanter og tekstur. Dette gjelder når vi zoomer inn på modellen og ser på nært hold også. Kan se tydelige opp mot samme detaljer som er i virkeligheten. Da kan man si at det er en god modell når konturene fremdeles er veldig bra samt at teksturen blir godt bevart.

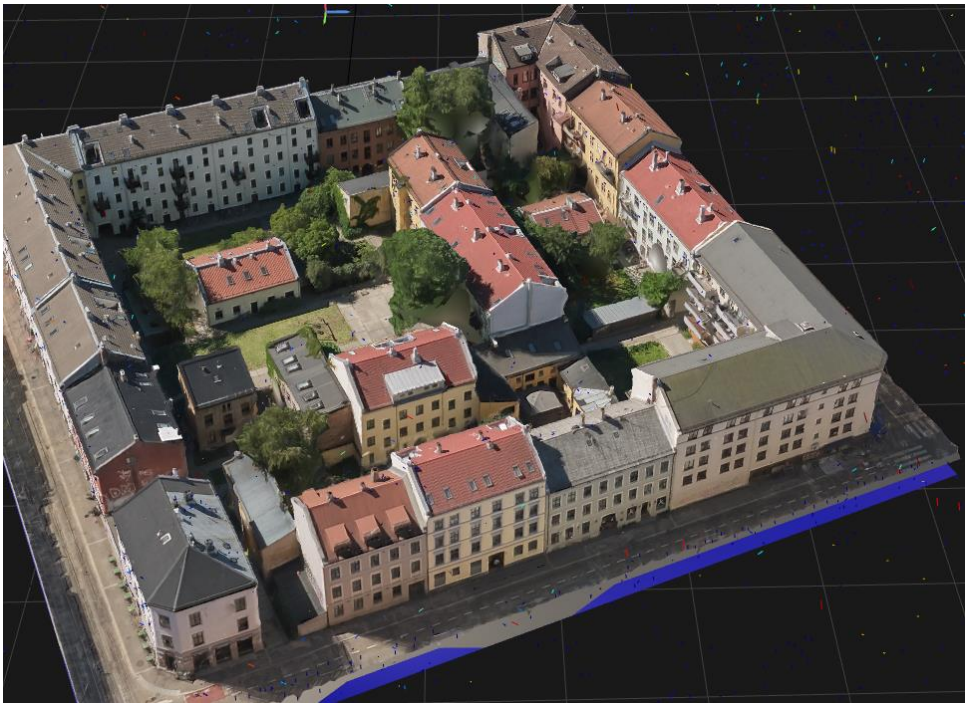
Figur Her ser vi at det er mulig å se hva folk i bygget har på kontorpulten sin.

Figur 36 testområde 2, vi ser tydelig hva som vises på vinduet og kan se det sitter en person der og hva han har på pulten. Vinduene med noe i er lettere å lage en solids på. Det er ser bedre ut med tekstur på.

Det ble også testet området med homogene overflater (nabobygget, figur 36), dette var noe programmet greide overraskende bra. Det har vært en liten utfordring i både pix4d og Agisoft Metashape. Dette er et eksempel på at programmet fyller ut alle mulig hull i punktskyen veldig bra og det klarer å lage en geometrisk riktig modell.

Programmet krever 2 bilder for å klare å lage en modell. Det krever da god overlapp mellom bildene for å klare å lage en modell.

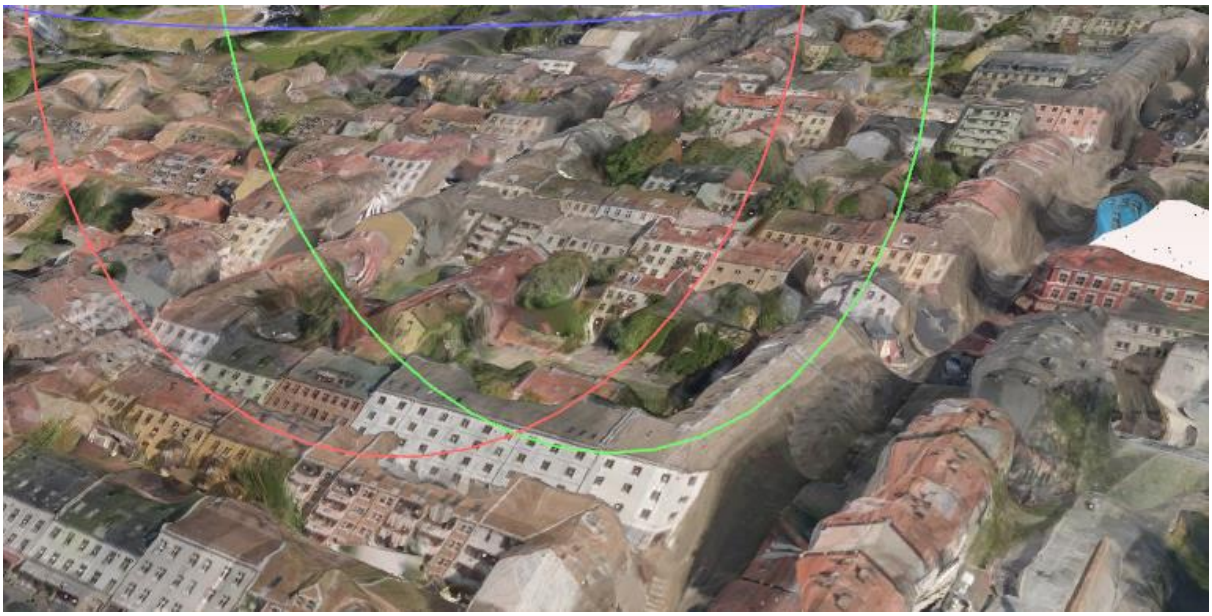
Reality Capture er bedre egnet til å skille ut vegetasjon og husvegg i forhold til Agisoft, der det var et problem. Algoritmen til Reality Capture ser ut til å klare å klassifisere ut punkter som hører sammen ved rekonstruksjon av punktskyen uten å måtte foreta en separat klassifisering først. Det ser ut som algoritmen har det innebygd, i forhold til hva Agisoft har. Dette er noe usikkert siden det ikke kan sies med sikkerhet hvilke algoritmer som er brukt for å bygge opp punktsky og 3D modell. Reality Capture vil holde det for seg selv og sier det konfidensielt. Det er også slik i Reality Capture som med Agisoft at det er enklere å lage en god rekonstruksjon med en tettere punktsky. Dette fram hever teksturen på bygget enklere og i Reality Capture blir det ikke laget runde flater som det ble i Agisoft.



Figur 37 Bilde av modell med tekstur, kvartal i fly perspektiv.

5.2.3 Pix4d Mapper

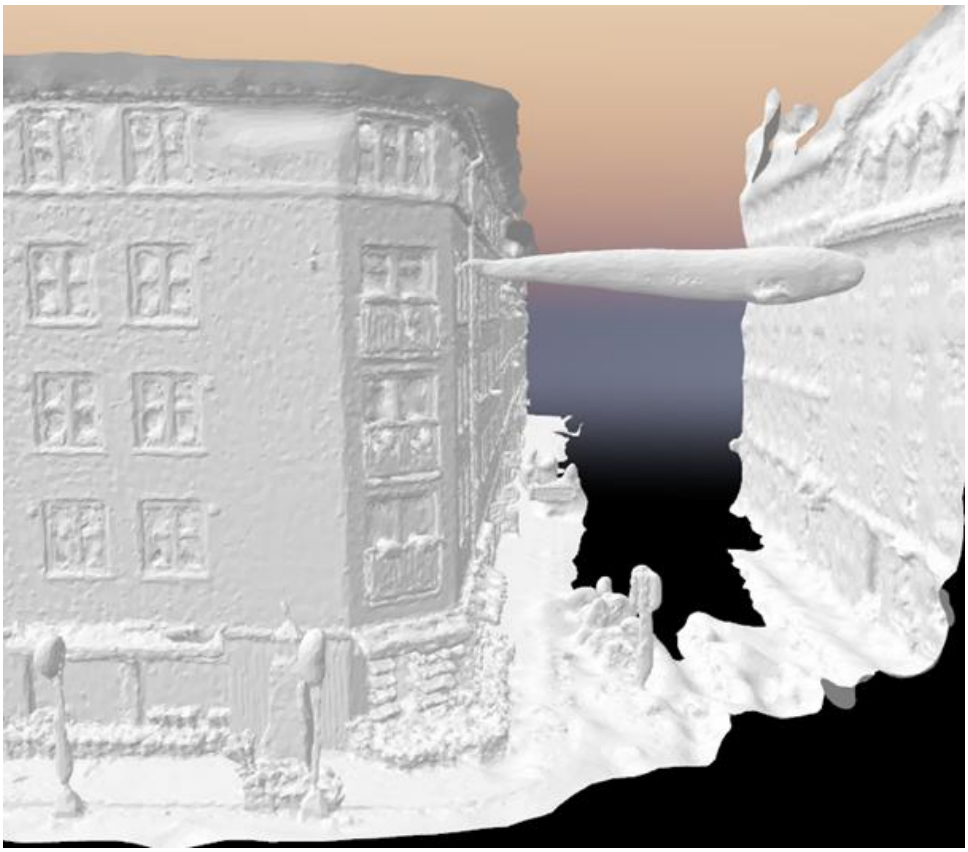
I pix4d kommer det to resultater en fra fly og en fra bakken på begge områdene. Ser tydelig forskjeller mellom de resultatene. Klarer ikke å slå sammen bakke og flybilder. Det har noe med for lite overlapp, men også med at programmet ikke liker bilder tatt med ulik kamera. Dette er en feil melding som kommer opp når det prøves å prosessere. Pix4d er et program egnet for dronebearbeiding, står det på deres nettside og Norconsult sier det også på sine sider. Norconsult er leverandøren for Pix4d i Norge. Noe som resulterer i at resultatet fra flybilder blir dårlig.



Figur 38 Testområde 1, Kvartalet. Ser at det er liggende bilder, dårlig til å takle flyfoto.

Figur 39 Testområde 1, med flybilder

Her kommer forvrenging av bildene tydelig fram. Det har nok med avstand til objekt som gjør inn zoomingen så dårlig. Pix4d er bygget på en algoritme SIFT, som skal kjenne igjen objekter og detaljer i bilder, som om bildet roteres, skaleres eller forvrenges. Samt så bruker pix4d Bundler adjustment metoden også kalt strålebuntutjevningen for å finne koordinater til felles punkter. Så kamerahøyden har noe å si for resultatet med disse algoritmene. Etter å ha fått lånt en lisens så kunne jeg ta inn originale flybildene, ble ikke kvaliteten spesielt mye bedre. Bedre resultat på testområde 2 men ikke veldig. Det kan legges inn fil med kameraposisjon X,Y, Z, omegha, phi og kappa for hvert bilde hadde det hjulpet noe med tanke på orientering av bilder i forhold til hverandre.



Figur 40 Testområde 1, solid, kun bakkebilder.

Figur 41 Testområde 2, bakkebilder

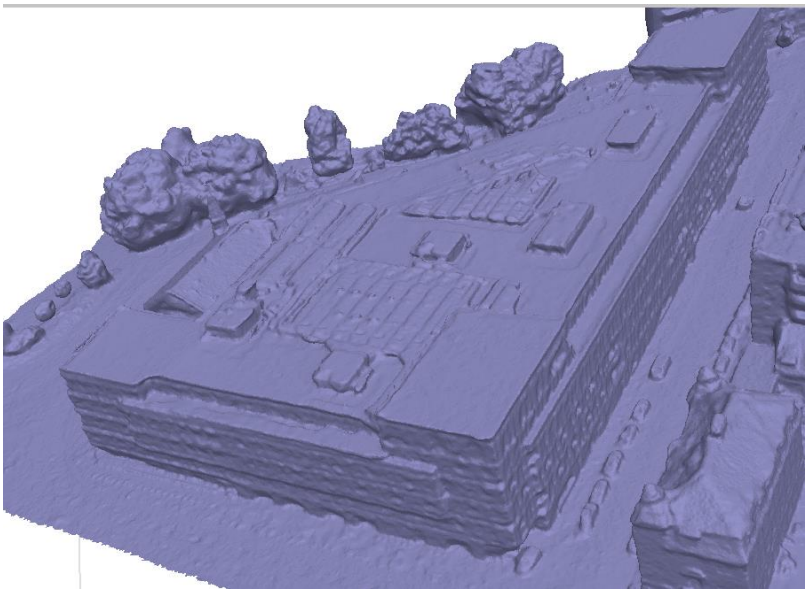
Ser vi på resultatet som ble av og bare bruke bakkebilder så er det et veldig godt resultat, ser tydelig kanter og linjer på bygget. Bygget blir detaljert. Her klarte programmet å ta inn bildene. Sliter litt med bildematching, noe som har med vinkelen opp på huset. Spesielt i hjørner, vanskelig å matche bilder i hjørnene slik at de klarte å matche neste vegg. Bilder i hjørner er kritiske steder, er det ikke lett å få god nok overlapp og dette programmet virker sårbar på over lapper i hjørner. Med det klarer ikke programmet å utnytte bildedatasettet maks. Dette er spesielt på homogene flater.

5.3 Vanntett solids:

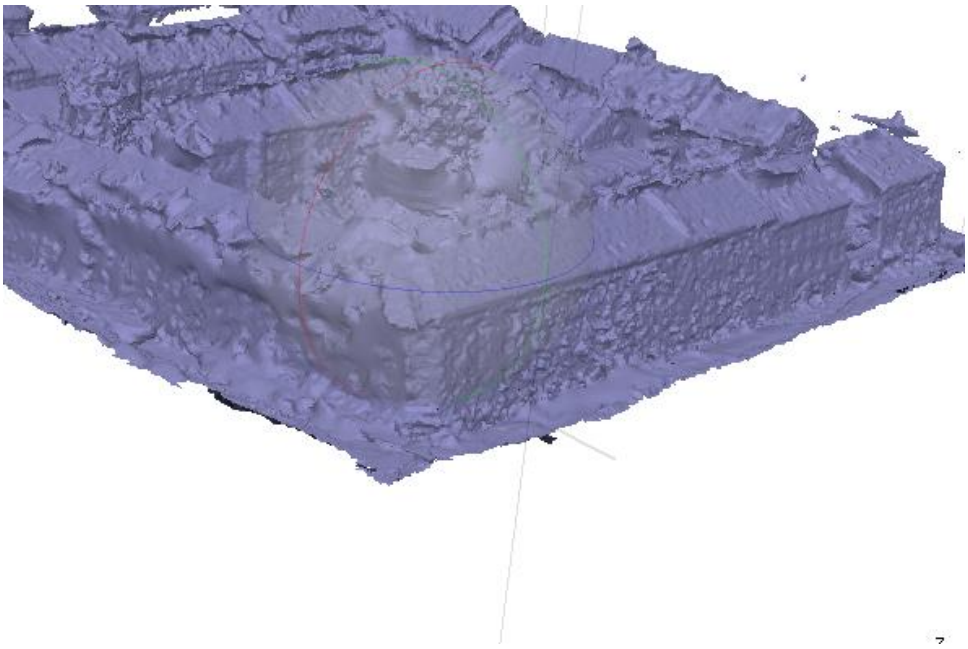
Med vanntett soldis menes en 3D modell som er uten hull og med da kan eksporteres og printes om ønskelig. Er det mulig å bruke dette i forhold til resultatet i NT 3D bygger. I alle programmene er med to resultat, en med bare flybilder og en med en kombinasjon av fly - og bakkebilder. Er modellen skalert og georeferert programmet.

5.3.1 Agisoft Metashape:

Med Agisoft Metashape er det varierende resultat ut fra hvilke datasett som blir brukt. Brukes det kun flybilder får man et vanntett solids uten å gjøre noe, men dårlig kvalitet på solids. Dette synes godt ved vinduer spesielt, de blir forvrengt helte nede ved bakken. For testområde 2, kan man se at det er enkelte steder at det er flyvende tak. Det har noe med at det ikke er mange nok flybilder som gjør det vanskelig for algoritmene å matche bilder sammen. Tak ligner homogene flater og med det veldig mye som er likt på tak. Her har belysning på når taket noe å si også.

