



BACHELOROPPGAVE

Oppgradering av datainnsamlingsystem for forskning på Ciona Intestinalis

Upgrading data acquisition system for research on Ciona Intestinalis

Andreas Kjøs Seljeset

Dennis Leonhardsen Dimmen

Håvard Faxvaag Johnsen

Automatisering med robotikk

Institutt for datateknologi, elektroteknologi og realfag

Fakultet for ingeniør- og naturvitenskap (FIN)

Bergen, 26. Mai 2022



Høgskulen
på Vestlandet



UNIVERSITETET I BERGEN

Andreas Kjøs Seljeset

Andreas Kjøs
Seljeset

Håvard Faxvaag Johnsen

Håvard Faxvaag
Johnsen

Dennis Dimmen

Dennis Leonhardsen
Dimmen

Dokumentkontroll

<i>Rapportens tittel:</i> BO22EB-04 – Oppgradering av datainnsamlingssystem for forskning på Ciona Intestinalis.	<i>Dato/Versjon</i> 26. mai. 2022/1.00
	<i>Rapportnummer:</i> B022EB-04
<i>Forfatter(e):</i> Andreas Kjøs Seljeset: 588254@stud.hvl.no Dennis Leonhardsen Dimmen: 585111@stud.hvl.no Håvard Faxvaag Johnsen: 584509@stud.hvl.no	<i>Studieretning:</i> AUTB_19
	<i>Antall sider m/vedlegg</i> 77
<i>Høgskolens veileder:</i> Yngve Thodesen – 55 58 75 32, yngve.thodesen@hvl.no	<i>Gradering:</i> Åpen
<i>Eventuelle Merknader:</i> Vi tillater at oppgaven kan publiseres.	

<i>Oppdragsgiver:</i> Sars International Centre for Marine Molecular Biology	<i>Oppdragsgivers referanse:</i> Jørgen Høyer
<i>Oppdragsgivers kontaktperson(er) (inkludert kontakinformasjon):</i> Jørgen Høyer – jorgen.hoyer@uib.no Marios Chatzigeorgiou – marios.chatzigeorgiou@uib.no Daniel Dondorp – danieldondorp@gmail.com	

Revisjon	Dato	Status	Utført av
0.10	14.01.22	Dokument opprettet. Kap. 1-3 lagt til.	Andreas, Dennis og Håvard
0.11	31.01.22	Forstudie, La inn vedlegg, ferdigstilt.	Håvard
0.20	03.02.22	Rapport opprettet, lagt inn deler av forstudiet.	Andreas, Dennis og Håvard
0.21	01.03.22	Lagt inn innhold og kapitler.	Håvard
0.22	04.04.22	Restrukturert innhold for et mer oversiktlig inntrykk.	Håvard
0.30	10.05.22	Første utkast.	Andreas, Dennis og Håvard
0.31	18.05.22	Endret kapittel 5 testing.	Andreas, Dennis og Håvard
0.40	20.05.22	Andre utkast.	Andreas, Dennis og Håvard
0.41	23.05.22	Fikset koblinger i appendiks.	Håvard
0.42	23.05.22	Fikset referanser.	Andreas, Dennis og Håvard
1.00	26.05.22	Endelig rapport.	Andreas, Dennis og Håvard

Forord

Denne rapporten beskriver arbeidet som er gjort i forbindelse med bacheloroppgaven *BO22EB-04 Oppgradering av datainnsamlingsystem for forskning på Ciona Intestinalis* våren 2022. Prosjektet er gjennomført som et avsluttende emne på studiet Automatisering med Robotikk ved Høyskolen på Vestlandet, Campus Bergen.

Arbeidet har gått inn på mange av fagområdene dekket av emnene på studiet som blant annet kretsteknikk, digitalteknikk, programmering og mikrokontrollere. Prosjektet har både vært lærerikt og spennende, og har i tillegg bydd på mange utfordringer som ikke hadde vært mulig å løse uten gode veiledere som har hjulpet oss i riktig retning.

En takk til Sars Internasjonale Senter for Marin Molekylærbiologi og Chatzigeorgiou Group for tillit til gjennomføring av oppgaven. Spesielt må oppdragsgiver Jørgen Høyer og Marios Chatzigeorgiou takkes for tett samarbeid og veiledning gjennom hele prosjektet, og avdelingsingeniør ved Sars, Athira Athira for hjelp avslutningsvis i arbeidet. Stor takk til vår interne veiler Yngve Thodesen og Høyskolen på Vestlandet samt venner og familie for hjelp og motivasjon underveis i rapportskrivningen og forarbeidet.

Sammendrag

Forskningsgruppen Chatzigeorgiou Group benytter et datainnsamlingssystem for å forske på bevegelser og oppførsler til én Ciona Intestinalis når den blir utsatt for ytre påvirkning. Arbeidet tilknyttet prosjektet innebar å oppgradere dette systemet for å kunne effektivisere forskningen til gruppen.

Det har resultert i et nytt system som gjør opptak av seks Ciona-er samtidig med et høyoppløsningskamera, og tilfører lik stimuli til alle. Alt av innstillinger og valg knyttet til forsøkene som skal gjennomføres kan velges ved hjelp av GUI-en fra det opprinnelige systemet. Programmet er også oppgradert slik at både det nye og gamle oppsettet kan kjøres på samme datamaskin samtidig, i tillegg til at feil og mangler fra originalprogrammet er rettet opp.

Den ferdige løsningen fungerer som den skal og ble brukt til å samle data for å trene opp en modell ved hjelp av DeepLabCut. Modellen ble til slutt funksjonell og fulgte bevegelsene til Ciona-ene til tross for at bildekvaliteten viste seg noe lavere enn forventet. Det ble derfor konkludert med at det er behov for et mer avansert kamera for å oppnå ønsket bildekvalitet, som igjen vil resultere i en mer nøyaktig modell i DeepLabCut.

Innhold

Dokumentkontroll	3
Forord.....	4
Sammendrag	5
1 Introduksjon	9
1.1 Oppdragsgiver.....	9
1.2 Problemstilling	9
1.3 Hovedidé for løsningsforslag	11
2 Kravspesifikasjon.....	12
3 Analyse av problemet.....	13
3.1 Utforming av mulige løsninger.....	14
3.1.1 Alternativer til det fysiske oppsettet	14
3.1.2 Alternativer til programvare.....	15
3.2 Vurderinger i forhold til verktøy og HW/SW-komponenter	15
3.2.1 HW/SW-Komponenter.....	15
3.2.2 Digitale verktøy.....	16
3.2.3 Fysiske verktøy	16
3.3 Konklusjon.....	17
3.4 Beskrivelse av tenkt løsning	18
4 Realisering av valgt løsning	21
4.1 Systemets sammenheng.....	21
4.2 Fysisk oppsett	22
4.2.1 Rammeverk.....	22
4.2.2 3D-Printer	26
4.2.3 Materialer.....	27
4.2.4 Agarose.....	27
4.2.5 Modellering	28
4.2.6 Base	28
4.2.7 Fester til øvre rammeverk	29
4.2.8 Holder til IR-filter	29
4.2.9 Holder til lysbeskytter og feste til kamera.....	30
4.2.10 Kamera, linse og filter	30
4.3 Komponenter til krets.....	32
4.3.1 Arduino Nano	32

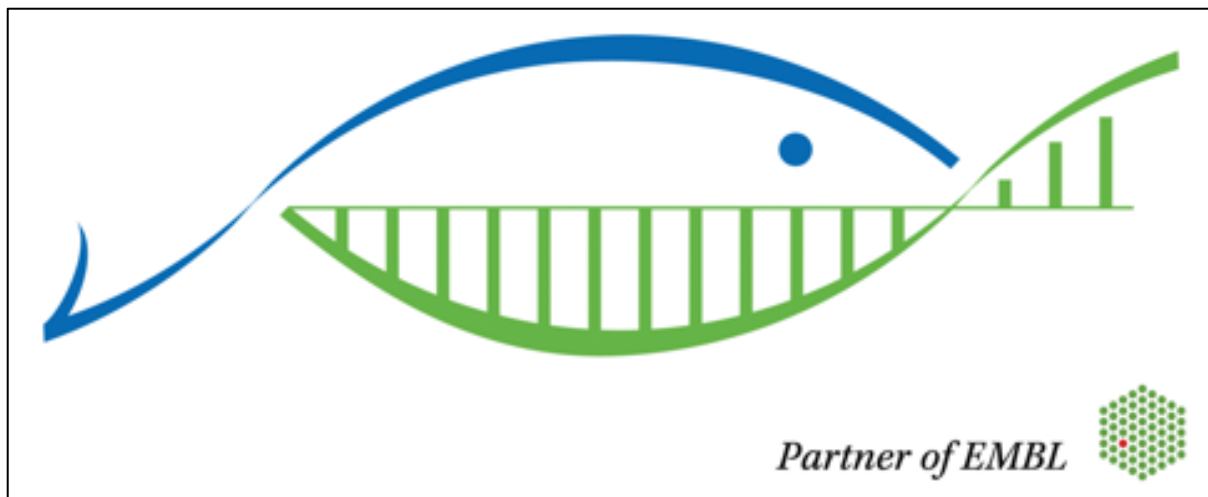
4.3.2	MOSFET	33
4.3.3	IR-LED	34
4.3.4	Lysstimuli 12 V 20 W Halogen Lyspære	35
4.3.5	Lysstimuli RGB NeoPixel Ring	35
4.3.6	Motor.....	36
4.3.7	Temperatursensor	36
4.3.8	Øvrige komponenter	37
4.4	Krets.....	38
4.4.1	Beskrivelse	38
4.4.2	Lodding	39
4.5	Programvare	43
4.5.1	Brukergrensesnitt (GUI).....	43
4.5.2	Programkode	47
4.5.3	Mikrokontrollerkode	48
4.5.4	Kommunikasjon	48
4.5.5	Main.py.....	48
4.6	DeepLabCut.....	49
4.6.1	Invertering	49
4.6.2	Markering	50
4.6.3	Trening.....	51
5	Testing.....	53
5.1	Test av bildekvalitet	53
5.2	Test av maskinvare.....	53
5.3	Test av programvare	54
5.4	Siste systemtest	54
5.5	Test av DLC.....	55
5.5.1	Modell 1.....	55
5.5.2	Modell 2.....	55
6	Diskusjon	57
6.1	Forslag til forbedringer	57
6.1.1	Bildekvalitet	57
6.1.2	Trening av DLC-modell.....	57
6.1.3	Kjøring av programvare	57
7	Konklusjon.....	59

Referanser	60
Appendiks A Forkortelser og ordforklaringer	63
Appendiks B Prosjektledelse og styring	64
B.1 Prosjektorganisasjon	64
B.2 Risikoliste	64
B.3 Fremdriftsplan	64
Appendiks C Brukerdokumentasjon	68
C.1 Brukerinstruksjoner	68
C.2 Drifts- og vedlikeholdsdokumentasjon	70
C.3 Vanlige Feil	70
Appendiks D Øvrige diagrammer og lister	72
D.1 Figurliste	72
D.2 Tabelliste	74
Appendiks E Kildekode, skjemadesign og komponentliste	75
E.1 Kildekode	75
E.2 Skjemadesign	75
E.3 Komponentliste	76
Appendiks F Linker til opptak	77
F.1 Modell 1	77
F.2 Modell 2	77

1 Introduksjon

1.1 Oppdragsgiver

Oppgavens oppdragsgiver er en forskningsgruppe ved Sars Internasjonale Senter for Marin Molekylærbiologi (Sars/Sarscenteret). Sars ble etablert i 1997, og har siden 2003 vært samarbeidspartner med det Europeiske Molekylærbiologiske Laboratoriet (EMBL). Senteret holder til på Høyteknologisenteret på Florida i Bergen og har tilknytning til Universitetet i Bergen (UiB). For tiden består Sarscenteret av åtte forskningsgrupper som studerer biologiske prosesser i marine organismer med utvikling og evolusjon som hovedfokus. [1] Forskingen ved Sarscenteret er finansiert av Norges Forskningsråd (NFR), Kunnskapsdepartementet og Universitet i Bergen. [2]



Figur 1-1 Sars Logo (Foto/ill.: Sars Center)

Oppgaven er formulert av forskningsgruppen Chatzigeorgiou Group, også kalt S13 [3]. De forsker blant annet på mekanismene nervesystemet benytter, fra det blir utsatt for ytre påvirkning, til det skapes fysisk bevegelse og oppførsel.

1.2 Problemstilling

For å forske på disse mekanismene har S13 utviklet et system for å spore bevegelser og aktivitet til et grønnsekkdyr, også kalt Ciona Intestinalis (CI) [4], i larvestadiet. Systemet benytter et høyoppløsningskamera til å overvåke og samle data av en enkelt CI, når den blir utsatt for lysstimuli, mekanisk stimuli eller ulike kjemikalier. På den måten er målet å identifisere de delene av nervesystemet som aktiveres i overgangen fra påvirkning til reaksjon [3].

Det eksisterende systemet benytter et grafisk brukergrensesnitt (GUI) for å sette opp forsøket som skal utføres. Der velges det type, varighet og styrke på stimuli som skal tilføres, i tillegg til lagringsplass for opptaket slik at det automatisk blir lagret på ønsket sted.

Oppgavens problemstilling består av å videreutvikle og oppgradere dette systemet slik at forskningen ved Sarsenteret effektiviseres. Det nye systemet skal, i likhet med det eksisterende systemet, kunne ta opptak og tilføre stimuli. Hovedforskjellen er at det skal oppgraderes slik at det kan filmes opp til seks Ciona samtidig med ett kamera. Det opprinnelige systemet kunne kun filme ett dyr med ett kamera av gangen.

Etter at det er samlet tilstrekkelig med opptak fra det nye systemet, skal denne dataen brukes til å trene opp en modell ved hjelp av dyplæringsrammeverket DeepLabCut (DLC) [5]. Hensikten med dette er at S13 skal kunne sammenlikne sine observasjoner med den trente modellen for å kunne estimere oppførselen og bevegelsene til hver CI, basert på stimuli de ble utsatt for.



Figur 1-2 Eksisterende løsning med tre oppsett. Tre kameraer filmer én CI hver. [6]

På grunn av dens unike egenskaper er CI, eller grønnsekkdyr, vanlig å forske på. Med en lengde på rundt en halv millimeter i larvestadiet, en utviklet hjerne og nervesystem, og i tillegg en reproduksjonstid på rundt 36 timer [7], er den godt egnet med tanke på håndtering og tilgang.

1.3 Hovedidé for løsningsforslag

Hovedidéen for løsningen vil være å bygge et tilsvarende oppsett med mulighet til å filme ovenfra. Kameraet vil filme ned på en base med plass til et reaksjonsbrett med seks dyr. Mekanisk stimuli vil bli tilført via to motorer som er festet til basen, mens lysstimuli vil bli tilført via lysemitterende dioder (LED) plassert ved kameraet. Begge blir styrt av en mikrokontroller, hvor innstillinger i form av type, varighet, og styrke blir valgt gjennom det eksisterende brukergrensesnittet. Endringer for å etterkomme oppgraderingen fra en større base kan være nødvendig, både for den fysiske og digitale løsningen.

2 Kravspesifikasjon

Kravspesifikasjonene til systemet er avtalt med eksternt veileder ved Sarsenteret. Gjennom flere møter er det formulert en rekke krav etter oppdragsgivers ønsker, som skal sørge for et best mulig resultat for prosjektet. Kravene som er avtalt, omhandler både det fysiske oppsettet og den digitale løsningen, men òg tilleggskrav for dokumentasjon av det ferdige resultatet. Se Tabell 2-1 for konkretiserte krav med forklaring.

For å filme grønnsekkdyrene skal det benyttes ett kamera som skal filme ett reaksjonsbrett. Dette brettet har seks brønner og er en standard som S13 benytter ved forskning. I hver brønn er det kun én CI som skal analyseres. Reaksjonsbrettet skal erstatte den enkle petriskålen med én brønn fra det tidligere systemet. Se Figur 2-1.

Stimulien som tilføres skal være lysstimuli og mekanisk stimuli hvor det benyttes henholdsvis LED-er og motorer for dette. Tilført stimuli skal påvirke alle seks CI-ene likt, og ved endt prosjekt skal hele det nye fysiske oppsettet passe inn i en inkubator på Sarsenteret for temperaturkontroll.

På grunn av de fysiske oppgraderingene og endringene må den eksisterende programkoden justeres. Det skal derfor videreutvikles eller utvikles ny kode for å etterkomme dette. Alle tekniske tegninger, komponenter og kildekode skal i tillegg være tilgjengelig på GitHub [8] e.l. etter prosjektslutt.

Dataen som samles fra det nye systemet, skal brukes til å trene opp en modell ved hjelp av dyplæringsrammeverket DLC.

Krav	Beskrivelse
Reaksjonsbrett	Skal benyttes et reaksjonsbrett med seks brønner. Én brønn per CI.
Stimuli	Skal kunne tilføres lik stimuli ved hjelp av mikrokontroller med LED-er og motor.
Størrelse	Skal passe inn i en inkubator for temperaturkontroll.
Programmering	Skal videreutvikle ev. utvikle kode for å passe nytt system.
Trening	Skal trene opp en modell ved bruk av dyplæringsrammeverket DeepLabCut.
Tilleggskrav	
Arkitekturdokumenter	Skal legge ved tekniske tegninger av krets, samt komponentliste.
Dokumentasjon	Skal benytte GitHub e.l. med dokumentasjon av kildekode.

Tabell 2-1 Krav og tilleggskrav.



Figur 2-1 Petriskål [9] og reaksjonsbrett med seks brønner [10].

3 Analyse av problemet

Under dette kapittelet skal det analyseres og beskrives utfordringer knyttet til løsningen. Det vil gjelde problemer for å etterkomme kravspesifikasjonene beskrevet i kapittel 2, men også utfordringer utover dette. For å dekke hovedutfordringene ved prosjektet dekomponeres analysen ved å ta utgangspunkt i de forskjellige kravene, og deretter se på utfordringer knyttet til hver av disse.

Oppgraderingen fra en enkelt petriskål til reaksjonsbrettet skaper utfordringer i flere deler av løsningen. Siden reaksjonsbrettet har et areal som er over seks ganger så stort som petriskålen fra det originale oppsettet, må plassering i forhold til kamera tenkes gjennom for å få plass til hele brettet i bildet. Det er også viktig med godt fokus og lysforhold for at bildekvaliteten skal bli så god som mulig, siden god kvalitet og kontrast mellom CI og bakgrunn er kritisk for at dataene skal kunne analyseres i etterkant. Forskning viser at grønnsekkdyrets svømmemønster blir påvirket av dagslys og LED-er, dermed vil det være hensiktsmessig at belysningen består av infrarødt (IR) lys som har en bølgelengde lenger enn for spekteret til synlig lys. [11]

Lik påvirkning av stimuli på hele reaksjonsbrettet er svært viktig for kvaliteten på forsøkene som skal gjennomføres av forskningsgruppen Chatzigeorgiou Group. Plassering av motorer og LED-er i forhold til reaksjonsbrettet vil påvirke hvorvidt det er mulig å oppnå lik distribusjon av stimuli. I tillegg til plasseringen, må stimuliene kunne komme med valgt styrke, varighet og tidspunkt ved bruk av et grafisk brukergrensesnitt. Samhandling gjennom GUI-en og en mikrokontroller skal sammen sørge for at valgte stimuli-innstillinger tilføres.

Hele det nye fysiske oppsettet skal passe inn i en inkubator på Sarsenteret som skal holde temperaturen lik for alle eksperimenter S13 skal gjennomføre. Inkubatoren er av typen Sanyo MPR-311D med innvendige mål på 720x1,435x350 mm [12] hvor det eksisterende systemet allerede er plassert. Temperaturen skal registreres og logges ved hjelp av sensorer koblet til mikrokontrolleren, i tilfelle avvik. Som et tillegg må være en mulighet til å feste kamera og komponenter på en enkel måte, samtidig som det sitter godt fast for at utskiftninger eller reparasjoner skal være effektivt.

For at det nye systemet skal virke uten feil er det nødvendig å videreutvikle eksisterende kode og utvikle noe ny kode. Utgangspunktet for programvaren er hentet fra dokumentasjonen fra det tidligere prosjektet ImMobilize [13] fra GitHub. Dette oppbevaringsstedet (EN. Repository/ «repo») inneholder flere filer med kode tilhørende den eksisterende programvaren, men òg bilder av systemet samt filer brukt for 3D-printing av deler. Blant programkodefilene er det noen mindre feil som gjør at deler av systemet ikke fungerer på best mulig måte. For at nytt oppsett med programkode skal virke, må noen av disse filene justeres noe.

Etter å ha finjustert kamerafokus og bildekvalitet skal det samles data for å kunne trene opp en modell ved hjelp av DLC. For å sikre at modellen fanger opp CI-enes bevegelser er det viktig å ha tilstrekkelig med data og være nøyaktig med prosessen i forkant av treningen. Det er flere faktorer som skal til for at en modell blir god, men både presisjon og tålmodighet under markering vil være viktig for treningens resultat.

3.1 Utforming av mulige løsninger

For å se på de mulige løsningene settes problemstillingene fra analysen i kapittel 3 opp mot kravene som er formulert i Tabell 2-1, i tillegg til å beskrive andre deler som er nødvendig. Siden det er mange måter å etterkomme disse kravene, blir det naturlig å se på løsninger for hvert av dem og på den måten løse problemstillingen på en ryddig og strukturert måte. Det skal tas valg i forhold til design av oppsettet, plassering av stimuli, plassering av IR-lys, mikrokontroller, programvare og hvordan programmet skal kjøres.

