

Magnus Renaa Kjørseng

Bacheloroppgave

3D-skanner for bruk i stabilisering av håndleddsbrudd

Juli 2021

NTNU

Norges teknisk-naturvitenskapelige universitet.
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for IKT og realfag

Bacheloroppgave

2021



Magnus Renaa Kjørseng

3D-skanner for bruk i stabilisering av håndleddsbrudd

Bacheloroppgave
Juli 2021

NTNU

Norges teknisk-naturvitenskapelige universitet.
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for IKT og realfag



Kunnskap for en bedre verden

Abstract

Denne rapporten tar for seg målet å forbedre dagens gipsingsprosess for håndleddsbrudd. Dagens prosess krever store mengder kunnskap hos de som legger gipsen, samt tid og trening for å få et resultat som støtter slik det skal og ikke fører til skader som følge av gipsen.

Forbedringen er sett som et komplett system som foretar en 3D-skann av pasientens underarm, bruker denne til å generere en støtteskinne som så kan tilvirkes ved hjelp av 3D-printing eller lignende teknologier. Denne rapporten tar kun for seg den første delen av denne prosessen.

Ved hjelp av en LiDAR sensor og en rig som holder denne i ønsket posisjon blir innskanningen gjort, denne blir så prosessert noe for å gi en modell.

Målet med oppgaven ble ikke nådd fullstendig, da løsningen ikke gir ut en 3D-modell i et brukbart format, men dette kan forbedres i fremtidige iterasjoner.

Innhold

Abstract	i
Begreper og terminologi	iv
Figurer	vi
Tabeller	vii
1 Innledning	1
1.1 Bakgrunn	1
1.1.1 Tidligere oppgaver	1
1.2 Problemstilling	2
1.3 Mål	2
1.3.1 Problemer ved tidligere oppgaver	3
1.3.2 Forbedringer i denne oppgaven	4
1.4 Rapportens innhold	5
2 Teoretisk grunnlag	6
2.1 Håndleddsbrudd	6
2.2 Fotogrammetri	6
2.3 3D-skanning	8
2.3.1 Punktskyer	8
3 Metode	9
3.1 Sensorvalg	9
3.1.1 Synlig lys kamera	9
3.1.2 3D kamera/dybdekamera	10
3.1.3 LiDAR	11
3.1.4 Håndholdt 3D-skanner	11
3.1.5 Vurderinger for dette prosjektet	11
3.2 Riggen	13
3.2.1 Stykkliste	14
3.3 Prinsipp for skanning	15
3.4 Programvare	15
3.4.1 Pythonbiblioteker	15
3.4.2 Annen programvare	16

4	Resultat	18
4.1	Oppgraderinger til riggen	18
4.2	Programvare	18
4.2.1	Fanging av dybdebilde	19
4.2.2	Konvertering til vektorliste	20
4.2.3	Dybdefiltrering	20
4.2.4	Endring av origo	20
4.2.5	Rotasjon	22
4.2.6	Kompilasjon	22
4.3	Testskann	23
5	Drøfting	26
5.1	Resultater	26
5.1.1	Opfylte mål	26
5.1.2	Mangler og utbedringer	26
5.2	Prosjektet	28
5.2.1	Utvikling	28
5.2.2	Tidsstyring	28
5.2.3	Gruppen	28
6	Konklusjon	29
	Kode	30
	Bibliografi	34
	Vedlegg	35
A	Forprosjektrapport	35
B	Ukesrapporter	47

Begreper og terminologi

C++ Programmeringsspråk

CAD/CAM Computer Aided Design/Computer Aided Manufacturing, data-programmer for tilvirkning av 3D-modeller og prosesser

Dybdekart En datatype som representerer dybde slik et vanlig kamera representerer farger

Fotogrammetri Prosesseringsmetode for å generere 3D-modeller fra 2D-RGB bilder

GUI Graphical User Interface, grafisk brukergrensesnitt. En del av et dataprogram som gjør datafremstilling og interaksjon med programmet enklere gjennom grafisk fremvisning.

IR Infrarød, en usynlig del av det elektromagnetiske spekteret

Kantfinning Matematisk metode for å finne skarpe endringer (kanter) i bilder. Brukes f.eks. for bakgrunnsfjerning.

LiDAR Light Detection and Ranging, en sensortype som benytter seg av en utsendt lysstråle og måler tiden det tar mellom strålen blir sendt ut til den kommer tilbake

Meshing Prosessen å gå fra en punktsky til et 3D-objekt

Nær-synlig lys Et annet begrep på nær-infrarødt lys. Lys like utenfor spekteret som er merkbart for mennesker

Parallax/parallakse Endringen i posisjonen til to stasjonære punkter relativt til hverandre sett av en observatør, som følge av observatørens bevegelse eller endring i posisjon.

Punktsky Representasjon av et objekt som et sett med punkter som ligger på objektets overflate

PyPI Python Package Index, samleplass for biblioteker og pakker til Python

Python Programmeringsspråk

SDK Source Development Kit, en programpakke som gjør det enklere å utvikle nye programmer som samarbeider med eksisterende programmer eller maskinvare

ToF Time of Flight, teknologi for avstandsmåling som tar tiden fra et signal blir sendt ut til det kommer tilbake, for å finne avstanden til et objekt

Wrapper Programvare som fungerer som et oversettelsesledd mellom f.eks. en SDK og et språk SDKen ikke er utviklet for

Åpen kildekode/open source Prinsipp for programvare- og maskinvareutvikling hvor all informasjon om sluttproduktet, som kildekode eller diagrammer for konstruksjon, er offentlig tilgjengelig

Figurer

2.1	Prinsippene bak fotogrammetri. Bilde fra det finske geotekniske forskningscenter (GTK)	7
3.1	Trommelen, kameraene ville her bli plassert i de beige brakkettene rundt trommelen med lysstriper inn i trommelen for å sørge for jevn belysning.	10
3.2	Riggen lagt ut flatt med alle delene synlig bortsett fra kulelagrene.	13
3.3	Intel RealSense Viewer overblikksbilde	16
3.4	De tre undersensorenes datastrømmer visualisert i RealSense Viewer	17
4.1	Innskanningsriggen klar for skanning. Objektet som skal skannes er satt på den røde platen. Riggen er her satt opp for skanning av vertikale objekter.	19
4.2	En illustrasjon av prosessen å endre origo, sett fra siden. . .	21
4.3	Flaske, skyene høyt og lavt i bildet er trolig støy på grunn av flaskens refleksivitet	24
4.4	En eske med synlig innside. Støyen i punktskyen bør være mulig å fjerne ved filtrering senere.	25

Tabeller

3.1	Krav til sensorvalg og sammenligning mellom potensielle sensorer	12
3.2	Stykkliste for innskanningsriggen	14

Kapittel 1

Innledning

Denne rapporten beskriver en videreutvikling av et prosjekt som har foregått ved NTNU i Ålesund over flere år. Prosjektet går ut på å utvikle en ny prosess for stabilisering av håndleddsbrudd ved hjelp av 3D-skanning og -printing. Prosjektet er ledet av Paul Steffen Kleppe ved IHB og Webjørn Rekdalsbakken ved IIR.

1.1 Bakgrunn

Håndleddsbrudd er den vanligste bruddskaden som forekommer i Norge [7], og blant de vanligste på verdensbasis[6]. Behandling mot håndleddsbrudd er i dag en tidkrevende prosess som krever stor erfaring hos den som behandler, enten det skal legges på en gips eller håndleddet behøver operasjon. Målet ved dette prosjektet er å lage en omfattende løsning som erstatter store deler av dagens prosess med legging av gips ved hjelp av 3D-teknologi.

1.1.1 Tidligere oppgaver

Det er flere oppgaver skrevet om dette prosjektet, med varierende fokusområde og suksessgrad. Avhengig av størrelsen på gruppene som har arbeidet med prosjektet har det gjerne blitt segmentert opp.

En av de tidligste oppgavene skrevet av M. Gya og A.D. Thorsen [4] benyttet seg av et bedriftsbesøk for å ordne innskannings-delen og har hovedsakelig fokusert på reduksjon av tilvirkningstiden av det endelige produktet. Resultatene fra Gya og Thorsen vil ikke bli brukt i denne rapporten, men kan med fordel benyttes i tilvirkningsfasen senere i prosjektet, da de nådde gode resultater både med tanke på komfort og reduksjon av materiale og tid for printing.

D. Sjøstad, V.A.J. Alvestad og O.H. Nedrelid [9] og T.A. Hoseth og A.N. Dale [5] tok for seg tilvirkningen av en 3D-skanner ved hjelp av flere synlig-lys kamera og fotogrammetri. Sjøstad et al. tok her for seg tilvirkningen av skanne-

ren mens Hoseth og Dale fokuserte på dataprosesseringen fra denne skanneren. Skanneren utviklet av disse er en tidlig prototype for senere prosjekter, den var lovende i resultater, men var teknisk komplisert både i oppbygging og prosessering. Hoseth og Dale benyttet seg av tredjepartsprogramvare for fotogrammetridelen. Dette gjør løsningen resursmessig litt tyngre på grunn av GUIer og unødvendige funksjoner i programvaren, samt at en må være mer forsiktig med bruken på grunn av lisensavtaler.

Denne rapporten er i stor grad basert på den siste oppgaven skrevet om prosjektet av S. Romunstad [8]. Romunstads rapport er i hovedsak en litteraturstudie om teknologiene nødvendig og er et utmerket sammendrag av de tidligere oppgavene skrevet om prosjektet. Dette er hovedgrunnen til at den blir brukt som basis for denne rapporten. En annen grunn er at Romunstad tilvirket et oppsett, en rigg, som forenklet innskanningsprosessen ved å kun bruke én sensor. Denne riggen er arvet fra Romunstad og videreutviklet her. Romunstad benyttet allikevel i hovedsak tredjepartsprogrammer for tilvirkning av modeller. Dette er som nevnt noe jeg ønsker å komme vekk fra ved dette prosjektet.

1.2 Problemstilling

Ålesund Sykehus har ytret et ønske om å effektivisere prosessen med å stabilisere håndleddsbrudd. Dette skal utføres ved hjelp av en større prosess.

1. Pasient blir vurdert å trenge stabilisering
2. Underarmen og håndleddet til pasienten blir skannet
3. En datamodell blir laget fra skanningen
4. Datamodellen blir automatisk prosessert og en støtteskinne generert
5. Støtteskinnen blir tilvirket på sykehuset ved hjelp av 3D-printing eller liknende teknologi

Denne rapporten tar for seg punkter 2 og 3 i lista over.

1.3 Mål

Målet for denne oppgaven er å skanne inn et objekt og gi en god 3D-modell videre til andre programmer for videre bearbeiding. En godmodell defineres som en modell med lite støy, som inneholder all informasjonen nødvendig for videre automatisk generering av skinne. Modellen skal være av høy nok oppløsning til å kunne videreutvikles. Den skal óg helst være i sanne dimensjoner; altså 1cm målt på det fysiske skann-objektet skal være 1cm i den digitale modellen.

1.3.1 Problemer ved tidligere oppgaver

De tidligere prosjektene har møtt noen hovedproblemer. Disse kan kategoriseres som følgende

- Kompleksitet
- Kostnad
- Kvalitet

Det finnes og mindre problemer, som brukervennlighet og utforming, men disse er ikke vurdert her i stor grad da denne oppgavens fokus er på den rent tekniske innskanningen, ikke sluttbruker.

Kompleksitet

Sjåstad et al. og Hoseth og Dale tilvirket en stor trommel med mange kameraer og flere Raspberry Pi'er. Dette gjorde at skanneren deres både tok opp stor plass og ble teknisk sett kompleks med mange komponenter som måtte arbeide sammen. Dette gir større mulighet for feil, både i den enkelte komponenten og i samkjøringen mellom dem. Det er stort potensiale for feil.

Kostnad

Kostnad har ikke vært et problem med de tidligere oppgavene som de har vært. Romunstad har benyttet noe dyre komponenter men disse var aldri ment for bruk i det ferdige produktet. Prisen på hele løsningen skal være så lav som mulig for å gjøre den attraktiv for sykehus og klinikker å anskaffe. Dersom det kreves enkeltkomponenter med prislapp på mange tusen vil dette gjøre senere adopsjon vanskeligere.

Et annet problem som kan falle under kostnad er lisensavtaler. Sluttproduktet blir mindre brukbart og mindre salgbart dersom det medfølger en kostbar lisensavtale for bruk, eller en løpende lisensavgift over tid. Selv om det finnes ferdige løsninger som har gunstige lisensavtaler vil den beste løsningen trolig være å kun benytte seg av biblioteker og løsninger med åpen kildekode og lisenser som tillater både akademisk og kommersiell bruk.

Kvalitet

Resultatene i form av modell fra skann og den ferdig tilvirkede skinna må være av høy nok kvalitet til å være brukbare. Dersom den skannede modellen er for lite virkelighetsnær vil det ikke være mulig å tilvirke et godt sluttprodukt. Tidligere prosjekter har nådd gode resultater i form av skann, men alle har så langt slitt med enten mye støy eller hull i innskanningen.

1.3.2 Forbedringer i denne oppgaven

Løsningen jeg vil komme med i denne rapporten benytter seg kun av én sensor. Dette trekker ned både kostnaden og kompleksiteten av sluttproduktet, samt at det gjør prosessering av data enklere da det kun er én datakilde.

Denne oppgaven benytter seg også av en annen type sensor enn de brukt tidligere. Denne sensoren har større dybdenøyaktighet og bedre evne til kantfinning, to egenskaper som hever potensialet for høyere kvalitet i sluttproduktet.

I tillegg vil jeg her utvikle et grunnleggende innskanningsprogram som kun benytter biblioteker og produkter med åpen kildekode og lisenser som tillater for fri bruk både i akademia og for kommersielle formål.

1.4 Rapportens innhold

Kapittel 2 - Teoretisk grunnlag

Kapittel 3 - Materialer og metoder

Kapittel 4 - Resultater

Kapittel 5 - Diskusjon

Kapittel 6 - Konklusjon

Kapittel 2

Teoretisk grunnlag

2.1 Håndleddsbrudd

Underarmen består av to bein, ulna og radius. Ulna ligger på utsiden av armen, samme som lillefingeren, mens radius ligger på tommelsiden. Brudd i radius mindre enn 4cm fra enden av beinet ved håndleddet betegnes på fagspråket som distalt radiusbrudd. Distal radiusbrudd er en av de vanligste bruddtypene, både i Norge [7] og i verden generelt [6].

For å legge til rette for god legning av håndleddsbrudd må bruddet først få eventuelle misstillinger korrigert. En misstilling er der delene som skal gro sammen har forskjøvet seg i forhold til hverandre, denne forskyvningen må rettes opp i for å unngå sammenvekst som er enten smertefull eller begrenser bevegeligheten eller livskvaliteten til pasienten.