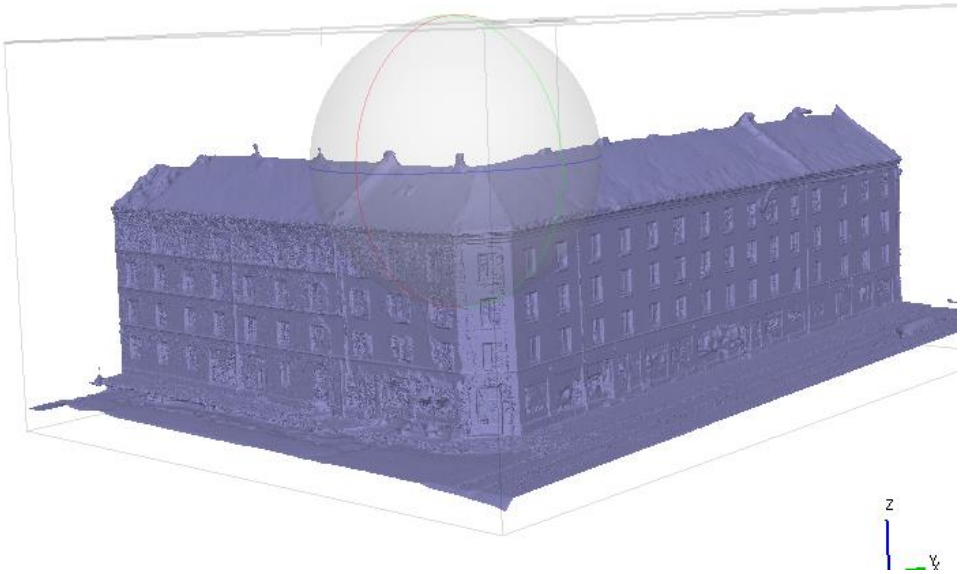


Figur 42 Testområde 2, flybilder



Figur 43 Testområde 1, flybilder, Agisoft Metashape

Ved bruk av kombinerte datasett er resultatet en mer detaljert solids, se forskjell mellom figur 42 og figur 43. Solids blir tettere og mer detaljert. Får mer rette skarpe linjer.



Figur 44 Testområde 1, kombinasjon med bakke - og flybilder.

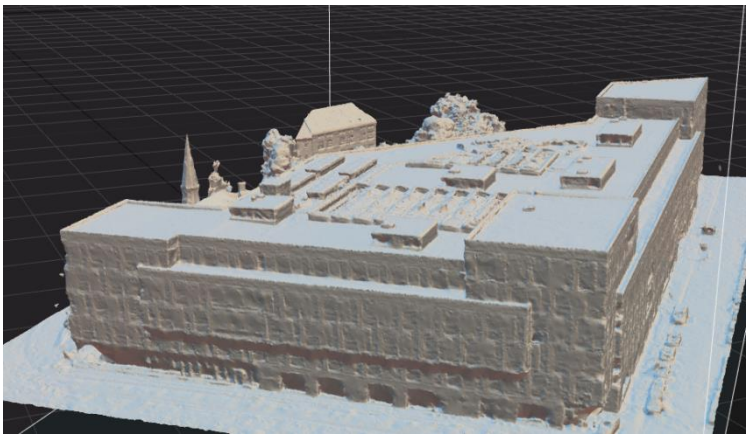
Modellen som kommer fra figur 44 er en mulig stedfortreder for modeller som blir laget i NT3D bygger. Siden det er rette kanter og tydelige vinkler.

Ser vi på forskjellen mellom figur figur 42 (testområde 2. flybilder) og figur 43 (testområde 1 flybilder) blir modell for nabobygget bedre kvalitet og tettere enn kvartalet med bare bruk av

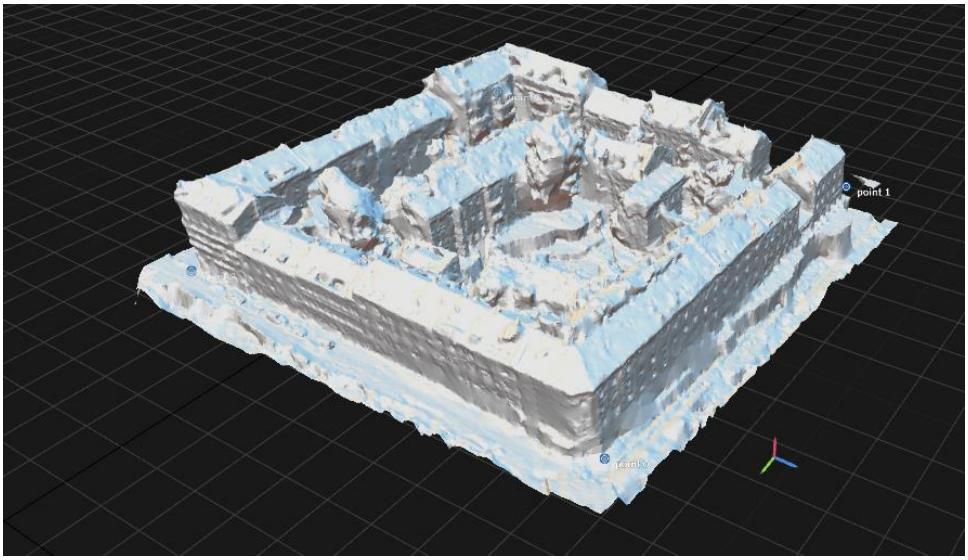
flybilder. Det blir klare linjer og kantene blir tydelig. Her kunne man nøye seg med kun bruk av flybilder for testområde 2. Det gir et resultat som kan brukes på lik linje med NT3D bygger. For testområde 2 så må det til mere bilder enn kun flybilder for å kunne få klare linjer og kanter.

5.3.2 Reality Capture:

Reality Capture klarer å tette byggene veldig bra. Selv om flater er homogene er det veldig håndterlig i Reality Capture. Det som er problemet er når det er mange vinduer og vinduer med størrelse på. Ser det veldig godt på nabobygget sitt resultat nederst i venstre hjørne på bygget.

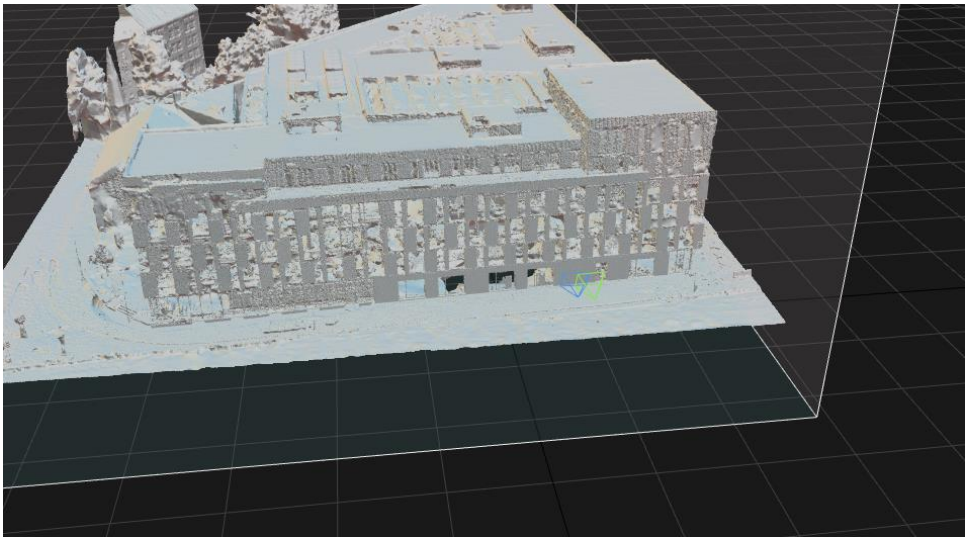


Figur 45 Testområde 2, flybilder, Reality Capture.



Figur 46 Testområde 1, flybilder

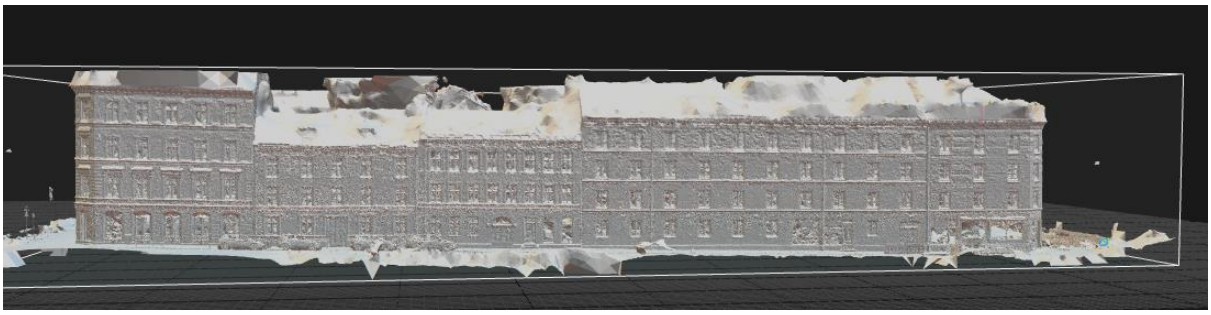
Her ser vi at det blir store hull og modellen blir tettet inne i bygget og greier ikke og trekken flate der det er vinduer. Vinduer er en vanskelig flate å beregne på og alle punkt på flaten er like. Det er fare for gjen skinn også, noe som gjør det enda vanskeligere å prøve å matche punkt på vindu. Det blir rett og slett ikke nok punkt på vindusflaten for å genere vanntett solid for bygget. Vinduer er en kritisk flate, det trengs utrolig mye punkt for å kunne dekke alt. Her hadde nok løsning vært laser som skyter mange punkter, og får en enda større punkttetthet enn med bildematching. Det blir markant forskjell når det legges til bakkebilder, men ikke nok til å klare og dekke vinduene.



Figur 47 Testområde 2, kombinasjon av bakke - og flybilder

Her ser det enklere siden det ikke er så store vinduer på bygget, her er vegetasjonen et større problem som forklart i 5.2.2

Det ble en markant forskjell når bakkebilder både inni bakgården og uti gaten ble med, hvor bedre kvalitet på solids ble.



Figur 48 Testområde 1, kombinasjon av bakke - og flybilder

5.3.3 Pix4d Mapper

Her ble det ikke noe resultat av vanntett solid, pga. Pix4d Mapper ikke klarte å lage en mesh av flybilder. Det skjer en forvregning av flybildene. Det blir liggende modell se figur 38 og 39. For et vanntett solids er jeg avhengig av flybilder og at det går ann å kombinere bilder fra ulike kameraer noe som ikke pix4d klarer. Det er heller ikke mulig å sette sammen punktskyer fra de ulike datasettene heller. Dette programmet krever at alt er tatt med samme kamera. Algoritmene klarer ikke kombinerer flere datasett fra ulik flyhøyde å match.

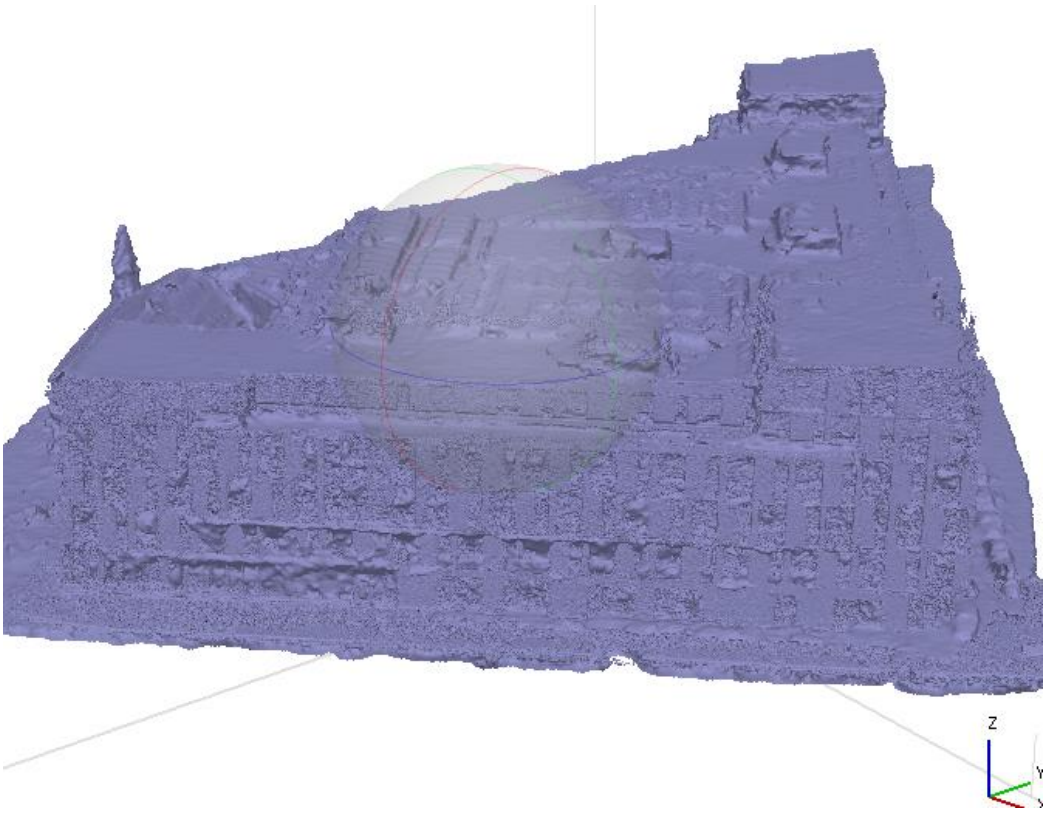
Ser at Pix4d er laget for bilder tatt nærmere enn de fra fly ut fra generering av gata. Pix4d klarer å genere en tydelig modell i forhold til bare bruk av bakkebildene. Dette er bilder som er nærmer på bygget noe som gjør at algoritmen i programmet takler det bedre. Flyhøyden spiller inn på sammenbindingspunkt i forbindelse med algoritmen STIF og bildematching algoritmen Bundler som brukes i programmet. Det står samtidig i programbeskrivelsen at programmet er egnet for droneflyving og ikke fly.

5.4 Sammenslåing av datasett:

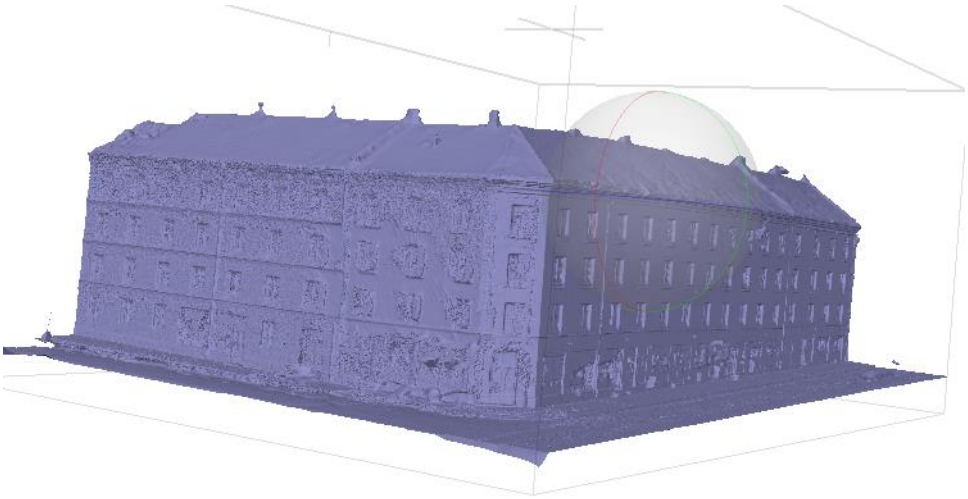
Hvor godt lar det seg gjøre å kunne kombinere flere ulike datasett tatt med forskjellige sensorer og i forskjellige høyder? Kan programmene kombinerer ulike datasett, slik at modellen fører til en bedre dense cloud eller en tie points, som igjen fører til en bedre mesh.