Oppdragsgiver har mye utstyr og komponenter liggende som er naturlig å benytte seg av. Dette gjelder mindre komponenter som transistorer, motstander og ledninger, men også mikrokontrollere og programvare. Siden oppgaven går ut på å oppgradere et eksisterende system, er det flere valg som er tatt med utgangspunkt i dette systemet. Forslag som beskrives under har med de generelle digitale og fysiske løsningene av prosjektet å gjøre. Når det gjelder detaljer til det fysiske oppsettet og den digitale løsningen blir disse beskrevet i kapittel 3.3 Konklusjon og kapittel 3.4 Beskrivelse av tenkt løsning.

3.1.1 Alternativer til det fysiske oppsettet

Det fysiske oppsettet skal være et rammeverk hvor kamera, reaksjonsbrett og andre komponenter kan festes fast til. Størrelsen på den ferdige fysiske riggen skal være slik at den passer inn i inkubatoren på Sarssenteret.

Alternativ 1 – Oppgradere eksisterende oppsett

Det første alternativet vil bestå av å benytte hele det opprinnelige oppsettet i inkubatoren på Sarssenteret, men å bytte ut nødvendige deler og justere innstillinger for å etterkomme kravene. Denne oppgraderingen innebærer at det gamle systemet må fjernes til fordel for den nye og noe større riggen.

Alternativ 2 – Bygge nytt oppsett

Et annet alternativ er å bygge et helt nytt oppsett fra bunnen. Det vil føre med seg å kutte materialer og skaffe alle deler til et nytt oppsett, men samtidig øker det valgfriheten med tanke på materialear og eventuelle endringer for å gjøre oppsettet bedre og enklere.

3.1.2 Alternativer til programvare

Programvaren vil være avgjørende for funksjonaliteten til den totale løsningen. Sammen med de fysiske komponentene skal programkoden sørge for enkle valg av innstillinger og tilføring av stimuli gjennom et brukergrensesnitt som skal være enkelt å bruke.

Alternativ 1 – Justere eksisterende programvare

Første alternativ vil bestå av å bruke den eksisterende programkode, men justere for å rette opp feil og mangler. Denne komplekse koden er skrevet i Python3 og er fordelt på fem forskjellige kodefiler i tillegg til én fil for kode til mikrokontroller som til sammen utgjør ImMobilize.

Alternativ 2 – Utvikle helt ny programvare

Alternativ nummer to er å utvikle helt ny programkode fra bunnen av i programmeringsspråket C# (C-sharp) [14]. Dette gjør det mulig å utvikle en ny kode som er tilpasset et nytt oppsett med en unik og intuitiv GUI med gruppens eget preg.

3.2 Vurderinger i forhold til verktøy og HW/SW-komponenter

For å etterkomme kravspesifikasjonene knyttet til den ferdige totale løsningen må det benyttes en rekke fysiske og digitale verktøy og komponenter. Valg rundt hvilket utstyr som egner seg best faller på tilgang, gruppens erfaring og hvor intuitive de er i bruk. Ettersom S13 har laget og jobbet med det tidligere oppsettet, finnes det en del ekstradelere og komponenter som kan benyttes til den nye løsningen.

3.2.1 HW/SW-Komponenter

Valget av mikrokontroller for å tilføre stimuli og IR-lys falt på Arduino Nano. Arduino benytter en noe forenklet versjon av programmeringsspråket C++ [15], og både enheten og tilhørende integrerte utviklingsmiljø (integrated development environment, IDE) [16] har gruppen erfaring med gjennom egne prosjekter og tidligere emner på studiet. Et annet alternativ er å bruke en mikroprosessor av typen Raspberry Pi [17] da denne har langt bedre ytelse og flere valgmuligheter. Enheten har blant annet støtte til programmeringsspråkene C++ og Python, men langt flere språk kan lastes ned om det skulle være nødvendig. Selv om Arduino er ment for læring og hobbyprosjekter, har den mer enn nok datakraft for oppgaven ettersom det kun skal styres et par motorer, LED-er og temperatursensorer. Det vil derfor være hensiktsmessig å benytte denne mikrokontrolleren istedenfor mikroprosessoren.

Kameraet som S13 har og som skal benyttes i prosjektet er av typen DMK-33UP1300 usb3 [18]. Dette er et høyoppløsningskamera som skal ta opptak av alle seks CI-ene i larvestadiet, mens stimuli blir tilført. Reaksjonsbrettet som er nevnt tidligere er såpass bredt det trengs en linse som gir et vidt nok

synsfelt for at hele brettet skal komme i bildet. Dette må likevel ikke komme på bekostning av bildekvaliteten som gjør at avstanden mellom reaksjonsbrett og kamera må være så liten som mulig for å sikre god oppløsning.

3.2.2 Digitale verktøy

For utvikling av spesialdelene skal det benyttes 3D-printer og delene som skal skrives ut må først designes digitalt. Ved bruk av tegneprogrammer som Fusion 360 [19], SketchUp [20] eller TinkerCAD [21] kan det tegnes detaljerte spesialtilpassede 3D-modeller. Det finnes flere gode programmer for modellering, men ettersom bachelorgruppen har mest erfaring med Fusion 360 gjennom tidligere emner på studiet, falt valget på dette programmet.

Det må brukes en 3D-printer for å skrive ut de forskjellige spesialdelene oppsettet trenger og som er designet i Fusion 360. Chatzigeorgiou Group disponerer en 3D-printer på sine lokaler på Sarsenteret, men dette er en noe utdatert printer av lav kvalitet i forhold til nyere printere. Valget falt derfor på å bruke printerne som Læringslabben [22] på Høyskolen på Vestlandet Campus Kronstad disponerer. De har flere printere av ulik kvalitet, men bachelorgruppen falt på en Original Prusa i3 MK3S [23] som har vist seg pålitelig og skriver ut med relativt høy hastighet og kvalitet.

For utvikling av kode er det gunstig at alle på bachelorgruppen har tilgang og redigeringsmulighet, så ved hjelp av GitHub og et eget oppbevaringssted, kan man enkelt jobbe sammen med prosjektet. Gruppen har forskjellig erfaring med verktøyet, der noen er erfarne og andre har ikke brukt det før, men totalt sett vil nyttheten av GitHub veie opp for den tiden det tar å sette seg inn i. Her vil alle ha redigeringsmulighet og alle kan se gjennom endringene til hverandre slik at alle får med seg hvilke oppdateringer som foregår. Selve kodeskrivingen vil skje i Visual Studio Code [24] som er et program som tillater kodeutvikling i mange forskjellige språk på en oversiktlig og strukturert måte.

Det endelige programmet skal kjøres på en ekstern datamaskin hvor både mikrokontroller og kamera skal kobles til. GUI-en skal kunne brukes og opptak skal i tillegg kunne lagres på denne datamaskinen. ImMobilize, det eksisterende systemet, benytter en stasjonær datamaskin for å styre den tilhørende GUI-en. Ved å koble til enda et system til denne datamaskinen kan man styre både det gamle og det nye systemet fra samme PC samtidig. Et annet alternativ er å ha en egen datamaskin for det nye oppsettet som kan plasseres ved siden av den datamaskinen ImMobilize nå benytter. På den måten kan man kjøre nytt og gammelt system parallelt uten at det går ut over ytelsen om man kun benytter én enkelt PC.

3.2.3 Fysiske verktøy

Når det kommer til de fysiske delene av oppsettet som for eksempel rammeverk og krets, må det brukes en rekke verktøy for måling, kutting og montering. Det må benyttes sag eller baufil for å kappe bygningsrammer i riktig lengde. Måling av lengre lengder benyttes målebånd eller meterstokk, for mindre lengder som for eksempel diametere og bredder kan det benyttes et skyvelære for økt presisjon og nøyaktige målinger. Disse rammene festes sammen med hjørnebeslag og skruer med muttere ved hjelp av en drill, skrutrekker eller unbrakonøkkel og skiftenøkkel. For å sikre nøyaktighet når rammen festes i inkubatoren benyttes også et vater for å sørge for at alt står rett. Siden kameraet skal filme rett ovenfra kan man være sikker på at kamera og base står helt rett i forhold til hverandre.

For montering av kretsen brukes en loddebolt og loddetinn som skal feste komponentene sammen på et eksperimentkort. Dette vil være en mer permanent løsning enn å montere komponentene på et koblingsbrett uten bruk av loddetinn. Da oppsettet skal brukes til forskning i fremtiden, er det gunstig å lage en immobil løsning som holder.

3.3 Konklusjon

I dette kapittelet blir de konkrete valgene av drøftingene over, beskrevet. Fra kapittel 3.1.1 falt valget på alternativ 2, å bygge et helt nytt oppsett fra bunnen. Dette gjør det mulig å finjustere oppsettet etter kravene formulert tidligere, i tillegg til at det eksisterende systemet fortsatt fungerer fint. Ettersom at inkubatoren ved Sarssenteret er såpass stor, vil det være mer enn nok plass til å bygge det nye oppsettet i underkant av det gamle. Ved å bygge rammeverket fra bunnen av gir det også økt valgfrihet til design og oppsett som kan være godt egnet med oppgraderingene som skal gjøres.

Når det kommer til alternativene for programvare fra kapittel 3.1.2, falt valget på alternativ 1, å justere den eksisterende programkoden. Denne koden er skrevet i Python og fungerer fint til det eksisterende oppsettet og med til sammen flere tusen linjer med programkode, tas det utgangspunkt i denne og endrer for å etterkomme kravene for utvidelse. Den eksisterende koden er noe uoversiktlig og har enkelte feil knyttet til motoren som sørger for mekanisk stimuli. Det må derfor utvikles noe ny kode og gjennomføres justeringer og tilpasninger. Programkoden skal til slutt sørge for at man kan gjøre et forsøk ved å velge stimuli og dens innstillinger. Etter at både type, lengde og styrke av stimuli er valgt, eventuelt en serie med flere typer stimuli, skal valgene tilførtes systemet og filmes.

Valget av kamera falt på hva S13 hadde brukt ved tidligere prosjekter og dermed hadde tilgjengelig. I tillegg til kamera hadde forskningsgruppen flere linser og avstandsstykker liggende, som alt er svært kostbart utstyr. Ved å teste forskjellige kombinasjoner av linse og avstandsstykke var det med en 25 mm linse med et 5 mm avstandsstykke som ga det beste resultatet. Avhengig av størrelsen på stykkene

flytter man fokuspunktet til kameraet nærmere eller lenger unna ettersom hva som er målet. Det er dette som gir høyest bildekvalitet i forhold til synsfeltet som trengs for å fange opp hele reaksjonsbrettet. Både kamera og mikrokontroller skal kobles til samme datamaskin, men for at kameraet skal virke må det også lastes ned drivere til kameraet på denne maskinen.

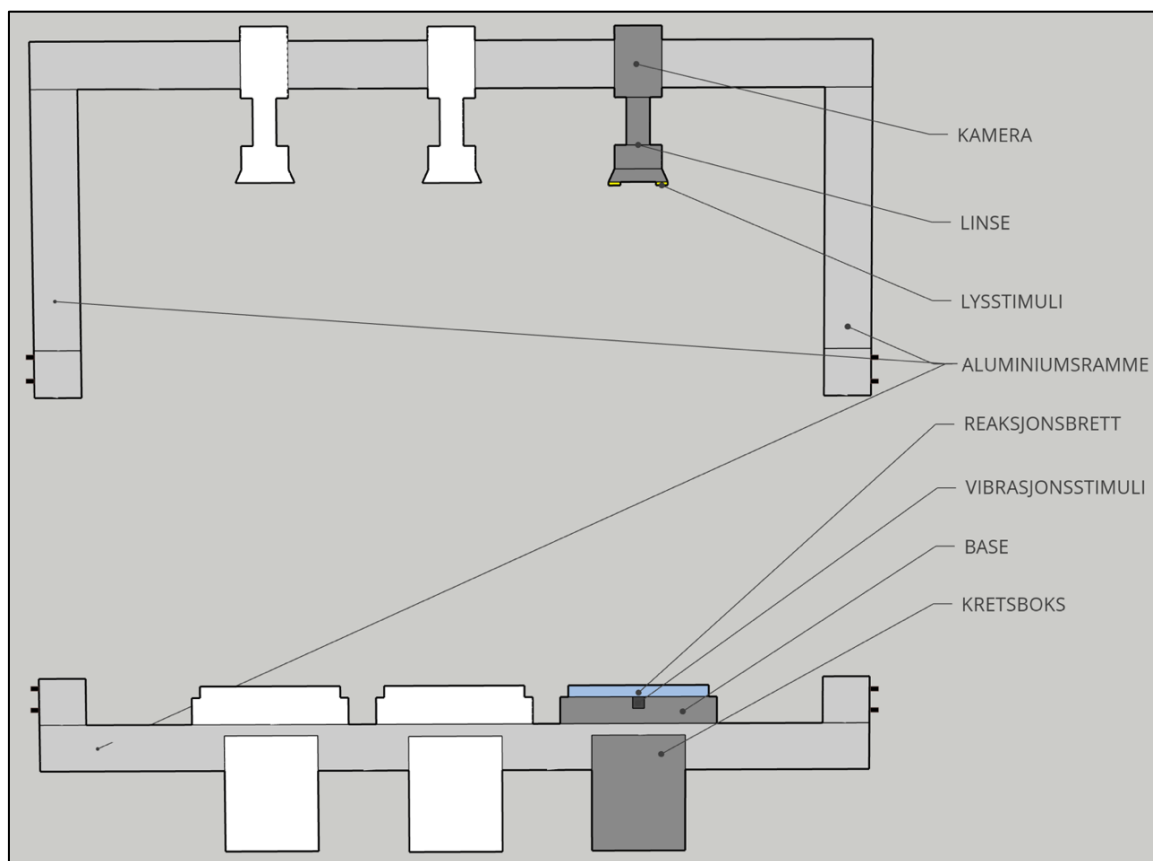
Løsningen på kjøring av programmet ble å koble det til det eksisterende systemet og justere programkoden for å tillate dette. Det vil altså være mulig å kjøre det nye og det gamle systemet fra samme PC på Sarsenteret samtidig. Den justerte koden og all brukerdokumentasjon vil være tilgjengelig for alle på GitHub etter endt prosjekt, for at både S13 og andre skal ha tilgang til å videreutvikle eller gjøre endringer [25].

3.4 Beskrivelse av tenkt løsning

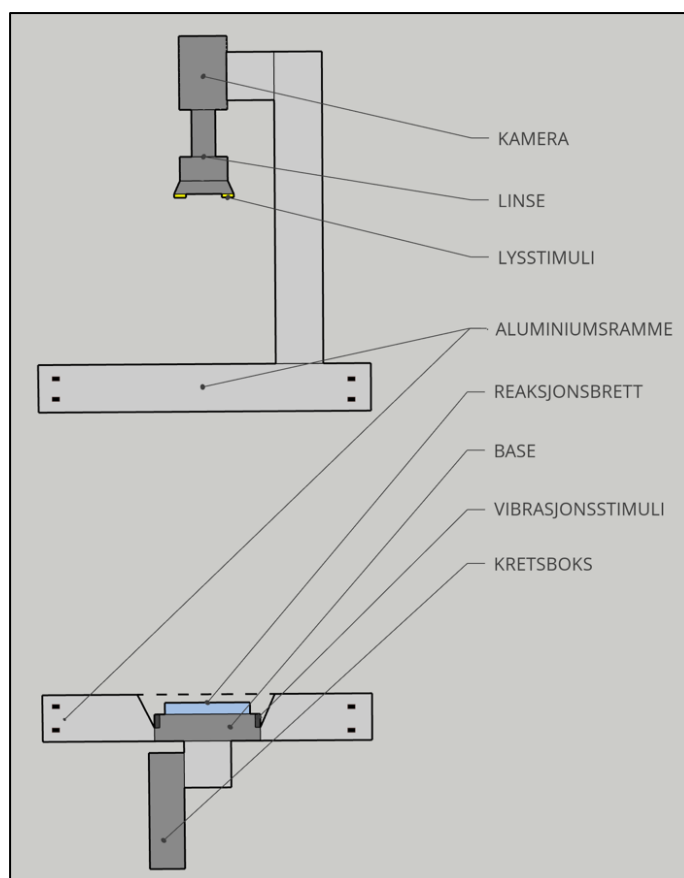
I dette kapitlet beskrives den tenkte fysiske løsningen generelt. Valgene vil sikre effektivitet i utviklingsfasen samt brukervennlighet og kvalitet av sluttproduktet. I kapittel 4 vil den fullstendige løsningen, både fysisk og digital, beskrives i detalj både når det gjelder gjennomføring og verktøy.

Fra Figur 3-1 ser man hvordan oppsettet er tenkt sett forfra, og fra Figur 3-2 fra siden. Det vil være to rammedeler i aluminium som er festet til innsiden av inkubatoren. Det er aluminiums-rammene markert med to sorte «tapper» fra Figur 3-1 og Figur 3-2) som skal festes i veggen. To lengre stenger på tvers vil holde hovedkomponentene for oppsettet. Den øverste delen vil holde kamera med tilhørende linse, i tillegg vil det være et IR-filter og LED-er for lysstimuli utenfor denne linsen. Reaksjonsbrettet vil være plassert på den nedre delen av rammen. For å sikre lik påvirkning av lys og mekanisk stimuli og like kamerainnstillinger for hvert forsøk, skal en base være festet i rammen. Man legger reaksjonsbrettet oppi denne basen for å sikre lik plassering hver gang. Kretsboxen med mikrokontrolleren vil også være festet i den nedre delen av rammen, i underkant av basen. Til denne boksen vil det kobles motorer for mekanisk stimuli, LED-er for lysstimuli, IR-LED-er og temperatursensorer.

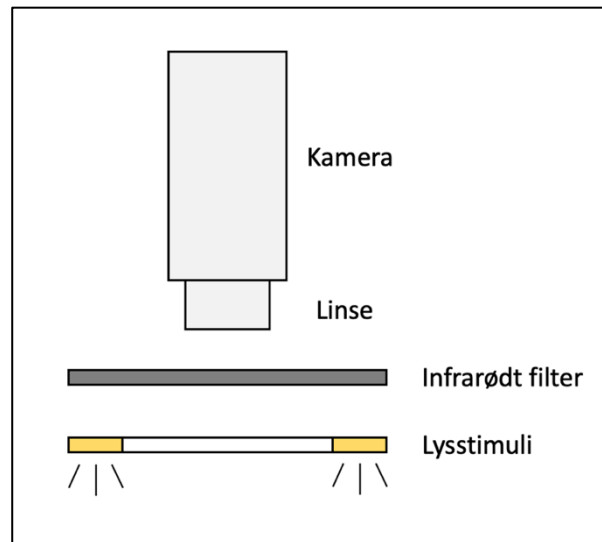
Når det kommer til kameraet vil dette kobles rett i datamaskinen, men med mulighet til å endre innstillinger via GUI-en. Fra Figur 3-3 ser man en illustrasjon av det fysiske kameraoppsettet. Både filter og LED for stimuli vil være festet i en 3D-printet innkapsling. Hele det fysiske oppsettet vil bli styrt av en GUI fra en datamaskin som nevnt tidligere. I kapittel 4.1, Systemets sammenheng, illustreres sammenhengene mellom det fysiske og digitale. En fullstendig komponentliste vil være tilgjengelig i appendiks E.3



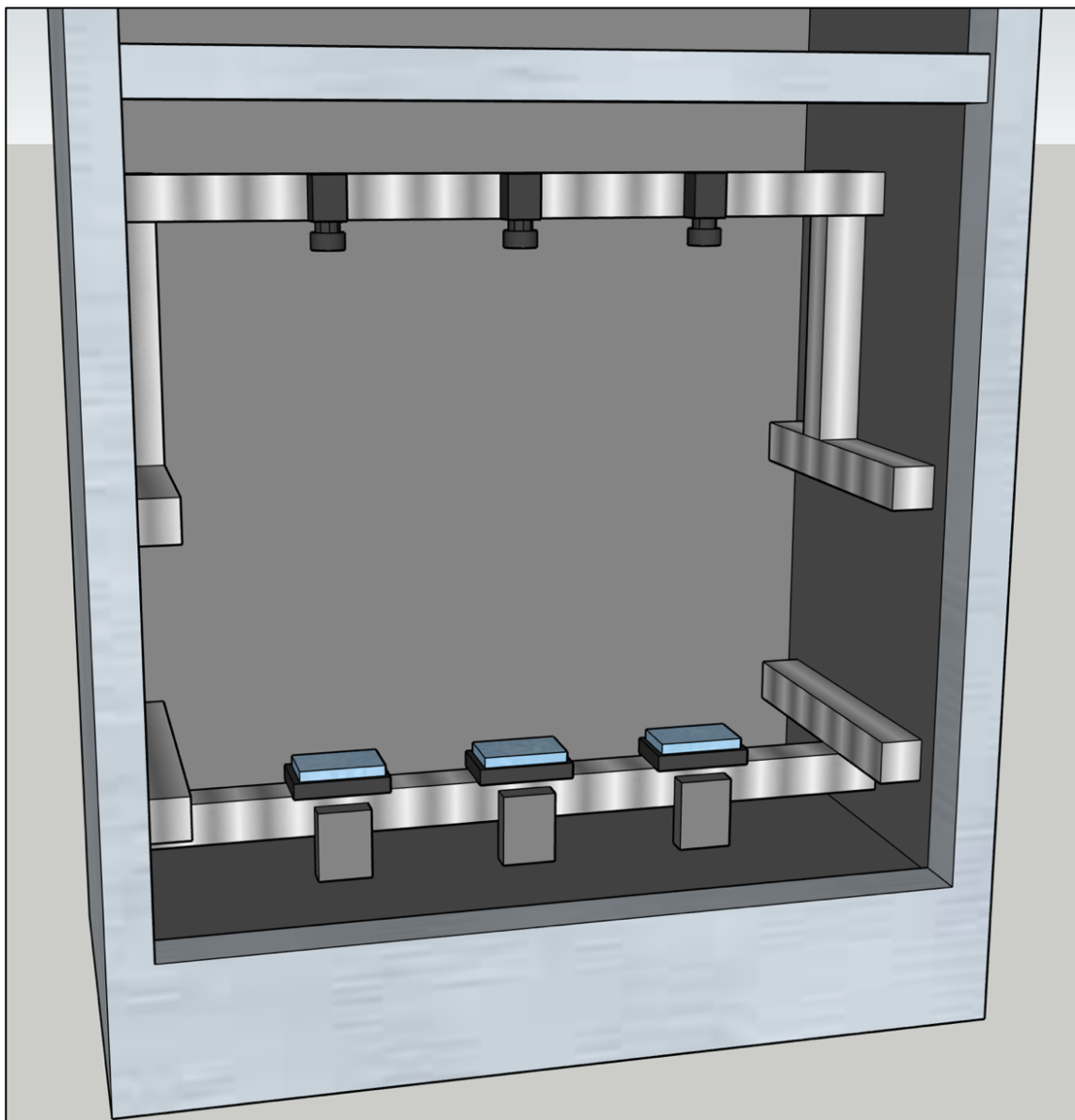
Figur 3-1 Skisse i 2D sett forfra.