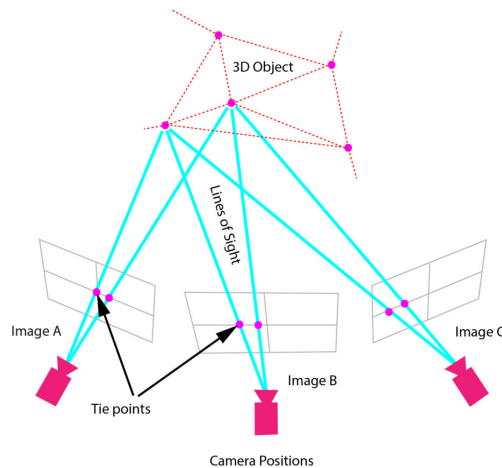
Etter at misstillingen er rettet opp i må leddet legges i riktig stilling og immobiliseres som best mulig. Det finnes flere måter å immobilisere ledd, men hovedsakelig brukes gips, enten kalkgips/pariser gips (eng: plaster of paris) eller glassfibergips. For mindre brudd som i fingre, eller i mer akutte settinger er også spjelker brukt, avstivende element som stålstenger blir lagt parallelt med den skadde delen, og de blir bandasjert sammen. Sluttmålet med immobiliseringen er uansett det samme, bruddet holdes i den optimale stillingen for at legningen skal få et så bra som mulig resultat, og ved å fjerne leddets mulighet for å bevege på seg vil også legningen gå raskere fordi bruddet ikke åpnes opp igjen.

Det finnes problemer med gipsing. Gips puster dårlig, dette gjør at huden under gipsen blir svett og bakterievekst går raskere, noe som kan føre til infeksjoner eller utslett. Trykkskader, varmeskader og andre skademekanismer er også potensielle problemer. [3]

2.2 Fotogrammetri

Fotogrammetri er en teknikk i bildebehandling hvor flere bilder tatt av samme objekt fra forskjellige vinkler blir satt sammen til en 3D-modell av objektet.

Dersom bildene er tatt hyppig nok og objektet av bildene er leselig nok for dataprogrammet kan en 3D-modell bli generert ved hjelp av parallakse, relativ bevegelse og kjennskap til sensoren brukt. Fotogrammetri benytter seg vanligvis av landemerker” (eng. landmarks) for å kunne sette sammen bildene. Et svært forenklet eksempel vil være en hvit sylinder med ett sort merke på seg. Merket på sylindere er i ett bilde i midten av sylindere, men i det neste bildet er det nærmere den venstre kanten av sylindere. En kan med denne informasjonen si at kameraet og sylindere har blitt rotert i forhold til hverandre, sylindere har blitt rotert med klokke sett ovenfra. Hvis en kjenner kameraets parametre, fokusdistanse, synsvinkel, etc., kan man finne ut plasseringen til kameraet relativt til sylindere, både avstand og vinkelforskjell. Disse bildene, med kjente opphavspunkt, kan så kombineres sammen til ett 3D-objekt. Med mange nok bilder vil dette kunne bli en god modell av det originale objektet.



Figur 2.1: Prinsippene bak fotogrammetri. Bilde fra det finske geotekniske forskningssenter (GTK)

Noen problemer har allikevel fotogrammetri, for best resultat krever fotogrammetri at objektet ikke er for reflekterende, da refleksjoner kan gi forvirrende informasjon til sensoren og programvaren. Fotogrammetri har også bedre resultater om objektet ikke er homogent i farge, enten naturlig, som en barberspiral, eller ved at operatøren setter på landemerker manuelt. Hvis objektet er fullstendig ensfarget og har gjentakende eller små størrelsesforskjeller, f.eks. en hvit kaffekopp, vil mange av bildene være umulige for fotogrammetriprogramvaren å plassere i rommet, og derfor vil det ikke være mulig å sy sammen bildene.

Fotogrammetri er altså avhengig av å først finne ut hvor i rommet bildene er tatt fra i forhold til hverandre, for så å bruke informasjonen som er felles mellom bildene for å sette sammen en modell.

2.3 3D-skanning

3D-skanning er en prosess for å generere datamodeller av ting i den fysiske verden. Modellene som blir generert skal ideelt sett ha alle de umiddelbart synlige detaljene til tingen, altså vil den digitale modellen og den fysiske tingen være formlike. Funksjon til ting, som for eksempel hengsler eller andre bevegelige deler, vil normalt ikke bli fanget opp av en skanner og må derfor legges til manuelt senere eller via andre prosesseringsmetoder.

De aller fleste CAD/CAM programmer benytter seg av størrelsesenheter som ikke er rent digitale. Altså de benytter centimeter, tommer, liter, etc.. For at en skanner skal være så effektiv som mulig i det her planlagte bruksområdet bør modellene den gir ut være like store som de faktiske objektene i virkeligheten er. Hvis en kopp blir skannet og koppen er 15cm høy og 8cm i diameter, bør den digitale modellen også ha målene 15x8cm.

Om den skannede modellen ikke har sanne dimensjoner er dette mest sannsynlig på grunn av forvrenging i bildet. Forvrenginger er et kronglete problem å komme over, dette på grunn av at det krever intrikat kunnskap om sensorens geometri. Det er allikevel ikke uoverkommelig da de fleste sensorprodusenter enten kompenserer for dette selv i en ferdig løsning eller gir all nødvendig informasjon for å kompensere der nødvendig.

2.3.1 Punktskyer

En punktsky er et sett punkter i rommet som representerer et 3D-objekt. [10] Hvert punkt har X, Y og Z koordinater. Punktskyer er gjerne en av de tidligste dataformatene en 3D-skanner gir ut, før denne har blitt prosessert til et vanntett objekt eller hva annet som er ønsket av skanningen. En punktsky alene forteller kun om posisjonen til objektet som har blitt skannet, hvert individuelle punkt er kun en posisjon, og har ingen annen informasjon. For å gjøre punktene mer brukbare i andre sammenhenger finnes det teknikker for å finne punktskyens normaler; altså normaliserte vektorer som står normalt på planet gjennom punktet som er parallelt med objektets overflate i det punktet.

Dersom en har normalene til en punktsky går det an å gjøre om punktskyen til et objekt med flater. Denne prosessen kalles meshing; da man lager en netting (eng. mesh), hvor alle krysningpunktene i nettingen er punktene i skyen, som definerer overflatene til objektet. Flatene til objektet blir da altså hullene i nettingen. Det finnes mange meshingalgoritmer, både de som behøver normaler og de som ikke gjør det. Algoritmene som benytter normaler er gjerne å foretrekke over de som ikke behøver det da disse har mer data å arbeide utifra, altså det blir mindre inferens. De forskjellige algoritmene er bedre på forskjellige ting. F.eks. gir en fouriermeshing en noe overglattet modell men med god nøyaktighet, mens IMLS (Implicit Moving Least Squares) gir en mindre glatt og dermed mer virkelighetsnær modell men kan lage hull som ikke var der originalt [1]. Avhengig av om punktskyen har punkter fra overalt rundt objektet, eller om den kun har punkter fra noen sider, kan man lage en vanntett modell i standard formater ved hjelp av meshingen.

Kapittel 3

Materialer og metoder

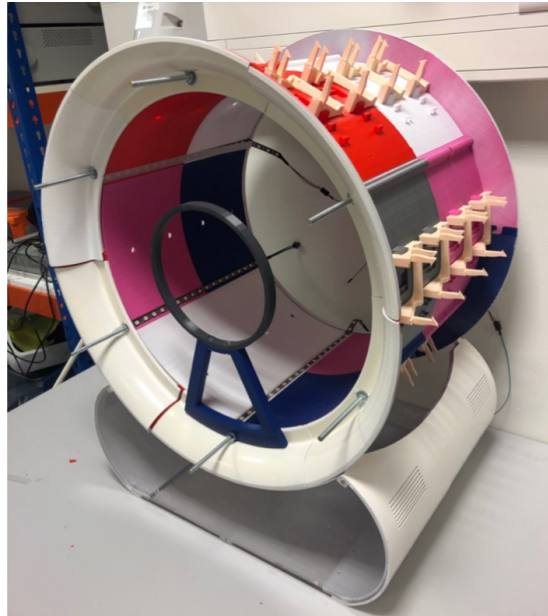
3.1 Sensorvalg

Tidligere forsøk på dette problemet har benyttet seg av tre typer sensorer. Synlig lys kameraer, 3D kamera, og LiDAR.

3.1.1 Synlig lys kamera

Et synlig lys kamera er hva en oftest vil tenke på ved bruk av ordet kamera: Det er en optisk sensor som fanger opp synlig lys og gir dette ut i form av enkeltbilder slik som i fotografi, eller flere bilder tatt i rask rekkefølge og satt sammen slik som i video. Prosjektet som benyttet seg av synlig lys kamera gjorde dette av en hovedgrunn: sensorene er svært vanlige og derfor svært billige, og svært robuste. Det er en teknologi som har blitt brukt i mange tiår, begrensningene og styrkene til teknologien er svært godt kjent. Dette ville i teorien gjøre det endelige resultatet både robust og lett å vedlikeholde.

Hovedproblemet ved bruk av synlig lys kamera til dette formålet er at sluttmålet er en 3D-modell, mens et vanlig kamera kun gir 2D-bilder. Dette ble forsøkt løst ved hjelp av fotogrammetri. En mengde kamera ble plassert i en trommel som kunne rotere rundt objektet. Dette ville gi mange bilder svært raskt, ikke ta opp stor plass, og i teorien gjøre fotogrammetrien enklere. Selv om løsningen i teorien var god, ble sluttresultatene umulige å bruke på grunn av kvaliteten. Dette er ikke en feil de som utførte prosjektet kan noe for, heller er det et problem med fotogrammetri som metode. Fotogrammetri krever svært mange bilder fra svært mange vinkler for å kunne gi gode resultater. I tillegg, siden fotogrammetri benytter seg av landemerker i bildene vil enkelte objekter være vanskeligere å sette sammen, for eksempel en arm som har nogenlunde jevn farge rundt hele. Fotogrammetriprogramvaren er rett og slett ikke sikker på hvilken vinkel et bildet er tatt fra og kan derfor ikke bruke det optimalt.



Figur 3.1: Trommelen, kameraene ville her bli plassert i de beige brakettene rundt trommelen med lysstriper inn i trommelen for å sørge for jevn belysning.

3.1.2 3D kamera/dybdekamera

Et 3D kamera er vanligvis to synlig lys kamera med kjente karakteristikk som er plassert en kjent avstand fra hverandre på en sammensatt sensor. Ofte har kameraet også en infrarød projektor som projiserer et kjent mønster ut i omgivelsene. Disse blir brukt av sensorens programvare for å gi et bedre resultat, tilsvarende landemerker i fotogrammetri. Jeg vil herfra benytte Intel RealSense D435 dybdekamera for konkrete eksempler.

D435 projiserer et rutenett av prikker. Dersom sensoren er pekt vinkelrett mot en flat overflate som ikke er for reflekterende vil mønsteret være perfekt ortogonalt og kvadratisk. Dersom sensoren ikke er vinkelrett mot overflaten vil mønsteret tilsynelatende være forskjøvet, altså der man ville forventet et kvadrat er det nå et trapes, eller mønsteret er ujevnt. Gjennom bildebehandlingsmetoder kan man finne ut av hva som er annerledes med mønsteret i forhold til hva som er forventet, og dermed finne ut hvordan sensoren er i forhold til objektene. Ved å kombinere denne teknikken og parallakse fra de to kameraene som har kjent avstand til hverandre, kan et dybdekart gis fra det enkelte bildet.

3D kamera som egen komponent har ikke blitt brukt annet enn som en vurdert sensor i prosjekter tidligere, inkludert dette prosjektet. Hovedproblemet til dybdekamera er nøyaktigheten. På lik linje med synlig lys kamera løsningen over bruker vanligvis dybdekamera en form for fotogrammetri for å generere dybdekartet, selv om denne gjerne er abstrahert for sluttbrukeren. Derfor vil

dybdekamera støte på mange av de samme problemene som synlig lys kameraer, men fordi mange kjente problemer har blitt vurdert og løst vil dybdekamera ha bedre resultater enn en hjemmelaget løsning.

3.1.3 LiDAR

LiDAR står for Light Detection and Ranging. Det er en liknende teknologi til RaDAR (Radio Detection and Ranging) som er brukt i mange sammenhenger. Jeg vil her ta for meg kun LiDAR, men prinsippene er de samme for RaDAR. Hovedforskjellen mellom de to er at RaDAR bruker radiobølger heller enn synlig eller nær-synlig lys.

LiDAR fungerer ved å sende ut en puls med lys mot et objekt. Objektet vil reflektere noe av dette lyset tilbake til sensoren. Avhengig av hvor lang tid det tar mellom at LiDARen har sendt ut pulsen og den får refleksjonen tilbake, kan avstanden til objektet bli regnet ut, da lysets hastighet er konstant i luft.

LiDAR har som hovedfordel at det er ekstremt nøyaktig og kan, avhengig av omstendighetene, gi svært gode resultater for avstandsnøyaktighet og kantfinning av objekter.

3.1.4 Håndholdt 3D-skanner

Noen av de tidligste forsøkene på dette prosjektet benyttet seg av en kommersielt tilgjengelig håndholdt 3D-skanner. Dette ble også vurdert for dette forsøket av prosjektet.

Hovedfordelen med en kommersiell løsning, som for eksempel de levert av selskapet Einscan, er at noen andre allerede har løst mange av de potensielle problemene som kommer opp. Det er ikke nødvendig for den som benytter en kommersiell skanner å tenke på sensorvalg, programvare eller begrensninger av løsningen, all informasjonen nødvendig ligger i lett tilgjengelige datablader og databehandling kan gjøres enten med programmer og APIer laget av produsenten eller via tredjepartsløsninger som mest sannsynlig eksisterer på internettet.

Hovedulempen er prisen. De kommersielle løsningene er gjerne testet både for robusthet og brukervennlighet. De har i tillegg trolig flere tusen arbeidstimer med design, utvikling og tilvirkning bak seg. Alt dette skal tjenes inn igjen, og den beste måten å gjøre dette på er ved å heve prisen mot sluttbruker.

For noen formål er kommersielle løsninger de beste løsningene, dette gjelder spesielt dersom bruker kommer til å være mindre teknisk anlagt, eller har mindre tid til feilsøking og utvikling selv. Det er mye enklere å kunne bestille et produkt og plukke dette ut av en eske, koble det i datamaskinen og bruke det umiddelbart. Da kan også kostnaden bli vurdert som verdt det.

3.1.5 Vurderinger for dette prosjektet

Opgaven her er å kunne skanne et objekt (en underarm) i for det meste kjente omgivelser. Dette er et objekt av omtrent kjente dimensjoner, da størrelsen på en underarm ikke varierer veldig fra person til person. Dette gjør at avstanden

	Kravspec	L515	TeraRanger Evo Mini	Einscan-SE
Pris	Så lav som mulig	≈ 3500NOK	≈ 270NOK	17.500 NOK
Nøyaktighet	< 5mm	≈ 1mm ved 0.5m	±15mm ved kort avstand	< 0.1mm
Skann-avstand	< 50cm	25 – 900cm	3 – 300cm	Laget for 3D-skanning på kort avstand
Lisenskrav	Ingen krav for universiteter	Har open-source SDK	Benytter standard filformater ut	Har SDK , lisenskrav usikkert
Filformat	Standard filformat ut	Ikke relevant	Standard filformater	Kan konverteres til standard filformat
Festemuligheter	Mulighet for festing	Standard kameraskruer + M3	M2 bolthull	Behøver brakett (følger med)

Tabell 3.1: Krav til sensorvalg og sammenligning mellom potensielle sensorer

som skannet skal utføres på er omtrent konstant. Lysforholdene vil også være mer eller mindre kjente.