5.4.1 Agisoft Metashape:

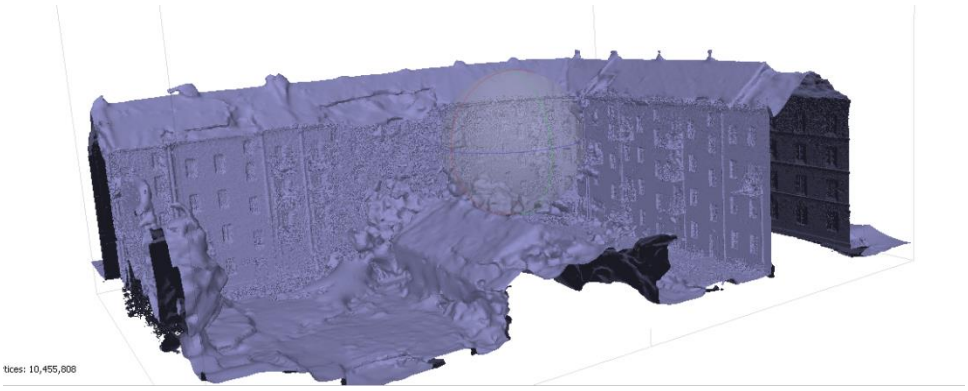
Det ble prøvd og kombinerer bilder, fra fly, bakke og skann, noe programmet ikke klarer uten at de må lages flere dense cloud og merge de sammen. Softwaren er i stand til å rekonstruere en modell ved flere bilder om vinkelen er mindre enn 15 grader og blir bedre hvor mindre vinkelen er. Mellom bakke- og flybilder er det en stor vinkel, større enn 15 grader. Så disse lot seg ikke overlappe og ble ikke matchet sammen. Mens bilder fra terrase og gata klarte å matche selv om det var tatt med forskjellige kameraer. Det ble tatt bilder fra terrasen til PBE for testområde 2. Dette ble ikke gjort det til testområde 1, der er det bakkebilder, flybilder og lasersky. For at vi skulle kunne ha matchet fly og de fra bakkenivå må det taes i flere høyder slik at det blir god overlapp. Hadde flyet flydd i flere høyder over og brukt droner i de nærmeste ville det ha blitt et godt resultat.



Figur 49 Testområde 2, kombinasjon av bakke - og flybilder. veggen mot PBE



Figur 50 Testområde 1, fra gata

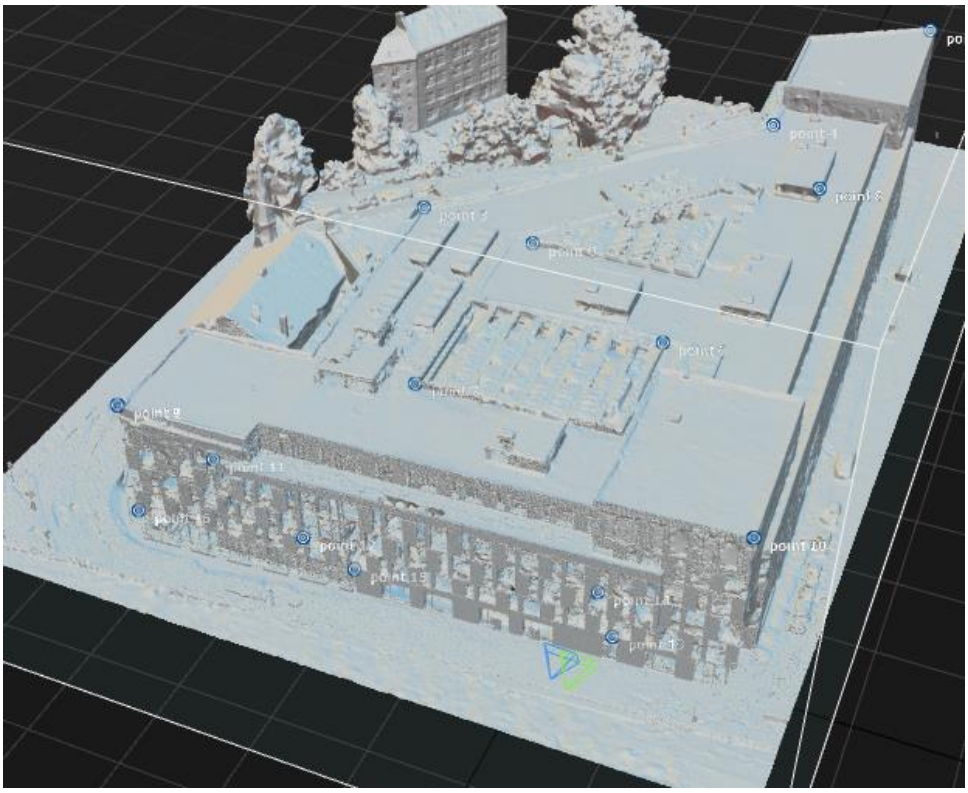


Figur 51 Testområde 1 fra bakgård, Det er kun veggen som er tatt bakkebilder på ikke det i bakgården.

Det ble også brukt en punktsky fra sporveien for testområde 1. Den ble lagt inn som egen dens cloud og merge sammen med alle dense clude slik at det ble en felles dense cloud å lage solid av. Siden det er forskjellige kameraer som er brukt og det er brukt laser på testområde 1 klarer ikke Agisoft Metashape å kombinere de datasettene uten å georeferer hvert enkelt datasett. Dette for å få en mer detaljert og nøyaktig modell. Alle dense cloudene ble georefert og med det skalert, se figurene over.

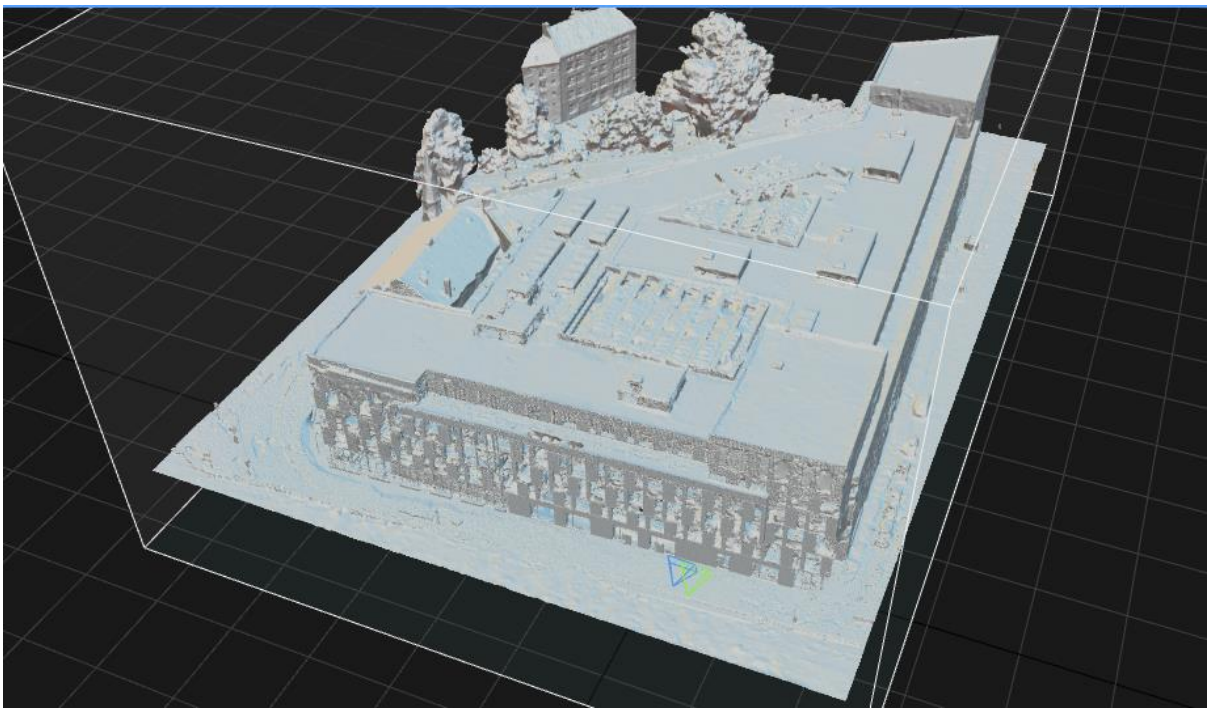
5.4.2 Reality Capture:

Flere variasjoner av bilder og skann blir testet. Programmet klarer å lage en god 3D modell med ulike datasett, men dårlig på å kombinere de sammen. Dette er spesielt mellom fly og bakkebilder. Programmet lager punktskyer i forskjellige komponter for å skille de ulike datasettene. Komponentene ble lagt sammen til en felles komponent for å kunne lage den beste 3D modell med alle datasettene. Dette ble gjort ved å legge inn 9 kontrollpunkter for to og to lag i alt 17 komponenter for testområde 2 og 21 kontrollpunkter for testområde 1. Se vedlegg Rapport. Det måtte så mange punkter for å kunne knytte datasettene sammen. Hadde vi mindre så ble det forvrenging. Det er enkelte av punktene som ikke klarte å matche sammen, så ble en del avvik. Der kommer det forskyvning i datasettene mellom dem inn.



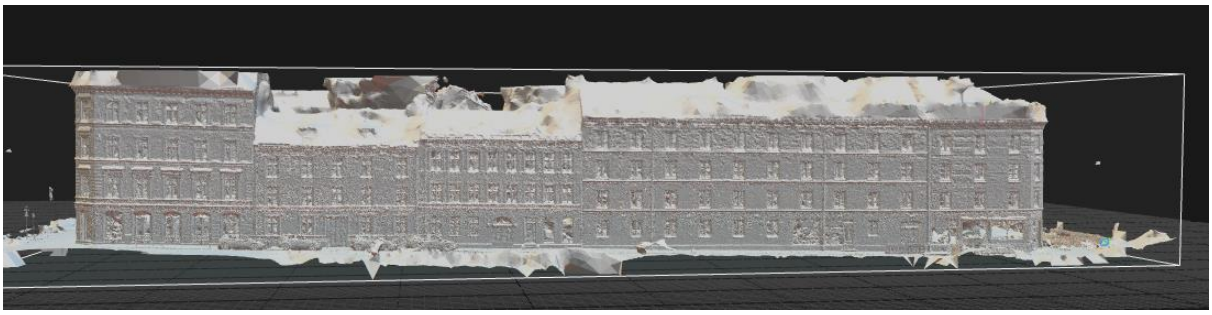
Figur 52 Testområde 2, viser kontrollpunkter.

Grunnen til at de ikke greier matche fly og bakke bildene er at vinkelen er for stor slik grunnen var i Agisoft Metashape også. I Reality Capture har en toleranse vinkelen mindre enn 15 grader som i Agisoft, uten at det kan sies med sikkerhet siden alt er konfidensielt. Det legges merke til ved at terrasser, gate og fly kommer i hver sin komponent og ikke klarer å kombineres i forbindelse ved testområde 2.



Figur 53 Testområde 2, Hele bygget med bakke - og flybilder.

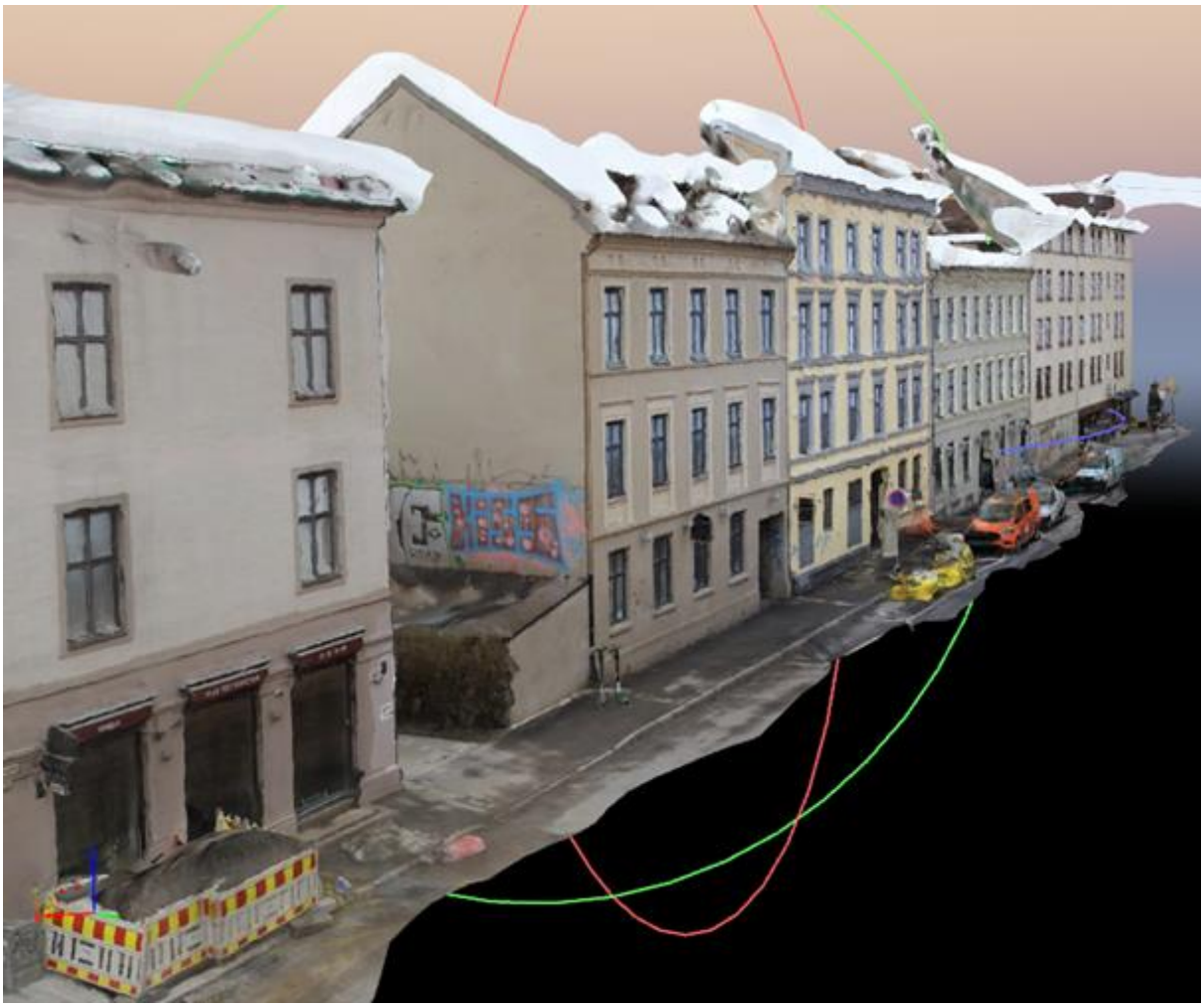
Punktskyen fra sporveien ble ikke brukt i Reality Capture, det var ikke lett å legge den sammen med de andre komponenter gikk ikke med kontrollpunkter. Ser vi på resultat ble det veldig bra uten å bruke punktskyen også.



Figur 54 Testområde 1, veggen ut mot gaten med kombinasjon av bakke - og flybilder.

5.4.3 Pix4d Mapper

Resultatet fra Pix4d ble to separate modeller, bakke og fly. Grunnen er at vinkelen mellom fly og bakkebilder er for stor om med det klarer ikke de å matche samme som i de andre programmene. Det er heller ikke mulig å kombinere bilder tatt med forskjellig kamera. Med det må de roteres med ekstern data, om rotasjon, posisjon til kamera. Med det legges alle datasettene i et lokat koordinatsystem og det ble heller ikke brukt targest så det er vanskelig å legge inn felles punkter i datasettene. Får med det ikke georeferert eller skalert modellen i programmet.



Figur 55 Testområde 1, bakkebilder, teksturering.

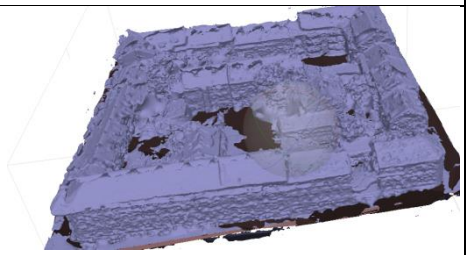
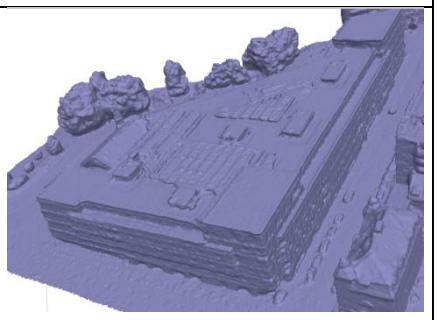
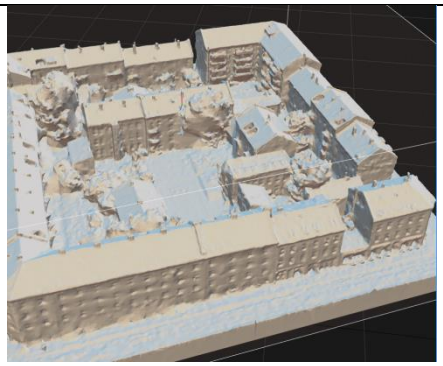
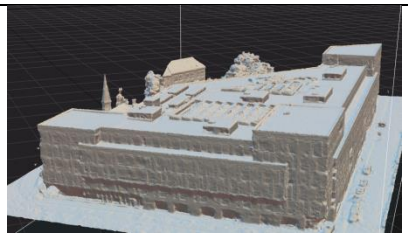
Programmet klarte heller ikke få med seg alle bildene som skal vært med. Det ble brukt under 50 % av bildene som ble lagt inn. se vedlegg x, rapport.