Figur 3-2 Skisse i 2D sett fra siden.



Figur 3-3 Illustrasjon av kamera med filter



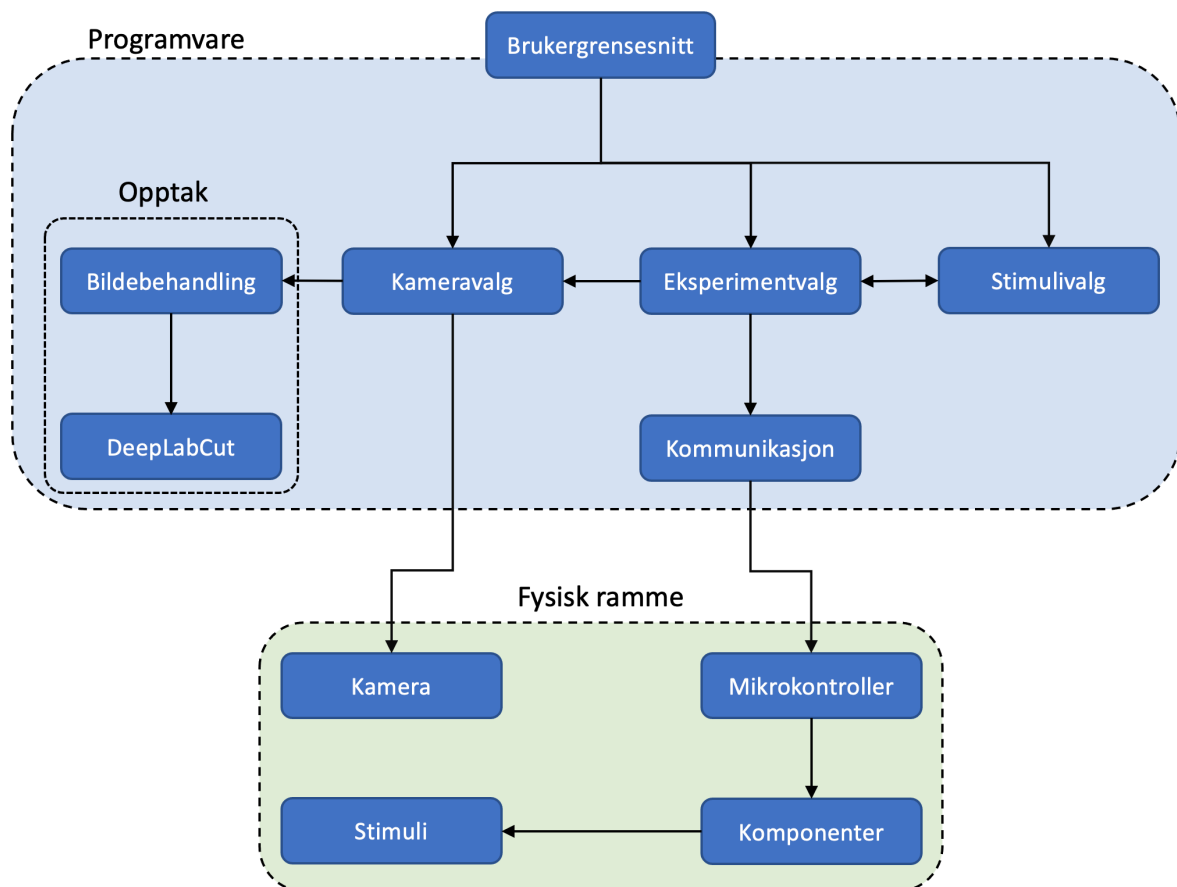
Figur 3-4 Skisse i 3D av oppsettet.

4 Realisering av valgt løsning

I dette kapittelet beskrives utviklingen av den totale løsningen med valgte verktøy i detalj. Det er i denne delen av oppgaven sammenhengen mellom det fysiske og digitale oppsettet tydeliggjøres. Først vil det fysiske oppsettet beskrives, deretter kretsen og til slutt den digitale løsningen.

4.1 Systemets sammenheng

Fra Figur 4-1 illustreres sammenhengen mellom systemets programvare og maskinvare. Prosjektet er en todelt oppgave med både fysiske og digitale utførelser som sammen skal kommunisere og brukes parallelt. Den digitale løsningen består av brukergrensesnitt, kommunikasjon mellom PC og mikrokontroller, og dyplæring ved hjelp av DLC. I GUI-en vil man kunne velge hvilket kamera som skal brukes, og hvordan eksperimentene med tilhørende stimuli skal være. Programvaren vil så kommunisere med en Arduino som tilfører valgte innstillinger til forsøket. Kameraet vil, med valgte kamerainnstillinger, ta opptak av forsøket og lagre det på et valgt sted på datamaskinen, deretter vil disse video-opptakene behandles og brukes i DLC. Detaljene i løsningen blir beskrevet i kapittel 4.2 til kapittel 4.6.



Figur 4-1 Sammenheng mellom programvare og maskinvare.

4.2 Fysisk oppsett

4.2.1 Rammeverk

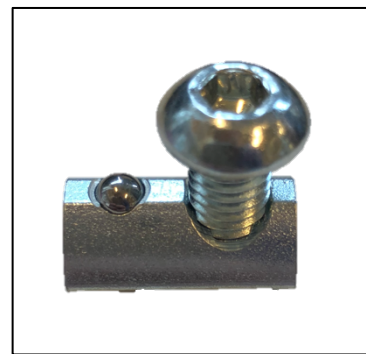
Som nevnt tidligere ble det valgt å bygge den fysiske løsningen fra bunnen i aluminium, slik som det opprinnelige oppsettet. Fordelen med byggerammene er at de kommer med spor som gjør montering og justering svært enkelt (Figur 4-4). Med rammene følger det med muttere og skruer (Figur 4-2) som passer i sporene og som enkelt kan festes godt sammen med hjørnebeslag (Figur 4-3). Rammen deles inn i to deler, en for kamera og en for basen. Fra Figur 3-1 og Figur 3-2 ser man illustrasjoner av rammeverket og som viser rammens plassering av kamera, base, reaksjonsbrett, stimuli og kretsboкс i forhold til hverandre.



Figur 4-3 Hjørnebeslag.



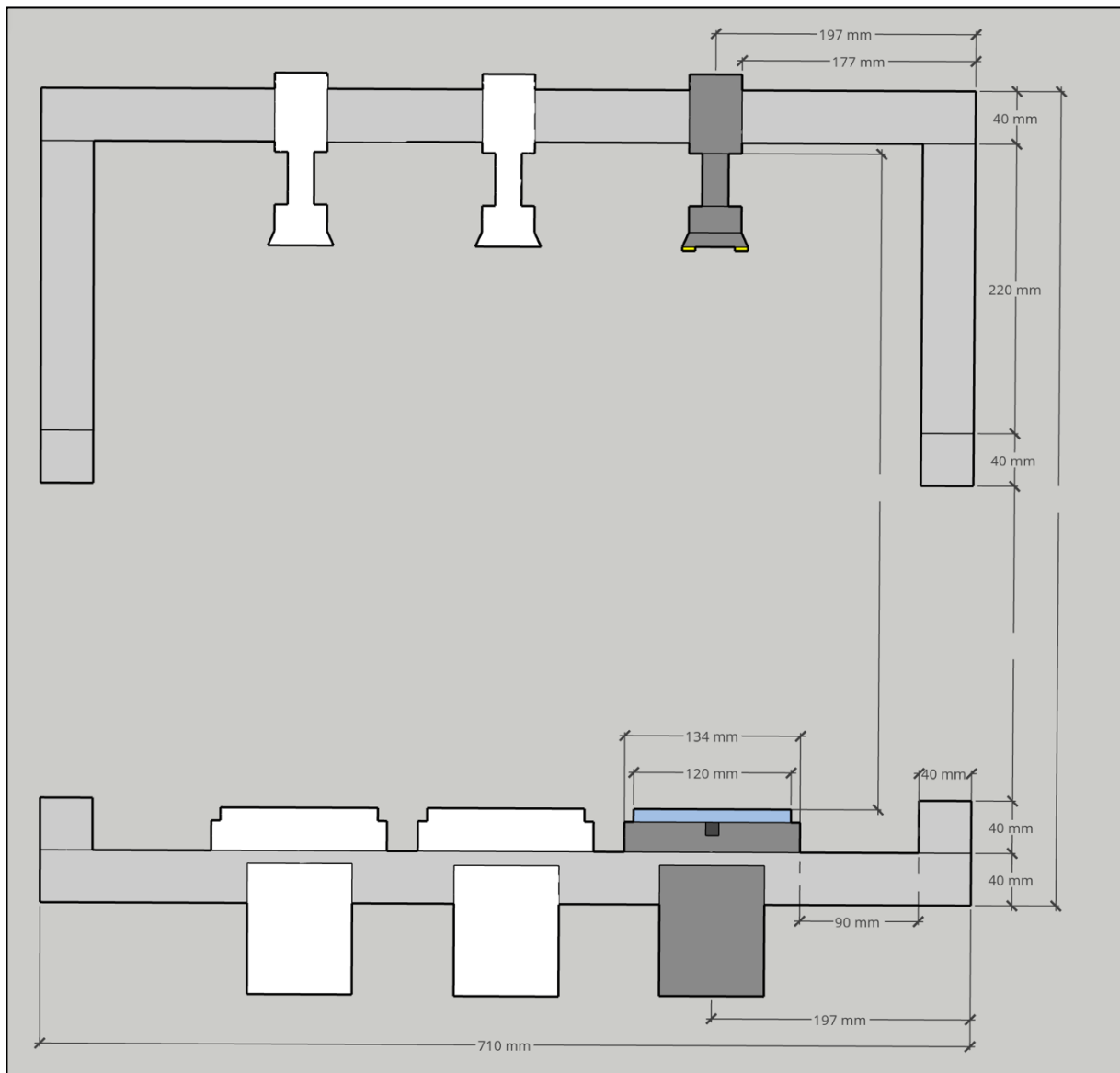
Figur 4-4 Aluminiums-ramme.



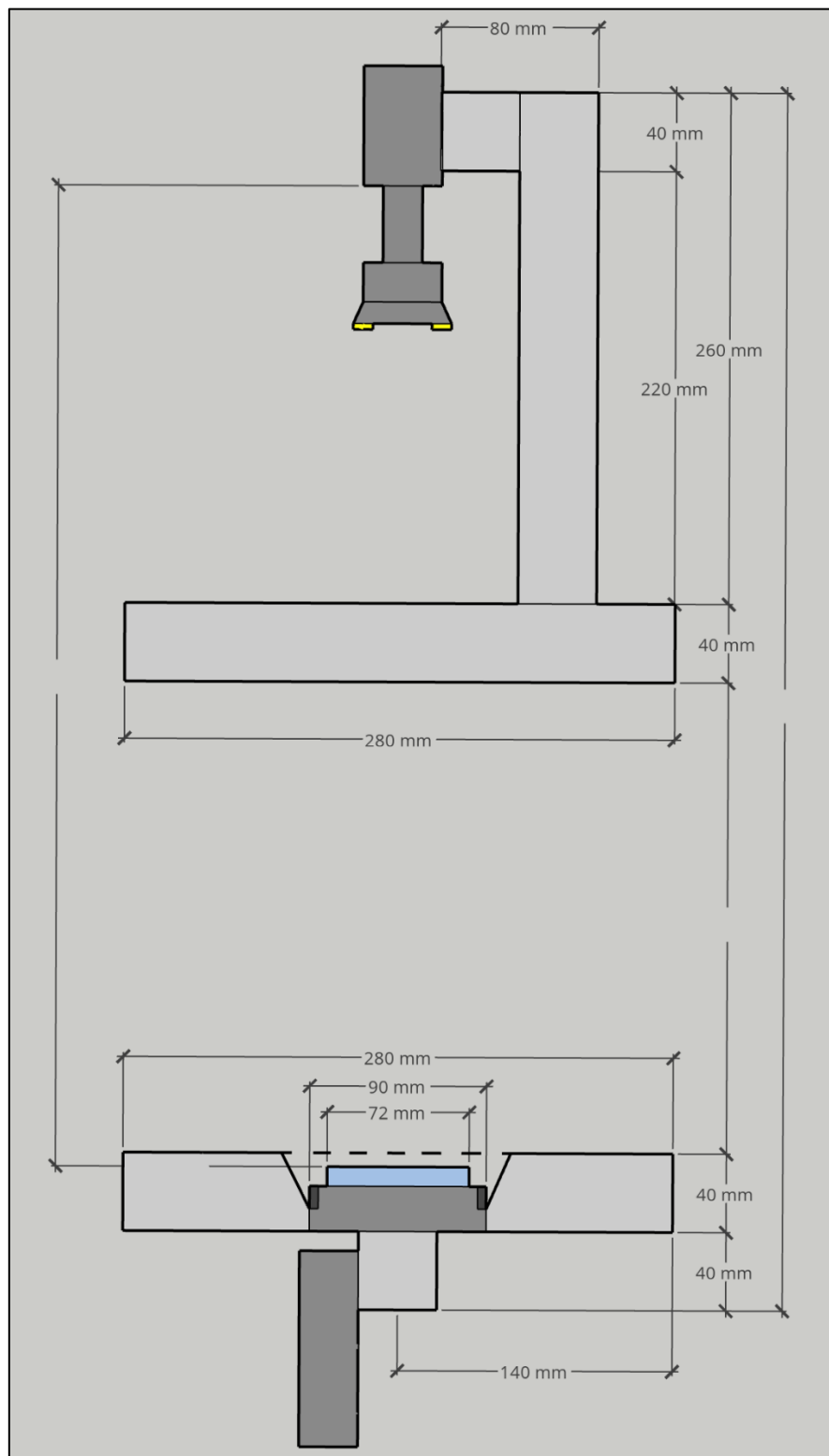
Figur 4-2 Skruer og mutter tilhørende rammen.

Som nevnt tidligere vil rammedelene bestå av to lengre stenger på tvers, med tilhørende festemuligheter til veggen på inkubatoren. Den øvre rammen vil i tillegg ha en vertikal justeringsmulighet om det er behov for en annen avstand mellom kamera og base. Mål for hele oppsettet er beskrevet i Figur 4-5 og Figur 4-6. Tverrsnittet til aluminiums-stengene er på 40x40 mm, men det er lengden på stengene som her er viktig. Avstanden mellom kamera og base er for å sette størrelsen til oppsettet i perspektiv. For den øvre rammen vil det i tillegg til kamera bli festet en lysbeskytter i rammen. Denne vil være et plastrør av typen polyvinylklorid (PVC) som går rundt kamera og nedover mot basen på samme måte som det eksisterende oppsettet fra Figur 1-2. Hensikten med dette er å blokkere ut dagslys siden dette kan påvirke forsøkene og resultatene ettersom CI-ene er lyssensitive.

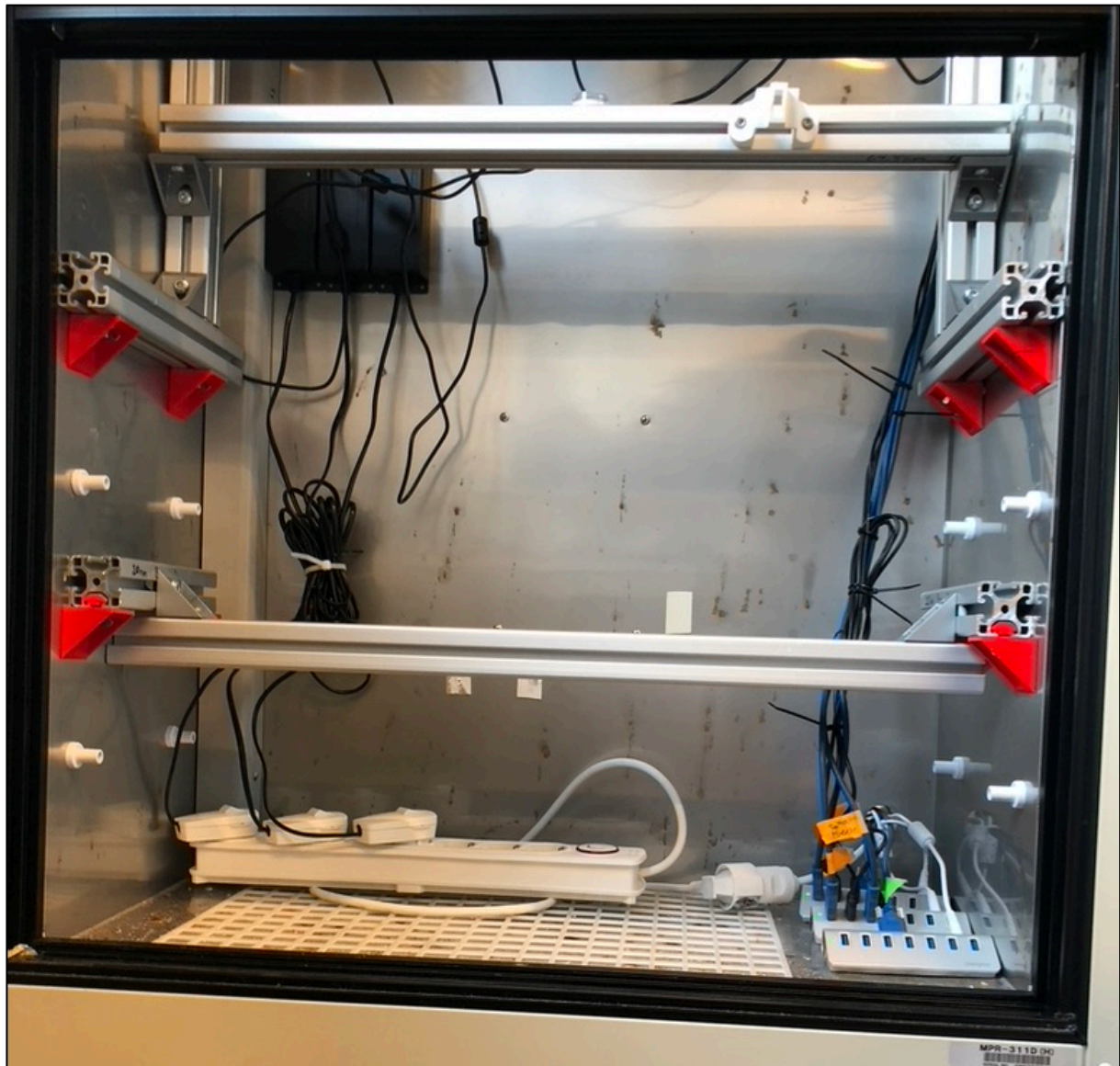
Det er totalt åtte rammedeler med en lengde på henholdsvis 2x71 cm, 2x26 cm og 4x28 cm fordelt på øvre og nedre ramme. S13 hadde aluminiums-stenger liggende som ble kuttet til med baufil der det var nødvendig. Lengdene på enkelte stenger var noe avvikende fra planen, men passet uten at det påvirket sluttresultatet og ble derfor brukt.



Figur 4-5 Skisse i 2D forfra med mål.



Figur 4-6 Skisse i 2D fra siden med mål.



Figur 4-7 Ferdigmontert rammeverk i inkubator.

Delene på 28 cm ble festet til veggen på inkubatoren og de andre delene til hverandre slik som figurene har vist tidligere, og de resterende stengene ble festet med hjørnebeslag og skruer. Fra Figur 4-7 ser man ferdig montert rammeverk i sin helhet der kabler i bakgrunnen tilhører det eksisterende systemet som er festet i overkant i samme i inkubator, men utenfor bildet. Figur 4-8 og Figur 4-9 viser montert rammeverk fra henholdsvis høyre og venstre side.



Figur 4-8 Rammeverk i inkubator. Venstre.



Figur 4-9 Rammeverk i inkubator. Høyre.

Som nevnt over blir det brukt et PVC-rør for å beskytte kameraet og CI-ene fra naturlig lys. Dette røret er 110 mm i diameter og blir festet rundt kameraet hengende ned mot selve basen ved bruk av et 3D-printet feste (Figur 4-19) og en skrue med muttere. Ganske likt som i Figur 1-2, men med en annen festemetode for røret.