De viktigste målene for godt resultat er en klar objektkant og god dybdenøyaktighet. Derfor ble LiDAR vurdert som antageligvis den beste teknologien å gå utifra. Allikevel var det fristende å ha en fullstendig ferdig løsning, og kommersielle, håndholdte skannere ble også vurdert. De tre sensorene som endte opp på kortlisten var da Intel RealSense L515 (LiDAR), Terabee TeraRanger Evo Mini (LiDAR), og Einscan-SE (programmert lys/dybdekamera).

Terabee sin sensor var fristende da denne hadde en ekstremt lav pris og svært nær skann-avstand, det ville derfor vært mulig å sette sammen mange sensorer til et stort nettverk som ikke ville tatt mye plass. Sammenlignet med trommelen i figur 3.1 ville en potensiell skanner laget med Terabee-sensorene vært praktisk talt lommestørrelse. TeraRanger Evo Mini ble valgt bort ved nærmere undersøkelse av utformingen til sensoren. Den har svært lav oppløsning og svært høy synsvinkel. Sensoren vil da alltid merke punkter på objektet som er høyere enn andre, og derfor vil den ikke kunne gi detaljerte nok skann uten ekstremt mye etterarbeid. Dette ble vurdert som uholdbart for formålet.

Av de to gjenværende sensorene ble Intel RealSense L515 valgt som den beste. Den er hovedsakelig laget for industrielle formål som benytter seg av ma-



Figur 3.2: Riggen lagt ut flatt med alle delene synlig bortsett fra kulelagrene.

skinsyn, men 3D-skanning er også et av de uttalte formålene til Intel for denne sensoren. L515 ble valgt da LiDAR ble vurdert som den beste teknologien for formålet, og med de beste mulige resultatene. Sensoren er også langt billigere enn den andre sensoren som ble vurdert så langt i prosessen, Einscan-SE, med tilsynelatende bedre resultater under forholdene som vil være. De bedre resultatene kan forklares av at Einscan-SE bruker programmert lys og synlig lys kamera, som er mer sårbare for støy. L515 bruker en infrarød laser som fremdeles kan oppleve støy i sollys eller andre miljøer hvor det er mye innfrarødt lys, men på grunn av laserens egenskaper og sensorens innebygde egenskap til å kompensere for mindre mengder sollys blir denne innvirkningen svært liten.

3.2 Riggen

Riggen er en mindre videreutvikling av den samme riggen brukt i Romunstads prosjekt tidligere år [8]. Denne består av et 3D-printet bordfeste med skruklemme, bordfestet har plass for en aksling i både horisontal og vertikal stilling. Akslingen har et 3D-printet lagerhus som er festet til akslingen ved hjelp av kulelager. Lagerhuset holder fast en aluminiumsprofil med t-slot som er festet normalt på akslingen. Denne profilen har en annen profil festet normalt på seg hvor sensoren sitter. Sensoren er festet i profilen ved hjelp av en brakett i t-slotten. En motorisert løsning er under utvikling.

Når riggen er satt opp for vertikal skanning opplever nesten ingen friksjon mellom armen og bordfestet. Denne modusen er ideell for testing under utvik-

Mengde	Beskrivelse
100cm	Aluminiumsprofil, 20x20mm, 6mm groove
2stk	Kulelager, plastikk, 20m ID, 42mm OD
1stk	Aksel, 20mm OD
1stk	Bordfeste, 3D printet
2stk	Akselstopper, 3D printet
2stk	Vinkelbraketter for 20x20 6mm profil
1stk	Lagerhus, 3D printet
1stk	Sensorbrakett, 3D printet
8stk	T-slot muttere med skruer, 6mm
4stk	M3x15 skruer med muttere

Tabell 3.2: Stykkliste for innskanningsriggen

ling, men vil også kunne fungere for skanning av vilkårlige objekter. Skanneren kan benyttes som en generell 3D-skanner, heller enn en spesielt for formålet her, men dette er ikke hovedmålet til prosjektet. Når riggen er satt opp i horisontal modus, slik den skal når implementert i det endelige resultatet, er hovedproblemet at profilene og sensoren utgjør et ikke-ubetydelig dreiemoment på akslingen. Dette er planlagt løst i den motoriserte løsningen ved å putte motor, styringskomponenter og batteri som motvekt til armen, på den andre siden av lagerhuset.

3.2.1 Stykkliste

Riggen er, som nevnt over, den samme som brukt i Romunstads prosjekt med kun mindre endringer. Jeg vil derfor her gjengi materiallisten hennes.

3.3 Prinsipp for skanning

RealSense SDKen kan en ved hjelp av ønsket språk, her er Python 3.9 brukt, gi tilgang til dybdebilder fra sensoren. Disse bildene kan enten tas enkeltvis eller strømmes kontinuerlig. Disse dybdebildene blir konvertert til punktskyer, som videre blir gjort om til en liste vektorer for hvert punkt. Hver vektor benytter da sensoren som origo og peker til hvert enkelt punkt i punktskyen. Origo blir endret til et punkt på samme høyde som sensoren men som er koaksialt med akslingen. Dette blir gjort for å gjøre kombinasjon av flere punktskyer enklere. Sensoren brukt har ikke innebygd mulighet for filtrering av punkter som er for langt unna, noe som kan være en ønskelig funksjon for fjerning av bakgrunn. Når punktskyen er representert som vektorer blir mangelen på filtreringsmulighet allikevel ikke et problem, da en enkelt kan fjerne punkter som er lengre unna enn ønsket ved hjelp av vektormatematikk. Vektorene som kommer ut av funksjonene brukt er gitt i meter, så dette er en intuitiv innstilling å endre på for videre utvikling.

For testing ble flere punktskyer funnet fra forskjellige, kjente vinkler. Disse ble så rotert utifra de kjente vinklene slik at alle vektorene blir effektivt funnet fra ett synspunkt, heller enn de mange vinklene som faktisk ble brukt. Etter rotasjonen ble de separate listene med vektorer kombinert til én stor liste, som så gir én stor punktsky. Denne kan så meshes og eksporteres i ønsket filformat for videre utvikling og utforming for produksjon av skinnene, eller andre produkt.

3.4 Programvare

For dette prosjektet er Python 3.9 brukt. Python ble valgt da dette er noe jeg har arbeidet med før, som gjør at læringskurven kun vil gå på bibliotekene brukt, ikke selve språket. 3.9 ble valgt vilkårlig da dette er siste stabile versjon av Python per dags dato. Dette valget kunne vært gjort mer intelligent da noen av bibliotekene brukt ikke har ferdigkompileerte versjoner til 3.9, kun til tidligere versjoner. Dette gjorde at noe manuell kompilasjon fra kildekode var nødvendig. I fremtiden er dette trolig ikke nødvendig, da bibliotekene mest sannsynlig vil få ferdigkompileerte versjoner lagt ut på PyPi innen kort tid. Allikevel burde bibliotekenes tilgjengelighet sjekkes mot versjonen av Python brukt før videre arbeid utføres.

3.4.1 Pythonbiblioteker

RealSense SDK

Sensoren valgt, Intel sin RealSense L515 LiDAR, har en SDK med åpen kildekode tilgjengelig for nedlasting. RealSense SDKen er hovedsakelig beregnet for bruk av C++, men det finnes wrappers tilgjengelig gjennom PyPi for bruk av Python. SDKen gir tilgang til alle funksjonene til sensoren i C++, og wrapperen gjør disse tilgjengelig i Python.

Open3D

Open3D er et pythonbibliotek med mange funksjoner og verktøy for prosessering av 3D-objekter. Her er Open3D brukt hovedsakelig for skriving av filer fra dataene prosessert ellers i prosjektet. I senere iterasjoner av prosjektet kan Open3D med fordel bli brukt for meshing av punktskyer.

Numpy

Numpy er det største og mest brukte matematiske biblioteket til Python. Det inneholder de aller fleste matematiske funksjoner og en god del konstanter. I tillegg inneholder numpy verktøy for arbeid med vektorer og matriser. Numpy er brukt i dette prosjektet hovedsakelig for vektormatematikken nevnt i forrige delkapittel.

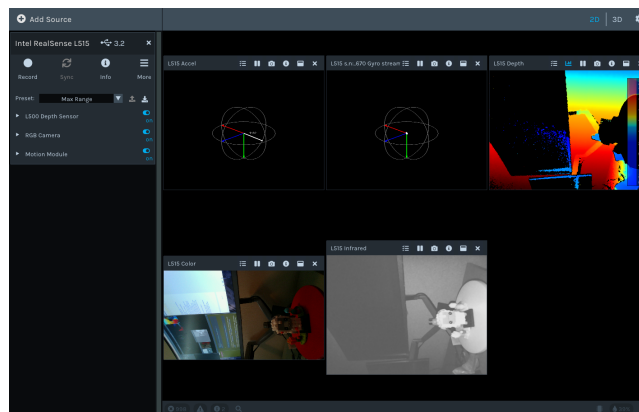
SciPy

SciPy er et av de største matematiske bibliotekene til Python, det inneholder noen andre verktøy og funksjoner som numpy ikke gjør. SciPy er her brukt for å rotere vektorene til samme referansebilde, som nevnt i forrige delkapittel.

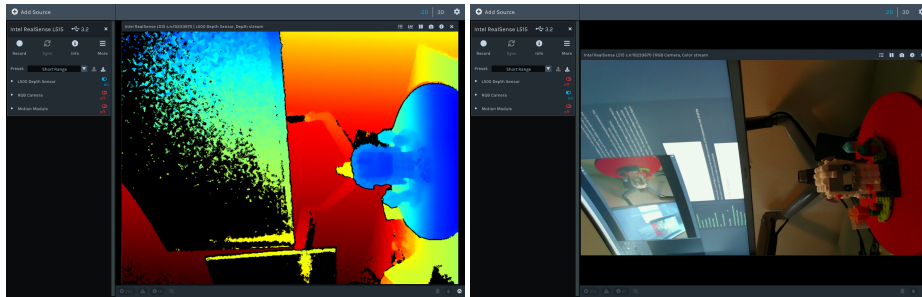
3.4.2 Annen programvare

Intel RealSense Viewer

Intel har tilgjengelig et eget program for bruk av sensorene. Disse gir tilgang på alle innstillinger sensoren er kompatibel med. RealSense Viewer er ikke brukt for noe annet enn grunnleggende testing og framvisninger i dette prosjektet da det ikke har gode verktøy for automatisk eksport av data. Det er også vanskeligere å kjøre og styre et GUI-basert program uten GUI, som er planen senere i prosjektet.

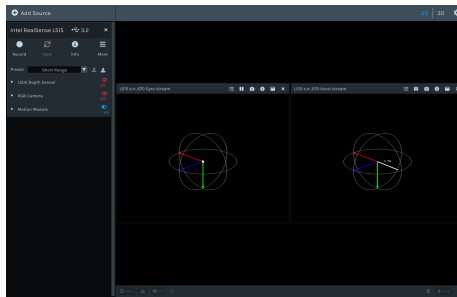


Figur 3.3: Intel RealSense Viewer overblikksbilde



(a) Dybdestrøm, blåere er nærmere, rødere er lengre unna.

(b) RGB strøm



(c) IMU strøm, Gyrometer på venstre i enheter rad/s og akselerometer på høyre i m/s^2

Figur 3.4: De tre undersensorenes datastrømmer visualisert i RealSense Viewer

Bildene fra RealSense Viewer og kommende bilder, vil være rotert 90° mot klokken, dette fordi sensoren er montert på siden. Dette er gjort hovedsakelig for at festbraketten skulle bli enklere å designe, men resultatet er at den bredeste synsvinkelen til sensoren blir i den interessante retningen, på bildene over er det i høyderetning, mens i senere bruk vil det bli langs en underarm.

RealSense Viewer har som nevnt tilgang på alle innstillinger til sensoren. Den har i tillegg tilgang til alle dataene sensoren har mulighet for å gi ut. Dette inkluderer rå dybde data, infrarødt kamera, RGB kamera, samt gyrometer og akselerometerdata fra den innebygde IMUen.

Kapittel 4

Resultat

4.1 Oppgraderinger til riggen

Riggen brukt for skanning er identisk med den brukt av Romunstad tidligere, med to unntak. Avstandsringene brukt mellom lagrene og bordfestet/skannplaten er blitt erstattet med aksellåser som fungerer både som avstandsringer men som også holder lagerhuset festet til akslingen.

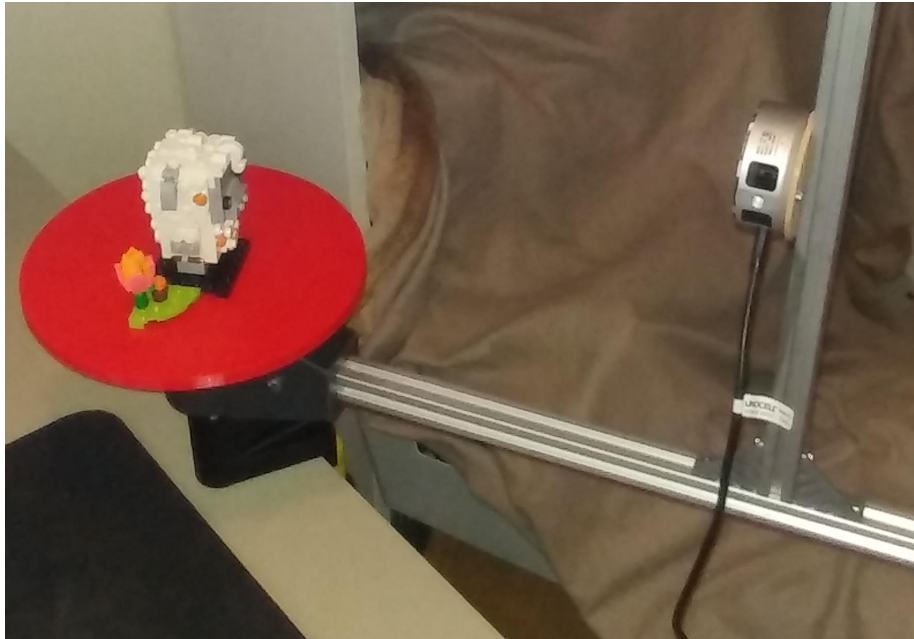
I tillegg er en brakett for sensoren tilvirket, denne har en festeanordning tilsvarende sporene i profilene. Dette gjør at braketten kan sklis opp eller ned sporene i profilen for justering av høyde, tilsvarende justeringen av skann-avstand mellom de to profilene.

Under utvikling er et system for batteridrift og motorstyring. Systemet er per dags dato ferdiglaget i førsteutgave, men har ikke blitt testet med riggen eller sensoren.

4.2 Programvare

Det ble skrevet et pythonskript for styring av riggen. Dette skriptet har følgende trinn

1. Et dybdebilde blir fanget fra sensoren
2. Dybdebildet blir konvertert fra RealSense-punktsky til en liste med vektorer
3. Listen med vektorer blir filtrert etter dybde, vektorer som er for lange (punkter for langt unna sensoren) blir nullstilt
4. Vektorene blir rotert utifra vinkelen dybdebildet har blitt tatt ifra
5. Vektorene får origo endret fra sensorens posisjon til punktet koaksialt med akslingen på samme høyde som sensoren $(0,0,z)$



Figur 4.1: Innskanningsriggen klar for skanning. Objektet som skal skannes er satt på den røde platen. Riggen er her satt opp for skanning av vertikale objekter.