Figur 56 Testområde 1, kun flybilder modell.

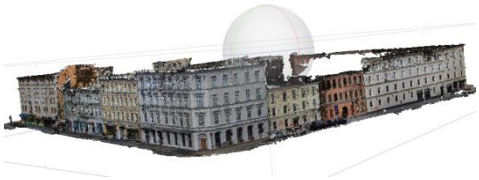
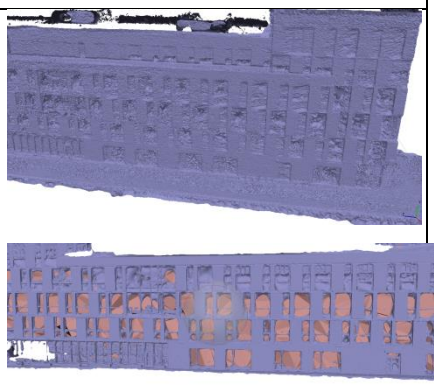
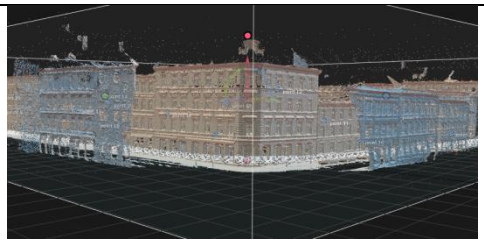
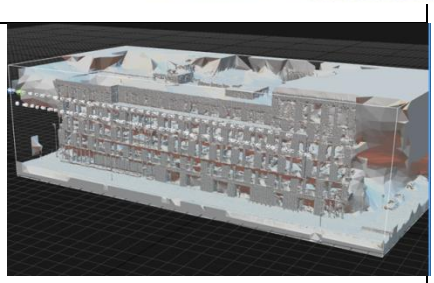
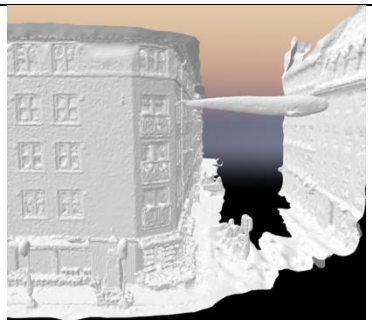
5.4.4 Solids med ulikt datasett:

Tabellen under viser ferdig solids med ulikt datasett. Den viser hvordan programmene klarer å utnytte data den får inn på en best mulig måte.

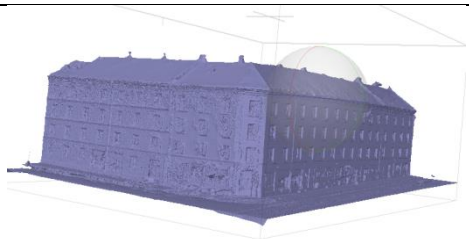
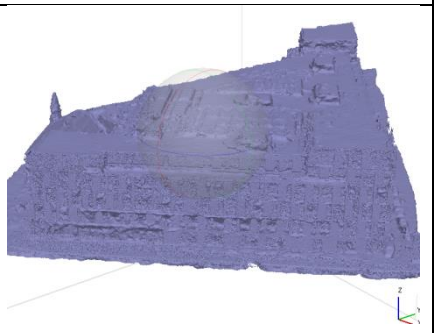
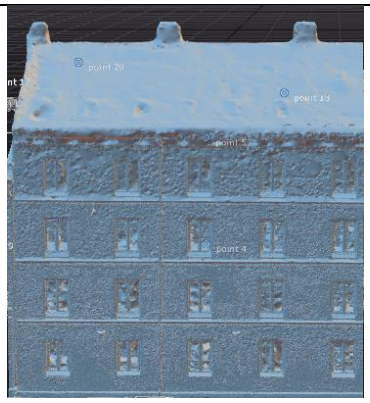
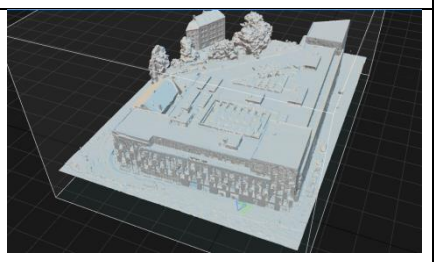
Tabell 2 flybilder, oppsummering

FLYBILDER	område	
	Testområde 1, kvartal	Testområde 2, nabobygget
Agisoft Metashape		
Reality Capture		
Pix4d		

Tabell 3 Bakkebilder, oppsummering

BAKKEBILDER	område	
	Testområde 1, kvartal	Testområde 2, nabobygget
Agisoft Metashape		
Reality Capture		
Pix4d		

Tabell 4 kombinasjon av datasett, oppsummering

KOMBINERT	område	
	Testområde 1, kvartal	Testområde 2, nabobygget
Agisoft Metashape		
Reality Capture		
Pix4d	Gikk ikke å kombinere	Gikk ikke å kombinere

Tabell 5 Oppsummering ved bruk av lasersky.

Data	**Område**	**Agisoft Metashape**	**Reality Capture**	**Pix4d**
Kombinasjon med punktsky til bilde	Kvartal		SJEKK OM MAN KAN TA INN FIL FRA AGISOFT	

5.5 Hvilken nøyaktighet kan oppnås:

Det varierer på nøyaktighet på ferdig solid ut fra hvilke datasett som blir brukt til å konstruere 3D modell. PBE ønsker et vanntett 3D solids slik at den kan 3D printes med en så god nøyaktighet som mulig i forhold til FKB data. Er dette den nye metoden å lage 3D – modeller på.

5.5.1 Resultat:

Tabell 6 Viser beregnet RMSD for alle målingene.

prosjekt/program	Agisoft Metashape	Reality Capture	Pix4d Mapper
Kvartal fly	0.061	0.104	
Kvartal kombi	0.085	6.273	
Nabobygg fly	0.073	15.144	
Nabobygg kombi	0.063	15.882	
Totalt:			

Utgangspunktet for alle modellene er at de er georeferert med samme bakkepunkt for hvert område. Referansemålene er fra FKB data for området.

5.5.1 Agisoft Metashape:

Nøyaktigheten i Agisoft Metashape er stabil uansett datasett som blir brukt, se tabell 6. Det svinger 2 cm i mellom flydatasettene og de kombinerte datasettene der bakkebilder er med. Det som er den største utfordringer er at det er noe “flyvende tak” som gjør det vanskelig å sjekke nøyaktigheten i enkelte deler av modellen. Ved at det ble brukt *close holse* og *smoothing* funksjonene gjorde at deler av flyvende tak ble minsket betraktelig.

Modellen skalerte seg med det godt når den blir georefererte, trengte ikke noe mer skalering for å kunne tilfredsstille kravene til nøyaktighet i XYZ. Kravet var på 20 cm. Dette ser og ved nøyaktigheten til bakkepunktene, se vedlegg C og D.

5.5.2 Reality Capture:

Ser vi på resultatet for Reality Capture i forhold til Agisoft Metashape er det betraktelig mye dårligere. Modellen med testområde 1 med kun flybilder ble ikke så ille som de andre modellene der RSMD ble 15-16 meter. Dette har nok med antall bilder. For det er kun 37 flybilder for testområde 1 og der ble det lagt inn kun 4 bakkepunkt. Med en gang det ble lagt inn flere bilder ble det dårligere nøyaktighet, men kvaliteten på modellen ble bedre. Selv om det ble lagt inn 20 bakkepunkter ble det ikke noe bedre resultat. Dette tyder på at Reality Capture ikke takler å georefere mange bilder på en gang. Når antall bilder kommer over 40 bilder må modellen eksporteres slik den er for å kunne holde kvaliteten.

Ved å eksportere modellen kan georeferering og skalering skje i et annet program som f.eks. FME. Men siden det ikke blitt gitt nok credits for testing av programmet så ble det vanskelig å eksportere modellen, siden de er så store. Fra Reality Capture fikk jeg og enn annen person teste programmet og 10 000 credits for eksportering av modeller. Siden modellene ligger på 4000-5000 credits for eksportering eller 20 dollar fikk vi ikke til å eksportere både mine og hans andre modeller. Men han andre testet å eksportere modell og georeferert og skalere i FME. Det ga et godt resultat.

Modellene er fine på alle andre områder enn akkurat skalering. Ut fra nøyaktighet på punkter ser det ut som modellen er georeferert, sjekk nøyaktighet bakkepunkter vedlegg E og F. Det er enkelte punkt som blir dårlig pga. vanskelig å finne rett plassering i bilde og det er få bilder med samme punkt som fører til dårligere nøyaktighet på punktene.

5.5.3 Pix4d Mapper:

Siden resultatet av modellene ble som beskrevet i de andre testkriteriene med liggende hus og ikke vanntett solid ved kombinasjon av datasett. Ble det ikke målt nøyaktigheten i dette programmet. Det ble ikke videre sett på bakkemodellen siden den har en visuell god kvalitet. I den modellen er nøyaktigheten bra, men kan ikke sammenligne siden utgangspunktet er et vanntett solid. Bakkemodellen mangler tak. Det som kan sies er at bruk av bilder i nærfotogrammetri hadde gitt et godt resultat ut fra å sett på bakkemodellen som har blitt laget.

5.5.4 Feilkilder:

Den viktigste feilkilden i denne oppgaven er målinger på modellen for det gjøres manuelt. Erfarte at det å måle direkte i modellen fungerte veldig bra, men det forutsetter litt lett analyse av modellen for å finne ut hvor kanter og andre detaljer faktisk befinner seg hvis teksturen er dratt utover. Det hadde vært ønskelig med en måte å kunne måle direkte i bildene også for sammenligning. Måling direkte i modellen i de tilfellene der avstanden man skal måle er udefinert som mellom de to modellene i samme testområde. Kan føre til feil plassering av målepunkter. Det har vist seg at vis man har uklare start - og endepunkt kan modellen snus så den sees fra innsiden. Med det ser man start og endepunkt tydeligere. Dette gir noe mer nøyaktige målinger. Kan forklares med at man ser detaljer fra et annet perspektiv og med det blir tydeligere.

For å begrense feilen ble det gjennomført på enkelte steder flere målinger der det har vært hensiktsmessig (modellene med flybilder). Det ble i enkelte program (Reality Capture) lagt inn flere bakkepunkter for å få prøvde å georeferere modellen samt flere sammenbindingspunkter for å kunne flette sammen flere av datasett bedre.

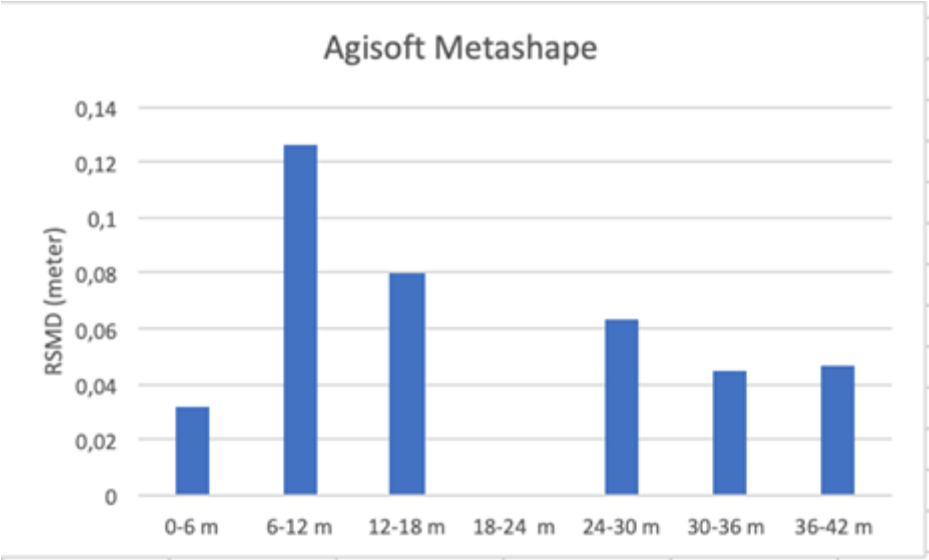
Siden i enkelte programmer blir lagt inn flere bakkepunkter eller sammenbindingspunkter gjør det at det blir forskjell i skaleringen. For jo flere punkter som blir lagt inn jo bedre blir skaleringen og georefereringen. Det kan gjøre at det blir feil mellom modellene fra de ulike programmene. Ut fra resultatet i tabell 5 ser vi at det er stor forskjell mellom programmene selv om det er brukt flere bakkepunkt i Reality Capture enn i Agisoft Metashape.

5.5.5 Del konklusjon:

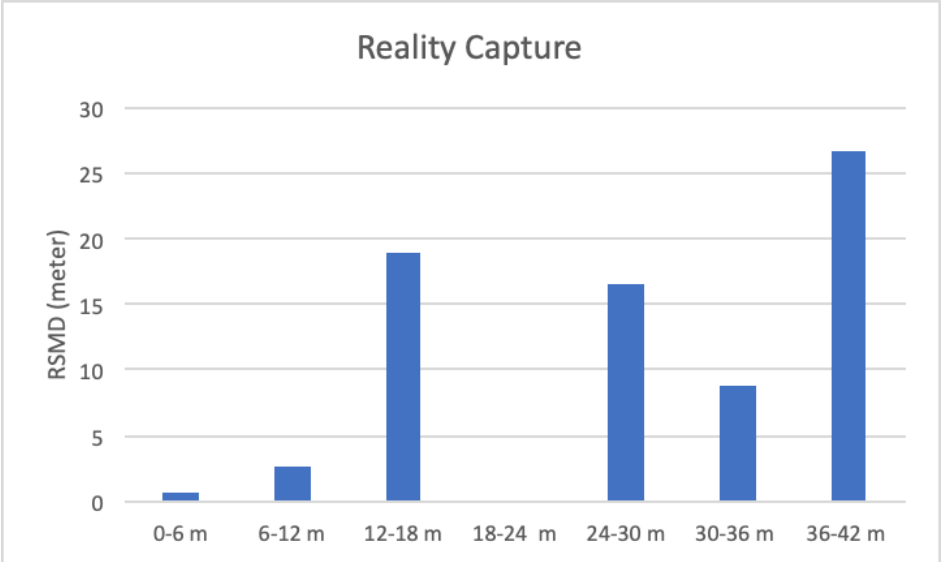
Alle programmene gir ulikt resultat. Dette kan sees ut fra tabell 5. Punkter som ble brukt for hver modell finnes i vedlegg A og B.

Tabell 5 viser at Agisoft Metashape gir et jevnt over bedre mindre feil enn de to andre programmene. Mens Reality Capture er gir varierende resultat ut fra hvilket testområde som blir sett på. Pix4d ga innen resultat på nøyaktigheten siden jeg ikke fikk et vanntett solid. Så på nøyaktighet ble fokus på Agisoft Metashape og Reality Capture. Det som utgjør den store forskjellen på disse to programmene er at i Agisoft har algoritmen klart å skalere modellen ut fra georefereringen i alle modellene. I Reality Capture klarte ikke algoritmen som ligger i

grunn å skalere ut fra georefereringen, slik som i Agisoft Metashape. Men anser at jeg har fått ut maksimalt av de programmene. Det som er løsningen for modellene fra Reality Capture er å eksportere modellene ut og skalere de i et annet program som f.eks. FME. Det som derimot vil ha noe ¨si er at jeg har jobbet lenge med de samme modellene og vil med det være i stand til å få et bedre resultat enn brukere som kun får en kort innføring.



Figur 57 RSMD (meter) fra Agisoft Metashape modellene.



Figur 58 RSMD (meter) fra Reality Capture modellene.