4.2.2 3D-Printer

For prosjektet ble det som nevnt tidligere brukt en Original Prusa i3 MK3S. Printerens har en fleksibel stålplate som holdt 60°C hvor delene ble skrevet ut på. Denne platen har en egenskap som gjør at PLA-plast ikke fester seg permanent fast i selve platen, og er gunstig ettersom det var denne typen plast som ble brukt i prosjektet. Med et byggevolum på 250x210x210 mm var den mer enn stor nok til prosjektets deler, og det kunne i tillegg skrives ut flere deler samtidig.

Etter å ha designet delene må de kompiles ved hjelp av en PrusaSlicer, hvilket er en programvare som gjør det mulig for en 3D-printer å forstå hvordan den skal skrive ut en modell. Det er i denne typen programvare det kan velges ytterligere innstillinger som bla. skrivetemperatur, tetthet og støtter. For dette prosjektet ble det brukt en tetthet på 10% og en temperatur på 215°C. Siden systemet ikke blir påvirket av noe mekanisk stress fra omgivelsene var det ikke nødvendig med en høyere tetthet. Eneste

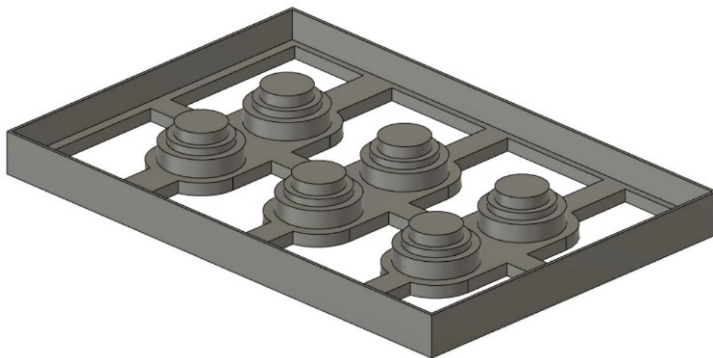
unntaket var platen som ligger mellom basen og reaksjonsbrettet. Her var det viktig med høy utfylling siden den skulle sørge for jevn belysning i basen. Med en tetthet på rundt 80% førte dette til en jevnere belysning enn med mindre tetthet. Resterende innstillinger var standardinnstillingene til printereren [26].

4.2.3 Materialer

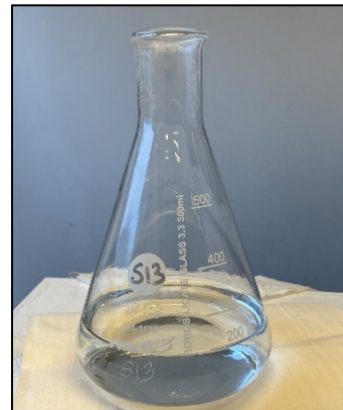
Ved 3D-printingen av basen til prosjektet ble det brukt PLA, som er et vanlig materiale for 3D-printing. Denne type PLA-plast var tilgjengelig på Læringslabben og er en lett og billig måte å produsere prototyper, gjøre justeringer og teste de spesialtilpassede delene til prosjektet. Miljøet som produktet vårt blir utsatt for kommer ikke til å ha mange eksterne påvirkninger som mekanisk stress eller store temperaturendringer. Dermed var det mulig å bruke denne type filament også i den avsluttende løsningen.

4.2.4 Agarose

I prosjektet var det nødvendig å bruke et stoff som heter agarose. Dette er et pulverlignende stoff som blandet med kunstig sjøvann og varmet i mikrobølgeovnen, stivner til en fast gele ved nedkjøling. Funksjonen til geléen er at den skal begrense svømmeområdet til dyrene mens de ble filmet, uten at dyrene setter seg fast i veggene. Det var noe forskningsgruppen hadde tidligere erfaring med at dyrene gjorde.



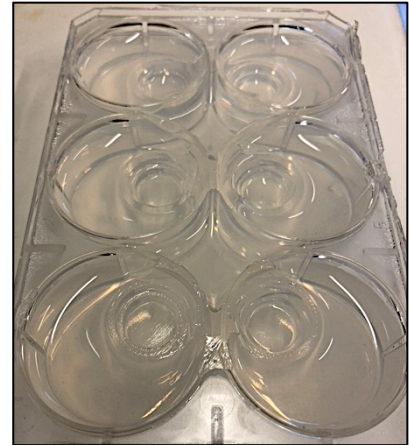
Figur 4-11 Agaroseform



Figur 4-10 Oppvarmet agarose

For å lage avgrensningen ble det laget en form som passet oppå reaksjonsbrettet. Ved å helle 1% løsning (1 g agarose pr. 100 ml sjøvann) av ferdigblandet agarose i reaksjonsbrettet og sette på formen, ville avgrensningene, eller brønnene, bli på 11.6 mm i diameter for hver CI. (Figur 4-12)

Ved bruken av denne typen gele har dyret en større sjanse for å bli svømmende rundt, og dermed mer data for å kunne studere bevegelsen til forskningsdyret. Etter mange tester med forskjellige brønnformer ble de bestemt at formen fra Figur 4-11 ga det beste resultatet.



Figur 4-12 Ferdig agarose i reaksjonsbrett.

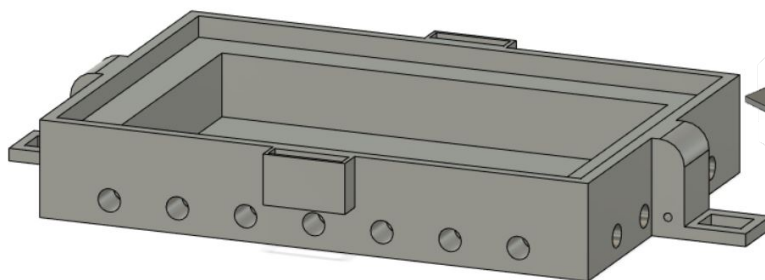
4.2.5 Modellering

For 3D-modellering brukes to forskjellige programvarer, Fusion 360 og Sketchup. Sistnevnte ble hovedsakelig brukt til tegninger og skisser av det fysiske oppsettet, også i 2D. Denne programvaren er gunstig for å kunne vise til det fysiske systemet på en oversiktlig måte, både med og uten mål og forklaringer. Fusion 360 ble brukt for å modellere spesialtilpassede deler til det fysiske oppsettet, som viste seg nødvendige for prosjektets resultat.

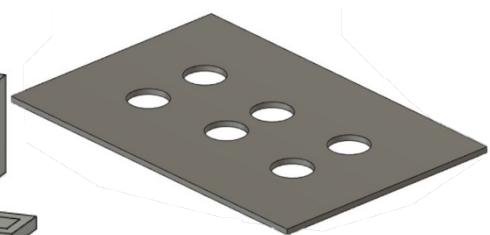
4.2.6 Base

Selve reaksjonsbrettet skal som nevnt plasseres ned i en 3D-printet beholder som skal sørge for lik plassering og påvirkning ved alle forsøk gjennomført av forskningsgruppen S13. Basen skal kunne festes direkte ned i aluminiums-rammeverket ved bruk av to skruer, en på hver kortside av beholderen. På hver langside er det en lomme som skal huse vibrasjonsmotorene og videreføre stimuliene skapt av motorene, gjennom resten av basen og opp til reaksjonsbrettet.

På hver kortside av basen er det fire hull, og hver langside har syv. De totalt 22 hullene er tilegnet for å holde IR-lyspærer som skal belyse innsiden av basen for å få en lik, men mild opplysning på undersiden av reaksjonsbrettet. Platen som vises i Figur 4-13 er skrevet ut helt hvit for å gi litt jevnere gjennomskinn fra IR-lysene opp til reaksjonsbrettet. Platen plasseres opp i selve basen og vil være



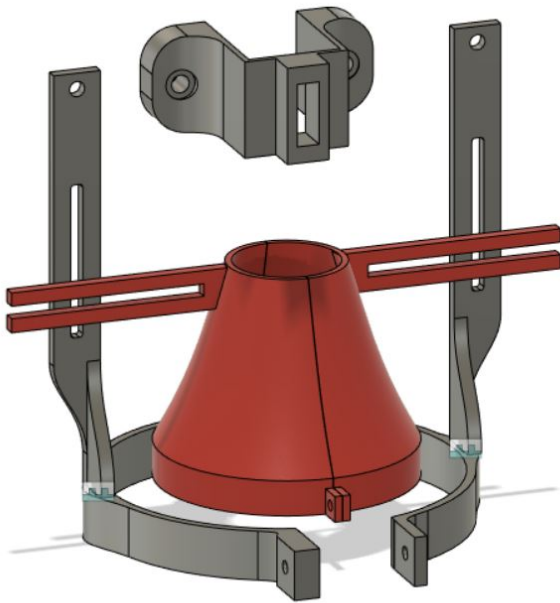
Figur 4-14 Base med hull på siden for IR-lys.



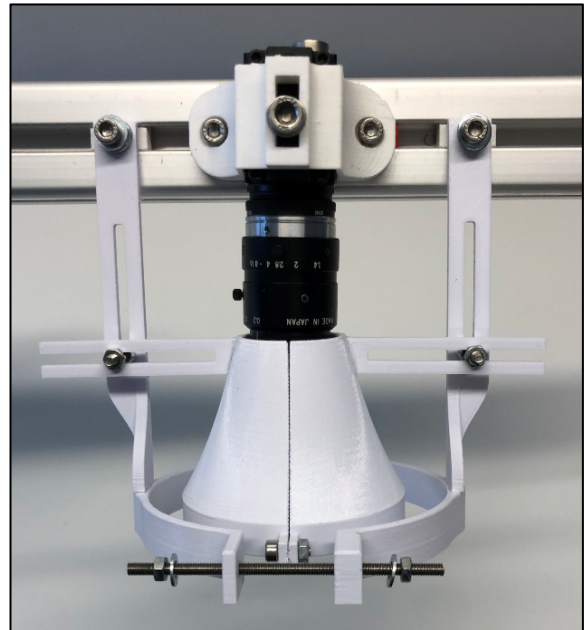
Figur 4-13 Plate inni Basen.

mulig å byttes ut, skulle det være nødvendig. Nederst i basen plasseres det en helt svart plastplate som skal fungere som en bakgrunn for hver av brønnene som kameraet skal filme ned i. Ettersom sortfargen absorberer mye IR-lys, sørger dette for en god bakgrunn å filme mot. Som det nevnes tidligere gjør det at kontrasten mellom dyrene og bakgrunnen er stor, hvilket er nødvendig for at DLC skal greie å fange opp bevegelsen og posisjonen til dyrene som blir filmet.

4.2.7 Fester til øvre rammeverk



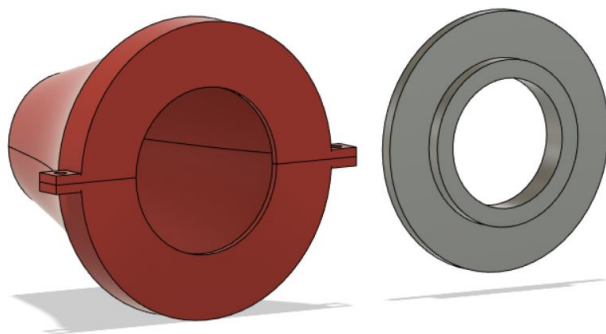
Figur 4-16 Fester til øvre rammeverk - 3D-Modell



Figur 4-15 Øvre System - Fysisk

Det øvre rammeverket består av åtte synlige 3D-printede komponenter som er designet for å holde kameraet, PVC-røret og det infrarøde filteret oppe. De forskjellige komponentene og deres hensikt blir forklart i kapitlene under.

4.2.8 Holder til IR-filter



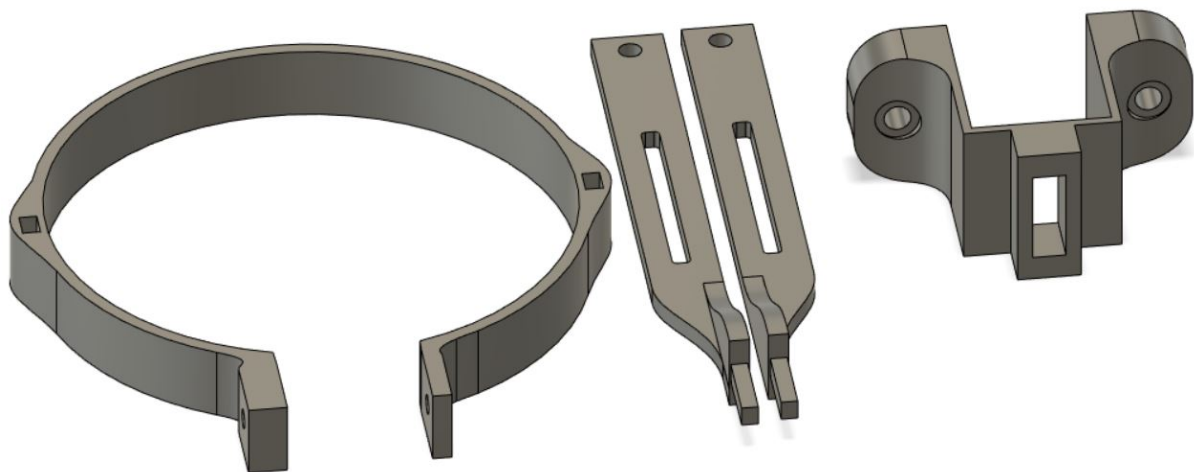
Figur 4-17 Holder til IR-filter, underside.



Figur 4-18 Holder til IR-filter, side.

Figur 4-17 og Figur 4-18 viser holderen til IR-filteret som skal være foran kameralinsen. Funksjonen til denne komponenten er både å holde et IR-Filter rett foran linsen til kameraet, men også beskytte mot annet lys fra utsiden som kan føre til gjenskinn og forstyrrelser i bildet. Holderen er todelt og festes sammen med to skruer med tilhørende muttere. For å sørge for at holderen står rett under kameraet er det i tillegg limt på ett feste til hver side av holderen. Disse skrues fast til resten av festene i det øvre rammeverket illustrert i Figur 4-16 og Figur 4-15. Den grå ringen vist i bildet fra Figur 4-17 skal plasseres inni den røde holderen til IR-filteret. RGB-LED-ene for stimuli limes på denne ringen som igjen plasseres inni holderen til IR-filteret. (Figur 4-31)

4.2.9 Holder til lysbeskytter og feste til kamera



Figur 4-19 Holder til lysbeskytter, og feste til Kamera.

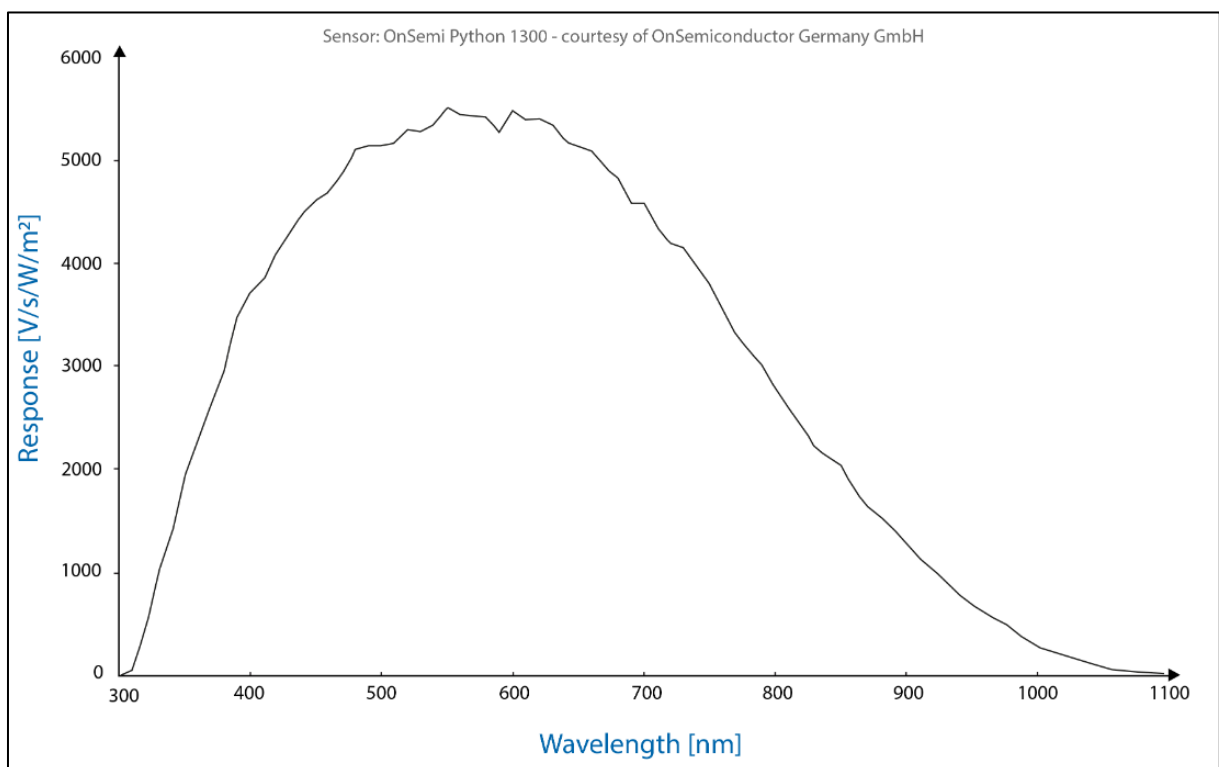
Komponentene på venstre side av Figur 4-19 er designet for å kunne holde oppe et 110 mm bredt PVC-rør for lysbeskyttelse. Funksjonen til dette røret er å blokkere så mye naturlig lys som mulig. Ringen som vises til venstre, skal gå rundt dette røret og strammes inn ved bruk av en skruer og muttere som skal gå igjennom det to hullene som er plassert parallelt ovenfor hverandre. Søylene som ligger til høyre for sirkelen skal plasseres hver opp i hvert av hullene som stikker direkte ned i rammen. Disse komponentene er dermed limt sammen og hele rammen skal kunne henge fast med to skruer som går direkte in i aluminiums-rammen. Mot høyre siden av bildet er kamerafestet som har to hull på hver side, festet direkte i aluminiums-rammen. I midten er det et avlangt hull som kommer til å holde selve kameraet oppe med mulighet til å finjustere høyden på kameraet istedenfor å måtte justere hele rammeverket.

4.2.10 Kamera, linse og filter

Kameraet som benyttes er av typen DMK-33UP1300 USB3.0 [18] vist i Figur 4-21. Det er et monokrom-kamera med 1.3-megapiksler og en ½ tomme sensor tilpasset IR-lys, som kan filme med opptil 210 bilder per sekund (Figur 4-20, [27]). Denne typen kamera var noe S13 hadde brukt ved tidligere

prosjekter og systemer og bla. ImMobilize var tilpasset DMK-kameraene. Som nevnt ble det brukt et avstandsstykke på 5 mm og en linse på 25 mm. Linsen tilsvarer MVL25M1 – 25 mm EFL, f/1.4 fra Thorlabs fra Figur 4-22 [28] og ga, sammen med avstandsstykket, et vidt bilde som får plass til hele reaksjonsbrettet.

Sammen med linse og kamera benyttes det et IR-filter som har til hensikt å blokkere ut synlig lys ettersom kameraet er svært følsomt mot denne typen lys kontra IR-lys. Dette vil hindre forstyrrelser i opptakene som kan være gjenskinn eller refleksjoner. I Kombinasjon med at CI også påvirkes av synlig lys vil RI-LED og IR-filter sammen være positivt for kvaliteten på både forsøket og video-opptaket som blir gjort.



Figur 4-20 Spektralsensitivitet til DMK-33UP1300.



Figur 4-21 DMK-33UP1300.



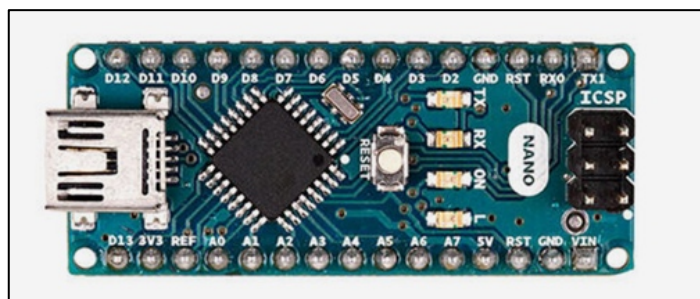
Figur 4-22 MVL25M1 – 25 mm EFL, f/1.4.

4.3 Komponenter til krets

Dette kapitlet vil gå inn på alle komponenter som ble brukt i kretsen. Komponentene som ble valgt falt i hovedsak på hva som var rast og enkelt å få tak i, men også hva S13 hadde liggende. Alle de forskjellige delene ble loddet sammen med ledninger på et eksperimentkort. Dette beskrives i kapittel 4.4. For de forskjellige komponentene som skal plasseres på ulike steder i oppsettet er det laget kontakter, slik at de enkelt kan kobles ut og inn etter behov. Kretstegningene i dette kapitelet illustrerer dette ved å ha en kontakt hvor det derfra kobles på for eksempel motorer eller LED-er for stimuli. Se Figur 4-28 for et eksempel. For IR-LED og motorene skal det kobles flere av samme komponent sammen, men det blir beskrevet med en egen illustrasjon i tillegg.