6. Vektorene blir lagt til i en hovedliste vektorer hvor alle vektorene har origo $(0,0,z)$
7. Gjenta punktene over fra så mange vinkler som ønsket

Resultatet blir da en punktsky definert av posisjonsvektorer fra et sentrum. Dette er ikke en overflate eller et objekt enda, men et objekt kan tilsynelatende bli tilvirket fra punktskyen med kun mindre justeringer.

Den fullstendige kildekoden er vedlagt.

Under er prosessen fra et objekt i riggen til en digital modell beskrevet.

4.2.1 Fangning av dybdebilde

Ved hjelp av enten SDKen eller Intels RealSense Viewer blir dybdebildet fanget opp fra sensoren. Til å begynne med er dette dybdebildet `realsense2.depth.frame` format. Dette formatet inneholder mye informasjon om bildet, blant andre dybdedata og normaldata. Dybdebilder kan med fordel benyttes i fremtidige implementasjoner av prosjektet for glatting, da det eksisterer avanserte glattingfunksjoner som en del av RealSense SDKen.

4.2.2 Konvertering til vektorliste

For å lettere kunne arbeide med dataene fanget i forrige punkt blir punkttskyen gjort om til en liste med vektorer. Dette er i grunn bare en annen måte å representere en punktsky da ingen dybde data er tapt. Hvordan en punktsky er formatert blir uvesentlig med dette trinnet, da nøyaktig den informasjonen som er ønskelig blir ekstrahert. Resultatet blir her et `numpy.ndarray` hvor hvert element er en posisjonsvektor til hvert punkt i punkttskyen. Nullvektoren i denne vektorlisten er et punkt på sensorens overflate. Jeg kaller dette punktet origo.

I den gjeldende implementasjonen er normaldata tapt i dette punktet. Normaldata kunne som nevnt i tidligere kapittel vært brukbart for meshing, og burde ikke bli forkastet i senere versjoner.

4.2.3 Dybdefiltrering

Sensoren brukt har en brukbar avstand mellom 0.3 og 9 meter. Siden sensoren kun er om lag 40-50cm unna det som skal skannes vil dette si at det er mye bakgrunn som blir plukket opp av sensoren og bør fjernes. Bakgrunnsfjerningen er gjort både for å eliminere de delene av resultatet som er fullstendig uinteressante, men også for å minimere datamengden som må bearbeides, og dermed redusere prosesseringstiden.

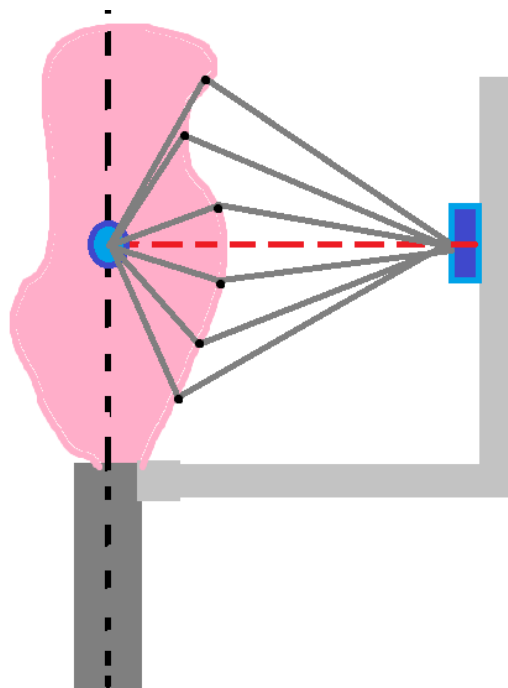
Siden hvert punkt er representert av en posisjonsvektor er det enkelt å svare på spørsmålet ”hvilke punkter er for langt unna sensoren” ved å sammenligne lengden til hver vektor med den ønskede maksimale avstanden som er interessant å fange opp. Denne maksavstanden kan endres på i koden, men siden dette prosjektet er laget uten GUI i tankene er det ikke svært sluttbrukervennlig.

Lengden til vektorene blir funnet ved hjelp av funksjonen `numpy.linalg.norm`. Denne funksjonen tar inn hele vektorlisten og gir ut en avstandsliste. Deretter er det mulig å finne hvilke vektorer i den originale vektorlisten som er for lange ved å sammenligne hver avstand i listen, og hvis avstanden er for stor nullstille den relevante vektoren. I denne sammenhengen med punktskyer og dybdekart er en nullvektor tilsvarende ingen data.

Vektorlisten blir ikke trimmet her for de overflødige punktene som peker til $(0, 0, 0)$ da det under utviklingsprosessen ble vurdert at det ville være enklere å kombinere vektorlistene dersom disse har samme størrelse. Dette kan endres på i senere iterasjoner av prosjektet da det var en feilaktig vurdering jeg kommer nærmere inn på senere.

4.2.4 Endring av origo

I figuren 4.2 representerer det mørkegrå rektangelet nederst akslingen til skaneren, den sorte stiplede linjen er senteraksen til akslingen. Det rosa er objektet som skal skannes. Armen er vist ved de lysegrå elementene, og sensoren ved det blå rektangelet på armen. Den røde stiplede linjen er en hjelpelinje normal med akslingens senterakse som går gjennom sensoren. De sorte punktene er punkter på overflaten til objektet. Den blå ovalen på senteraksen er det nye origo til



Figur 4.2: En illustrasjon av prosessen å endre origo, sett fra siden.

vektorene, mens de grå linjene er vektorene fra det nye origo og sensoren til punktene på overflaten.

Vektorene som blir gitt av dybdebildet tatt med sensoren er tatt med sensoren som nullpunkt. Dette vil si at nullvektoren peker til der sensoren har sin munning. For å forenkle prosessene med rotasjon og kompilasjon, som kommer senere, ønsker jeg å gjøre nullvektoren til bildene fra alle vinkler til ett felles punkt. Dersom rotasjon og kompilasjon hadde blitt gjort uten endringen av origo ville det resulterende objektet vært konkavt, heller enn konvekst som ønsket. Objektet hadde vært ”vrenget” med sensoren som sentrum og objektet strukket ut i en sirkel rundt sensoren, heller enn objektet i midten og sensoren roterende rundt.

Rent praktisk blir endringen i origo gjort som ved vektorsubtraksjon. Avstanden fra sensoren til senteraksen er kjent, i den nåværende implementasjonen er denne 40cm. Altså vil vektoren $[0 \ 0 \ 0.40]$ være vektoren fra sensoren til det nye origo, jeg kaller denne sentervektoren. Ved å subtrahere enhver vektor i vektorlisten med sentervektoren vil den resulterende listen med vektorer være de tidligere vektorene men nå med nullpunkt i punktet $(0, 0, 0.40)$. Disse koordinatene er absolutte med armen som referansepunkt, det vil si at når armen blir rotert rundt objektet må referansepunktet roteres for at punktskyene fra forskjellige vinkler ikke overlapper mer enn de skal.

4.2.5 Rotasjon

Rotasjonen er gjort for at bildene tatt skal stemme overens med rotasjonen av armen rundt objektet. Dersom armen blir rotert en kvart omdreining rundt objektet blir også bildet tatt fra den vinkelen rotert med en kvart omdreining samme vei.

I kode er rotasjonen gjort utifra rotasjonsvektorer. En rotasjonsvektor, eller Eulervektor, er en måte å representere rotasjoner i rommet, hvor hvert element i vektoren representerer en rotasjon rundt den aksen. Det vil si at en rotasjonsvektor $[1 \ 2 \ 3]$ har en rotasjon rundt x-aksen med 1 radian, rundt y-aksen med 2 radianer og rundt z-aksen med 3 radianer.

Fordi riggen ikke er motorisert enda, er det under testing gjort manuelle rotasjoner med en kjent vinkel og manuell fanging av bilder. F.eks. ble armen rotert fra startpunktet med $\frac{\pi}{4}$ radianer, for så å fange et nytt bilde med den nye vinkelen $\frac{\pi}{4}$ radianer som input til programmet.

4.2.6 Kompilasjon

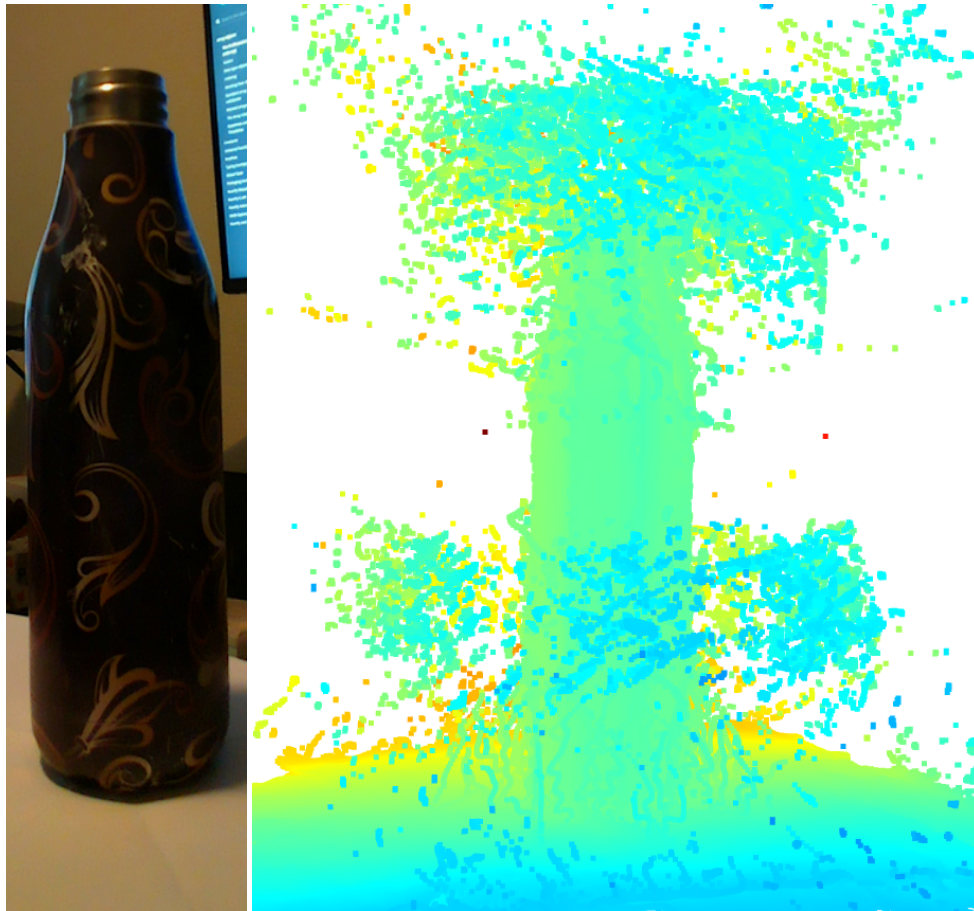
Etter at punktene har blitt rotert den passende vinkelen blir de lagt til i en stor felles liste som representerer det endelige skannet. Dette er nå i teorien en komplett punktsky av objektet.

4.3 Testskann

Flere testskann har blitt gjort for en grov vurdering av kvaliteten det går an å forvente fra sluttproduktet. Disse skannene har mange mangler, blant annet er vinkelforskjellen mellom bilder kun gjort visuelt med enkle hjelpemidler. Avstanden til sentrum er også tatt omtrentlig ved hjelp av et målebånd og øyemål. Omgivelsene brukt er heller ikke ideelle. Jeg mener allikevel at disse skannene kan være hjelpsomme som en grov vurdering av brukbarheten til sensoren i sin helhet, og har derfor inkludert dem her.

Her har jeg inkludert bilder tatt med sensorens RGB kamera, samt dybdebilder tatt ved hjelp av programmet jeg har skrevet, visualisert ved hjelp av Open3D.

På grunn av faktorene nevnt over, samt mangelen på motorisering, var det vanskelig å utføre skann av underarmer, slik det er planlagt å bruke skanneren.



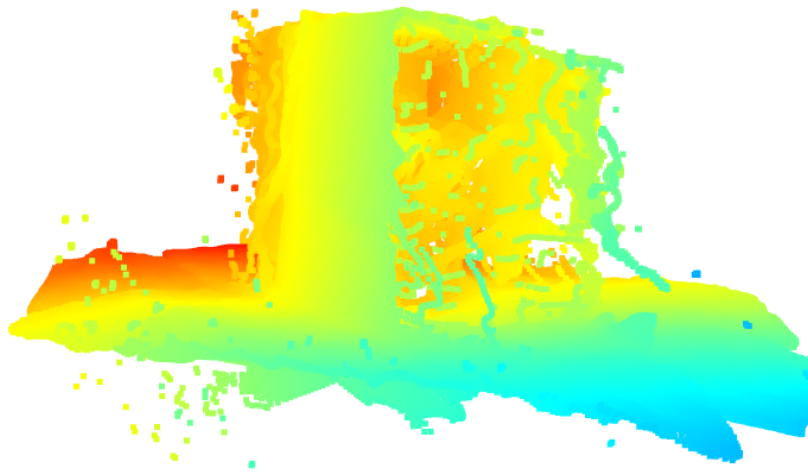
(a) En drikkeflaske

(b) Punktskyen til flasken

Figur 4.3: Flaske, skyene høyt og lavt i bildet er trolig støy på grunn av flaskens refleksivitet



(a) En åpen eske med innside



(b) Punktskyen til esken

Figur 4.4: En eske med synlig innside. Støyen i punktskyen bør være mulig å fjerne ved filtrering senere.

Kapittel 5

Drøfting

5.1 Resultater

Problemstillingen til oppgaven var å videreutvikle og ferdigstille en prototyp for skanning av fortrinnsvis underarmer. Denne problemstillingen har blitt delvis oppfylt. Prototypen er under god utvikling, men er enda ikke fullstendig klar for bruk.

5.1.1 Opfylte mål

Romunstads prosjekt fra 2019 [8] gav en indikasjon til løsningen jeg har brukt her. Hun brukte sensorpakken i fremsiden på en iPhone X, spesifikt de delene av denne som brukes for ansiktsgjenkjenning. Disse er LiDAR baserte og førte til sensorvalget gjort. Derfor ble en LiDAR enhet valgt for dette prosjektet. Enheten valgt har lavere kostnad, tilsynelatende tilsvarende resultater og er enklere å feste til annen maskinvare.

De resulterende innskanningene som er mulige å gjøre med riggen slik den er viser svært lovende resultater. Det er relativt lite støy i overflatene og kvaliteten burde være høy nok til å kunne videre bearbeides dersom noen andre problemer med denne løsningen blir løst først.

Dette prosjektet har også kommet frem til en grunnleggende løsning for automatisering av prosessen. Programvaren som er skrevet har lett for å bli utvidet og videreutviklet, samt at den bør være uproblematisk å implementere i en større prosess, skjult for sluttbruker.

5.1.2 Mangler og utbedringer

Jeg vil her nevne flere problemer som riggen har slik den er i dag. Noen er enkle å løse rent i programvare, andre behøver større eller mindre endringer i maskinvare for å gi gode resultater.

Meshing

Meshing er den største mangelen i prosjektet per dags dato. Uten meshing er det ikke mulig å videre bearbeide punktskyen som blir fanget for generering av skinnen eller for andre formål.

For å utføre meshing bør punktnormalene i fremtiden ikke forkastes slik de blir her fordi normalene forenkler meshingprosessen.

På grunn av oppløsningen i punktskyene som resulterer her er det vurdert at en ball pivot meshing algoritme troligvis er den beste. Dette er på grunn av at ball pivoting er en rask algoritme med små krav til minne, som allikevel gir gode resultater sammenlignet med andre meshingalgoritmer [2]. Det finnes funksjoner i Open3D som gjør meshing ved hjelp av ball pivoting mulig. Disse ble grunnleggende utforsket og vurdert gode for videreutvikling av prosjektet.