Valgte å ikke sette de i samme figur siden da ville resultatene fra Agisoft Metashape blitt liggende på null i forhold til Reality Capture. Klarer ikke å se noen spesiell trend på intervallene i de ulike programmene. Noe av det har med at modellene i Reality Capture ikke har blitt skalert slik det har blitt i Agisoft Metashape. Så det blir vanskelig å sammenligne modellene mellom de ulike programmene. Sees det bare på Agisoft Metashape alene kan man se at den ligger på 4 cm i alle intervaller unntatt mellom 6-18 meter. At de spriker fra de andre

kan ha med å gjøre at måner som måles og i flybildemodellen har flyvende tak som gjør at det blir vanskelig å måle korrekt. Det gjør at det blir dårlige målinger på disse. Reality Capture er det ikke noe å se på, siden modellen tydelig ikke blir skalert i riktig størrelse. Så er intervaller mellom de dårlig. Det er på intervallet 0-6 meter det er begynnelse innen for toleranse på nøyaktighet. Men dette er et program som bør testet videre med tanke på skalering og georeferering. Det er også bare testet på to ulike områder, noe som bør testet mer med. Inn legging av GPS data fra bildene er også noe som ville ha bedret modellen betraktelig. Da blir bildene satt på riktig plass i forhold til hverandre. Siden algoritmen til Reality Capture er konfidensiell er det vanskelig å si akkurat hva som skulle gjøre den bedre. Det kan hende det å legge inn GPS data ville ha hjulpet.

5.6 Andre data og bruksområder:

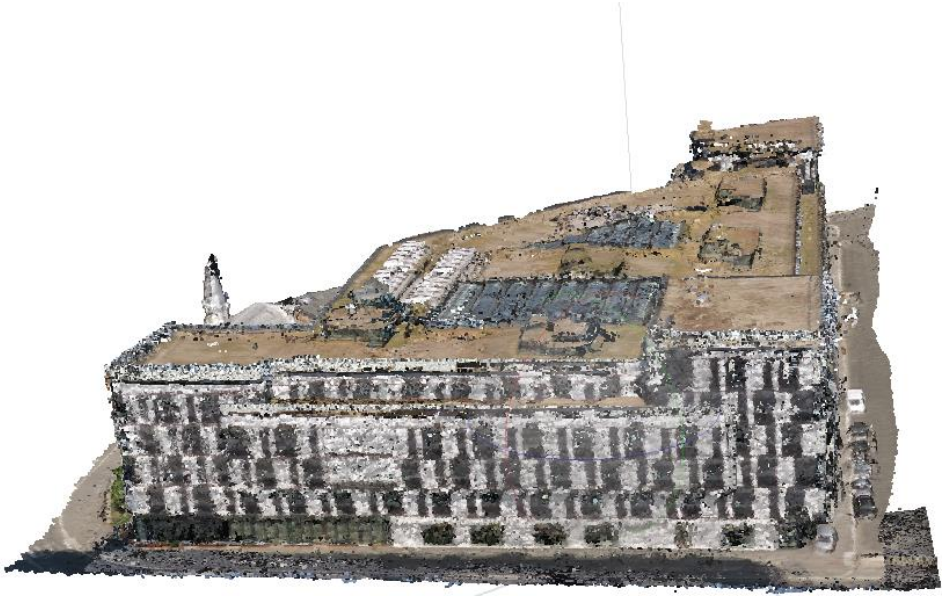
5.6.1 Sammenligning av Feature detection og matchingsalgoritmer:

Agisoft Metashape og Pix4d er de eneste programmene hvor vi med sikkerhet vet hvilke algoritmene som gjennomfører *feature detection*. Programmene gir varierende tilgang til dataene. I Agisoft Metashape gjennomfører som sagt *feature detection* med *pased based metode*. Mens Pix4d bruker SIFT algoritmen. Prosjektet som blir brukt er nabobygget. Resultatene tyder på at algoritmene for *feature detection* er forskjellig mellom programmene.

5.6.2 Agisoft Metashape:

Programmet kan helt klart benyttes til å genere lokale overflatemodeller eller 3D modeller over et større området. Det kan begrunnes i at man kan benytte veldig mange bilder og punktskyer og at punktskyen som generes inneholder lite støy. Samtidig kan man eksporter ut punktskyen og sette de inn i annet programvare. Programmet har mange ulike filformater og eksportere til. Vi har testet et bildesett med 37 bilder over kvartalet. Dette ga en punktsky som dekker et stort område. Med flybilder er det fotografert i fem vinkler i en og samme posisjon, slik at det dekker mest mulig. I hver posisjon blir det fotografert i rett ned, nord, øst, sør og vest. Dette fører til tørre dekning av området og det blir med mer detaljer. Hvor stort området man dekker vha. mengden bilder. Dette er tilfellet at det er tatt ut bilder med buffersone på 10 meter ut fra kvartalet.

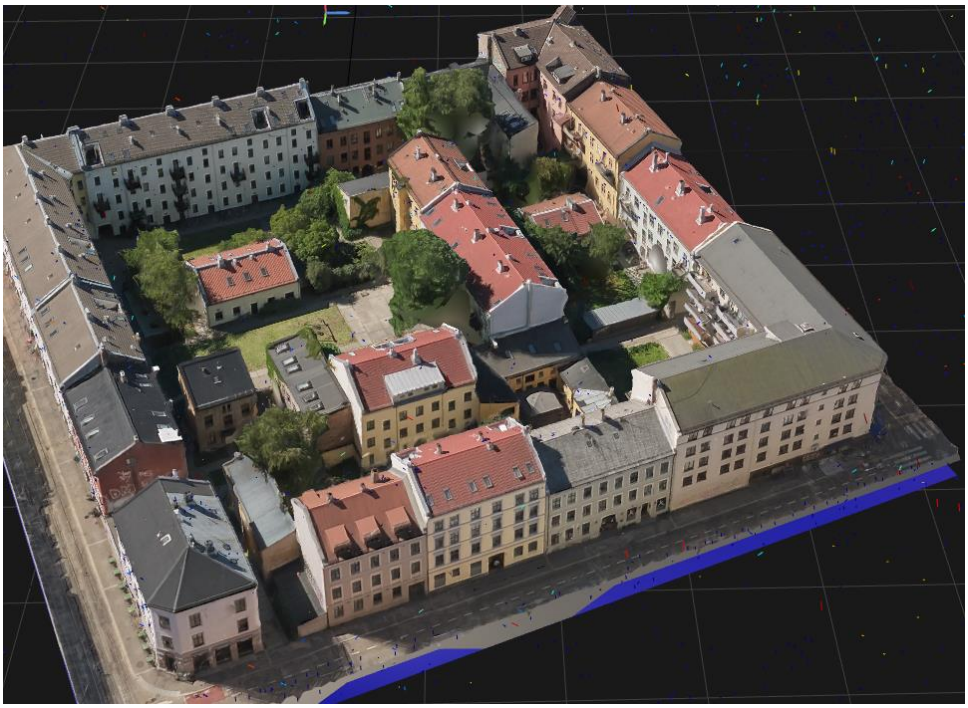
Figur 59 Dense cloud for testområde 1, kvartalet, 292 121 777 punkt.



Figur 60 Dense cloud testområde 2, nabobygget, 40 740 909 punkt.

5.6.3 Reality Capture:

Programmet selv kan beregne en automatisk 3D modell fra samme antall bilder som det ble lagt inn i Agisoft Metashape. Det som ble gjort er at det ble klipte ut kvartalet fra punktskyen som ble generert av de 37 flybildene figur 61. Resultatet blir bra i fly perspektiv.



Figur 61 Testområde 1,kvartal, 78 652 punkt. Figuren blir bedre når det blir lagt på struktur.

Som sakt tidligere har programmet en begrensning på visning til 40M polygoner som gjør det vanskelig når man skal beregne store bygninger og kvartaler med mye detaljer. For å vise bygget kan det eksporters og tas inn i andre programmer. En siste mulighet er å bruke "simplify" funksjonen etter 3d modellen er bygget for å slå sammen enkelte av polygonene.

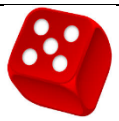
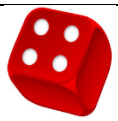
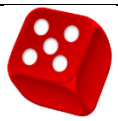
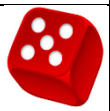
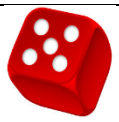
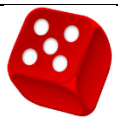
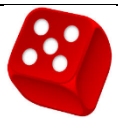
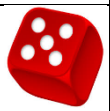
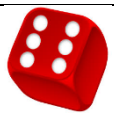
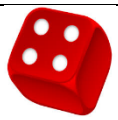
5.6.4 Pix4d:

Programmet er beregnet for å genere objekter som er fotografert eller skannet med drone og nærme objektet. Det er program som ikke klarer å bruke flybilder. Kun bruk av bakkebildene som forklart tidligere. Det er heller ikke egnet for solid, siden teksturering blir satt på automatisk. Det er ikke mulighet for å fjerne tekstureringen.

5.7 Oppsummering av resultatet:

Har vurdert de enkelte testresultatene og satt de inn i tabell 7. Se forklaring av tabell 7 under tabell 8.

Tabell 7 Oppsummering av resultatet

	Test 1	Test 2	Test 3	Test 4	Test 5
Agisoft Metashape					
Reality Capture					
Pix4d					

Test 1 Grad av automatisering

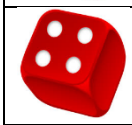
Test 2 Ivaretakelse av geometri og tekstur

Test 3 Vanntett solids

Test 4 Sammenslåing av datasett

Test 5 Hvilken nøyaktighet kan oppnås

Tabell 8 Forklaring på symboler

Terningkast	Forklaring:
	Fremragende bra
	Meget bra, men noe å utsette
	Bra, en del å utsette
	Mindre bra, noen mangler i funksjoner
	Dårlig/ ikke fullstendig funksjoner
	Elendig/fullstendig manglende

Tabell 7 viser at totalt sett kommer programmene bra ut på de to første testkriteriene så er det mer ujevnt blant programmene på de neste testkriteriene. Tar vi for hos hvert testkriteriet så ser vi at på testkriteriet 1 har fått godt terningkast for grad av automatisering. Dette har med hvor godt modellene lager seg selv, for lett forklarlig er programmene. Veldig enkel forklart hvordan bilder blir tatt inn og laget til en modell ved bildematching. I Pix4d er det enklest, det som er litt negativt der at de ikke går ann å skifte parametere, de er bestemt på forhånd noe som gjør at det ikke kan justeres og få et bedre resultat. Pix4d er fremdeles mer automatisert enn Reality Capture og Agisoft Metashape. I de programmene må man gjennom 2-3 prosesser for å få en modell, men her kan det skiftes parametere. Med det få en bedre modell, til slutt når man finner den perfekte balansen mellom parameterne.

Testkriteriet 2, ivaretakelse av geometri og tekstur, skårer alle programmene veldig bra når det gjelder bakkebilder, eller skann. Med flybilder ser man store forskjeller mellom programmene. I Agisoft Metashape er det veldig bra med kombinasjon av flybilder og bakkebilder, men dårlig om vi bare bruker flybilder. Det synes i form av ikke rette og tydelige kanter og en del flyvende tak. Reality Capture takler det mye bedre i begge kombinasjonen av bilder. Programmet får også et bedre resultat med flybilder. Det kan her gjemmes mye med å

legge på tekstur. Men det er ingen av programmene som klarer det kjempe bra. Pix4d får ett veldig godt resultat med bare bakkebildene, mens flybilde modellen blir forvrengt og mer liggende. Det er ikke lett å se klare kanter og linjer i den modellen.

Testkriteriet 3, vanntett solid: både Agisoft Metashape, og Reality Capture får nesten vanntette solids, i uansett kombinasjon av datasett materialet. Det at de ikke får topp score på dette testkriteriet er at i enkelte datasett (flybilder) må funksjonen close holes brukes for å kunne få et vanntett solid. Ingen av programmene greier det helt selv ved få bilder eller enkelte steder med lite dekning av data som inne i bygget på testområde 2 der det er mye homogene flater. Det blir også noe flyvende tak i Agisoft Metashape modellen der kun flybilder er brukt. Pix4d klarer ikke å kombinerer datasett fra to ulike kameraer så med der får jeg ikke vanntett solids. Med at taket ikke blir dekket. Flybilder blir lagt inn, men ser ut til at grunnen er for høy flyhøyde for den algoritmen programmet er bygd på. Det hadde blitt ett bedre resultat om det hadde blitt lagt inn GPS data for alle fotograferingsposisjonene. Dette gjelder for alle programmene, men spesielt for Pix4d.

Testkriteriet 4, Sammenslåing av datasett: Agisoft Metashape og Reality Capture klarer å ta inn all datasettene. Det som trekker ned her at man må finne sammenbindingspunkter mellom de ulike datasettene slik at de legger seg sammen. Ellers må alle datasettene være georeferert før de blir lagt inn i programmene. Ingen av programmene greier å legge alt i samme lokale koordinatsystem når de blir lagt inn i programmene. Det må gjøres manuelt å legge inn sammenbindingspunkter mellom datasettene. I Reality Capture blir de ulike datasettene lagt i hver sin komponent og med det merge sammen til en komponent. Pix4d klarte ikke sette sammen datasett, samt det ble forvrenginger i flymodellene, synes veldig godt i modellen. Dette gjør at det ikke får tak på modellene til bakkebilder.

Testkriteriet 5, hvilken nøyaktighet er det mulig å oppnå: Agisoft Metashape fikk man skalert modellene for testområdene riktig og fikk en nøyaktighet som var innenfor kravene som PBE ønsket. Mens Reality Capture klarte ikke skaleringen i programmet. Det ble avviket på 15-16 meter i enkelte deler av modellen. Modellene var fin visuelt, men ikke riktig skalering og georeferering. Løsningen blir å eksportere modellene til et annet program å skalere og georeferer, f.eks. FME. Siden det ikke ble noen ferdig vanntett modell i Pix4d pga. forvrenginger. Ble det heller ikke sjekket nøyaktigheten i modellene for de det var veldig vanskelig å måle

6.KONKLUSJON:

I problemstillingen stilte jeg spørsmålet om hvor godt egnet er fotogrammetri til å lage tette 3D-objekter for en bymodell? I min undersøkelse kommer det fram at det er fult mulig å lage 3D – objekter med fotogrammetri for bymodeller, relativet rask og med et godt resultat. Samtidig var det varierende resultat ut fra hvilket datasett som ble brukt til å lage 3D-objektet. Det programmet som skiltet seg ut på kvalitet blant de ulike typer datasett var Reality Capture. Det programmet fikk jevnt over et godt resultat. Reality Capture lager den beste modellen med minst antall bilder, kvalitetsmessig. Det er ikke helt automatisert, det må settes i gang flere prosesser for å skape et 3D-objekt. Likevel vil man kunne få ut en modell som PBE ønsker, pga. at det er mulighet å skifte på defaltverdier slik at modellen blir bra. Det som er minus med dette programmet at georeferering og skalering må skje i et annet program. Selv om det ble testet med over 20 bakkepunkter ble ikke nøyaktigheten i modellen bra. Modellene fra Reality Capture har best kvalitet og lager den mest vanntette soliden, noe som er viktig for PBE.

Agisoft Metashape gir et godt resultat på kvalitet, men trenger flere bilder for å lage en like god modell som Reality Capture. Dette programmet er heller ikke helt automatisert, må igjennom flere prosesser for å få laget et 3D objekt. Fordel med dette er at det går ann å skifte default verdier slik at modellen får en bedre kvalitet. Den største fordelen er at det er muligheter for georeferering og skalering som kommer innenfor et aksepterbart nøyaktighetsmål. Det er et godt alternativ for Reality Capture, men programmet trenger større datasett enn Reality Capture for å få et like godt resultat.

I pix4d fikk jeg problemer med å legge inn flybilder. Flybildene ble forvrengt og det ble mer liggende hus enn stående. Dette har noe med høyden til objektet. Prøvde å prosessere en modell med bare bakkebilder og det ga et helt annet resultat. Her økte kvalitet på modell betraktelig og kunne fått et vanntett solid om det ble fotografert med drone på takene. Pix4d er et automatisert program, er ikke muligheter å skifte på defaltverdier, kun velge hvilke type modell man ønsket. Det er mulighet for georeferering om det blir lagt inn en fil med bakkepunkter som blir plassert manuelt i modellen.