4.3.1 Arduino Nano

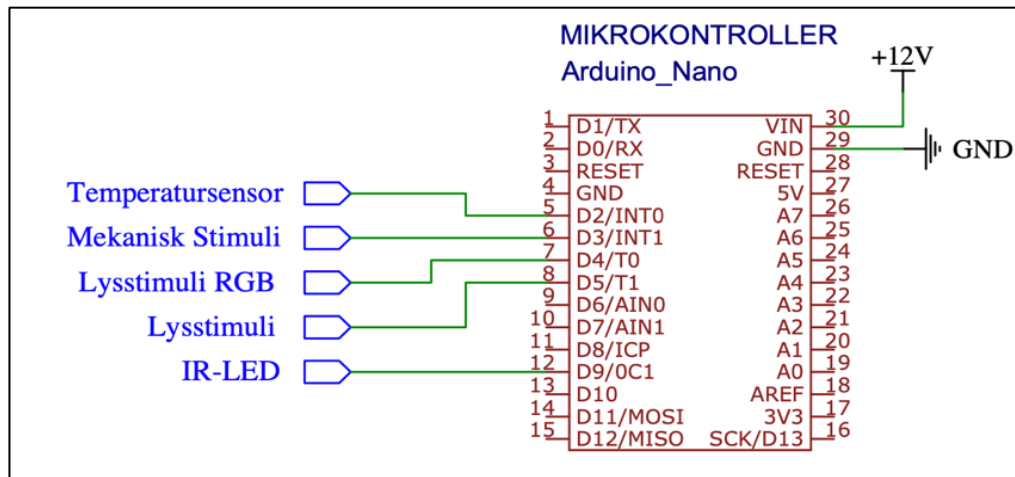
Valget av mikrokontroller falt som nevnt tidligere på en Arduino Nano [29]. Dette er bindeleddet mellom programvaren og kretsen som skal tilførsel av stimuli. Mikrokontrolleren er en liten kompakt enhet med mål på 18x45 mm. Arduino kan bli brukt til alt fra hobbyaktiviteter til større eksperimenter og er svært enkel å bruke. Til dette prosjektet ble programmeringsspråket C++ og Arduino IDE [16] brukt. Arduino Nano har 22 input/output (I/O) -pinner og kobles til en datamaskin gjennom en mini-B USB kabel, som også sørger for seriell kommunikasjon mellom brettet og datamaskinen den er koblet til. Se Tabell 4-1 for spesifikasjoner til Arduino Nano.



Figur 4-23 Arduino Nano.

Microcontroller	ATmega328
Operating voltage	5 V
Input voltage	7-12 V
Flash memory	32 KB (2 KB used by bootloader)
Clock speed	16 MHz
DC current per I/O pins	40 mA (I/O Pins)

Tabell 4-1 Spesifikasjoner, Arduino Nano. [29]

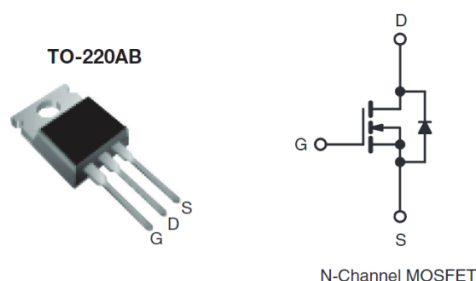


Figur 4-24 Arduino i forhold til andre komponenter.

Fra Figur 4-24 ser man hvordan mikrokontrolleren er plassert i forhold til resterende deler av kretsen. Figuren illustrerer hvordan sensorer, LED-er og input-spenning skal kobles til de forskjellige pinnene på mikrokontrolleren.

4.3.2 MOSFET

MOSFET-en som er brukt i kretsen er av typen IRL510 MOSFET. Det betyr at den bare lar strøm gå gjennom Drain og Source når det er en positiv spenning på Gatens. Output-strømmen varierer på input-spenningen som gjør at MOSFET-en blir brukt som en regulator i kretsen. Denne modellen har en output-grense på 100 V og 5 A som er mer enn nok for denne kretsen ettersom at deler av kretsen kun blir drevet av en 12 V 2 A strømtilførsel. Arduino-en sender et styresignal til MOSFET-ens input. Denne bruker styresignalet, på opp mot 5 V, til å styre en større spenning på opptil 12 V som IR-lys og deler av lysstimuli trenger [30].



Figur 4-25 MOSFET IRL510.

PRODUCT SUMMARY		
V_{DS} (V)	100	
$R_{DS(on)}$ (Ω)	$V_{GS} = 5.0$ V	0.54
Q_g (Max.) (nC)	6.1	
Q_{gs} (nC)	2.6	
Q_{gd} (nC)	3.3	
Configuration	Single	

Tabell 4-2 MOSFET IRL510, produktbeskrivelse.

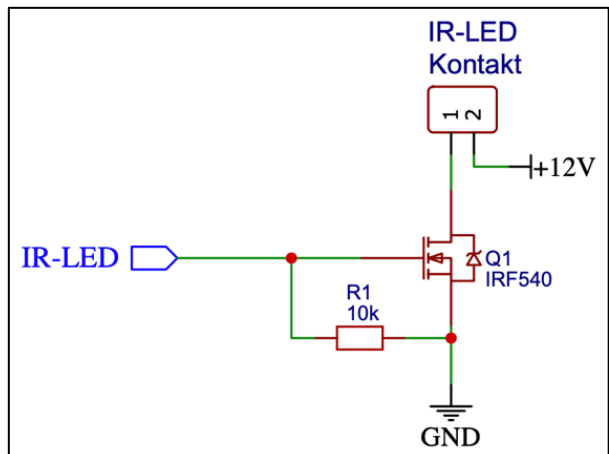
4.3.3 IR-LED

For å sørge for gode lysforhold for kamera og for å hindre påvirkning av svømmemønsteret til CI-ene, falt valget på en IR-LED. IR-lysdiodene som S13 hadde var ikke nummerert utenom at de har en bølgelengde på 850 nm med en forspenning på mellom 1,4 – 1,5 V, men de tilsvarer LED-en vist i Figur 4-27.

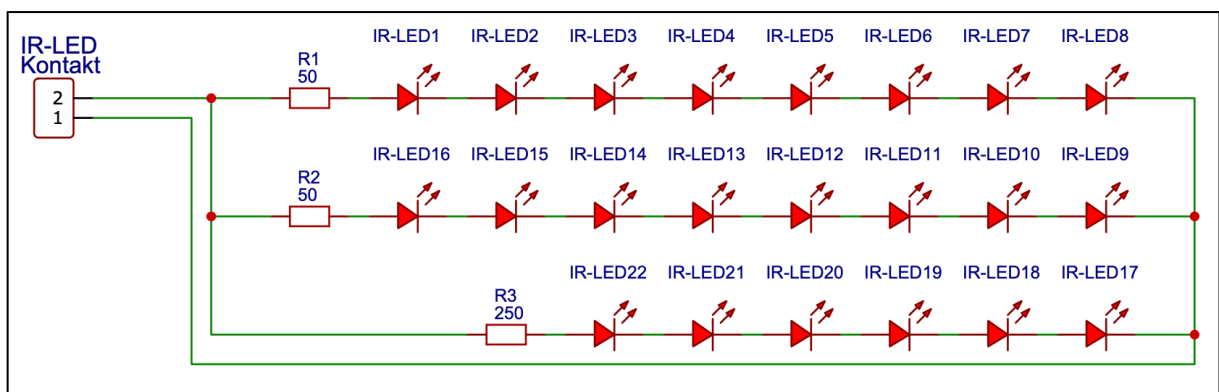
Som nevnt tidligere benyttes det totalt 22 IR-LED-er for å sørge for jevn og god belysning i basen. Fra Figur 4-28 illustreres hvordan kontakten er tilkoblet resten av kretsen på koblingsbrettet. For å koble sammen alle IR LED-ene ble det valgt å koble de sammen i tre serier. Med to serier på åtte og en serie med seks LED-er, ble det med noen motstander jevn og god belysning i basen. (Figur 4-26) Motstandene hindrer at IR-LED ryker om strømmen gjennom denne delen av kretsen blir for høy.



Figur 4-27 IR LED.



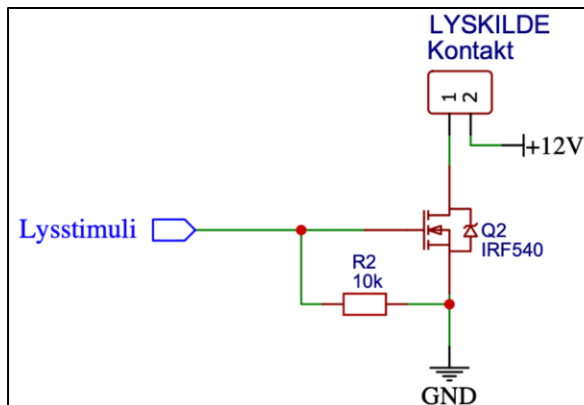
Figur 4-28 Krets for IR LED på koblingsbrett.



Figur 4-26 Krets av IR LED-er som festes rundt basen.

4.3.4 Lysstimuli 12 V 20 W Halogen Lyspære

For å simulere dagslys benyttes en 12 V 20 W halogen lyspære [31]. Dette vil være en form for lysstimuli som skal være mulig å velge som en del av forsøkene S13 vil gjennomføre. Lyspæren vil i tillegg være festet til en kontakt [32] som gjøre den lett å byttes ut om pæren skulle ryke etter lengre perioder med bruk. Figur 4-29 viser hvordan pære og kontakt er festet sammen og dette vil kobles på resten av kretsen vist i Figur 4-30.



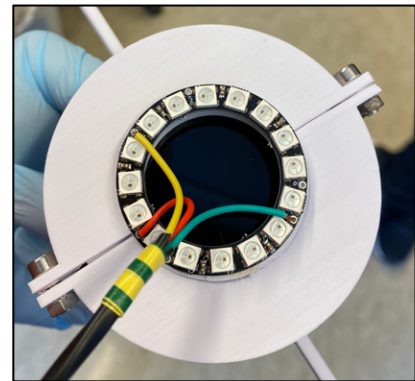
Figur 4-30 Krets for halogen lysstimuli.



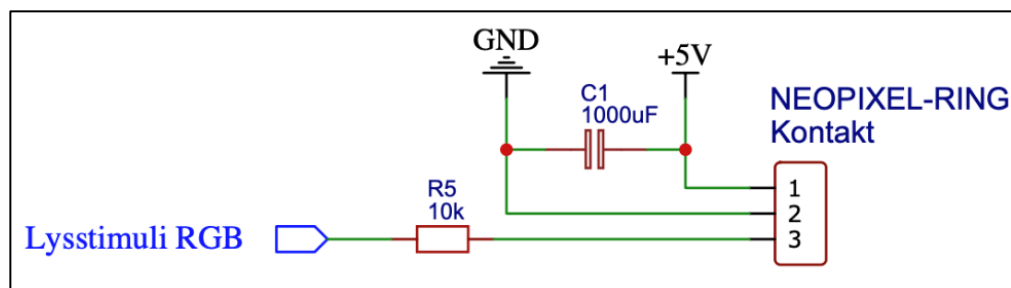
Figur 4-29 Lyspære i kontakt.

4.3.5 Lysstimuli RGB NeoPixel Ring

For resterende lysstimuli benyttes en RGB NeoPixel Ring med 16 LED-er. Denne er plassert rundt filteret på ringen i Figur 4-17, for å sørge for lik distribusjon av stimuli. LED-ringene har en 5 V spenning og er i tillegg festet til jord og input data. Sistnevnte skal sørge for farge og styrke valgt i GUI-en ved et forsøk, skal bli rett. Kretsen rundt NeoPixel ringen blir forklart i senere kapitler, men fra Figur 4-32 ser man denne delen.



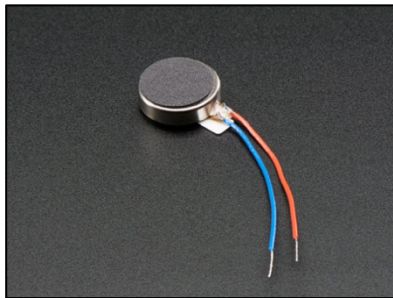
Figur 4-31 NeoPixel Ring 16.



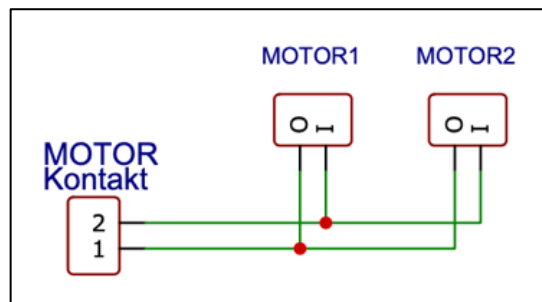
Figur 4-32 Krets for RGB-lysstimuli.

4.3.6 Motor

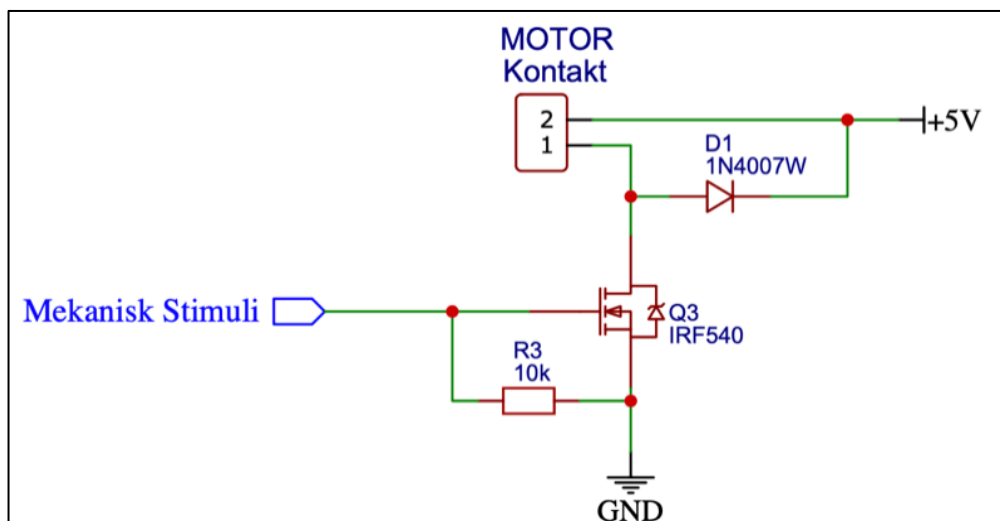
For mekanisk stimuli benyttes to små parallellkoblede motorer som kan vibrere med en gitt frekvens (Figur 4-33) [33]. Disse er kun 10 mm i diameter og 2.7 mm tykke som er en motor som S13 har brukt på det opprinnelige oppsettet, og har fungert mer enn godt nok til forsøkene de har gjort. Valget falt på denne ettersom S13 hadde mange slike liggende og har vist seg pålitelig til jobben den skal gjøre. Motoren tar fra 3 V til 5 V og kan operere i temperaturer fra -20°C til 70°C som er godt innenfor vårt bruk. Fra Figur 4-35 ser man motorens plassering i forhold til resten av kretsen som har med mekanisk stimuli å gjøre.



Figur 4-34 Vibrerende minimotor.



Figur 4-33 Motorer koblet.

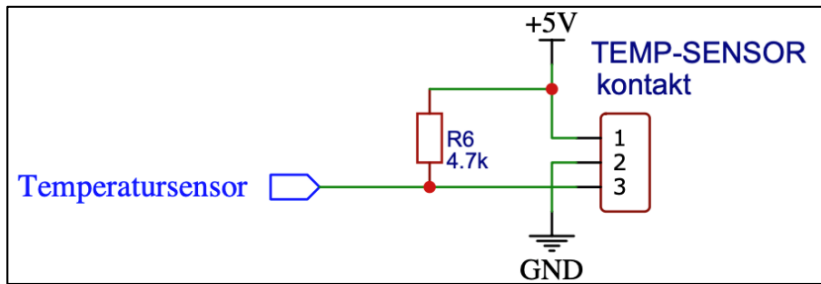


Figur 4-35 Krets for motorer.

4.3.7 Temperatursensor

For å kunne måle temperaturen blir det brukt to temperatursensorer av typen VMA324 DS18B20. Hver sensor har sin egen kontakt koblet direkte inn i kretsen. Sensoren tar en spenning på mellom 3.0 V og 5.5 V, og som har muligheten til å måle temperaturer mellom -55°C til $+125^{\circ}\text{C}$ [34]. Dette er mye kaldere og varmere nivåer enn det systemet kommer til å bli utsatt for og sensoren kommer dermed aldri til å måtte nå sitt minimale eller maksimale punkt, og sjansen for feillesinger blir da mindre. Selv

om oppsettet er plassert i inkubatoren så er hensikten med sensorene, som nevnt tidligere, å kunne logge og kartlegge temperaturen rundt reaksjonsbrettet ved forsøkene som skal gjennomføres.



Figur 4-37 Krets for temperatursensor.



Figur 4-36 Temperatursensor.

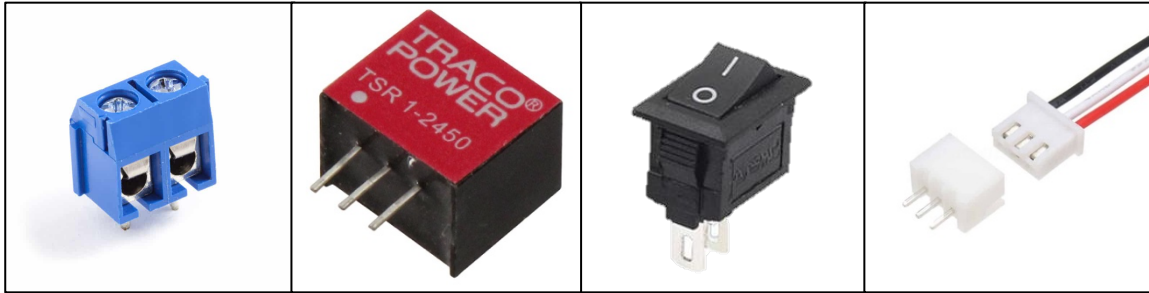
4.3.8 Øvrige komponenter

De resterende komponentene brukt i kretsen omfatter motstander, kondensatorer, terminaler for 12 V spenning, 12 V forsyning og kontakter for å koble sensorer og lys til resten av kretsen på en enkel måte. Alt var noe som S13 hadde tilgjengelig som gjort det enkelt å velge dette. Virkemåten og plassering i kretsen blir forklart i kapittel 4.4.

Motstandene som blir brukt i kretsen varierer fra mindre motstander 50-250 Ω til høyere verdier mellom 4,7 k Ω – 10 k Ω . Kondensatoren brukt i kretsen er på 1000 μ F og brukes for å redusere støy i forbindelse med lysstimuli, slik at det ikke blir noe avvik mellom valgt og observert verdi. Til å forsyne komponentene som trenger mer spenning enn det Arduino-en klarer å gi (3,3 V), brukes en 12 V DC, 2 A strømforsyning koblet rett i veggen. Denne tilsvarer en vanlig PC-lader som S13 hadde i sine lokaler. For å føre spenningen til resten av kretsen anvendes en terminal [35] hvor alt som trenger 12 V blir koblet til. Mellom strømforsyningen og terminalen er det også en bryter [36] hvor man har mulighet til å kutte strømmen om noe skulle skje.

Siden IR-LED-ene og den halogene lyspæren krever 12 V spenning og temperatursensor, NeoPixel Ring og motorer trenger 5 V spenning trengs det også en konverter. Med en Traco Power TSR 1-2450 [37] konverteres 12 V til 5 V og fordeles videre i kretsen (Figur 4-39).

Alle sensorer og stimulkilder er koblet til resten av kretsen ved hjelp av to- og tre-pins mini-kontakter [38]. De er 2 mm brede og benyttes for at man lett skal kunne bytte ut sensorer og kilder i sin helhet om det skulle bli nødvendig i fremtiden.



Figur 4-38 Terminal.

Figur 4-39 Konverter.

Figur 4-40 Bryter.

Figur 4-41 Tre-pin kontakt.

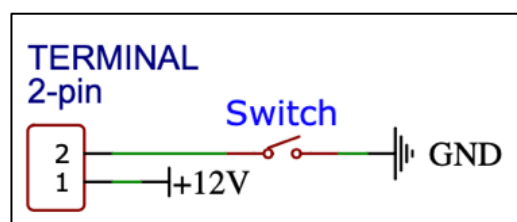
4.4 Krets

Den fullstendige kretsen vil i dette kapitlet blir beskrevet og forklart. Fra kapitlene over har komponenter, valg rundt disse og plassering i forhold til mikrokontrolleren blitt belyst, men her vil den fullstendige kretsen bli illustrert. Ettersom oppgraderingen av systemet i hovedsak har dreid seg om å effektivisere forskningen, kunne noe av det tidligere oppsett benyttes. Dokumentasjonen av den tidligere kretsen og komponenter var mangelfull, som gjorde at det ble mye jobb med å finne rette komponenter og oppsett for at kretsen skulle fungere som den skulle.

4.4.1 Beskrivelse

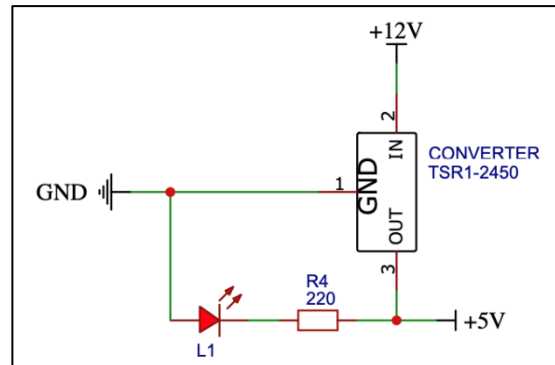
Figurene fra kapittel 4.3.1 til 4.3.7 som beskriver plasseringer av komponenter dekker det meste av den totale kretsen for dette prosjektet. For de resterende delene har med strømforsyning og konvertering av spenning å gjøre. Den fullstendige kretsen er lagt til som bilde (Figur 4-47) og som vedlegg til rapporten.