Motorisering

Motorisering var originalt et håp å få implementert i denne versjonen, men det ble dessverre ikke mulig på grunn av tilvirkningstiden av de nye delene. Det ble vurdert at det var bedre å fortsette prosjektet med riggen uten motorisering for å komme nærmere en funksjonell prototyp. Det vil så være mulig å føye inn motorstyring senere når skanne-delen var så og si ferdig.

Glatting

Glatting har ikke blitt utført i dette prosjektet på grunn av tidsmangel. Det ble vurdert flere muligheter for glatting, og tilsynelatende er enten fourier- eller poissonglatting de beste alternativene for dette prosjektet. Det er usikkert om glattingen burde skje for hvert enkelt dybdebilde fanget, eller kun for det ferdig kompilerte objektet, eller begge. Det finnes verktøy i RealSense SDKen for glatting som ble vurdert, men disse ble ikke tatt i bruk i dette prosjektet.

Nøyaktig avstand til senteraksen

For endringen i origo nevnt i 4.2.4, er den eksakte avstanden fra sensoren til midtaksen nødvendig å kjenne til. Denne er vanskelig å finne nøyaktig i den nåværende versjonen. Dersom avstanden ikke er nøyaktig vil det gi unøyaktigheter i sammensettingen av bildene og de vil ikke matche med hverandre.

Det er mulig å komme rundt dette problemet på flere måter, den enkleste er å bygge en ny arm med kjente og ikke-endrende dimensjoner slik at det er enkelt å stille inn avstanden fra senteraksen til sensoren.

Et annet alternativ er ved å printe en ny del som kan brukes som målepunkt for midtaksen, for eksempel en plugg som puttes i enden på akslingen og gir et flatt målepunkt. Dette kan være den beste løsningen for denne riggen uten større endringer, men siden denne riggen kommer til å bli nesten fullstendig forkastet i senere versjoner virker dette for meg som bortkastet arbeid.

5.2 Prosjektet

Jeg vil i denne seksjonen drøfte rundt selve utførelsen av prosjektet og hvordan dette har gått.

5.2.1 Utvikling

Utviklingen av programvaren ble gjort ved hjelp av iterativ utviklingsmetodikk. Dette ser ut til å ha fungert godt og gitt gode muligheter for å implementere de endringer som ble sett som nødvendig, enten for brukervennligheten eller for funksjonaliteten sin del.

5.2.2 Tidsstyring

Det virker for min del som om prosjektet ble utført til akseptabel grad. Prosessen med sensorvalg tok mye lengre tid enn hva den i retrospekt burde ha gjort, denne kunne vært kortet ned med antagelig 2 uker.

Begynnelsen på utviklingen tok lengre tid enn forventet og håpet på grunn av bibliotekene valgt. Som nevnt i kapittel 2 var det per dags dato ikke tilgjengelig ferdigkompileerte versjoner av bibliotekene planlagt å brukes tilgjengelig. Disse måtte da kompiles manuelt, en prosess jeg ikke var kjent med fra før av som tok lengre enn den burde ha gjort.

Prosjektmøter med styringsgruppa var i begynnelsen en gang i uken, dette ble nedjustert til en gang annenhver uke, eller ved behov. Dette viste seg å fungere godt.

5.2.3 Gruppa

Det er usikkert hvorvidt flere gruppedeltakere hadde vært hjelpsomt. Av erfaring var selve utviklingsjobben en relativt enkel affære når den først var påbegynt, og jeg tror det hadde vært et problem med mange kokker dersom flere skulle arbeidet med utviklingen av programvaren. Flere deltakere hadde imidlertid kunnet fokusert på forskjellige områder, og ting som meshing og motorstyring hadde kanskje vært gjort ferdig med flere i gruppa. Jeg føler allikevel ikke noe stort savn etter flere gruppemedlemmer nå når jeg ser tilbake.

Kapittel 6

Konklusjon

Målet for oppgaven var å videreutvikle en maskin for innskanning av underarmer og å generere datamodeller fra dette. Dette ble gjort som en del av et større prosjekt hvor målet er å erstatte dagens prosess for gipsing og stabilisering av underarmsbrudd.

En prototyp fra tidligere forsøk på oppgaven ble tatt som utgangspunkt og videreutviklet. Testing av maskinen viser stort potensial for videre bruk, men det er allikevel problemer med løsningen slik den er i dag. Nøyaktigheten til maskinen er god, og videre bearbeiding av resultatene bør være enkel å få til så lenge noen av de ovennevnte problemene blir løst. Resultatet til maskinen per dags dato er en sammensatt punktsky av armen som er forsøkt skannet.

Det ble ikke i denne oppgaven funnet en måte å meshe en modell av punktskyen. Dette er det største hinderet for at prototypen skal være mulig å bruke videre. Det ble heller ikke funnet en måte å motorisere skanneren, noe som ikke er et problem ved testing men er essensielt for at sluttproduktet skal være brukbart.

Denne oppgaven tar kun for seg det første punktet i det større prosjektet om å erstatte dagens gipsingsprosess, det er derfor ikke mulig å kommentere på det større prosjektets levedyktighet eller mulighet for å bli fullført. Resultatene fra denne oppgaven er allikevel lovende, og det gjenstår kun lite arbeid for å regne denne delen av prosjektet som løst.

Kode

```
1 import pyrealsense2 as rs
2 import numpy as np
3 import open3d as o3d
4 from scipy.spatial.transform import Rotation as R
5
6
7 #####
8 # # This program captures a multiple point clouds from the
9 # # depth sensor from multiple different angles,
10 # # filters out points that are too far away,
11 # # and then saves it to disk as a .ply file.
12 #####
13
14 #####
15 # Constants
16 #####
17
18 #set max clipping distance
19 clipping_distance_in_meters = 0.4 #40cm clipping
20
21 # Distance between the far end of the axle and the back end of the sensor arm
22 # This should be changed as necessary with regards to FOV
23 axle_to_arm_distance = 0.378 #37.8cm
24 # distance between the front of the sensor and the back of the sensor arm
25 sensor_to_arm_distance = 0.048 #4.8cm
26 axle_outer_radius = 0.02 #2cm
27 # The real distance from the front of the sensor to the centerline of the axle
28 distance_to_center = axle_to_arm_distance - axle_outer_radius -
  ↳ sensor_to_arm_distance
29 center_vector = np.array((0,0,distance_to_center))
30
31
32
33 def get_depth_vectors (pipeline):
34     #####
```

```

35     # Takes in a realsense pipeline
36     # extracts a depth frame and then
37     # gives out a numpy array of vectors for
38     # each depth point in the depth frame.
39     #####
40
41     pc = 0
42
43
44     #First we take a depth image with the sensor
45     frames = pipeline.wait_for_frames()
46     depth_frame = frames.get_depth_frame()
47
48     #turn depth frame into RS pointcloud
49     pcd = rs.pointcloud()
50     points = pcd.calculate(depth_frame)
51
52     #turn point cloud into numpy array for working with
53     array = np.asanyarray(points.get_vertices()).copy()
54     pc = array.view(np.float32).reshape((array.size, 3))
55
56     return pc
57
58
59 def filter_depth (depth_vectors, depth):
60     #####
61     # Sets vectors in depth_vectors greater than depth to zero-vectors
62     #####
63
64     #the lengths of each vector
65     distance = np.linalg.norm(depth_vectors, axis=1)
66
67     zeroVect = np.zeros((1,3)) #If a point has a value of 0 it's considered not
68     → a point and this can be used for filtering background
69     distance_3d = np.dstack((distance, distance, distance))
70     filtered_vectors = np.where(distance_3d <= depth, depth_vectors, zeroVect)
71     filtered_vectors = filtered_vectors[0,:,:] #removes extra dimension that
72     → appears for some reason
73     return filtered_vectors
74
75 def rotate_vector_about_z(vector, theta):
76     #####
77     # Rotates a vector in 3D space about the z-axis by theta radians
78     #####
79     r = R.from_rotvec(np.array((theta,0,0)))
80     return r.apply(vector)

```

```

80 def get_centered_vectors(pipeline, clipping_distance, theta):
81     #get teh vectors to be worked with
82     vectors = get_depth_vectors(pipeline)
83
84     #filter out vectors that have a magnitude greater than clipping distance
85     filtered_vectors = filter_depth(vectors, clipping_distance)
86
87     #Change the vectors' reference zero to center_vector as opposed to the
88     ↪ sensor
89     centered_vectors = np.subtract(filtered_vectors, center_vector)
90
91     #rotate the vectors to fit into the desired reference frame
92     rotated_vectors = rotate_vector_about_z(centered_vectors, theta)
93     return rotated_vectors
94
95 #set up pipeline
96 pipe = rs.pipeline()
97 config = rs.config()
98
99 #depth sensor configuration
100 config.enable_stream(rs.stream.depth, 640, 480, rs.format.z16, 30)
101 #config.enable_stream(rs.stream.pose, rs.format.motion_xyz32f)
102
103 #start streaming
104 profile = pipe.start(config)
105
106 #colorizer for representation later
107 colorizer = rs.colorizer()
108
109 #set device preset, this allows for close range operation,
110 #but makes long range results less reliable
111 depth_sensor = profile.get_device().first_depth_sensor()
112 depth_sensor.set_option(rs.option.visual_preset,
113     ↪ rs.1500_visual_preset.short_range)
114
115
116
117 numer = 0 #The starting multiple of pi
118 num_of_steps = 8 # each image is taken pi/4 rad apart for a total of 8 images
119
120 vecs = []
121 while numer <= num_of_steps/2:
122     prompt = "Rotate to " + str(numer) + "pi/" + str(int(num_of_steps/2)) + "
123     ↪ then hit enter"
124     input(prompt)
125
126     vectors = get_centered_vectors(pipe, clipping_distance_in_meters, numer *
127     ↪ np.pi/4)

```

```
123     vecs.append(vectors)
124     numer+=1
125     vecs = np.array(vecs)
126
127     vecs = vecs.reshape((int(vecs.size/3), 3))
128
129     # Visualisation
130     pcd_out = o3d.geometry.PointCloud()
131     pcd_out.points = o3d.utility.Vector3dVector(vecs)
132     o3d.io.write_point_cloud("4.ply", pcd_out)
133     o3d.visualization.draw_geometries([pcd_out])
134
135     pipe.stop()
```

Bibliografi

- [1] M Berger, JA Levine, LG Nonato, G Taubin, and CT Silva. A benchmark for surface reconstruction. *ACM Transactions on Graphics*, 2013. doi: doi:10.1145/2451236.2451246.
- [2] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin. The ball-pivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 5(4):349–359, 1999. doi: 10.1109/2945.817351.
- [3] AS Boyd, HJ Benjamin, and C Asplund. Principles of casting and splinting. *American Family Physician*, 2009.
- [4] M Gya and AD Thorsen. Spesiiltilpasset gips for h ndleddsbrudd ved bruk av dagens 3d-teknologi. V r 2017.
- [5] TA Hoseth and AN Dale. 3d-tilpasset st tteskinne ved h ndleddsbrudd. V r 2019.
- [6] KW Nellans, E Kowalski, and KC Chung. The epidemiology of distal radius fractures. *Hand Clinics*, 2012. doi: doi:10.1016/j.hcl.2012.02.001. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3345129/>.
- [7] NHI. H ndleddsbrudd, 03 2018. URL <https://nhi.no/sykdommer/muskelskjelett/beinbrudd/handleddsbrudd/>.
- [8] S Romunstad. Integrering av 3d teknologi for brukertilpasset gips ved  lesund sykehus.
- [9] D Sj stad, VAJ Alvestad, and OH Nedreid. Brukertilpasset gips. V r 2019.
- [10] Tech27. What are point clouds?, 02 2018. URL <https://tech27.com/resources/point-clouds/>.
- [11] N Wongwaen, S Tiendee, and C Sinthanayothin. Method of 3d mesh reconstruction from point cloud using elementary vector and geometry analysis. *Proceedings - ICIDT 2012, 8th International Conference on Information Science and Digital Content Technology*, 06 2012.

Vedlegg

A Forprosjektrapport

FORPROSJEKT - RAPPORT

FOR BACHELOROPPGAVE

TITTEL:

Bruk av 3D teknologi for tilvirkning av brukertilpasset gips ved Ålesund Sykehus

KANDIDATNUMMER(E):

Magnus Renaa Kjørseng

DATO:

Vår 2021

EMNEKODE:

IE303612

EMNE:

Bacheloroppgave

DOKUMENT TILGANG:

- Åpen

STUDIUM:

AUTOMATISERINGSTEKNIKK

ANT SIDER/VEDLEGG:

11/0

BIBL. NR:

- Ikke i bruk -

OPPDRAGSGIVER(E)/VEILEDER(E):

NTNU Biomekaniske laboratorium
Paul Steffen Kleppe
Webjørn Rekdalsbakken

OPPGAVE/SAMMENDRAG:

Håndleddsbrudd er den vanligste bruddtypen i Norge med over 15.000 tilfeller hvert år. Den vanligste behandlingen for håndleddsbrudd er pålegging av gips, en tidkrevende prosess som behøver erfarent personell for å gjøre uten fare for pasientens helse mens gipsen fremdeles støtter som nødvendig.

Denne oppgava er en del av et større prosjekt for å lage en helhetlig løsning for produksjon av individuelt tilpassede støtteskiner som erstatning for den tradisjonelle gipsen. Produksjonen er delt inn i delene innskanning, behandling og tilvirkning. Denne oppgava tar for seg kun innskanningsdelen hvor målet er å skape en digital modell av underarmen og håndleddet til pasienten. Denne modellen vil så bli brukt videre av andre for å gjøre den videre behandlingen av modellen og tilvirkning av den ferdige støtteskinna.

Oppdragsgiver er NTNU Ålesund, ved Paul Steffen Kleppe og Webjørn Rekdalsbakken.

Det endelige formålet ved oppgava er å forkorte prosesstida fra en pasient møter på legevakt eller akuttmottak med håndleddsbrudd til de er ute av døra med en personlig tilpasset støtte. En del av formålet er også å gjøre prosessen billigere ved å kreve mindre tid fra høyt utdannet personale.

Postadresse
Høgskolen i Ålesund
N-6025 Ålesund
Norway

Besøksadresse
Larsgårdsvegen 2
Internett
www.hials.no

Telefon
70 16 12 00
Epostadresse
postmottak@hials.no

Telefax
70 16 13 00

Bankkonto
7694 05 00636
Foretaksregisteret
NO 971 572 140

Denne oppgaven er en eksamensbesvarelse utført av student(er) ved NTNU i Ålesund.