I mine studier av prosessen 3D – rekonstruksjon med fotogrammetri har jeg fått innblikk i hvordan fotogrammetri er egnet til å lage 3D objekter. Utfordringer har i bunn og grunn dreid

seg om å finne balansen mellom antall bilder og hvilke parametere som gir det beste resultatet i de ulike programmene. Hvilket bildemateriale trengs for å lage den beste modellen.

Det kommer tydelig fram i mine tester at det er nærfotogrammetri som gir det beste resultatet i alle programmene og at det ikke trengs mer enn et speilrefleks kamera for at modellene skal bli bra. Det som man må være klar over er kravene som stilles til billedsekvensene for at en rekonstruksjon skal være gjennomførbart. Spesielt vinkler og overlapping mellom bilder. Dette er noe som er avgjørende i algoritmene for programmene. Blir det for dårlig overlapp eller for stor vinkel greier ikke algoritmene i programmene å matche bildene.

Det ble testet en lasersky fra sporveien som har en punkttetthet på 20 000 punkt per kvm. Noe som ga et veldig bra resultat. Det å kunne bruke laserskyer som dekker hele objektet er noe som bør testes videre. Det ga gode kvalitet og nøyaktighet på resultatet. Det ville ha gitt ett bedre resultat enn med ren fotogrammetri.

Alle programmene som ble testet innehar nødvendig funksjonalitet for å gjennomføre målinger i modellene. Men det er forskjellige måter dette ble gjort på. Begge metodene er å måle direkte i modellen, men den ene så må man sette start og slutt punkt. Mens den andre ikke trenger det. Man trykker der det ønskes og start og der man ønsker og slutte så får man opp avstanden mellom «trykkene».

Ingen av de programmene som jeg har testet innehar alle kvalitetene PBE ønsker for å kunne lage 3D objekter som skal erstatt hvordan solids blir laget i dag. Selv om Reality Capture lager den modellen som gir best kvalitet og utnytter dataen best med å lage et vanntett solid, så er programmet dårlig på georeferering og skalering. Selv om jeg ikke fant det ideelle programmet som dekker alle behov, vil det komme med årene siden programmet er utvikling.

Til slutt vil jeg bare understreke at jeg mener alle programmene vil gi et resultat PBE kan bruke til å lage 3D objekter for printing så lenge det er riktig bildematerialet. Også med de funksjonalitetene de har til nå, selv om det trengs støtte program for georeferering og skalering.

Oppgaven min har gitt meg en veldig god innsikt i hva som trengs av data og type programmer for å lage 3D objekt som er helt vanntett. Og at dette er i sterk utvikling enda.

7.VEIEN VIDERE:

Denne oppgaven har gitt meg en innsikt i prosessen som må gjennomføres for å få til en 3D – reduksjon. Har gjort meg noen tanker på hva som bør utvikles videre i framtiden for å kunne få et enda bedre resultat.

I Agisoft Metashape har vi fått testet Pased based metode for bildematching. Pix4d bruker Bundle adjustment metode. Selv om jeg ikke har funnet ut konkret dokumentasjon på det siste programmet jeg testet, tyder det på at den har noe av dem samme algoritmen som de to andre. Dette kan tyde på at jeg har testet samme algoritme opp imot hverandre, men med forskjellig innpakning bare. Det som kunne være spennende å se på er mer av funksjonene til Reality Capture for i forumene heveds det at det går ann å georeferere og skalere modellen. Den måten som ble testet funket ikke, selv om punktnøyaktigheten var bra. Løsningen er her å eksportere modellen og georeferere og skalere i et annet program. Siden visuelt og på alle andre måter var modellen bra og klar. Det hadde også vært spennende å kunne testet ut mer bruk av scann i dette programmet. Prøvd drone scanning slik at man hadde fått en god og vanntett modell. Ut fra nettsiden til Reality Capture er programmet fremdeles under utvikling.

Agisoft Metashape gir en mulighet for PBE å få det ønskete resultat med vanntett solid, skalert og georefert modell. I dette programmet er det også mulighet for å legge tekstur på modellen som også kan brukes videre i VR. Her hadde det vært aktuelt å teste ut enda flere funksjonalister med andre type datasett som dronescannet et helt bygg eller dronefoto. Hvor mye bedre hadde resultatet blitt om all data ble tatt med samme kamera, slippe å kombinere flere typer datasett som ble gjort i dette forsøket.

På nettsidene til Pix4d så er det et program som er laget for drone fotografering, det var tenkt til det. Noe som blir vist i oppgaven har skapt problemer med flybilder. Bakkebilder blir veldig finne. Så det å kunne laget en modell med bilder tatt nærmere objektet gjør at resultatet av et vanntett solid hadde blitt bedre.

Det å kunne kombinere de ulike programmene hadde gjort at resultatet hadde blitt best. Det hadde blitt et “drømme program”. Da hadde jeg ønsket automatiseringen til pix4d, modell visualiseringen og tekstureringen til Reality Capture. Fra Agisoft Metashape hadde man ønsket georefereringen og skaleringen og nøyaktigheten det programmet gir.

REFERANSELISTE:

[Oslo kommune, 2019] Oslo Kommune Plan - og bygningsetaten, (2019) *3D-pilot Hovinbyen, kravspesifikasjoner.*

[Szeliski, 2010] Szeliski.R,. (2010). *Computer Vision: algorithms and Applications*, URL: http://szeliski.org/Book/drafts/SzeliskiBook_20100903_draft.pdf

[Lowe, 2004]. Lowe David G., (2004). *Distinative Image Features from Scale – Invariant Keypoints.* URL: https://www.cs.ubc.ca/~lowe/papers/ijcv04.pdf

[Luhmann et al., 2019] Lugman T., Robson S., Kyle S. and Beohm J. (2019). *Close –Range photogrammetry and 3D imaging.*

[Furukawa et al. 2008] Furukawa Y. and Ponce J. (2008) *Accurate, Dense, and Robust Multi-View Stereopsis.* URL: https://www.di.ens.fr/willow/pdfs/cvpr07a.pdf

[Kim et al. 2005] Kim J., Kolmogorov V.,Zabi R. (2005). *Visual Correspondence Using Energy Minimization and Mutual Information,* URL: http://www.cs.cornell.edu/~rdz/Papers/KKZ-ICCV03.pdf

[Gehrke et al.2008] Gehrke S., Morin K., Downey M., Boehrer N., Fuchs T. (2008). *Semi – global matching an alternative to lidar for DSM generation?* URL: https://pdfs.semanticscholar.org/5b36/5279c5ac32a3dc158b57a37f6827011b1be4.pdf

[Hirschmuller ,2005] Hirschmuller H. (2005). *Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutal Information,* URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.88.8897&rep=rep1&type=pdf

[Dick, 2020] Dick Ø.B., Mæhlum L. (2020) *Fotogrammetri,* URL: https://snl.no/fotogrammetri (Besøkt 18.04.2020)

[Reality capture support, 2019] Reality Capture Support (2019). *Taking pictures for Photogrammetry* URL: https://support.capturingreality.com/hc/en-us/articles/115001528211-Taking-pictures-for-photogrammetry (besøkt 01.04.2020)

[Photomodler] PHOTOMODELER, (u.d.). *FACTORS AFFECTING ACCURACY IN PHOTOGRAMMETRY.* URL: WWW.PHOTOMODELER.COM/KB/ENTRY/63/

[kartverket, 2014] Kartverket, Geovekst, (2014). *Felles KartdataBase (FKB),* URL: https://kartkatalog.geonorge.no/metadata/felles-kartdatabase-(fkb)/0e90ca71-6a02-4036-bd94-f219fe64645f (Besøkt 18.03.2020)

[Agisoft, 2020] Agisoft, (2020) *Agisoft Metahsape User Manual Professional Edirion, Version 1.6* URL: https://www.agisoft.com/pdf/metashape-pro_1_6_en.pdf

[relaity capture, 2015] Reality Capture (2015), *Process lasers & Images Automatically*. URL: https://www.capturingreality.com/Explore-Lasers-Images (Besøkt 18.03.2020)

[pix4d, 2020] pix4d. (2020). *Pix4d mapper* URL: https://www.pix4d.com/product/pix4dmapper-photogrammetry-software (Besøkt 15.03.2020)

[pix4d mapper, 2020] pix4d. (2020).*Proessing steps* URL: https://support.pix4d.com/hc/en-us/articles/115002472186-Processing-steps (Besøkt 15.03.2020)

[Recap,u.d.] Recap, *Autodesk Recap*, (u.d.). URL: https://www.autodesk.com/products/recap/overview (Besøkt 19.03.2020)

[safe software, u.d.] Safe Software, (u.d.). *FME - Feature Manipulation Engine* URL: https://www.safe.com/ (Besøkt 18.03.2020)

[Geodata, u.d.] Geodata, (u.d.). *NT3Dbygger*, URL: https://geodata.no/geodataprodukter/nt-3dbygger (Besøkt 20.03.2020)

[Kartverket, 2018] Kartverket, (2018). *FKB Bygning*, s. 72-73 URL: https://register.geonorge.no/data/documents/Kartleggingsinstrukser_Fotogrammetrisk%20registreringsinstruks%20for%20FKB-Bygning_v2_fotogrammetrisk-fkb-bygning-v461-2018-01-01_.pdf

[kartverket, 2011] Kartverket, (2011), *Hjelpetabell over forventet stedfestningsnøyaktighet for fotogrammetrsk FKB fra flyfotografering med analogt kamera*. URL: https://www.kartverket.no/globalassets/standard/sosi-kap3-produktspesifikasjoner/fkb-4.02-2011-12-01/produktspesifikasjonfkb-samletabell-v402-forventet-noyaktighetskrav-analogt-kamera-2011-12-01.pdf

[kartverket 2020] Kartverket, (2020) *Produktspesifikasjon for ortofoto*, versjon 5.0, side 22. URL: https://register.geonorge.no/data/documents/Produktspesifikasjoner_digitale-ortofoto_v8_produktspesifikasjon-for-ortofoto-v5-0-geodata_.pdf

VEDLEGG:

Vedlegg A: Koordinater til signalerte posisjoner for testområde 1: .. Feil! Bokmerke er ikke definert.

Vedlegg B: Koordinater til signalerte posisjoner for testområde 2: .. Feil! Bokmerke er ikke definert.

Vedlegg C rapport Agisoft Metashape for testområde 1 (del av rapporten): ..... Feil! Bokmerke er ikke definert.

Vedlegg C.1 Flybilder:..... Feil! Bokmerke er ikke definert.

Vedlegg C.2 Flybilder og bakkebilder: ..... Feil! Bokmerke er ikke definert.

Vedlegg D rapport Agisoft Metashape for Testområde 2: .. Feil! Bokmerke er ikke definert.

Vedlegg D.1 Flybilder: ..... Feil! Bokmerke er ikke definert.

Vedlegg D.2 Flybilder og bakkebilder:..... Feil! Bokmerke er ikke definert.

Vedlegg E rapport fra Relaity Capture testområde 1 (del av rapporten): . Feil! Bokmerke er ikke definert.

Vedlegg E.1 flybilder: ..... Feil! Bokmerke er ikke definert.

Vedlegg E.2 flybilder og bakkebilder:..... Feil! Bokmerke er ikke definert.

Vedlegg F rapport fra Relaity Capture testområde 2 (del av rapporten): . Feil! Bokmerke er ikke definert.

Vedlegg F.1 flybilder: ..... Feil! Bokmerke er ikke definert.

Vedlegg F.2 flybilder og bakkebilder:..... Feil! Bokmerke er ikke definert.

Vedlegg A: Koordinater til signalerte posisjoner for testområde 1:

Kilde: https://www.norgeskart.no/?_ga=2.88833243.1339642745.1588680032-1685840598.1588680032#!?project=norgeskart&layers=1002&zoom=6&lat=6781644.19&lon=320924.65

B = bakkepunkt
K = kontrollpunkt for nøyaktighet
L = Linje

Koordinater for posisjonene fra FKB-data:

	X-koordinat	Y-koordinat	Z-koordinat
B1	598360.17	6643718.98	10.83
B2	598260.24	6643715.70	11.62
B3	598252.94	6643811.67	12.63
B4	598353.21	6643816.13	12.29
B5	598337.85	6643732.20	11.83
B6	598319.68	6643732.13	11.12
B7	598309.94	6643731.99	11.06
B8	598275.16	6643781.75	12.18
B9	598283.98	6643781.93	12.22
B10	598274.82	6643797.21	12.18

	X-koordinat	Y-koordinat	Z-koordinat
K1	598355.68	6643718.11	24.98
K2			
K3	598352.48	6643723.92	28.47
K4	598356.86	6643758.60	25.01
K5	598345.51	6643758.18	24.73
K6	598348.26	6643808.66	27.39
K7	598270.380	6643800.360	26.37
K8	598270.820	6643729.390	26.49

	Avstand (m)
L1	9.01
L2	34.47
L3	1.61
L4	3.56

Vedlegg B: Koordinater til signalerte posisjoner for testområde 2:

B = bakkepunkt
K = kontrollpunkt for nøyaktighet
L = Linje

Koordinater for posisjonene fra FKB-data:

	X-koordinat	Y-koordinat	Z-koordinat
B1	598572.01	6643374.93	7.64
B2	598559.07	6643379.59	7.78
B3	598551.73	6643389.17	7.89
B4	598542.40	6643455.30	8.09
B5	598550.29	6643530.96	9.07

B6	598548.92	6643537.84	9.29
B7	598640.15	6643421.19	8.62
B8	598372.22	6643598.29	8.43
B9	598382.60	6643606.57	8.49

	X-koordinat	Y- koordinat	Z-koordinat
K1	598575.9	6643375.0	27.7799765
K2	598565.6	6643388.40	27.7799765
K3	598558.32	6643522.44	32.0499765
K4	598575.64	6643499.78	32.0499765
K5	598636.13	6643420.52	32.2699765

	Avstand (m)
L1	41.32
L2	17.43
L3	16.71
L4	25.88

Vedlegg C rapport Agisoft Metashape for testområde 1 (del av rapporten):

Merk: det har blitt laget tre prosesser fram til dense cloud før de ble satt sammen. Med det får vi ikke helt stemmende bilde antall fra den kombinerte fila, siden det ble lagt inn etter prosessen.

Vedlegg C.1 Flybilder:

Survey Data

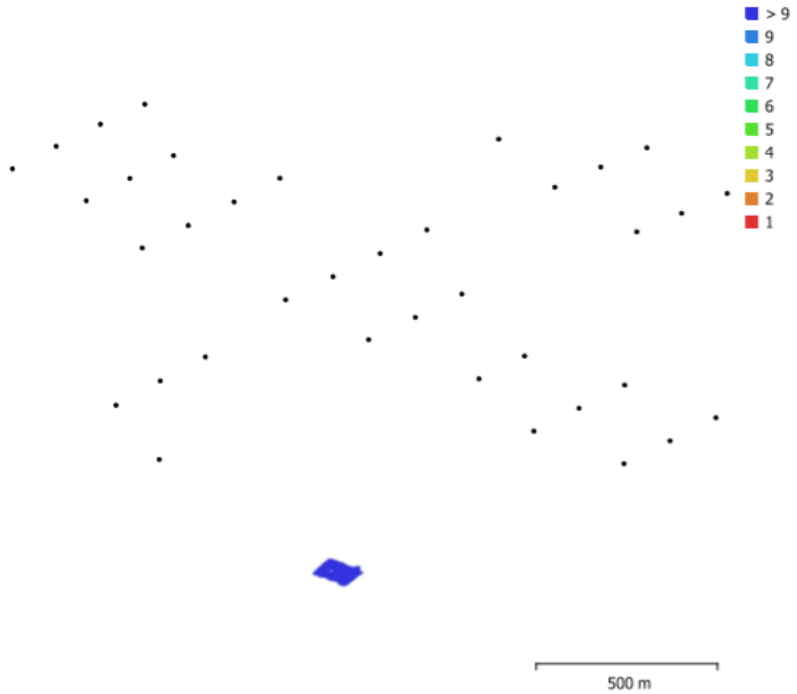


Fig. 1. Camera locations and image overlap.