Mikrokontrolleren som oppsettet benytter, Arduino Nano, leverer en 3,3 V spenning. Dette er for lite til å drive alle komponentene og det er grunnen for å benytte en ekstern kilde som nevnt tidligere. Fra Figur 4-42 ser man hvor 12 V spenningen er koblet til og det er fra denne terminalen hvor komponenter som trenger det, henter spenning fra. Ved å legge til en bryter kan man i tillegg koble fra og kutte forsyningen raskt om noe skulle skje.



Figur 4-42 Krets terminal med bryter.

For komponentene som trenger 5 V brukes en Step-Down konverterer som har inputspenning på 12 V og output på 5 V som Figur 4-43 viser. I tillegg til dette er det koblet til en LED som indikerer om kretsen har spenning. Om bryteren fra Figur 4-40 er på, vil LED-en lyse, og om den er av vil LED-en være slukket. Dette gir en rask indikasjon om systemet har spenningen den trenger eller ikke.

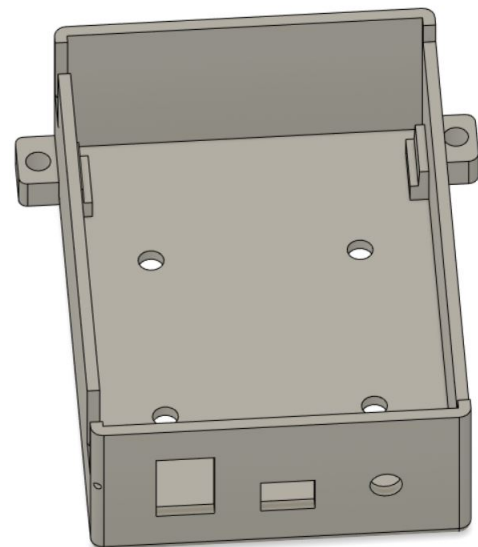


Figur 4-43 Step Down Konverterer.

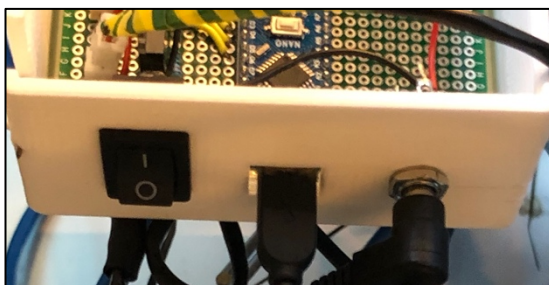
4.4.2 Lodding

Prosjektet innebar mye lodding til å skape den tilpassede kretsen som skulle passe det nye og større oppsettet. For å lodde ledninger og komponenter sammen, ble det brukt loddebolt og tinn. Motorer, IR-LED og temperatursensorer ble loddet med ekstra kabel for å rekke dit de skulle plasseres i oppsettet.

Hele koblingsbrettet er plassert i en kretsbox som fra Figur 4-45. På undersiden av boksen er det tre hull i ulik størrelse. Hullet til venstre er for å plassere bryteren til strømforsyningen. Denne forsyningen kobles til en kontakt i hullet til høyre. I midten er det plass til USB-kabelen som skal kobles til PC-en. I tillegg til boksen fra figuren er det også et lokk som hører til. Det er festet sammen og henges opp med skruer på hver side av boksen. Se Figur 4-44 for bilde av ferdig fysisk kretsbox.



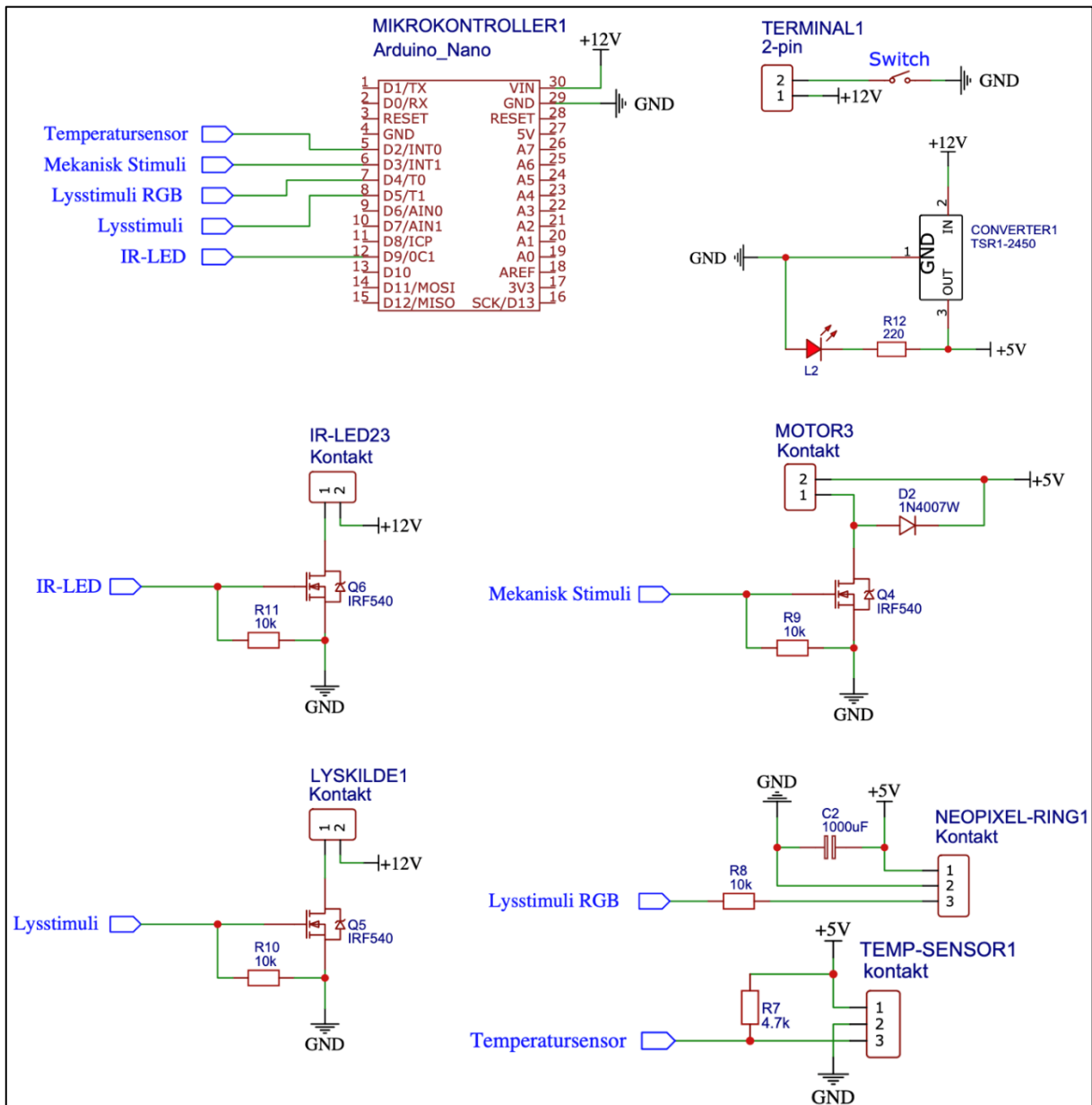
Figur 4-45 Kretsbox uten lokk.



Figur 4-46 Kretsbox underside.



Figur 4-44 Kretsbox for oppsettet.



Figur 4-47 Fullstendig krets.



Figur 4-49 Plassering av kretsboкс.



Figur 4-48 Plassering av kretsboкс, side.



Figur 4-50 Oppsett med kretsbox åpen. Alt tilkoblet.

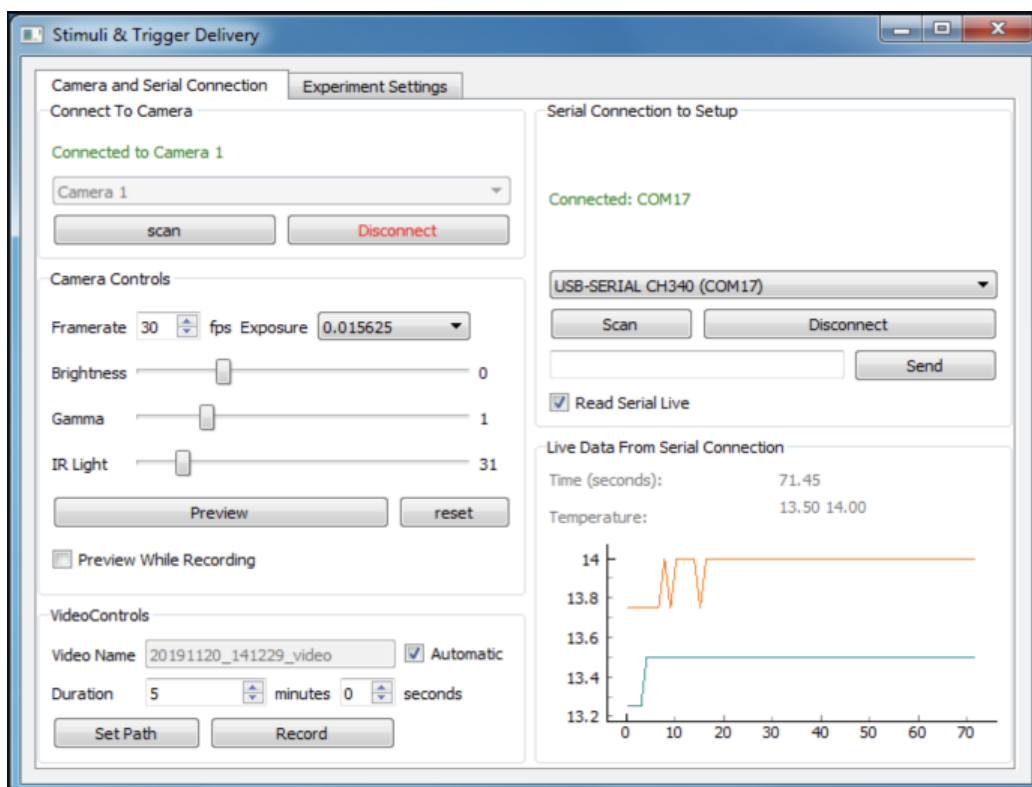
4.5 Programvare

I dette kapittelet vil programvaren fra ImMobilize [13] bli beskrevet, hvilket er programkoden utviklet av S13. Det vil også gå inn på videreutviklingen av programvaren som er gjort. Målet er å vise hvordan det ferdige systemet virker, og beskrive de endringer som gruppen har gjort. Det opprinnelige oppsettet med tilhørende kode ble alt kjørt på en datamaskin på Sarseneret hos S13. Denne maskinen kjører med en Windows 7-versjon og er frakoblet internett, ettersom at enhver oppdatering på maskinen vil føre til at kommunikasjonen med kameraet, datamaskinen og ImMobilize ikke ville fungere. Dette gjorde at arbeid og endringer tilknyttet programkoden og kamera var tidvis krevende.

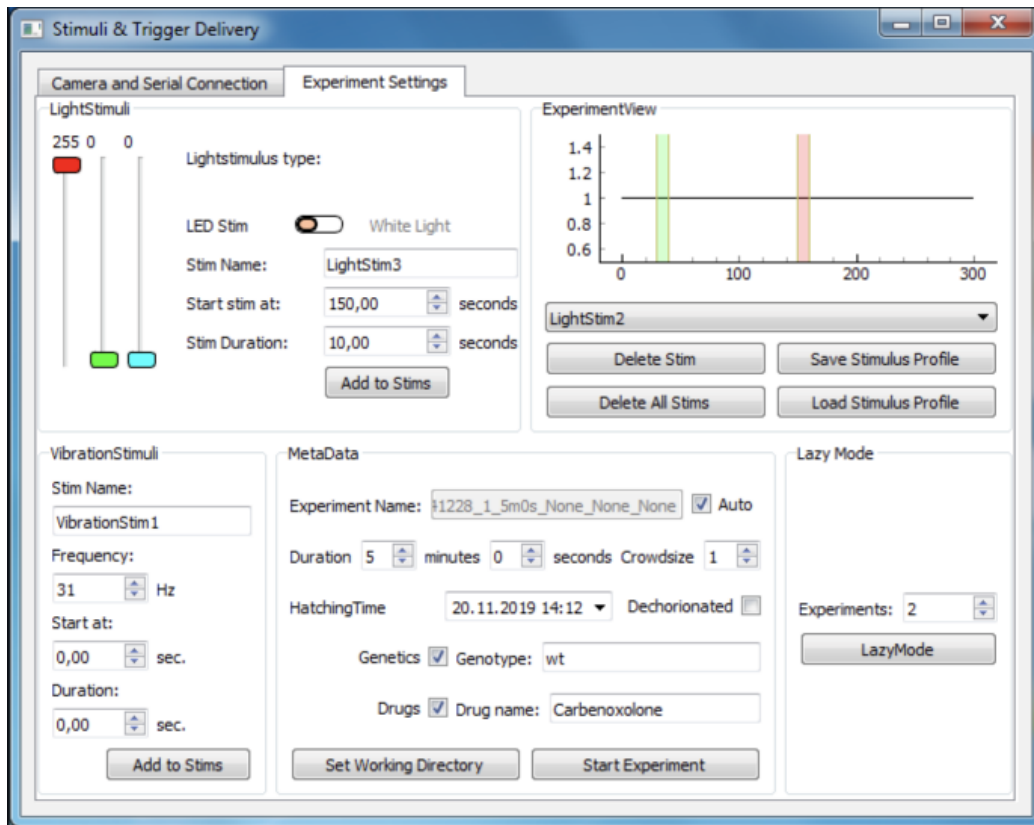
Siden det eksisterende systemet skulle oppgraderes og at det ble valgt både det nye og det gamle oppsettet skulle kjøres på samme datamaskin, ble dette i tillegg en utfordring for å sørge for at alt ville fungere samtidig. Selv om gruppen har jobbet mye med programkoden, har det i hovedsak dreid seg om funksjonaliteten til systemet, men også dypplæring beskrevet i kapittel 4.6.

4.5.1 Brukergrensesnitt (GUI)

For å velge alt av innstillinger for et forsøk benytter man seg av en GUI som er tilgjengelig på datamaskinene og er koblet til systemene. GUI-en består av to faner «Camera and Serial Connection» (Figur 4-51) og «Experiment Settings» (Figur 4-52).



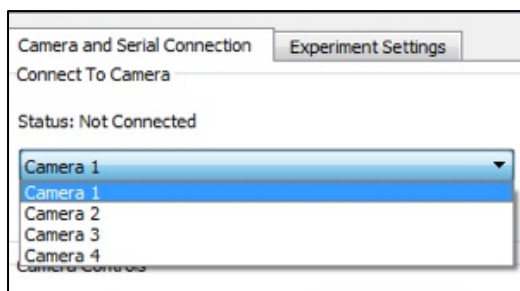
Figur 4-51 Brukergrensesnitt "Camera and Serial Connection".



Figur 4-52 Bruergrensesnitt "Experiment Settings".

Den første fanen tar for seg tilkoblinger og innstillinger til både kamera og fysisk oppsett. Den andre er hvor man kan velge alt av innstillinger for hvert forsøk som skal gjennomføres i forbindelse med forskning. Figurene nevnt over viser til sammen GUI-en i sin helhet.

Fra «Camera and Serial Connection» ser man tilkoblingsmuligheten for kamera under «Connect to Camera». Her er det en nedtrekksmeny som inneholder mulige kameraer å koble til. Figur 4-54 viser et eksempel hvor det er fire kameraer tilgjengelig i menyen. Om det skulle mangle et kamera er det også en mulighet til å hente alle kameraer som er fysisk tilkoblet ved å trykke «scan». Nedtrekksmenyen vil da oppdateres og inneholde tilgjengelige kameraer.



```
def detect_cameras(self):
    self.ui.comboBoxConnectedCameras.clear()
    for ii in range(4):
        cap = cv2.VideoCapture(ii)
        if not cap.isOpened():
            pass
        else:
            cam = "Camera "+str(ii +1)
            self.ui.comboBoxConnectedCameras.addItem(cam)
```

Figur 4-54 Meny med tilgjengelige kameraer. Figur 4-53 Programkode for å hente tilgjengelige kameraer.

Figur 4-53 viser programkoden bak menyen. Det tidligere systemet hadde kun tre kameraer tilkoblet samtidig. Ettersom oppsettet nå har fire kameraer (tre fra gammelt oppsett og ett fra nytt oppsett), er koden oppdatert slik at alle de fire kameraene som er tilkoblet vises i menyen. Den opprinnelige programkoden tillot kun de tre kameraene fra det originale oppsettet å vises i nedtrekksmenyen.

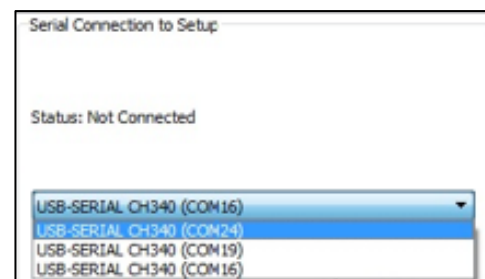
For tilkobling til mikrokontroller er det en tilsvarende programkode som henter tilgjengelige serielle porter. Figur 4-55 viser koden brukt til dette. Disse portene vises på samme måte i en meny slik som fra Figur 4-56, men her med et eksempel med tre ledige porter. Siden kameraet og mikrokontrollen ikke er koblet til hverandre direkte, må man sørge for at riktig kamera kobles til riktig mikrokontroller. Det er derfor laget en liste som forteller hvilken port tilhører hvilket kamera, Se Tabell 4-3.

```
def update_serial_available(self):
    """Scans the available comports and updates the combobox with the result"""
    self.ui.comboboxSerialAvailable.clear()
    self.arduino.scan_comports()
    for port in list(self.arduino.port_dict.keys()):
        self.ui.comboboxSerialAvailable.addItem(port)
```

Figur 4-55 Programkode for å hente tilgjengelige serielle porter.

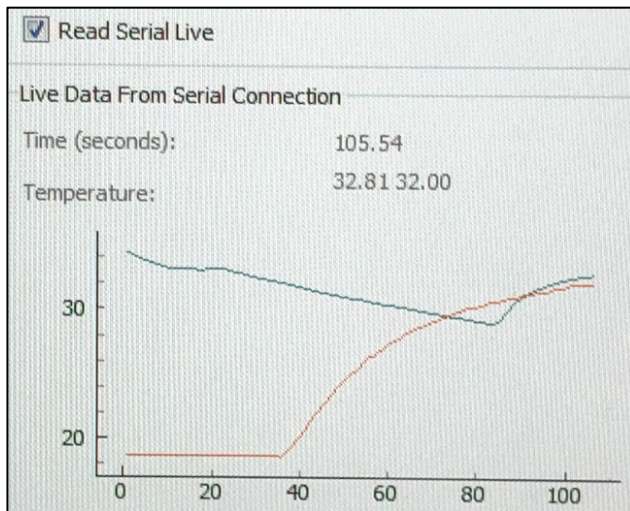
Kamera: → Seriell Port:	
Kamera 1 →	USB-SERIAL CH340 (COM24)
Kamera 2 →	USB-SERIAL CH340 (COM16)
Kamera 3 →	USB-SERIAL CH340 (COM19)
Kamera 4 →	USB-SERIAL CH340 (COM11)

Tabell 4-3 Kamera/COM-port - tilkobling.

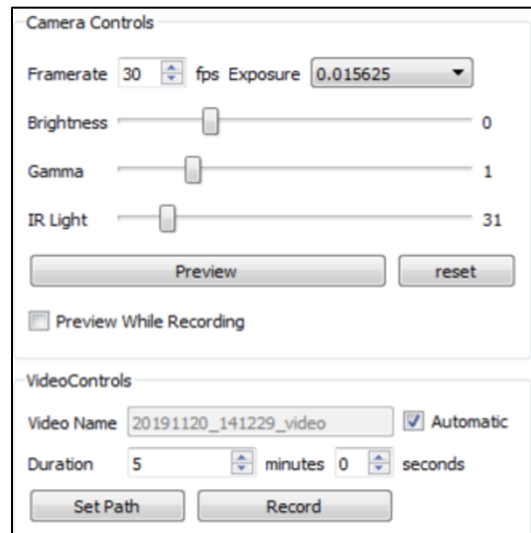


Figur 4-56 Meny med serielle porter.

Når kamera og mikrokontroller er tilkoblet, er det mulig å endre kamerainnstillinger og se direkte data fra temperatursensorene på siden. Ved å huke av avkryssingsboksen «Read Serial Live» vil det under «Live Data from Serial Connection» komme temperaturer fra to sensorer i tillegg til en tidtaker som forteller hvor lenge dataen har blitt hentet, Se Figur 4-58.



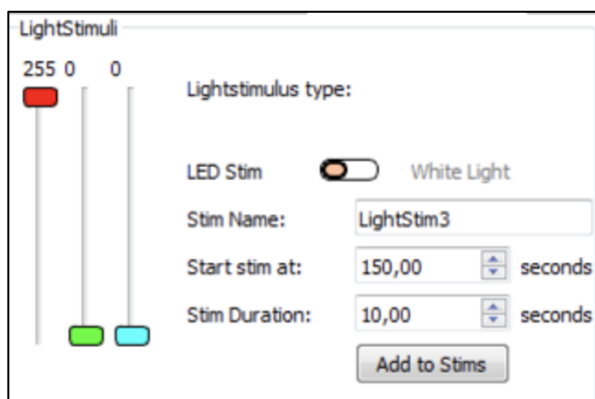
Figur 4-58 Live Data.