INNHold

INNHold	3
1 INNLEDNING	4
2 BEGREPER	4
3 PROSJEKTORGANISASJON	4
3.1 PROSJEKTGRUPPE	4
3.1.1 <i>Oppgaver for prosjektgruppen - organisering</i>	5
3.2 STYRINGSGRUPPE (VEILEDER OG KONTAKTPERSON OPPDRAGSGIVER)	5
4 AVTALER	5
4.1 AVTALE MED OPPDRAGSGIVER	5
4.2 ARBEIDSSTED OG RESSURSER	5
4.3 GRUPPENORMER – SAMARBEIDSREGLER – HOLDNINGER	5
5 PROSJEKTBEKRIVELSE	6
5.1 PROBLEMSTILLING - MÅLSETTING - HENSIKT	6
5.2 KRAV TIL LØSNING ELLER PROSJEKTRESULTAT – SPESIFIKASJON	6
5.3 PLANLAGT FRAMGANGSMÅTE(R) FOR UTVIKLINGSARBEIDET – METODE(R)	6
5.4 INFORMASJONSINNSAMLING – UTFØRT OG PLANLAGT	7
5.5 VURDERING – ANALYSE AV RISIKO	7
5.6 HOVEDAKTIVITETER I VIDERE ARBEID	8
5.7 FRAMDRIFTSPLAN – STYRING AV PROSJEKTET	9
5.7.1 <i>Hovedplan</i>	9
5.7.1.1 Forprosjekt og kartlegging	9
5.7.1.2 Informasjonssamling.....	9
5.7.1.3 Design av ny rig.....	9
5.7.1.4 Programmering av rig	9
5.7.1.5 Testing av rig	9
5.7.1.6 Rapportering og loggføring.....	9
5.7.2 <i>Styringshjelpemidler</i>	10
5.7.3 <i>Utviklingshjelpemidler</i>	10
5.7.4 <i>Intern kontroll – evaluering</i>	10
5.8 BESLUTNINGER – BESLUTNINGSPROSESS	10
6 DOKUMENTASJON	10
6.1 RAPPORTER OG TEKNISKE DOKUMENTER.....	10
7 PLANLAGTE MØTER OG RAPPORTER	10
7.1 MØTER	10
7.1.1 <i>Møter med styringsgruppen</i>	10
7.1.2 <i>Prosjekt møter</i>	10
7.2 PERIODISKE RAPPORTER	10
8 PLANLAGT AVVIKSBEHANDLING	11
9 UTSTYRSBEHOV/FORUTSETNINGER FOR GJENNOMFØRING	11
10 REFERANSER	11

1 INNLEDNING

Håndleddsbrudd er den vanligste bruddtypen i Norge (NHI, 2018) med over 15.000 tilfeller hvert år. Den vanligste behandlingen for håndleddsbrudd er pålegging av gips, en tidkrevende prosess som behøver erfarent personell for å gjøre uten fare for pasientens helse mens gipsen fremdeles støtter som nødvendig.

Denne oppgava er en del av et større prosjekt for å lage en helhetlig løsning for produksjon av individuelt tilpassede støtteskinner som erstatning for den tradisjonelle gipsen. Produksjonen er delt inn i delene innskanning, behandling og tilvirkning. Denne oppgava tar for seg kun innskanningsdelen hvor målet er å skape en digital modell av underarmen og håndleddet til pasienten. Denne modellen vil så bli brukt videre av andre for å gjøre den videre behandlingen av modellen og tilvirkning av den ferdige støtteskinna.

Oppdragsgiver er NTNU Ålesund, ved Paul Steffen Kleppe og Webjørn Rekdalsbakken.

Det endelige formålet ved oppgava er å forkorte prosestetida fra en pasient møter på legevakt eller akuttmottak med håndleddsbrudd til de er ute av døra med en personlig tilpasset støtte. En del av formålet er også å gjøre prosessen billigere ved å kreve mindre tid fra høyt utdannet personale.

2 BEGREPER

LiDAR	Light Distance And Ranging – bruk av lys for å finne avstand til punkter
Tomografi	Gr. Tomos – seksjon, grafo – tegne, måte å rekonstruere et mer komplisert objekt ved hjelp av flere, enklere seksjoner av objektet, gjerne skiver eller linjer av objektet.
CAD	Computer Aided Design, programvare for tilvirkning av og redigering av 3D modeller
3D-printing	Tilvirkning av digitale 3D modeller

3 PROSJEKTORGANISASJON

3.1 Prosjektgruppe

Studentnummer(e)
Magnus Renaa Kjørseng (studnr. Kommer)

Tabell: Studentnummer(e) for alle i gruppen som leverer oppgaven for bedømmelse i faget ID 302906

3.1.1 Oppgaver for prosjektgruppen - organisering

Da gruppa kun har ett medlem vil alle oppgaver falle på denne.

3.2 Styringsgruppe (veileder og kontaktperson oppdragsgiver)

Paul Steffen Kleppe - Veileder

Webjørn Rekdalsbakken – Veileder

Andreas Fagerhaug Dalen - kontaktperson

4 AVTALER

4.1 Avtale med oppdragsgiver

Oppdragsgiver er NTNU ManuLab/Biomekaniske Laboratorium, ved Paul Steffen Kleppe og Webjørn Rekdalsbakken.

4.2 Arbeidssted og ressurser

Arbeidsplass vil hovedsakelig være enten lesesal/lab for informasjonsanking og prosessering, samt 3D-modellering av deler nødvendig. I tillegg vil lab være nødvendig for utforming og tilpassing av det fysiske produktet som er endemålet.

Magnus Renaa Kjørseng kan nåes ved magnurk@stud.ntnu.no .

Ingen informasjon i rapporten er unndratt offentlighet.

Rapportering er avtalt med veiledere som minst ett ukentlig sammendrag over e-post, samt et møte annenhver uke.

4.3 Gruppenormer – samarbeidsregler – holdninger

Slik jeg har forstått en automasjonsingeniørs oppgave i verden, er denne i hovedsak å finne en måte å få den virkelige, fysiske verden, til å kommunisere med den idealiserte og kontrollerbare digitale verden. I så henseende vil denne oppgava definitivt falle under denne paraplyen. For å oppnå dette mener jeg en av de viktigste holdningene å holde er en klarhet over hvor virkeligheten og modellene avviker. Å vite nøyaktig hvor modellen ikke kan brukes lengre, eller hvor virkeligheten blir for kompleks for å bearbeide. Siden målet er å få de to til å samarbeide, er det essensielt å få dem til å «prate samme språk».

En annen viktig holdning er å ikke bli for fastkjørt i ett enkelt tankemønster. Den virkelige verden er ekstremt kompleks, problemene automasjonsingeniører forsøker å løse er gjerne problemer hvor grensa mellom et trivielt problem og et umulig ett er syltynn. Det å holde et åpent sinn for å kunne finne løsninger eller andre innfallsvinkler for prosjektet virker for min del svært hjelpsomt, og vil kunne hjelpe i produktivitet.

«Det finnes ikke ubrukelig kunnskap» heter det i et ordtak. Dette er gjerne sant, men spesielt innen ingeniørfag generelt vil jeg legge til «... men det finnes irrelevant kunnskap». Evnen til å vite nøyaktig hvilken informasjon som er viktig for problemet, for prosjektet, for å nå målet, og hvilken informasjon som kan være relevant, og hvilken informasjon som er irrelevant, tenker jeg er viktig for en ingeniør, og er et perspektiv på verden jeg ønsker å oppfostre i meg selv.

5 PROSJEKTBEKRIVELSE

5.1 *Problemstilling - målsetting - hensikt*

Håndleddsbrudd (distal radius fraktur) er den vanligste formen for brudd i Norge (NHI, 2018) med rundt 15.000 tilfeller årlig, og blant de vanligste typene brudd på verdensbasis. Den vanligste behandlingen av håndleddsbrudd er per i dag immobilisering ved pålagt gips eller halvgips i enten kalkgips eller glassfiber. Dette er under best mulige omstendigheter en god måte å stabilisere bruddet på, men krever god kunnskap og mye erfaring med å legge gips for å kunne både ivareta den ideelle stillingen til bruddet, samtidig som pasienten ikke blir skadd eller unødvendig plaget av gipsen.

Ved Ålesund Sjukehus har det blitt ytret et ønske om å strømlinjeforme denne prosessen, og gjøre at den ikke lengre krever like mye kunnskap og erfaring fra høyt utdannet personell.

Dette bachelorprosjektet har som oppgave å lage en 3D-skanner for innskanning av solide objekt, fortrinnsvis underarmer og håndledd, som skal brukes videre i det større prosjektet. Denne skanneren skal ha en standardisert utputt som kan lett importeres til verktøyene som brukes videre, hvilket vil gjøre den endelige løsningen mer sømløs.

5.2 *Krav til løsning eller prosjektresultat – spesifisering*

Aller viktigst skal løsningen gi et akseptabelt resultat i form av en punktsky. Denne punktskyen må være nøyaktig nok til å gi et godt sluttresultat, og ikke være for støyete.

Løsningen skal ikke være for stor eller for dyr. Dette er et prosjekt ment for bruk i legevakter og akuttavdelinger på sykehus og må derfor være liten nok til å kunne f.eks. puttes i et skap eller på et bord i et hjørne. Løsningen kan ikke kreve et helt rom.

Løsningen skal ikke kreve mye tid og bearbeiding fra faglært personell. Den skal være så autonom som mulig, helst kun én knapp.

Innskanninga skal være rask. For å få en god støtte for håndleddsbrudd er det viktig at håndleddet er i riktig posisjon. Det er vanskelig for en pasient å holde håndleddet stille og i samme stilling over lengre tid, spesielt uten støtte. Løsningen må derfor være rask, helst under 30 sek, og gjerne ha støtter slik at det ikke er like vanskelig for pasienten å undergå skanninga.

Løsningen skal være så lite komplisert som mulig. Helst skal delene være så «off-the-shelf» som mulig, noe som gjør service og reparasjon enklere. Det vil også føre til at løsningen blir mer hardfør, da et produkt som eksisterer for kommersielt kjøp gjerne har mer QA bak seg enn hva en studentløsning kan komme med.

5.3 *Planlagt framgangsmåte(r) for utviklingsarbeidet – metode(r)*

Begynnelsen til prosjektet vil gå i informasjonssamling. Idéen er å lage en prototyp til en skanner. I motsetning til tidligere oppgaver ønsker jeg her å benytte LiDAR for å lage en skanner basert på tomografi. Informasjonssamlingen vil derfor gå hovedsakelig i banene av fotogrammetri og tomografi, og deres relaterte teknologier og teknikker.

Avhengig av resultatene fra informasjonssamlingen fra tomografi-delen vil så en prototyp bli designet som en etterligning av eksisterende tomografiapparat.

5.4 Informasjonsinnsamling – utført og planlagt

Tidligere rapporter for samme prosjekt er samlet og vil bli gjennomgått i de tidlige ukene i hovedprosjektet. Deres referanser vil også bli undersøkt og brukt om nødvendig og relevant.

5.5 Vurdering – analyse av risiko

Prosjektet er utført under koronaviruspandemien, i vårsemesteret av 2021. På grunn av dette er det mye usikkerhet rundt hvilke muligheter som er åpne for gjennomføring av prosjektet. Jeg vil her gå gjennom scenarier jeg ser for meg som realistiske og urealistiske, samt deres eventuelle løsninger.

Alt åpnes opp og blir som før pandemien: I dette tilfellet vil alt av labber være åpent og tilgjengelig. Fysiske møter og veiledning vil være en mulighet. Dette vil gjøre de praktiske elementene av oppgaven enklere.

Restriksjoner fortsetter som nå: Per i dag (18/01/21) sier Regjeringens smitteverntiltak at alle som har mulighet for hjemmekontor skal benytte seg av dette. Da jeg er alene om prosjektet vil samarbeid i rent digital arbeidsflyt ikke være et problem. Allikevel er labber og kollektivtrafikk åpne så lenge smittevernradene er fulgt. Dette gjør det mindre praktisk å gjennomføre prosjektet, men vil ikke sette noen begrensning annet enn på meg personlig. Prosjektet burde ikke lide noen nød av dette. Viktig å holde i mente her er effektiv tidsbruk. Jeg har ingen adskilt kontor plass i hjemmet mitt, som gjør at det er viktigere med planlegging av hvilken tid som skal brukes til arbeidet, og så gjennomføre disse planene. Dette alternativet vil også være mulig å gjennomføre dersom jeg mot formodning skulle bli smittet selv, jeg vil da bare arbeide med ting som kan gjøres uten tilgang til lab.

Restriksjoner blir strengere (Fullstendig lockdown): Under disse forholdene vil alt arbeid foregå digitalt, med fysisk labvirksomhet umulig. I dette scenariet vil oppgaven måtte bli justert noe og heller bli en slags planleggingsoppgave med større fokus på datamodeller av det ferdige produktet og utforskning av hvordan inndata fra en skanner kan behandles, heller enn å arbeide med å lage en fysisk prototyp av skanneren. Samme krav og fokus som over, men igjen enda vanskeligere mtp. tidsavgrensning.

I tillegg til pandemiproblematikk finnes det og andre problemer som kan oppstå. Dersom det ikke er mulig å anskaffe skanner eller sensorer vil det fremdeles være mulig å lage en prototyp for riggen som vil holde pasientens arm og skanneren i forhold til hverandre, samt å programmere denne for å gjøre det å sette sammen den eventuelle skanner og rig enklere.

Alle LiDAR sensorer har et avstandsområde de fungerer innenfor. For å gjøre skanneren til et praktisk brukbart produkt må ikke dette avstandsområdet være for stort. Altså dersom sensoren brukt kun er nøyaktig mellom 40cm-120cm vil dette gi en skanner med total diameter på minst en meter, noe som kan bli upraktisk å benytte i en klinisk sammenheng. Dersom tilstrekkelig sensor med et passende avstandsområde ikke blir funnet kan dette sette en stopper for prosjektets fysiske del, men det digitale designet kan fremdeles utføres.

Dersom større dataproblematikk oppstår vil det være vanskelig å utføre prosjektet på noe vis, da både 3D-modellering av riggen og programmering av inndatastrømmen fra skanner er digitalt arbeid.

Da et av målene ved oppgava er å lage en punktsky i standardformat vil ikke inkompatibilitet i filformat være et realistisk problem, men i tilfellet det blir det vil det trolig finnes løsninger allment tilgjengelig på nettet for konvertering av punktskyer fra et dataformat til et annet. Alternativt er det en større sjans for at dersom en allerede kommersielt tilgjengelig skanner blir brukt, at denne vil bruke et eget spesialformat. Dette kan løses enten ved å bruke produsentens format og programvare for å konvertere filene, eller ved å ikke benytte et ferdig produkt i den grad i det hele tatt.

5.7 Framdriftsplan – styring av prosjektet

5.7.1 Hovedplan

5.7.1.1 Forprosjekt og kartlegging

Forprosjektarbeidet og kartlegginga av oppgava er første del av prosjektet, som hovedsakelig er arbeidet med å definere og avgrense hva oppgava rommer. Forprosjektet, som er summert av denne rapporten, fungerer som en oppsummering av kartlegginga og arbeidet allerede gjort.

5.7.1.2 Informasjonssamling

Informasjonssamlinga vil som tidligere nevnt hovedsakelig gå i å lese artikler om fotogrammetri og tomografi, samt LiDAR og dets virkemåte. Informasjonssamlinga vil også inkludere undersøkelse av tilgjengelige og brukbare sensorer, og gjøre en beslutning for hvilke sensorer og hva slags annet utstyr som vil være nødvendig for å utføre oppgava.

Hovedpunktet A2 er lagt til lang tid inn i oppgava, dette for å reflektere kontinuerlig arbeid med informasjonssanking innenfor relevante tidsskrifter, bøker og liknende. Det vil få mindre fokus når andre punkter starter opp, men er allikevel en viktig post å holde i mente.

Jeg har også her lagt testinga av den gamle riggen, laget av forrige oppgave (Romundstad, 2020), da denne trolig kan brukes om i noen grad her.