Number of images:	37	Camera stations:	37
Flying altitude:	1.16 km	Tie points:	47,735
Ground resolution:	5.41 cm/pix	Projections:	103,660
Coverage area:	6.02e+03 m²	Reprojection error:	0.339 pix

Ground Control Points

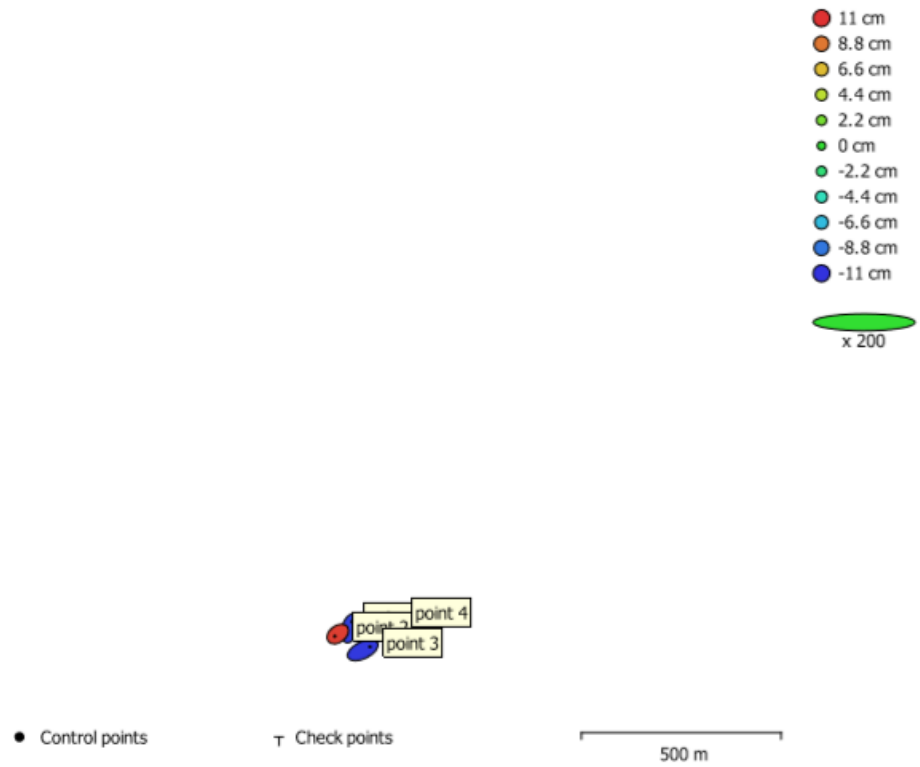


Fig. 4. GCP locations and error estimates.
Z error is represented by ellipse color. X,Y errors are represented by ellipse shape.
Estimated GCP locations are marked with a dot or crossing.

Label	X error (cm)	Y error (cm)	Z error (cm)	Total (cm)	Image (pix)
point 1	4.70873	16.1287	-10.5451	19.837	9.749 (14)
point 2	-7.44218	-5.65904	10.591	14.1273	5.898 (18)
point 3	18.2677	8.827	-10.8023	22.9851	10.590 (17)
point 4	-15.5343	-19.2966	10.7565	27.0069	11.239 (18)
Total	**12.7728**	**13.6238**	**10.6743**	**21.5103**	**9.571**

Table 5. Control points.
X - Easting, Y - Northing, Z - Altitude.

Tetthet:

Resolution: 10.8 cm/pix

Point density: 85.3 points/m²

Processing parameters	
Generel	
Cameras	37
Aligned cameras	37
Markers	4
Coordinate system	ETRS89/UTM zone 32N (EPSG: 25832)
Rotation angles	Yaw, Pitch, Roll
Point cloud	
points	47,735 of 52,162
RMS reprojection error	0.0691409
Max reprojection error	0.216441
Mean key point size	3.53739 pix
Point colors	3 bands, unit 8
Key points	No
Average tie point multiplicity	2.2273
Dense point cloud	
points	3,280,106
Points color	3 bands, unit 8
Model	
Faces	440,278
Vertices	221,940
Vertex colors	3 bands, unit 8

Vedlegg C.2 Flybilder og bakkebilder:

Survey Data

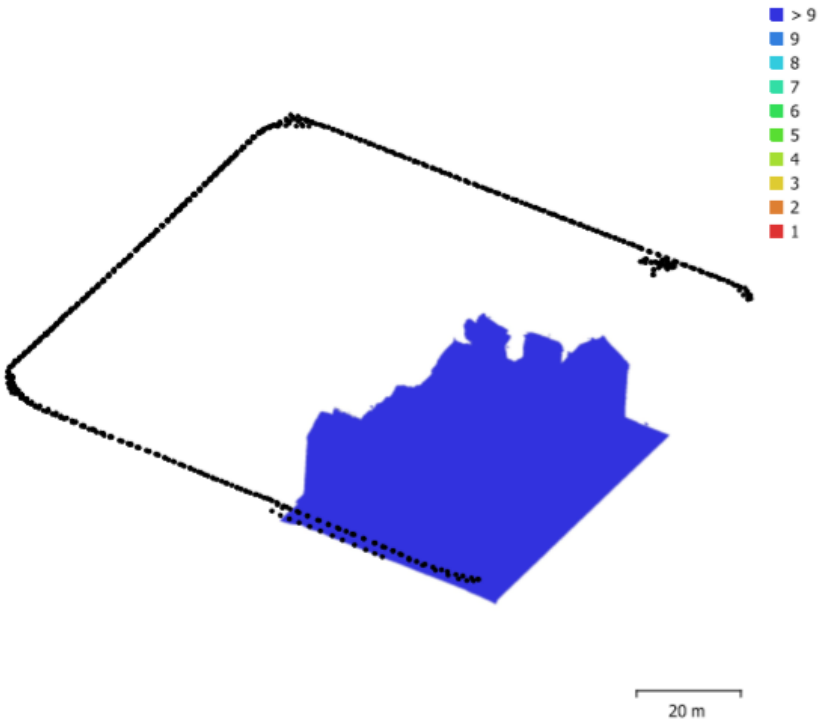


Fig. 1. Camera locations and image overlap.

Number of images:	961	Camera stations:	616
Flying altitude:	8.94 m	Tie points:	507,716
Ground resolution:	2.98 cm/pix	Projections:	1,476,232
Coverage area:	2.29e+03 m²	Reprojection error:	1.09 pix

Merk: figur for bakkebilder I gaten.

Survey Data

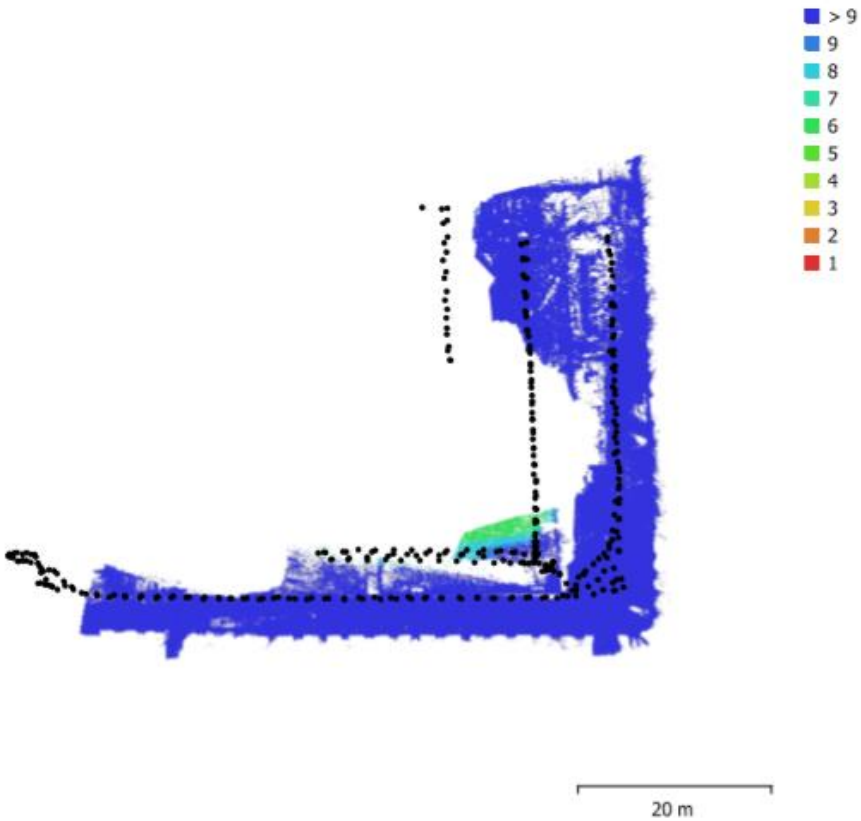


Fig. 1. Camera locations and image overlap.

Number of images:	345	Camera stations:	345
Flying altitude:	8.96 m	Tie points:	335,700
Ground resolution:	3.02 mm/pix	Projections:	901,582
Coverage area:	799 m²	Reprojection error:	1.07 pix

Merk: figur fra bakkebilder I bakgården.

Ground Control Points

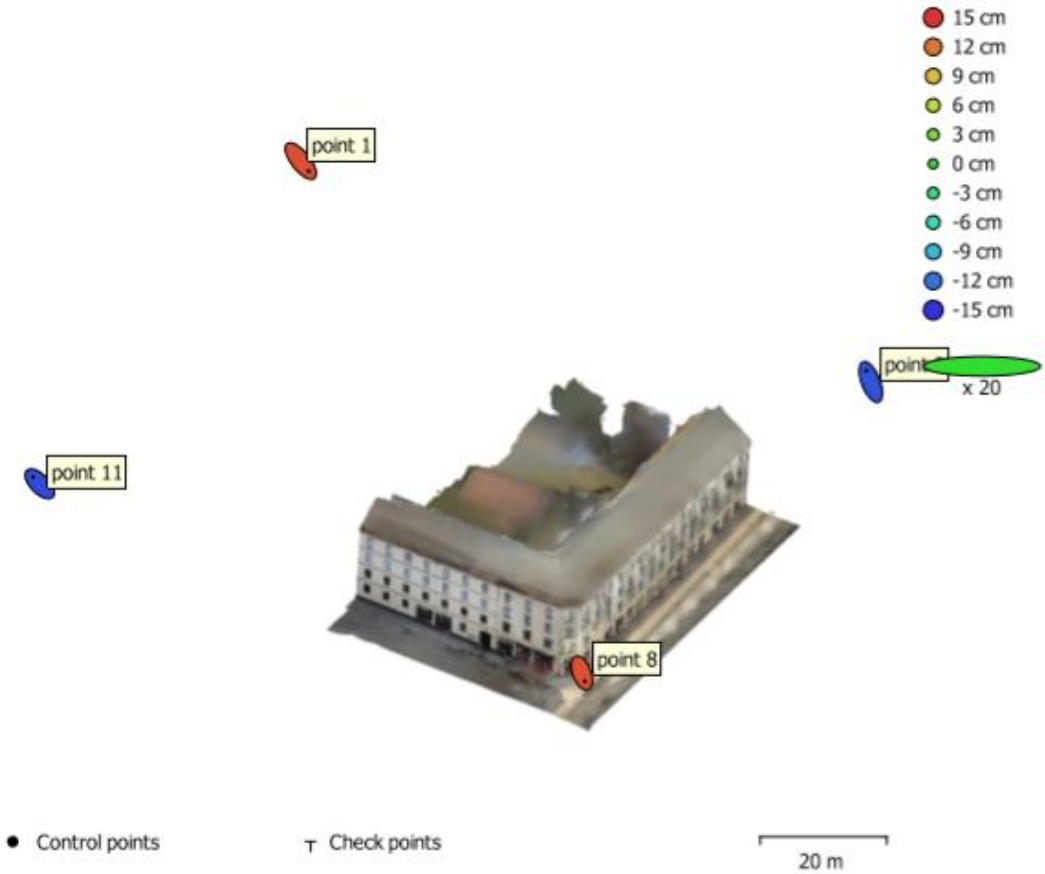


Fig. 4. GCP locations and error estimates.
Z error is represented by ellipse color. X,Y errors are represented by ellipse shape.
Estimated GCP locations are marked with a dot or crossing.

Label	X error (cm)	Y error (cm)	Z error (cm)	Total (cm)	Image (pix)
point 1	12.6609	-15.8881	13.8002	24.5597	4.938 (2)
point 8	4.86934	-12.9676	13.6751	19.4648	2.343 (7)
point 9	-7.17209	18.1845	-13.5706	23.7966	0.453 (11)
point 11	-10.3581	10.6712	-13.9048	20.3595	0.221 (23)
Total	**9.2566**	**14.7066**	**13.7383**	**22.152**	**1.451**

Table 5. Control points.
X - Easting, Y - Northing, Z - Altitude.

Tetthet:

Resolution: 7.91 mm/pix

Point density: 1.6 points/cm²

Processing parameters	
Generel	
Cameras	1030
Aligned cameras	1030
Markers	4
Coordinate system	ETRS89/ UTM zone 32N (EPSG: 25832)
Rotation angles	Yaw, Pitch, Roll
Point cloud	
points	507,716 of 1,088,209
RMS reprojection error	0.22217
Max reprojection error	0.700599
Mean key point size	4.57679 pix
Point colors	3 bands, unit 8
Key points	No
Average tie point multiplicity	3.16102
Dense point cloud	
points	292,121,777
Points color	3 bands unit 8
Model	
Faces	20,843,723
Vertices	10,455,808
Vertex colors	3 bands, unit 8

Vedlegg D rapport Agisoft Metashape for Testområde 2:

Vedlegg D.1 Flybilder:

Survey Data

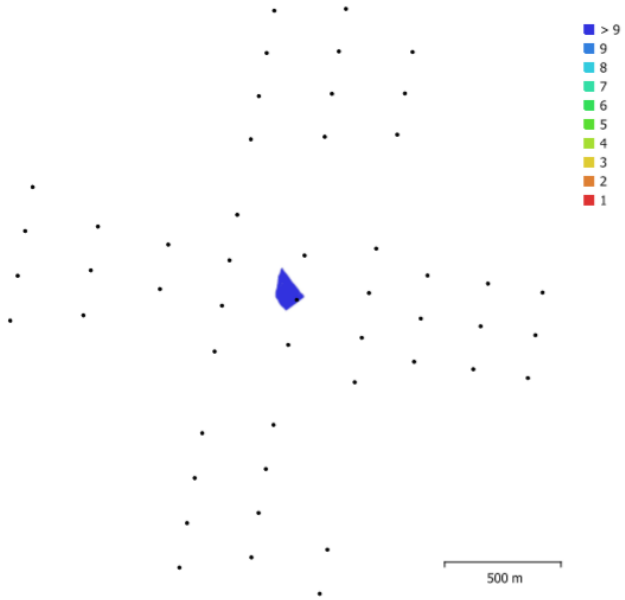


Fig. 1. Camera locations and image overlap.

Number of images:	50	Camera stations:	50
Flying altitude:	1.18 km	Tie points:	59,048
Ground resolution:	5.41 cm/pix	Projections:	132,687
Coverage area:	0.0287 km²	Reprojection error:	0.379 pix

Ground Control Points

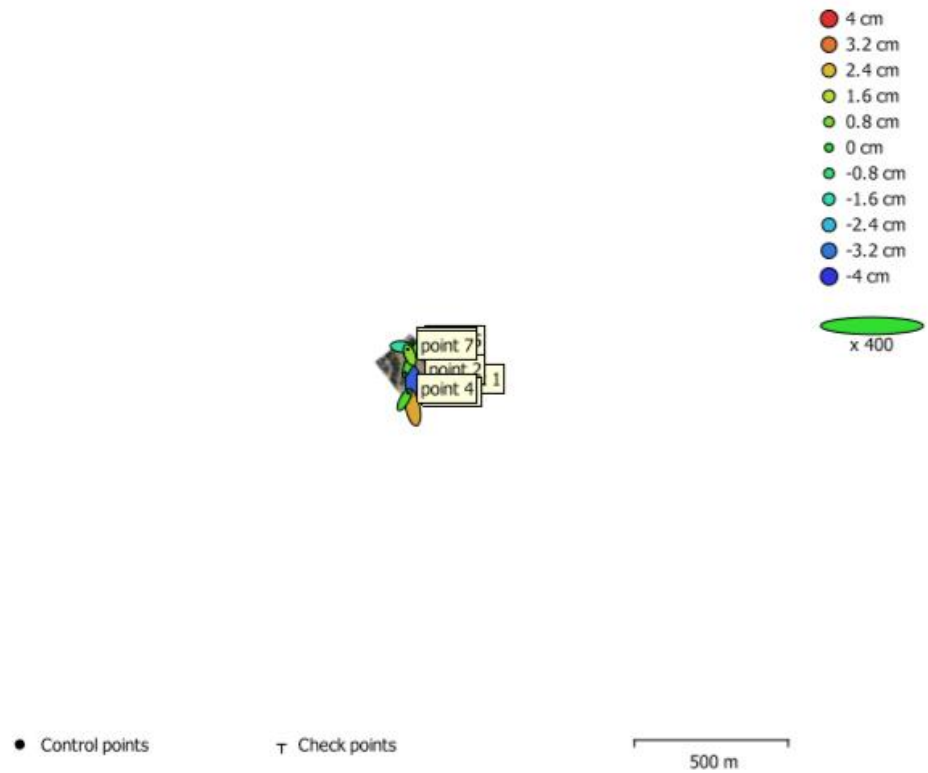


Fig. 4. GCP locations and error estimates.
Z error is represented by ellipse color. X,Y errors are represented by ellipse shape.
Estimated GCP locations are marked with a dot or crossing.