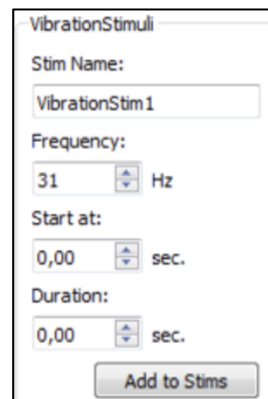
Figur 4-57 Kamerainnstillinger og optak.

Det er i denne fanen også mulig å velge kamerainnstillinger for et forsøk. Man kan endre de forskjellige innstillingene for å gjøre opptak tydeligere og med mer kontrast. Ved å trykke på «Preview», fra Figur 4-57, kan man direkte se hvilke innstillinger som påvirker hva. Det er her også mulig å ta opptak uten å tilføre noe form for stimuli.

I eksperimentvinduet fra Figur 4-52 «Experiment Settings» er det mulig å justere på alle de forskjellige stimuli-funksjonene for et forsøk. Det kan velges mellom lysstimuli og mekanisk stimuli i form av vibrasjon.



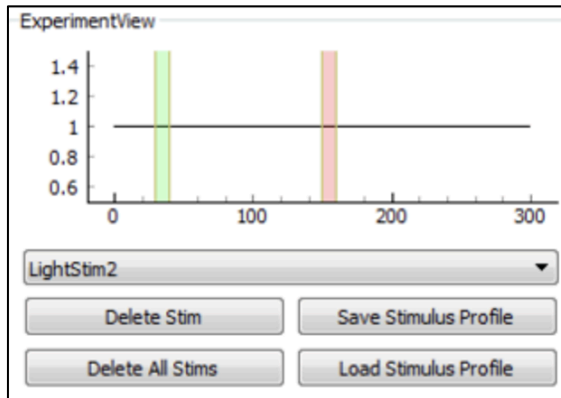
Figur 4-60 Lysstimuli.



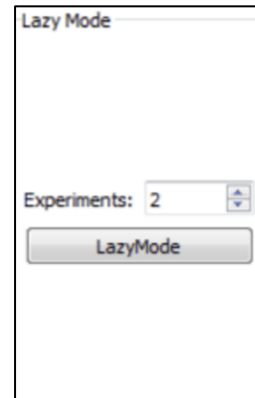
Figur 4-59 vibrasjonsstimuli.

Ved å justere på de røde, grønne og blå knappene fra Figur 4-60 justerer man fargen til NeoPixel-ringen. I dette tilfellet ville den ha lyst helt rødt. Om man trykker på knappen mellom «LED Stim» og «White Light» byttes det mellom LED-Ringen og lyspæren for å simulere dagslys. Det kan velges starttid for stimuli og hvor lenge det skal vare noe som også gjelder for vibrasjonsstimuli, men her kan det velges frekvens på vibrasjonen istedenfor lys og lysstyrke, Se Figur 4-59.

Etter man har valgt stimuli vil det dukke opp som en sekvens under «ExperimentView» (Figur 4-61). Systemet har en mulighet for å lagre de forskjellige stimuli-innstillingene, skulle en ønske å bruke de flere ganger i ettertid. Når det gjelder «Lazy Mode» (Figur 4-62) kan det brukes om det ønskes opptak av samme dyr med samme innstillinger, flere ganger. Denne vil da gjenta et forsøk et gitt antall ganger uten at det manuelt trengs å startes på nytt.



Figur 4-61 ExperimentView.



Figur 4-62 Lazy Mode.

Når alle innstillinger er valgt vil «Start Experiment» under «MetaData» starte et opptak av et fullt forsøk. Det er noe tilleggsinformasjon som kan velges som S13 bruker til sin forskning. Dette vil lagres som en tekstfil i samme mappe som opptaket, når forsøket er ferdig.

Figur 4-63 MetaData.

4.5.2 Programkode

Den opprinnelige programkoden fra ImMobilize ble skrevet i Python med unntak av mikrokontrollerkoden. Programkoden kan deles inn i fem hoveddeler som består av én hovedkode, én

for brukergrensesnittet, én for å tilføre stimuli, én for kameraet og én mikrokontrollerkode. Gruppen hadde liten erfaring med Python fra tidligere. Ettersom det kun skulle gjøres mindre endringer for å etterkomme krav, ble det likevel valgt å skrive i samme programmeringsspråk. Det hadde vært langt mer tidkrevende og utfordrerne tidsmessig å skrive all kode på nytt, selv med et språk gruppen kjenner bedre til. Datamaskinen på Sarsenteret benytter IDE-en Spyder3 [39] til å kjøre ImMobilize. Denne programvaren har et kommandovindu hvor alt av valg fra GUI-en plottes underveis i løpet av et forsøk.

4.5.3 Mikrokontrollerkode

Arduino-en fra det tidligere prosjektet har en allerede fungerende programkode som styrer det forskjellige stimuliene og IR-lysene. Denne koden kommuniserer med Python-koden, og via brukergrensesnittet kan en bruker justere eller forandre de forskjellige innstillinger nødvendig for å gjennomføre diverse tester. Hvordan kommunikasjonen foregår blir beskrevet i kapittel 4.5.4. I mikrokontrollerkoden brukes det en funksjon kalt «analogwrite», denne funksjonen blir brukt for å skape et pulsbreddemodulasjon-signal (PWM) med en skala på 0-255. Dermed blir mange av innstillingene som RGB-lys og IR-lys satt til maksimum 255 i brukergrensesnittet.

4.5.4 Kommunikasjon

Kommunikasjonen mellom mikrokontroller og Python-programmet brukt i oppsettet, er seriell kommunikasjon via en Micro-B USB-kabel. For å skille forskjellige innstillinger blir det sendt kommandoer som en streng til Arduino-en, som deles opp og tolkes. En kommando består av et startsignal etterfulgt av åtte bits, til slutt skriver programmet ut et sluttsignal til kommandovinduet. For eksempel «vXXXXXXXX» starter vibrasjonsstimuli, men om koden starter med «n» ville det signaliseres at NeoPixel-ringen skal skrus på, mens «w» starter lyspæren for simulasjon av dagslys. De fire første X-ene står for starttiden, og de siste fire X-ene står for hvor lenge tilført stimuli skal vare etter starttiden. For eksempel vil kommandoen «v50009000» etter fem sekunder starte mekanisk stimuli som varer i 9 sekunder, og «n00005000» vil med en gang starte RGB-lysstimuli som varer i fem sekunder. Disse kodene er ikke noe som manuelt skrives inn, men noe som GUI-en sender til Arduino-en basert på hva som velges av bruker.

4.5.5 Main.py

Den originale Python-koden som var laget av et tidligere medlem av forskningsgruppen S13, hadde en funksjonsfeil i det at motoren som skapte vibrasjonsstimuliene til systemet ikke ville slå seg av når programmet var ferdig. Dette førte til at brukeren manuelt måtte resette og koble fra mikrokontrolleren for at motoren til å stoppe. Det viste seg at «__main.py__» hadde en kommando for å avslutte og en kommando for start av vibrasjonene. Når koden skulle motta start-signalet så ble programmet sittende fast i en løkke. Dette var på grunn av at både start og slutt kommandoen begynte

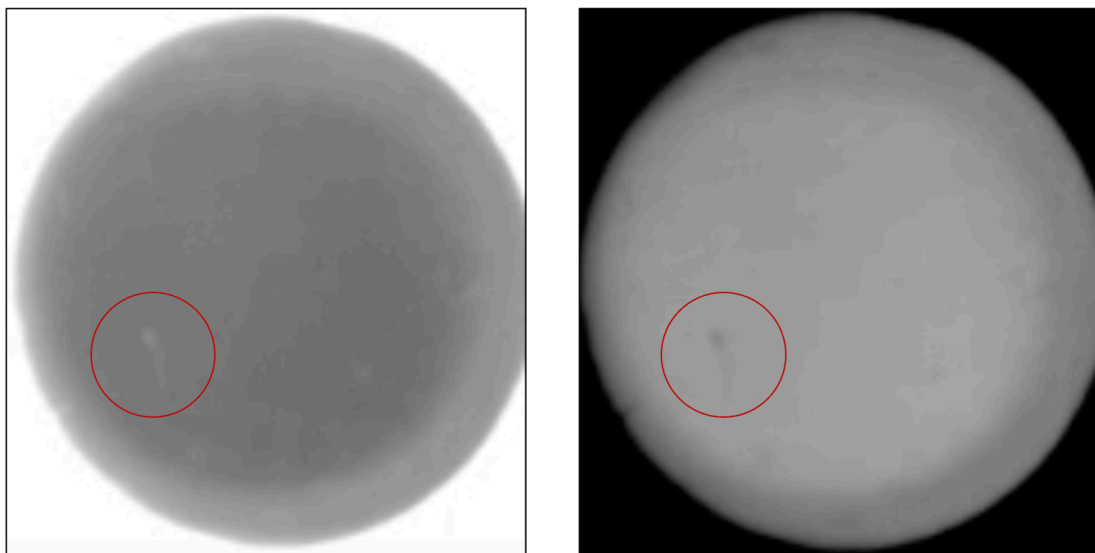
med samme signal. Da programmet leste av strengen som den mottok og kom til samme signal, tolket programmet det som start-kommandoen og begynte den løkken på nytt og på nytt. Dette måtte da fikses og ved å bytte avslutningskommando, ble feilen rettet opp og programmet kjørte uten større feil.

4.6 DeepLabCut

DeepLabCut (DLC) er et dyplæringsrammeverk som brukes til å estimere posisjoner i hovedsak til dyr. Det var dette rammeverket S13 ønsket ble benyttet for å lære opp en modell som de senere skulle bruke til sin forskning. DLC er en veldokumentert programvare på GitHub [40] og har mulighet til å spore flere dyr samtidig, slik oppgaven forventer. Prosessen av å lære opp en modell innebærer å samle data ved å ta et opptak ved hjelp av systemet som er laget, deretter må det hentes ut bilder fra videoene, markere det som skal spores i hvert bilde og til slutt trene opp modellen basert på tidligere punkter. I kapitlene under beskrives prosessen etter at videoene er tatt.

4.6.1 Invertering

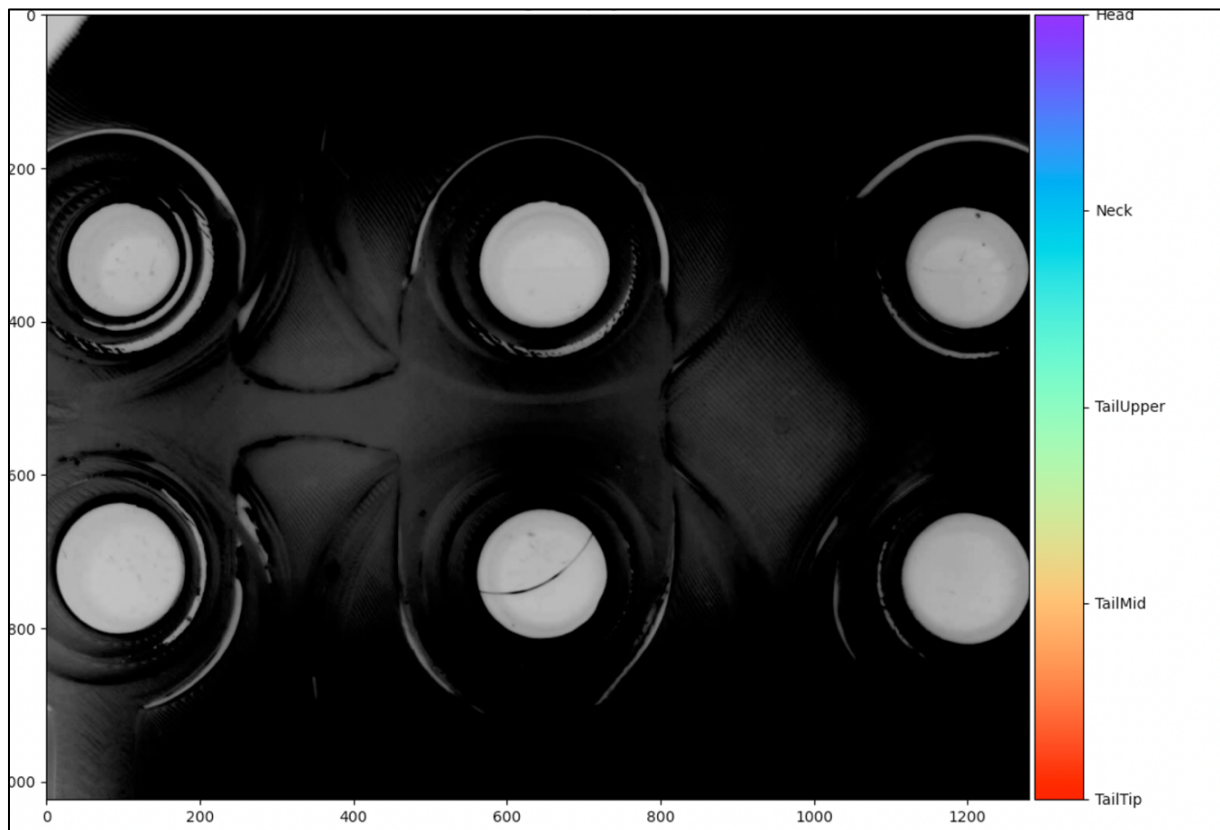
Ekstern veileder anbefalte at opptakene ble invertert for å få bedre kontrast mellom CI og bakgrunn. De ble kjørt gjennom et program som Chatzigeorgiou Group hadde utviklet for å invertere fargene på hele videoer. Selv om dette ikke er en del av gjennomføringen av DLC, har det likevel positivt utslag på sluttproduktet. Se Figur 4-64 for illustrasjon.



Figur 4-64 Eksempel på invertert bilde av CI.

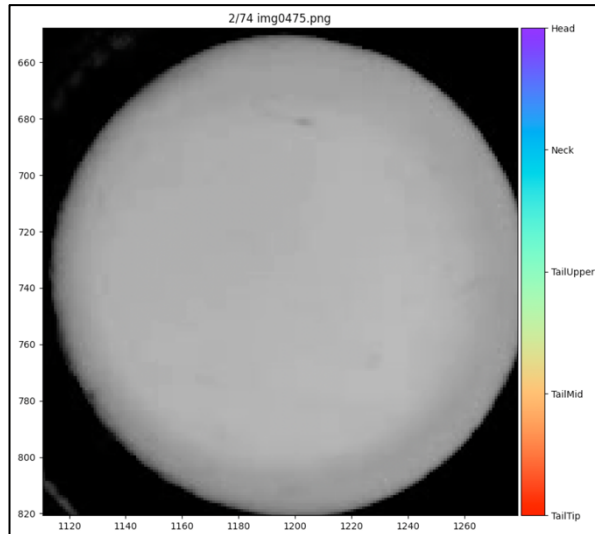
4.6.2 Markering

For å lære opp modellen må det hentes ut et bestemt antall bilder eller «frames», fra en video, og manuelt makere flere punkter på hver CI. DLC bruker da disse dataene til å trene opp modellen. Det er en del faktorer som påvirker nøyaktigheten til en modell som i hovedsak er antall bilder, antall videoer bildene er hentet fra, og hvor nøyaktig hver markering eller «labeling» er. Ettersom CI-ene blir så små på bildene er det valgt å markere fem punkter per CI. Det vil være hode, nakke og tre punkter på halen.

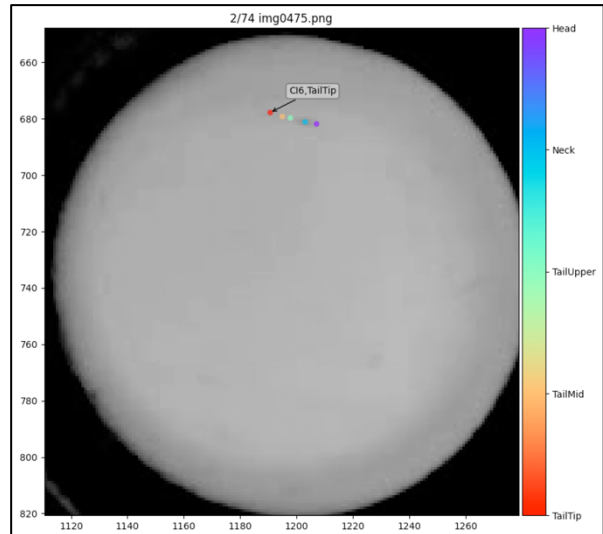


Figur 4-65 Én "frame" fra invertert video uten markeringer.

For at en modell skal bli nøyaktig nok er det anbefalt å ha mellom 100 til 150 bilder fordelt på flere videoer. Dette skaper mer variasjon i bildene og gir modellen mer å trene på ettersom det er flere unike tilfeller. Figur 4-67 og Figur 4-66 viser et forstørret bilde av en brønn før og etter markering.



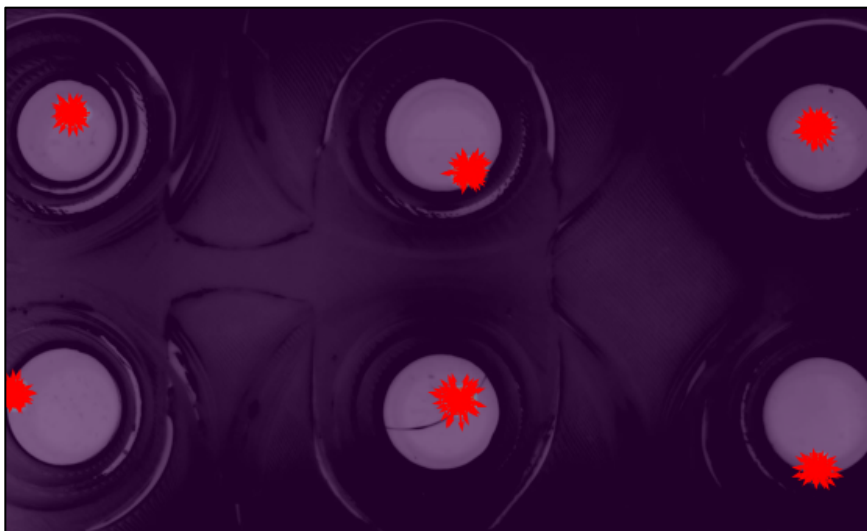
Figur 4-66 Forstørret "frame" uten markeringer.



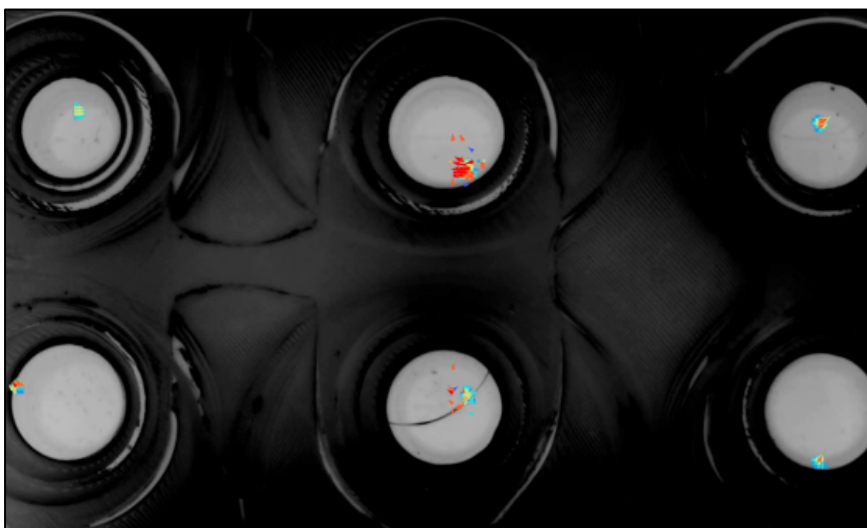
Figur 4-67 Forstørret "frame" med markeringer.

4.6.3 Trening

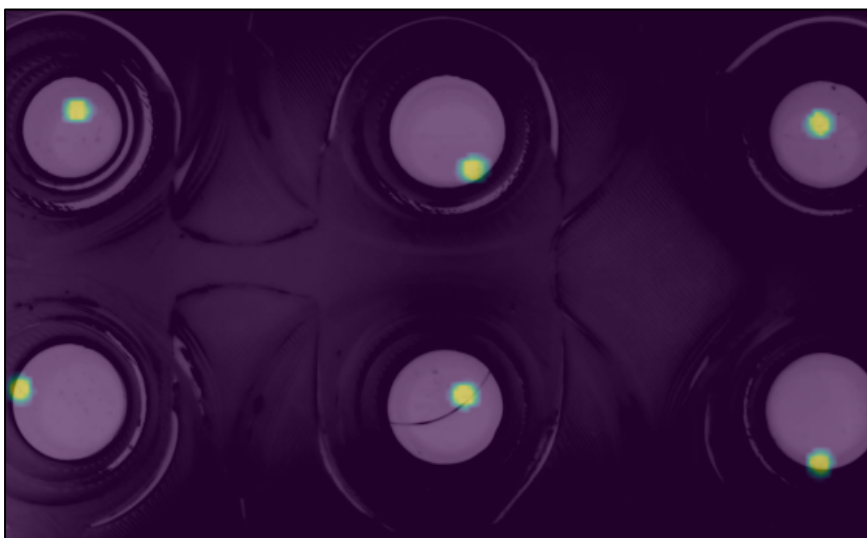
Etter at alle markeringer på alle bildene er ferdig kan treningen begynne, men det krever mye datakraft for å trene opp en modell. S13 disponerer en datamaskin med grafikkprosessor (GPU) som egner seg godt til oppgaver med store mengder grafikkdata og maskinlæring. I motsetning til en vanlig prosessor (CPU) er en GPU laget for å håndtere tunge matematiske beregninger [41]. Med mengden bilder og markeringer tar det opp mot ett døgn å trene opp en modell gjennom DLC. Etter fullført trening brukes modellen på en ny video for å se hvor godt den faktisk gjenkjenner en CI. Basert på treningen beregnes sannsynligheten for at en CI er i et område for et bestemt bilde. Figurene under illustrerer denne sannsynligheten fra en trent modellen. Se Figur 4-68, Figur 4-69 og Figur 4-70. Under kapittel 5.5 forklares opprenningen med antall bilder og videoer, og resultatet av dette.