5.7.1.3 Design av ny rig

Her vil enten videreutvikling og modifisering av den tidligere riggen havne, eller nyproduksjon av en annen rig, avhengig av resultatene fra forrige post. For eksempel vil design og 3D-printing av en brakett for å holde den bestemte sensoren havne under denne posten.

Dersom en fullstendig ny rig blir valgt som beste mulighet vil 3D- og elektronisk design av denne havne under denne posten, samt den faktiske tilvirkningen.

5.7.1.4 Programmering av rig

Da tanken er å bruke tomografisk prinsipp og LiDAR for å skanne inn et legeme vil programmeringen trolig ta mye tid. Under dette punktet vil jeg tilvirke inndataprosessen for å gå fra vilkårlige punkter til en brukbar punktsky.

5.7.1.5 Testing av rig

Her, etter at riggen er ferdig tilvirket og i fungerende tilstand praktisk talt vil det bli gjort tester for å undersøke nøyaktigheten i innskanning som riggen kan oppnå, samt om de resulterende punktskyene er av akseptabel kvalitet.

5.7.1.6 Rapportering og loggføring

Loggføring er gjort for å holde kontroll på alt av informasjon som er hentet inn, samt å holde styr på mengden tid brukt i de forskjellige delene av prosjektet. Rapportering vil bli gjort i form av et sammendrag for arbeid planlagt, påbegynt og fullført, samt en vurdering av hvordan arbeidet er gjort i forhold til tidsplanen over.

Rapportskriving er gjort til en større post for å reflektere at denne vil bli kontinuerlig føyd til og utformet, men hovedparten av skrivinga vil trolig ende i slutten av hver hovedposts tid for å oppsummere de relevante delene av hovedrapporten da.

5.7.2 Styringshjelpemidler

Et Gantt-diagram, vist i seksjon 5.6 er brukt som styringshjelpemiddel. Dette vil bli oppdatert etter hvert som oppgava utvikler seg, og vil være hovedhjelpemiddel for å holde oppgava i rute.

5.7.3 Utviklingshjelpemidler

Se punkt 9 for oppsummering av dette.

5.7.4 Intern kontroll – evaluering

Intern kontroll vil bli gjort i sammenheng med ukentlig rapportering, hvor nåværende arbeidsoppgaves framgang vil bli sammenlignet med gjenværende tid oppsatt. Det vil også være en kort oppsummering gjort ukentlig i tillegg til den ukentlige rapporten for å passe på at alt går i rute.

5.8 Beslutninger – beslutningsprosess

Da gruppa bare har ett medlem vil hovedparten av beslutninger og feider i så henseende være et ikke-problem. Større beslutninger vil bli drøftet med styringsgruppa ved nærmeste møte samt over e-post dersom akutte problemer dukker opp.

6 DOKUMENTASJON

6.1 Rapporter og tekniske dokumenter

Alle ukesrapporter vil bli samlet etter hvert som prosjektet fortsetter. Disse er som tidligere nevnt en statusoppdatering på arbeid utført tidligere uke og sammenligning av denne og oppført tidsplan.

En Bachelorrapport vil bli skrevet og utformet i løpet av prosjektet, denne vil leveres inn ved slutten av prosjektet.

7 PLANLAGTE MØTER OG RAPPORTER

7.1 Møter

7.1.1 Møter med styringsgruppen

Statusmøter med styringsgruppa er satt til torsdager i partallsuker, kl. 13:00, med forbehold om mulighet for møter i oddetallsuker samme tid, dersom nødvendig.

7.1.2 Prosjektmøter

Da prosjektet kun har ett medlem vil prosjektmøter være redundant.

7.2 Periodiske rapporter

Rapporter vil bli skrevet og sendt ukentlig, onsdag, innen kl. 14:00 til styringsgruppa, via epost. Disse vil inkludere nødvendig informasjon til neste møte dersom dette er planlagt dagen etter.

8 PLANLAGT AVVIKSBEHANDLING

Da gruppa kun har ett medlem vil alt ansvar falle på denne.

Avviksbehandling vil gå i å oppdage problemer, trolig under ukesrapporten eller oppsummeringen (nevnt i 5.7.4), etter dette må roten til avviket finnes, vurderes og løses. Dersom avviket er stort, skal styringsgruppa kontaktes for veiledning og assistanse.

Å lage en konkret plan for avviksbehandling er vanskelig, da avvik kan være forårsaket av så mangt. Målet vil være å ha avvik løst innen påfølgende uke. De vil så få førsteprioritet på arbeidstid.

9 UTSTYRSBEHOV/FORUTSETNINGER FOR GJENNOMFØRING

Utstyr:

- LiDAR
- Intel RealSense kamera
- Evt. Kommersielt tilgjengelig 3D-skanner (se under)
- CAD-lisens
- Tidligere rig
- Motor
- Raspberry Pi og strømforsyning som kontroller

En 3D-skanner kan vurderes innkjøpt. F.eks. Einscan. Disse vil forenkle oppgaven drastisk og gjøre sannsynligheten for et fullført og vellykket prosjekt mye større. En Einscan-SE skanner bør være bra for et vellykket prosjekt. Prosjektet kan også forsøkes gjennomført med andre typer sensorer, fortrinnsvis en LiDAR og et allerede anskaffet Intel RealSense 3D-kamera.

10 REFERANSER

- NHI. (2018, 03 06). *Håndleddsbrudd*. Hentet fra Norsk Helseinformatikk:
<https://nhi.no/sykdommer/muskelskjelett/beinbrudd/handleddsbrudd/?page=1>
- Romundstad, S. (2020). *Integrering av 3D teknologi for brukertilpasset gips ved Ålesund Sykehus*.

B Ukesrapporter

Rapportene er skrevet for hver uke og sendt til styringsgruppa. Disse dokumenterer arbeidet gjennomført foregående uke.

Periode: Uke 5

Hovedmål med perioden som var:

- Vurdering av sensorvalg
- Logg og rapportføring
- Generell informasjonsanking

Planlagte aktiviteter denne perioden:

- Søken etter passende sensorer
- Leting og lesing av relevante artikler

Faktisk utførte aktiviteter:

- Samtlige

Avviksbeskrivelse:

- N/A

Endring i framdriftsplan:

- N/A

Hovederfaringer fra perioden:

- Det finnes flere mulige framgangsmåter mtp. sensorer. Enten bruk av svært gode, men dyre sensorer, med minimal til ingen etterbehandling av data; eller bruk av flere billigere, mindre gode sensorer for så å sy sammen flere mindre gode datasett til ett større, bedre datasett.

Hovedfokus neste periode:

- Bestilling av sensorer
- Videre undersøkelse av sensorvalg og muligheter de gir

Planlagte aktiviteter neste periode:

- Lese manualer og relevante dokumenter til de valgte sensorene
- Begynne undersøkelse av den gamle riggen
- Finne flere relevante artikler

Behov fra styringruppa:

- Diskusjon rundt bestilling av sensorer og sensorvalg.

Periode: Uke 6

Hovdemål med perioden som var:

- Bestilling av sensorer
- Videre undersøkelse av sensorvalg og muligheter de gir

Planlagte aktiviteter denne perioden:

- Lese manualer og relevante dokumenter til de valgte sensorene
- Begynne undersøkelse av den gamle riggen
- Finne flere relevante artikler

Faktisk utførte aktiviteter:

- Laget sammenligningstabell for de aktuelle sensorene
- Funnet og gjort grov undersøkelse av SDKer tilgjengelig
- Funnet ut av lisensproblemer

Avviksbeskrivelse:

- Det er vanskelig å gå i dybden av manualer og støttedokumenter når sensor ikke er eksplisitt valgt enda
- Tilgang til den gamle riggen ble ikke diskutert ved forrige møte og kom derfor ikke opp.

Endring i framdriftsplan:

- Ingen

Hovederfaringer fra perioden:

- Intel-sensoren virker mer attraktiv enn den var tidligere og virker nå som en god motpart mot Einscan-sensoren. Begge har tilgjengelige SDKer, Intel sin er garantert open-source, Einscan sin har ingen lisensavtale og direkte spørsmål om dette til Einscan har ikke gitt gode svar. Hvordan dette skal tolkes må diskuteres med de som vet mer om opphavsrett og liknende. Einscan-sensoren er mer attraktiv i at den vil kunne gi ut en fullstendig, ferdig modell av en scan, men den kan kreve markører for å få en god scan på grunn av en underarms relativt få merkverdige landemerker for scanning, den er nemlig fotogrammetrisk. Intel-sensoren har akseptabelt feilområde, og bruker en laser heller enn hvitt lys, dette vil kunne gi mindre interferens fra andre lyskilder, og gi et bedre resultat rent i datainnsamlinga, men disse dataene må såklart bearbeides mer videre.

Hovedfokus neste periode:

- Lese manualer og relevante dokumenter til de valgte sensorene
- Begynne undersøkelse av den gamle riggen

Planlagte aktiviteter neste periode:

- Lese opp på SDK docs til den valgte sensoren
- Undersøke riggen på labben
- Finne flere relevante artikler

Behov fra styringruppa:

- Felles tenking rundt hvilken sensor som er best å bruke.

Periode: Uke 7 + 8

Hovdemål med perioden som var:

- Lese manualer og relevante dokumenter til de valgte sensorene
- Begynne undersøkelse av den gamle riggen

Planlagte aktiviteter denne perioden:

- Lese opp på SDK docs til den valgte sensoren
- Undersøke riggen på labben
- Finne flere relevante artikler

Faktisk utførte aktiviteter:

- Har mottatt og begynt arbeid med rig.
- Har begynt undersøkelse av SDK ved hjelp av RealSense D435
- Har vurdert muligheter for braketter til riggen for festing av sensorer
- Har vurdert muligheter for dataoverføring (Sensorene er visstnok problematiske med lengre/lavere kvalitet kabler)
- Har fortsatt skriving i rapporten.

Avviksbeskrivelse:

- Literaturstudie har utgått til fordel for arbeid med riggen.
- Ukesrapport uke 7 utgikk ved forglemmelse

Endring i framdriftsplan:

- Pkt. A3 vil praktisk talt utgå fullstendig da den nåværende riggen er vurdert god nok til å videreføre uten større modifikasjoner. Noen braketter må ordnes, men dette vil ikke ta opp den oppsatte tida.

Hovederfaringer fra perioden:

- Visstnok er festeskruene til sensorene svært svake og ødelegges lett. Dette må tas med i vurderinga for festing av sensorer til riggen.
- Sensorene krever korte, høykvalitetskabler for å fungere optimalt. Det kan være nødvendig med en mellommann for å kunne gjøre om dataene til noe mer robust (RasPi 4, f.eks.)

Hovedfokus neste periode:

- Feste sensorer bedre til riggen
- Få større kjennskap til nødvendig programvare
- Videre literaturstudium
- Videre arbeid med rapport
- Vurdere behov for mellommann mellom sensorer og datamaskin

Planlagte aktiviteter neste periode:

- D2esigne og tilvirke braketter for festing av sensorer
- Arbeide mer med SDKen og programvaren til Intel
- Føre inn mer av nåværende "løse tanker" i rapporten for å få dokumentasjon på utført arbeid
- undersøke muligheter for dataoverføring fra sensorer til datamaskin og hva som eventuelt må mellom.
- Regne og finne ut hvor stor og hva slags motor som kreves, samt kontrollbehov for denne.

Behov fra styringruppa:

- Ny bordfesteordning for riggen
- 3D-filer

- Diskusjon rundt RasPi eller alternativ for dataoverføring

Periode: Uke 9

Hovedmål med perioden som var:

- Feste sensorer bedre til riggen
- Få større kjennskap til nødvendig programvare
- Videre litteraturstudium
- Videre arbeid med rapport
- Vurdere behov for mellommann mellom sensorer og datamaskin

Planlagte aktiviteter denne perioden:

- Designe og tilvirke braketter for festing av sensorer
- Arbeide mer med SDKen og programvaren til Intel
- Føre inn mer av nåværende "løse tanker" i rapporten for å få dokumentasjon på utført arbeid
- undersøke muligheter for dataoverføring fra sensorer til datamaskin og hva som eventuelt må mellom.
- Regne og finne ut hvor stor og hva slags motor som kreves, samt kontrollbehov for denne.

Faktisk utførte aktiviteter:

- Begynt arbeid med braketter
- Regnet ut behov for motor (1:1 giring behøver cirka 0.5Nm, giring kan gi mulighet for mindre kraftig motor og større nøyaktighet i bevegelse)
- Rapportarbeid
- Tittet noe på krav til sensorer og SDK

Avviksbeskrivelse:

- Jeg har ikke satt av nok tid til arbeid denne uken pga. personlige aktiviteter. Dette har gjort at jeg har hatt mindre tid til arbeid.
- Jeg er ikke sikker på hvilken løsning som vil være best med braketter, dette har sakkert arbeidet på den fronten.

Endring i framdriftsplan:

- Ingen endring nødvendig

Hovederfaringer fra perioden:

- Arbeidsflyt er viktig. Jeg må sette av mer tid og være mer effektiv i tiden jeg arbeider for å faktisk komme i mål.
- Uker burde ha færre hovedfokus for å faktisk kunne fokusere på arbeidsoppgavene.

Hovedfokus neste periode:

- Fulføre braketter
- Videre arbeid med SDK og sensorinndata
- Begynne vurdering av sammenføring av data.

Planlagte aktiviteter neste periode:

- Fulføre design og tilvirkning av braketter
- Feste sensorer på riggen
- Få sensorene til å samarbeide
- Få ut brukbare data fra sensorene

Behov fra styringruppa:

- Ingen foreløpig

Periode: Uke 10

Hovedmål med perioden som var:

- Fulføre braketter
- Videre arbeid med SDK og sensorinndata
- Begynne vurdering av sammenføring av data.

Planlagte aktiviteter denne perioden:

- Fulføre design og tilvirkning av braketter
- Feste sensorer på riggen
- Få sensorene til å samarbeide
- Få ut brukbare data fra sensorene

Faktisk utførte aktiviteter:

- Fulført og tilvirket braketter
- Festet sensorer
- Vurdert samarbeid mellom sensorer og sensorinndata

Avviksbeskrivelse:

- SDKer har ikke blitt vurdert noe denne uka pga. at det er enklere å forstå programvaren med praktiske eksempler, og praktiske eksempler er vanskelig å få til uten at sensorene er festet til riggen.

Endring i framdriftsplan:

- N/A

Hovederfaringer fra perioden:

- En fullstendig 3D printet brakett uten profil-deler ble vurdert det lureste, denne ser ut til å holde godt uten å skli i profilen.

Hovedfokus neste periode:

- Starte arbeid på ordentlig med sensordata

Planlagte aktiviteter neste periode:

- Undersøke SDK etter løsning for å sammenføre dataene fra de to sensorene
- Gjøre test-skanninger manuelt
- Vurdere skanningskvalitet og dybde

Behov fra styringgruppa:

- Helhetsvurdering av prosjektet så langt

Periode: Uke 11

Hovedmål med perioden som var:

- Starte arbeid på ordentlig med sensordata

Planlagte aktiviteter denne perioden:

- Undersøke SDK etter løsning for å sammenføre dataene fra de to sensorene
- Gjøre test-skanninger manuelt
- Vurdere skanningskvalitet og dybde

Faktisk utførte aktiviteter:

- Vurdert skanningskvalitet
- Fått brukbar inndata
- Satt opp utviklingsmiljø på datamaskin
- Laget disposisjon til rapport og arbeidet videre med denne

Avviksbeskrivelse:

- Det tok lengre tid enn forventa å sette opp utviklingsmiljøet på maskina da dette måtte kompileres manuelt. Dette førte til at det ikke var mulig å gjennomføre test-skanninger på noe bedre vis enn bare å teste sensorene med Intel sin egen software. Denne testinga gav imidlertid gode resultater med stillestående variasjon i dybde på rundt 1mm ved den planlagte avstanden.
- Sammenføring av sensorer er vurdert unødvendig, se under.