Label	X error (cm)	Y error (cm)	Z error (cm)	Total (cm)	Image (pix)
point 1	-0.51419	3.9146	0.653478	4.00194	1.410 (17)
point 2	1.66019	-19.8107	-3.50594	20.1869	5.080 (23)
point 3	-3.82301	18.5824	2.63906	19.1543	4.614 (26)
point 4	6.07976	10.9436	0.277545	12.5221	3.450 (22)
point 5	-9.18357	-22.0594	0.201038	23.8955	6.362 (22)
point 6	9.25965	-0.692248	-1.28144	9.37349	2.507 (21)
point 7	-3.49007	9.12048	1.01613	9.81816	2.416 (25)
Total	**5.81696**	**14.3583**	**1.79186**	**15.5952**	**4.091**

Table 5. Control points.
X - Easting, Y - Northing, Z - Altitude.

Tetthet:

Resolution: 10.8 cm/pix

Point density: 85.5 points/m²

Processing parameters	
Generel	
Cameras	50
Aligned cameras	50
Markers	7
Coordinate system	ETRS89/ UTM zone 32N (EPSG: 25832)
Rotation angles	Yaw, Pitch, Roll
Point cloud	
points	59,048 of 64,642
RMS reprojection error	0.0736425
Max reprojection error	0.230708
Mean key point size	3.58768
Point colors	3 bands unit 8
Key points	No
Average tie point multiplicity	2.3191
Dense point cloud	
points	6,182,774
Points color	3 bands, unit 8
Model	
Faces	1,236,495
Vertices	662,113
Vertex colors	3 bands, unit 8

Vedlegg D.2 Flybilder og bakkebilder:

Survey Data

Number of images:	526	Camera stations:	329
		Tie points:	187,787
		Projections:	486,987
		Reprojection error:	0.913 pix

Ground Control Points

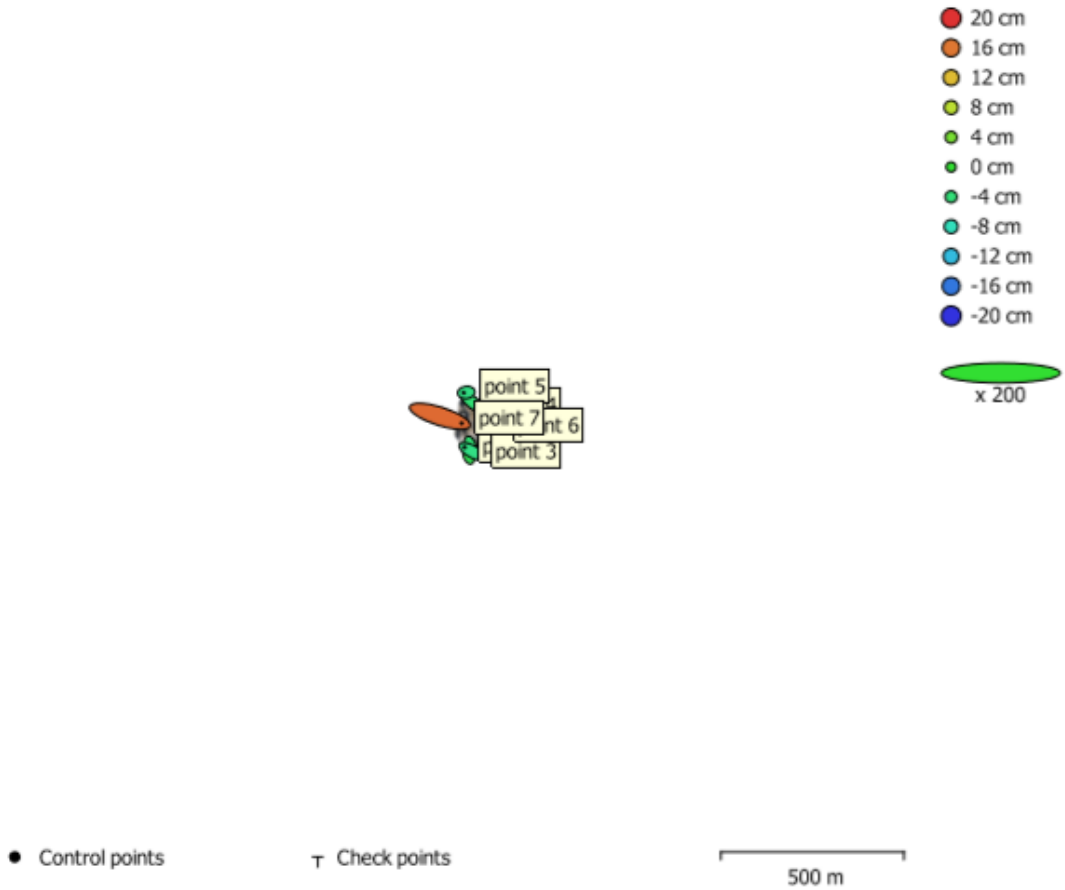


Fig. 4. GCP locations and error estimates.
Z error is represented by ellipse color. X,Y errors are represented by ellipse shape.
Estimated GCP locations are marked with a dot or crossing.

Label	X error (cm)	Y error (cm)	Z error (cm)	Total (cm)	Image (pix)
point 1	4.25261	-14.8218	-3.7638	15.8725	4.008 (23)
point 2	-3.10596	10.8851	-1.93883	11.4844	2.784 (26)
point 3	-21.5004	11.8486	-4.76334	25.0069	6.699 (22)
point 4	-22.559	19.0362	-3.76371	29.7565	5.758 (31)
point 5	-7.2022	-1.03109	-4.70306	8.66334	1.951 (30)
point 6	-9.04057	-6.91822	2.18872	11.5924	3.364 (21)
point 7	59.1822	-19.0032	16.7474	64.3749	21.364 (17)
Total	**25.7323**	**13.3683**	**7.19289**	**29.8764**	**7.932**

Table 5. Control points.
X - Easting, Y - Northing, Z - Altitude.

Tetthet:

Resolution: 2.1 cm/pix

Point density: 0.227 points/cm²

Processing parameters	
Generel	
Cameras	576
Aligned cameras	576
Markers	7
Coordinate system	ETRS89/ UTM zone 32N (EPSG: 25832)
Rotation angles	Yaw, Pitch, Roll
Point cloud	
points	13,393 of 439,723
RMS reprojection error	0.0779611
Max reprojection error	0.240614
Mean key point size	3.66855
Point colors	3 bands, unit 8
Key points	No
Average tie point multiplicity	2.2417
Dense point cloud	
points	82,804,242
Points color	3 bands, unit 8
Model	
Faces	8,148,181
Vertices	4,116,281
Vertex colors	3 bands, unit 8

Vedlegg E rapport fra Relaity Capture testområde 1 (del av rapporten):

Vedlegg E.1 flybilder:

Processering parametere:	
Seleted component	
Camera count	37
Point count	85 871
Control point count used	4
Constraint count used	0
Total projections	196 366
Averge track legth	2.286721
Max error (pixel)	1.999858
Median error (pixel)	0.562355
Mean error (pixel)	0.676672
Geo - referenced	Yes
metric	Yes
Modell:	
Trinagle count	2.9M
Vertex count	1.5M

	B1	**B2**	**B3**	**B4**
type	Ground control	Ground control	Ground control	Ground control
weight	10	10	10	10
Coordinate system	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N
X (m)	598 360.17	598353.21	598252.94	598260.24
Y (m)	6643718.98	6643816.13	6643811.67	6643715.70
Z (m)	10.83	12.29	12.63	11.62
Error X (m)	-0.02302	-0.002664	0.012786	0.693146
Error Y (m)	-0.011506	0.038539	-0.009637	-0.797755
Error Z (m)	0 011506	-0.013697	0.010400	-3.642409
totalt	0.031109	0.040997	0.019096	3.792708
maximal projection error	0.693922	0.693150	0.753992	27.427099

Vedlegg E.2 flybilder og bakkebilder:

Processering parametere:	
Seleted component	
Camera count	997
Point count	1 511 034
Control point count used	31
Constraint count used	0
Total projections	4 433 202
Averge track legth	2.937195
Max error (pixel)	1.999997
Median error (pixel)	0.468709
Mean error (pixel)	0.590332
Geo - referenced	Yes
metric	Yes
Modell:	
Trinagle count	167.6M
Vertex count	84.0M

	B1	**B2**	**B3**	**B4**
type	Ground control	Ground control	Ground control	Ground control
weight	10	10	10	10
Coordinate system	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N
X (m)	598 360.17	598353.21	598252.94	598260.24
Y (m)	6643718.98	6643816.13	6643811.67	6643715.70
Z (m)	10.83	12.29	12.63	11.62
Error X (m)	0.001137	0.005353	-0.025026	0.017928
Error Y (m)	0.019808	-0.009271	0.037600	-0.008965
Error Z (m)	-0.029986	-0.043063	0.035585	-0.020453
totalt	0.035959	0.044375	0.057511	0.028641
maximal projection error	1.145836	1.114200	2.307539	1.352747

	B5	**B6**	**B7**	**B8**
type	Ground control	Ground control	Ground control	Ground control
weight	10	10	10	10
Coordinate system	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N
X (m)	598337.850000	598319.680000	598309.940000	598275.160000
Y (m)	6643732.200000	6643732.130000	6643731.990000	6643781.750000
Z (m)	11.830000	11.120000	11.060000	12.180000
Error X (m)	-0.185168	-0.086977	1.308326	0.097309
Error Y (m)	0.064380	-0.039385	0.772976	0.808916
Error Z (m)	0.385859	-0.171552	5.113846	0.190851
totalt	0.432829	0.196345	5.334974	0.837027

maximal projection error	2.129239	2.905344	9.820313	15.050974

	B9	**B10**
type	Ground control	Ground control
weight	10	10
Coordinate system	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N
X (m)	598283.980000	598274.820000
Y (m)	6643781.930000	6643797.210000
Z (m)	12.220000	12.180000
Error X (m)	-0.391276	0.030478
Error Y (m)	0.599068	-0.070709
Error Z (m)	0.599068	0.036877
totalt	3.292961	0.085393
maximal projection error	17.119829	3.946280

Vedlegg F rapport fra Relaity Capture testområde 2 (del av rapporten):

Vedlegg F.1 flybilder:

Processering parametere:	
Seleted component	
Camera count	49
Point count	107 133
Control point count used	9
Constraint count used	0
Total projections	241 011
Averge track legth	2.249671
Max error (pixel)	1.999769
Median error (pixel)	0.539753
Mean error (pixel)	0.651742
Geo - referenced	Yes
metric	Yes
Modell:	
Trinagle count	3.9M
Vertex count	1.9M

	B1	**B2**	**B3**	**B4**
type	Ground control	Ground control	Ground control	Ground control
weight	10	10	10	10
Coordinate system	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N
X (m)	598572.01	598559.100000	598551.730000	598542.400000
Y (m)	6643374.930000	6643379.680000	6643389.200000	6643455.300000
Z (m)	7.640000	7.780000	7.890000	8.090000
Error X (m)	-0.007885	0.054575	-0.048287	-0.000080
Error Y (m)	-0.005961	0.034330	-0.033160	-0.000052
Error Z (m)	-0.060162	0.037443	-0.021878	0.000138
totalt	0.060969	0.074574	0.062545	0.000168
maximal projection error	1.284718	2.275233	2.101719	0.463393

	B5	**B6**	**B7**	**B8**
type	Ground control	Ground control	Ground control	Ground control
weight	10	10	10	10
Coordinate system	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N
X (m)	598550.290000	598548.570000	598640.150000	598372.220000
Y (m)	6643530.96	6643538.400000	6643421.190000	6643598.290000
Z (m)	9.070000	9.310000	8.620000	8.430000
Error X (m)	-0.032933	0.046077	-0.000472	-0.001405

Error Y (m)	0.009825	0.020987	0.000240	0.000338
Error Z (m)	-0.006029	-0.040793	-0.002658	0.002081
totalt	-0.006029	0.065032	0.002711	0.002534
maximal projection error	3.692099	0.774445	0.059698	0.030006

	B9
type	Ground control
weight	10
Coordinate system	Epsg: 25832 – ETRS89/UTM zone 32N
X (m)	598382.600000
Y (m)	6643606.570000
Z (m)	8.490000
Error X (m)	0.000040
Error Y (m)	-0.000091
Error Z (m)	-0.000005
totalt	0.000100
maximal projection error	0.037961

Vedlegg F.2 flybilder og bakkebilder:

Processering parametere:	
Seleted component	
Camera count	571
Point count	641 618
Control point count used	36
Constraint count used	0
Total projections	1 844 818
Averge track legth	2.872140
Max error (pixel)	1.999999
Median error (pixel)	0.410524
Mean error (pixel)	0.540188
Geo - referenced	Yes
metric	Yes
Modell:	
Trinagle count	32.8M
Vertex count	16.4M

	B1	**B2**	**B3**	**B4**
type	Ground control	Ground control	Ground control	Ground control
weight	10	10	10	10
Coordinate system	Epsg: 25832 – ETRS89/UTM	Epsg: 25832 – ETRS89/UTM	Epsg: 25832 – ETRS89/UTM	Epsg: 25832 – ETRS89/UTM

	zone 32N	zone 32N	zone 32N	zone 32N
X (m)	598572.01	598559.100000	598551.730000	598542.400000
Y (m)	6643374.930000	6643379.680000	6643389.200000	6643455.300000
Z (m)	7.640000	7.780000	7.890000	8.090000
Error X (m)	-0.005794	0.035970	0.057274	-0.17634
Error Y (m)	0.097838	0.161892	0.012666	-0.964223
Error Z (m)	0.022816	-0.045263	-0.083381	0.155601
totalt	0.100657	0.171951	0.101956	0.992762
maximal projection error	3.144117	2.603708	4.657136	21.814642

	B5	**B6**	**B7**	**B8**
type	Ground control	Ground control	Ground control	Ground control
weight	10	10	10	10
Coordinate system	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N
X (m)	598550.290000	598548.570000	598640.150000	598372.220000
Y (m)	6643530.96	6643538.400000	6643421.190000	6643598.290000
Z (m)	9.070000	9.310000	8.620000	8.430000
Error X (m)	0.054312	-0.117862	-0.088787	0.290737
Error Y (m)	0.023307	0.168060	-0.073684	0.321901
Error Z (m)	-0.009444	0.003727	-0.113337	0.321901
totalt	0.059868	0.205362	0.161757	0.701921
maximal projection error	4.131413	5.874272	3.146979	7.112076

	B9	**B10**	**B11**	**B12**
type	Ground control	Ground control	Ground control	Ground control
weight	10	10	10	10
Coordinate system	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N	Epsg: 25832 – ETRS89/UTM zone 32N
X (m)	598382.600000	598559.090000	598561.050000	598597.200000
Y (m)	6643606.570000	6643540.920000	6643538.600000	6643499.100000
Z (m)	8.490000	9.190000	9.170000	8.840000
Error X (m)	0.086892	-0.022986	0.194710	-0.419951
Error Y (m)	-0.066329	-0.038578	-0.047868	1.641484
Error Z (m)	0.040181	-0.074830	-0.135970	-0.173692
totalt	0.116494	0.087277	0.242310	1.703707
maximal projection error	3.686573	0.961842	6.167354	28.111959

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

Postboks 5003
NO-1432 Ås
Norway