Endring i framdriftsplan:

- Ingen endring nødvendig enda.

Hovederfaringer fra perioden:

- Har compilert en SDK manuelt, som er en god erfaring for fremtidig liknende arbeid.
- Pga. den vide synsvinkelen til sensorene har det blitt vurdert å bare bruke én sensor. Dette vil forenkle inndataprosessen, men gi mindre data å arbeide med som kan føre til mindre data å trekke snitt over for glatting og liknende. Jeg tenker allikevel at det er bedre å arbeide med kun én sensor, både fra et robusthetstandpunkt (det er mindre sjanse for feil med ferre feilmomenter) og fra et utviklingsstandpunkt.

Hovedfokus neste periode:

- Arbeide videre med maskinell inndata
- Arbeide videre med rapport

Planlagte aktiviteter neste periode:

- Finne koordinatsystem som vil være gunstig for sensordata
- Tolke dybdekart (fra sensoren) til punkttskyer
- Finne måte å kombinere punkttskyer fra flere vinkler til én stor punkttsky
- Arbeide med filtrering av punkter for minimering av datamengde

Behov fra styringruppa:

- Dersom styringsgruppa har elementer de mener er viktige å nevne i den endelige rapporten er disse ønsket

Periode:

Hovedmål med perioden som var:

- Arbeide videre med maskinell inndata
- Arbeide videre med rapport

Planlagte aktiviteter denne perioden:

- Finne koordinatsystem som vil være gunstig for sensordata
- Tolke dybdekart (fra sensoren) til punktskyer
- Finne måte å kombinere punktskyer fra flere vinkler til én stor punktsky
- Arbeide med filtrering av punkter for minimering av datamengde

Faktisk utførte aktiviteter:

- Har arbeidet noe med rapport
- Planlagt og kartlagt nødvendige endringer i planen pga. nye koronarestriksjoner.

Avviksbeskrivelse:

- Pga. endringer i smittesituasjon i Ålesund har denne uken praktisk talt gått bort. Noe arbeid har allikevel blitt utført med arbeid med disposisjon til rapport og noe skrijving på selve rapporten. Neste periode vil ta over fra denne for det meste.

Endring i framdriftsplan:

- Det er blitt vurdert om framdriftsplanen bør endres på. Enn så lenge har jeg fremdeles ifølge planen 4-5 uker resterende med rent arbeid på riggen og utstyret. Jeg vurderer dette som tilstrekkelig enn så lenge, avhengig av hvordan arbeid i den endrede situasjonen vil fortsette. Alt utstyr nødvendig er fremdeles tilgjengelig, men noe oppsett vil bli nødvendig å gjøre på nytt.

Hovederfaringer fra perioden:

- Hvorfor kan ikke bare folk følge restriksjona satt opp...

Hovedfokus neste periode:

- Arbeide videre med maskinell inndata
- Arbeide videre med rapport

Planlagte aktiviteter neste periode:

- Finne koordinatsystem som vil være gunstig for sensordata
- Tolke dybdekart (fra sensoren) til punktskyer
- Finne måte å kombinere punktskyer fra flere vinkler til én stor punktsky
- Arbeide med filtrering av punkter for minimering av datamengde

Behov fra styringruppa:

- Ingen behov vurdert enda. Neste møte planlagt torsdag etter påske (8. april) da jeg har planer om å faktisk ta en ferie for hjernen min.

Periode: Uke 13-14

Hovedmål med perioden som var:

- Arbeide videre med maskinell inndata
- Arbeide videre med rapport

Planlagte aktiviteter denne perioden:

- Finne koordinatsystem som vil være gunstig for sensordata
- Tolke dybdekart (fra sensoren) til punkttskyer
- Finne måte å kombinere punkttskyer fra flere vinkler til én stor punkttsky
- Arbeide med filtrering av punkter for minimering av datamengde

Faktisk utførte aktiviteter:

- Satt opp arbeidsomgivelser på nytt på grunn av fysisk flytting mtp. corona
- Begynt arbeid med tolkning av dybdekart fra sensordata
- Begynt kombinasjon av punkttskyer ved hjelp av forhåndsskapt programvare.

Avviksbeskrivelse:

- Koordinatsystem har blitt vurdert ikke nødvendig å finne da sensoren har flere sensorer enn forventet (se under)
- Arbeid er kun begynt, ikke fullført, både på grunn av størrelsen på prosjektene, men også på grunn av påske.
- Filtrering av punkter har ikke blitt begynt, men dette virker enkelt utifra resultatene så langt
- Rapport er ikke arbeidet med i perioden på grunn av prioritering mot de morsommere elementene (leiking med sensorer)

Endring i framdriftsplan:

- Motostyring er usikkert om jeg klarer å implementere, men dette burde bli enkelt å bare putte inn i fremtiden.

Hovederfaringer fra perioden:

- Sensoren har akselerometer og gyroskop innebygd! Dette gjør at den kan til en viss grad spore sin egen bevegelse. Siden De mulige bevegelsene til sensoren er kjent (den vil kun kunne bevege seg på en sirkel i ett plan) vil det være godt mulig å finne sensorens posisjon relativt til der den tidligere har vært ved hjelp av disse sensorene!

Hovedfokus neste periode:

- Arbeide videre med maskinell inndata
- Arbeide videre med rapport

Planlagte aktiviteter neste periode:

- Skrive videre i rapporten
- Arbeide med bevegelsesdata fra sensoren, se om det er mulig å få gode data ut av dem
- Lage skript for enkle skanninger med riggen.

Behov fra styringruppa:

- Møte i morgen (08/04) for generell diskusjon rundt framgang

Periode: Uke 15

Hovdemål med perioden som var:

- Arbeide videre med maskinell inndata
- Arbeide videre med rapport

Planlagte aktiviteter denne perioden:

- Skrive videre i rapporten
- Arbeide med bevegelsesdata fra sensoren, se om det er mulig å få gode data ut av dem
- Lage skript for enkle skanninger med riggen.

Faktisk utførte aktiviteter:

- Fått til konfigurasjon av sensor i python
- Fått fanget opp enkeltrammer i punktskyformat
- Noe arbeid med filtrering av dybde
- Startet arbeid med bevegelsesdata

Avviksbeskrivelse:

- Rapporten har falt til sides da jeg har kommet inn i en flyt med arbeidet her og vil helst ikke avbryte den mens jeg holder på
- Bevegelsesdata ble nedprioritert under det å faktisk få sensoren til å gi meningsfulle data fra dybdesensoren ble sett som viktigere.
- Skanning er enda ikke en mulighet, men det nærmer seg

Endring i framdriftsplan:

- ifølge planen er det nå kun 3 uker igjen av programmering av riggen. Det kan hende dette må utvides med en uke. Dette burde ikke påvirke sluttresultatet noe.

Hovederfaringer fra perioden:

- Har fått større og bedre kjennskap til dokumentasjonen til sensoren, samt det å arbeide med denne ikke via gui.
- Tidlige resultater viser at dybdekart er noe støyete, det finnes flere løsninger på dette, som alle går ut på gjennomsnittsbilder. Alternativene i det fulførte skannet er å enten ta bilder med liten nok vinkelforskjell til at man da kan bruke de n-nærmeste bildene og glatte på den måten, eller så kan kameraet være stillestående i enkelte posisjoner og ta flere bilder fra samme posisjon for så å glatte disse ut før de blir sydd sammen med bilder fra andre vinkler. Dette må vurderes videre.

Hovedfokus neste periode:

- Få til et grovt skann.

Planlagte aktiviteter neste periode:

- Finne system for å sy sammen bilder fra forskjellige vinkler
- Arbeide videre med dybdefiltrering
- Arbeide videre med bevegelsesdata

Periode: uke 16

Hovdemål med perioden som var:

- Få til et grovt skann.

Planlagte aktiviteter denne perioden:

- Finne system for å sy sammen bilder fra forskjellige vinkler
- Arbeide videre med dybdefiltrering

- Arbeide videre med bevegelsesdata

Faktisk utførte aktiviteter:

- Rapportarbeid
- Noe arbeid med dybdefiltrering (løst men ikke implementert)
- Noe arbeid med bevegelsesdata
- Teoretisk arbeid for sammensying av bilder

Avviksbeskrivelse:

- Et fullstendig scan er enda ikke en realitet, men det er ikke langt unna. Jeg antar at rundt en uke til vil gjøre et skan mulig. Fra tidligere av er punktskyer og manipulering av data en god mulighet og veldig tilgjengelig, men selve arbeidet med å få en god modell ut av dette er fremdeles ikke kommet så langt.

Endring i framdriftsplan:

- Ingen endring nødvendig

Hovederfaringer fra perioden:

- Det er lettere og raskere å arbeide direkte med dybdedata når det kommer inn, for så å sette dette sammen senere, enn å forsøke arbeid med egne filer.
- Sensoren har, i motsetning til alle andre RealSense sensorer, ingen eksplisitt innstilling for maksimumsavstand. Dette er noe irriterende.
- Riggeren er svært mobil, selv i sin nåværende tilstand. Det er ikke vanskelig å demontere den, ta den med til en annen plass, og sette den opp igjen med gode resultat.

Hovedfokus neste periode:

- Få til et grovt skann

Planlagte aktiviteter neste periode:

- Fulføre arbeid med dybdefiltrering
- Finne måte å konvertere koordinatsystem på (for sammensetning av punkter)
- Starte arbeid med å gå fra sett med punkter i rommet til et brukbart filformat (ply, stl, pcd, etc.)

Behov fra styringgruppa:

- Statusmøte i morgen (22/04) for diskusjon rundt fremgang

Periode: Uke 17

Hovedmål med perioden som var:

- Få til et grovt skann

Planlagte aktiviteter denne perioden:

- Fulføre arbeid med dybdefiltrering
- Finne måte å konvertere koordinatsystem på (for sammensetning av punkter)
- Starte arbeid med å gå fra sett med punkter i rommet til et brukbart filformat (ply, stl, pcd, etc.)

Faktisk utførte aktiviteter:

- Fulført arbeid med dybdefiltrering
- Funnet måte å konvertere koordinatsystem
- Fulført arbeid med å gå fra punktsky til brukbart format
- Funnet ut at å bruke IMU er ikke gjørbart da denne ikke er nøyaktig nok

Avviksbeskrivelse:

- Koordinatsystemet som blir laget er "vrengt", jeg har forsøkt å benytte et felles nullpunkt for alle vektorer, men dette ser ut til å ha blitt valgt feil. Dette må tenkes mer på.

Endring i framdriftsplan:

- Det gleder meg å rapportere at dette er et enormt fremskritt. Om noe bør framdriftsplanen endres for å utvides

Hovederfaringer fra perioden:

- IMU er ikke nøyaktig nok, og sensoren har ikke prosesseringskraft til å gi brukbar bevegelsesinformasjon ut. Dette er trist, men vil bli løst med maskinell styring.
- Dybdefiltrering og omgjøring til filformat er svært enkle oppgaver og tar ingen tid å implementere
- Nøyaktigheten i det sammensatte bildet er ikke fantastisk ved manuell kontroll, men ved maskinell styring vil dette trolig gi bedre resultater.

Hovedfokus neste periode:

- Glatte resultatet for å gjøre det enda bedre
- Rapportskrivning

Planlagte aktiviteter neste periode:

- Glatte det sammensatte bildet for å fjerne støy
- Komme nærmere et førsteutkast av rapport

Behov fra styringruppa:

- Møte i morgen 29/04, planlagt fysisk men dette bør vurderes mtp. smittesituasjonen og lokale forskrifter

Periode: Uke 18

Hovdemål med perioden som var:

- Glatte resultatet for å gjøre det enda bedre
- Rapportskrivning

Planlagte aktiviteter denne perioden:

- Glatte det sammensatte bildet for å fjerne støy
- Komme nærmere et førsteutkast av rapport

Faktisk utførte aktiviteter:

- Skrevet en del på materialer og metode
- Skrevet noe om teori
- Kommet nærmere en disposisjon for resultater og diskusjon
- Funnet en mulig løsning for glatting og meshing av punktsky

Avviksbeskrivelse:

- N/A

Endring i framdriftsplan:

- Hovedparten av tiden fremover vil bli brukt til å arbeide ut en rapport. Det kommer til å gå, men det er usikkert om tiden vil strekke til både rapport og videre arbeid.

Hovederfaringer fra perioden:

- Bibliotekene brukt har veldig mange fine innebygde funksjoner
- LaTeX matriser er kjipe å arbeide med
- Ved å benytte et tidsfilter kan "en" dybderamme bli glattet over tid for å få et mer sannhetsnært dydbilde med mindre støy. Dette kan være enklere å gjøre enn å forsøke å minimere støy i punktskyen, da punktskyfiltrering ser ut til å være et nesten uløselig problem, matematisk sett.

Hovedfokus neste periode:

- Rapportskrivning
- Implementere glatting av resultater

Planlagte aktiviteter neste periode:

- Fulføre førsteutkast for kapittel 2 og 3
- Implementere glatting ved hjelp av ball pivoting
 - Behøver normaler for punktskyen
- Starte arbeid med meshing

Behov fra styringruppa:

- Faglig vurdering om prioritering mellom et bedre resultat fra prosjektet eller en bedre rapport
- Møte Torsdag 06/05, 1300 som vanlig.

Periode: Uke 19

Hovdemål med perioden som var:

- Rappportskriving
- Implementere glatting av resultater

Planlagte aktiviteter denne perioden:

- Fulføre førsteutkast for kapittel 2 og 3
- Implementere glatting ved hjelp av ball pivoting
 - Behøver normaler for punktskyen
- Starte arbeid med meshing

Faktisk utførte aktiviteter:

- Ferdig førsteutkast for kapittel 2 og 3
- Halvferdig førsteutkast for kapittel 4
- nesten ferdig kapittel 1

Avviksbeskrivelse:

- Det har blitt vurdert at tida som er igjen må gå kun til rapport. Det er nå en uke igjen og en del arbeid gjenstår. Videre arbeid med prosjektet kan enten fortsettes av andre senere eller over sommeren dersom det er mulig.

Endring i framdriftsplan:

- Ingen endring

Hovederfaringer fra perioden:

- Lite nevneverdig

Hovedfokus neste periode:

- Ferdigstilling av rapporten

Planlagte aktiviteter neste periode:

- førsteutkast kapittel 4, 5 og 6.
- Redigere førsteutkast
- Ferdigstille rapporten
- Innsending

Behov fra styringruppa:

- Jeg forstår ikke helt forskjellen mellom Kapittel 2, 3 og 4. Enten går 2 og 3 i ett med ren bakgrunnsinformasjon, eller så går 3 og 4 i ett med hva som er utført og utrettet. Om jeg kan få noe mindre veiledning på dette hadde det hjulpet.

Behov fra styringruppa:

- Umiddelbare tanker og meninger
- Ingen møte vurderes nødvendig denne uken. Ved møte neste uke (22. april) skal det i teorien være mulig å gjøre et skann med riggen.