Figur 4-68 Sannsynlig plassering, kart 1.



Figur 4-70 Sannsynlig plassering, kart 2.



Figur 4-69 Sannsynlig plassering, kart 3.

5 Testing

Dette kapittelet beskriver testing av det fullstendige systemet. Systemet skal testes slik at alt av bildekvalitet og komponenter fungerer, og skal i tillegg stress-testes for å se om både programvare og krets tåler påkjenningen det er laget for. Det beskrives en plan for testingen, hvordan den er utført og i tillegg håndtering av resultater. Målet med mye av testingen er å forbedre bildekvaliteten og kontrasten slik at videoen blir mest mulig leselig for DLC.

5.1 Test av bildekvalitet

Formålet med disse testene var å se om skarpheten og kontrasten mellom bakgrunn og CI-ene var god nok. Det er disse to faktorene som utgjør om bildekvaliteten til slutt blir bra nok for forskningen S13 skal gjennomføre.

Det første som ble testet var linser og avstandsstykker for å finne rett bildeareal i tillegg til fokusområdet. Ved å ha kameraet så nære reaksjonsbrettet som mulig kan man forhindre noen uklarheter ved zooming inn på en CI. For å få et skarpt bilde med så liten avstand til reaksjonsbrettet som mulig ble det testet med ulike linser og avstandsstykker. Etter en del testing ble det beste resultatet en 25 mm linse med et 5 mm avstandsstykke. Dette ga til slutt klart best skarphet i bildet med kameraet som ble brukt, og var en viktig observasjon.

For å øke kontrasten mellom CI og bakgrunn ble det også gjennomført flere tester som gikk på både bakgrunn og belysning i basen. For belysningen ble det først testet å ha lyskildene rett under ifra. Dette direkte lyset fra IR-lysene ga en skarp og ujevn belysning, og dermed ble løsningen for basen fra kapittel 4.2.6 valgt. Et lag med 3D-printet plast skilte lyset og fordelte det jevnt rundt i basen som ga CI-ene en tilstrekkelig belysning. For kontrasten ble det først brukt en hvit 3D-printet plate som bakgrunn, men mønsteret i platen var tydelig i bildet, og gjorde at kontrasten ikke var tilstrekkelig. Ettersom CI i larvestadiet er mer eller mindre gjennomsiktige, viste det seg at en sort bakgrunn ga god kontrast.

5.2 Test av maskinvare

Det ble gjort en del tester av hver stimuli-komponent i systemet og alt virket som det skulle. Etter at motorene ble festet til basen ble det gjennomført en visuell test for jevn vibrering i reaksjonsbrettet. Alle dyrene ble likt påvirket og resultatet var dermed positivt for mekanisk stimuli. En rekke tester ble gjort med formen som blir brukt til å lage brønnene i agarosen. Forskjellige vinkler på denne formen ga ulike resultater, der noen ga sterkt gjenskinn som påvirket bildekvaliteten. Etter flere tester med forskjellige utforminger, falt valget på formen beskrevet i kapittel 4.2.4. Denne ga den beste bildekvaliteten og det konkludert med at en brønn fylt tilstrekkelig med vann, førte til klare avgrensninger for dyret.

5.3 Test av programvare

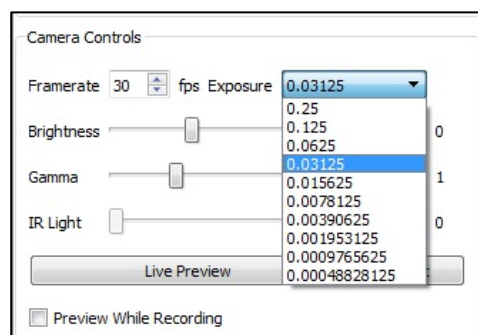
Ettersom det eksisterende oppsettet hadde et problem med at motorene ikke skrudde seg av når de skulle, ble det gjennomført en flere tester for å finne en løsning. Som nevnt tidligere ble motorene kontinuerlig slått på siden start og sluttsignalet var likt. Det gjorde at koden «satt fast» og fortsatte å tilføre mekanisk stimuli på nytt og på nytt. Resultatene etter endringer av programkoden var til slutt positive og programmet virket som det skulle etter oppgraderingene som ble gjort. Dette var en viktig del av prosjektet ettersom det både var positivt for det nye og det gamle oppsettet. Det gamle systemet ble nemlig også fikset da nye og oppdaterte programkode ble lastet opp på PC-en. For å se om det var flere ukjente feil, ble programmet kjørt med flere kombinasjoner. Resultatene her var også positive, og det ble ikke funnet flere store feil.

5.4 Siste systemtest

Her ble sluttproduktet testet og justert for gunstig bildekvalitet. Gjennom tilstrekkelige tester og dialog med ekstern veileder ble det konkludert at det trengs en sterk IR-belysning for å få en tilstrekkelig god kontrast. Dette vil si IR-lys-innstillingen burde være på 220-240 istedenfor 20 som det tidligere oppsettet benyttet. Dette er en skala som går fra 0 til 255. Videre ble det funnet ut at innstillingene som ga den beste bildekvaliteten var de som vist i Tabell 5-1.

Valg:	Innstilling:
Bildefrekvens	30
Eksposering	0.03125
Gamma	1
Lysstyrke	0
IR-lys	240

Tabell 5-1 Kamerainnstillinger



Figur 5-1 Eksponeringsinnstillinger.

Systemet ble avslutningsvis utsatt for en stresstest. Der ble alle stimuliene satt på fullt med ulike innstillinger over tid, uten at det skjedde noen feil. Stimuliene kom til rett tid og varte så lenge som valgt. Dette ble gjort flere ganger og programmet kjørte fint. Sluttproduktet viste seg pålitelig basert på testen som ble gjennomført.

5.5 Test av DLC

Siden bildekvaliteten til systemet er relativt dårlig, ble det først valgt å lage en enkel modell med lavt antall bilder for å se om programvaren klarte å fange opp CI-ene i det hele tatt. Siden den første modellen oppfattet bevegelsene overaskende godt, ble det laget enda en modell med langt flere bilder å jobbe med. En fellesfaktor for begge var at i enkelte bilder fra videoene var kvaliteten så lav at det var vanskelig å skille CI fra bakgrunn. For begge modellene ble det brukt standard innstillinger for både å lage treningsdatasett, og for trening av datasettet. Hovedforskjellen er antallet videoer og totalt antall bilder fra videoene som ble brukt. Det er disse to faktorene som gir størst utslag på kvaliteten av modellen.

Måten DLC trener en modell på er ved hjelp av et treningssett som sammenliknes opp mot et testsett. Treningsdataen består av 95% av de markerte bildene, mens test-dataen består av de resterende 5% av bildene. Etter treningen er ferdig regnes det ut et treningsavvik og et testavvik, og ved å sammenlikne dette avviket for en modell kan det gi en indikator på hvor god den trente modellen er. Om forskjellen mellom trening- og testavvik for en modell er liten kan dette også tyde på en relativt god modell. En perfekt modell vil ha null i avvik, men dette vil være svært vanskelig å oppnå.

5.5.1 Modell 1

Av den første modellen som ble trent opp ble det brukt én video og kun 40 bilder fra videoen, noe som var færre enn anbefalt antall bilder, men i hovedsak for å se om programvaren klarte å følge CI-ene. Med fem markeringer per dyr per bilde, ble det totalt 1200 forskjellige markeringer å trene ut ifra. Etter over 20 timer med trening, ga det likevel lite variasjon for modellen og gjorde at den presterte relativt dårlig på en «usett» video, men presterte overaskende bra på videoen de markerte bildene er hentet fra. Resultatet ble et treningsavvik på 2,99 piksler og et testavvik på 8,7 piksler, noe som passer med observasjonene gjort fra videoene.

Antall videoer	Antall bilder fra video	Totalt antall markeringer	Treningsavvik [px]	Testavvik [px]
1	40	1200	2,99	8,7

Tabell 5-2 Innstillinger og resultater DLC modell 1.

5.5.2 Modell 2

Den andre modellen ble trent opp ved hjelp av totalt 150 bilder fordelt på to forskjellige videoer. Med totalt 4500 markeringer hadde den langt mer data å gå på enn den første modellen, og ga etter over 30 timer med trening, et resultat som håndterte variasjoner bedre. Denne modellen presterte langt sterkere på «usette» videoer, men var relativt lik sammenliknet med den første modellen når det kom

til videoene bildene var hentet fra. De observerte resultatene gjenspeiler dette da treningsavviket ble 2,90 piksler og testavviket ble 4,58 piksler. Opptak fra begge modellene er lagt ved som vedlegg.

Antall videoer	Antall bilder fra video	Totalt antall markeringer	Treningsavvik [px]	Testavvik [px]
2	150	4500	2,90	4,58

Tabell 5-3 Innstillinger og resultater DLC modell 2.

6 Diskusjon

Det ferdige oppsettet med tilhørende programvare etterkommer alle krav og fungerer som det skal, likevel er det enkelte ting som kan forbedres i fremtiden. I dette kapittelet vil det diskuteres forslag til forbedringer til forskjellige deler av selve prosjektet, mens diskusjon rundt gruppearbeidet vil diskuteres i Appendiks B

6.1 Forslag til forbedringer

6.1.1 Bildekvalitet

Et problem som fulgte prosjektet, var kvaliteten i selve videoen som kameraet fanget opp. Da systemet ble utvidet og kameraet skulle filme et mye større arealområde, ble bildekvaliteten på hver av brønnene drastisk mye dårligere. Selv ved å plassere hver brønn så nære hverandre som mulig, ble allikevel ikke kvaliteten gunstig nok. Dette problemet var forventet ettersom siden hvis man øker synsfeltet vil bildekvaliteten oppleves noe uklart, og det krever dermed høyere kvalitet av kameraet. Uansett om kvaliteten ble dårligere, var fortsatt målet å få et godt nok bilde til at DLC klarte å fange opp lokasjonen og bevegelsene til dyret. Dette viste seg å fungere, men med en noe unøyaktig modell. Et forslag er å oppgradere kameraet til DMK-33UP5000 USB3.0 [42] som er et kamera med fem megapiksler, som gjør at bildekvaliteten blir langt bedre. Fordelen med dette kameraet er at det er av samme typen som det nåværende systemet bruker, og det vil kun være å plugge inn det nye for at det nye systemet skal virke. Et annet alternativ er å bruke en enkel petriskål som det kan lages til flere brønner. Problemet her er at det kun er plass til fire og ikke seks brønner i den enkle petriskålen S13 benytter, men det kan eventuelt brukes en noe utvidet petriskål som kan inneholde opptil seks brønner samtidig. Da vil kameraet kunne plasseres nærmere dyrene, som ville økt bildekvaliteten. Det enkleste vil være å bytte ut kamera som i tillegg påvirke DLC-modellen positivt. Selv om modellen fungerer, vil resultatet bli langt bedre med høyere oppløsning ettersom markering av bildene vil bli mer nøyaktig.

6.1.2 Trening av DLC-modell

Med høyere bildekvalitet burde det tas flere opptak med bredere variasjon, der hver CI er mer aktive for å sørge for at treningen består av så mange ulike hendelser som mulig. Ved å bruke flere enn to videoer vil dette også sørge for mer variasjon og en modell av høyere nøyaktighet.

6.1.3 Kjøring av programvare

Som nevnt tidligere kjører programmet på en datamaskin med Windows 7 uten internett. En oppdatering på operativsystemet eller annen programvare på PC-en vil føre til at ImMobilize slutter å fungere. Siden programvaren er så sårbar for oppdateringer hadde det vært gunstig å lage en ny kode som tillot oppdateringer. En løsning på dette kan være å bruke Docker [43]. Dette er en plattform som pakker programvare inn i en «boks» som gjør det mulig å kjøre et program på både Linux- og Windows-

baserte operativsystemer i form av en virtuell maskin. Ved å benytte seg av Docker og gjøre om programkoden, er ikke systemet begrenset til en spesiell PC noe som gjør det mindre sårbart.

7 Konklusjon

Denne bacheloroppgaven beskriver oppgraderingen av et datainnsamlingssystem for forskning på Ciona Intestinalis. Ved å bygge et nytt oppsett og justere eksisterende programvare har bachelorgruppen klart å lage et nytt oppsett som fungerer med det gamle. Det er i tillegg rettet opp enkelte feil med den gamle programvaren. Løsningen på oppgaven etterkommer alle hovedkravene formulert innledningsvis og har i tillegg brukerdokumentasjon og arkitekturtegninger, som var tilleggskrav.

Det svakeste leddet i det ferdige oppsettet var bildekvaliteten som hadde akkurat tilstrekkelig kvalitet til å fungere med DLC. Dette vil si at S13 kan benytte nåværende system, men burde investere i et kamera med bedre oppløsning for å gjøre forskningsarbeidet lettere og bedre.

Selv om bildekvaliteten viste seg å være en utfordring mot slutten, tror vi at Chatzigeorgiou Group kan benytte seg av løsningen og erfaringene våre i sin forskning og videreutvikling av systemet.

Bachelorgruppen har gjennom prosjektet lært ekstremt mye. Både faglige ting som noe programmering i Python og 3D-modelering, men også om planlegging og struktur. Ved å planlegge godt, forholde seg til egne frister og internjustis har gruppen kommet i mål med et produkt vi er fornøyde med.

Referanser

- [1] «Sars International Centre for Marine Molecular Biology», *University of Bergen*. <https://www.uib.no/en/sarssenteret> (åpnet 25. januar 2022).
- [2] «S_109-21Sarssenteret_Organisering_og_finansiering.pdf». Åpnet: 7. februar 2022. [Online]. Tilgjengelig på: https://ekstern.filer.uib.no/ledelse/universitetsstyret/2021/2021-09-09/S_109-21Sarssenteret_Organisering_og_finansiering.pdf
- [3] «Chatzigeorgiou Group», *University of Bergen*. <https://www.uib.no/en/sarssenteret/114771/chatzigeorgiou-group> (åpnet 14. januar 2022).
- [4] «*Ciona intestinalis*», *Wikipedia*. 25. desember 2021. Åpnet: 18. januar 2022. [Online]. Tilgjengelig på: https://en.wikipedia.org/w/index.php?title=Ciona_intestinalis&oldid=1061999272
- [5] «DeepLabCut», *The Mathis Lab of Adaptive Motor Control*. <http://www.mackenziemathislab.org/deeplabcut> (åpnet 7. februar 2022).
- [6] ChatzigeorgiouGroup, *imMobilize*. 2021. Åpnet: 27. januar 2022. [Online]. Tilgjengelig på: https://github.com/ChatzigeorgiouGroup/imMobilize/blob/c54856ad0fcd7d340864791d3004d814e043bce/Hardware/images/setup_incubator.jpg
- [7] A. Alcaraz, «*Ciona intestinalis*», *Animal Diversity Web*. https://animaldiversity.org/accounts/Ciona_intestinalis/ (åpnet 25. januar 2022).
- [8] «GitHub: Where the world builds software», *GitHub*. <https://github.com/> (åpnet 21. april 2022).
- [9] Fybikon.no, «Petriskål, plast, diameter: □90 mm, pk. á 10 stk - FYBIKON». <https://www.fybikon.no/laboratorieutstyr/plast-og-gummi/petriskaal-plast-diameter-90-mm-pk.-10-stk> (åpnet 27. januar 2022).
- [10] «Reaksjonsbrett, 6 brønner, plast, pk a 5». <https://no.frederiksen.eu/shop/product/reaksjonsbrett--6-broenner--plast--pk-a-5> (åpnet 27. januar 2022).
- [11] J. Rudolf, D. Dondorp, L. Canon, S. Tiew, og M. Chatzigeorgiou, «Automated behavioural analysis reveals the basic behavioural repertoire of the urochordate *Ciona intestinalis*», *Sci Rep*, bd. 9, nr. 1, s. 17, des. 2019, doi: 10.1038/s41598-019-38791-5.
- [12] «SANYO MPR-311D(H) SPECIFICATION SHEET Pdf Download | ManualsLib». <https://www.manualslib.com/manual/149006/Sanyo-Mpr-311d-H.html> (åpnet 10. februar 2022).
- [13] ChatzigeorgiouGroup, *imMobilize*. 2021. Åpnet: 15. februar 2022. [Online]. Tilgjengelig på: <https://github.com/ChatzigeorgiouGroup/imMobilize>
- [14] BillWagner, «C# docs - get started, tutorials, reference.» <https://docs.microsoft.com/en-us/dotnet/csharp/> (åpnet 2. mars 2022).
- [15] «cplusplus.com - The C++ Resources Network». <https://www.cplusplus.com/> (åpnet 28. februar 2022).
- [16] «Software». <https://www.arduino.cc/en/software> (åpnet 21. april 2022).
- [17] R. Pi, «Teach, Learn, and Make with Raspberry Pi», *Raspberry Pi*. <https://www.raspberrypi.org/> (åpnet 21. april 2022).

- [18] «DMK 33UP1300 Technical Reference Manual», s. 35.
- [19] «Fusion 360 for Personal Use | Fusion 360 | Autodesk». <https://www.autodesk.com/products/fusion-360/personal> (åpnet 2. mars 2022).
- [20] «3D Design Software | 3D Modeling on the Web», *SketchUp*. <https://www.sketchup.com/page/homepage> (åpnet 2. mars 2022).
- [21] «Tinkercad | Create 3D digital designs with online CAD», *Tinkercad*. <https://www.tinkercad.com/> (åpnet 2. mars 2022).
- [22] «Læringslab». <https://www.hvl.no/alu/medielab/laringslab/> (åpnet 2. mars 2022).
- [23] «Original Prusa i3 MK3S+ kit | Original Prusa 3D printers directly from Josef Prusa». <https://www.prusa3d.com/product/original-prusa-i3-mk3s-kit-3/> (åpnet 21. april 2022).
- [24] «Visual Studio Code - Code Editing. Redefined». <https://code.visualstudio.com/> (åpnet 3. mars 2022).
- [25] «General · havardfj/BO22EB-04-UpgradingDataAcquisitionSystem», *GitHub*. <https://github.com/havardfj/BO22EB-04-UpgradingDataAcquisitionSystem> (åpnet 23. mai 2022).
- [26] «prusa3d_manual_mk3s_en_3_11.pdf». Åpnet: 18. mai 2022. [Online]. Tilgjengelig på: https://www.prusa3d.com/downloads/manual/prusa3d_manual_mk3s_en_3_11.pdf
- [27] «DMK 33UP1300 - USB 3.0 monochrome industrial camera». <https://www.theimagingsource.com/products/industrial-cameras/usb-3.0-monochrome/dmk33up1300/> (åpnet 21. april 2022).
- [28] «Thorlabs - MVL25M1 25 mm EFL, f/1.4, for 1" C-Mount Format Cameras, with Lock». <https://www.thorlabs.com/thorproduct.cfm?partnumber=MVL25M1> (åpnet 19. mai 2022).
- [29] «Nano | Arduino Documentation». <https://docs.arduino.cc/hardware/nano> (åpnet 24. april 2022).
- [30] «sihl510.pdf». Åpnet: 19. mai 2022. [Online]. Tilgjengelig på: <https://www.vishay.com/docs/91297/sihl510.pdf>
- [31] «ACCU-SCOPE 3253 12V 20W Halogen Bulb», *New York Microscope Company*. <https://microscopeinternational.com/accu-scope-12v-20w-halogen-bulb-a3253/> (åpnet 25. april 2022).
- [32] «10 Pack Bioluz LED MR16 Socket, GU5.3 Socket, LED Halogen Lamp Ceramic Wire Connector Base Socket Adapter MR16 MR11 G4 GU5.3 Socket (Pack of 10) - Amazon.com». https://www.amazon.com/Bioluz-LED-Halogen-Ceramic-Connector/dp/B0743SYWGG/ref=sr_1_5?keywords=mr16%2Breplacement%2Bsocket&qid=1648122118&sr=8-5&th=1 (åpnet 25. april 2022).
- [33] A. Industries, «Vibrating Mini Motor Disc». <https://www.adafruit.com/product/1201> (åpnet 26. april 2022).
- [34] «Temperature probe ds18b20 & arduino compatible adapter». <http://www.velleman.eu/products/view/?id=439184&country=no&lang=no> (åpnet 27. april 2022).
- [35] «2 Pin Screw Terminal Block Connector».

https://www.twinschip.com/2_Pin_Screw_Terminal_Block_Connector (åpnet 27. april 2022).

[36] «Bryter 1pol mini», *Elektronikk-Service*. <https://webshop.elektronikk-service.no/produkt/bryter-1pol-mini/> (åpnet 27. april 2022).

[37] «TSR 1-2450 - DC/DC Converter, ITE, 1 Output, 5 W, 5 V, 1 A, TSR 1 Series». <https://no.farnell.com/tracopower/tsr-1-2450/converter-dc-dc-5v-1a-sip/dp/1696320> (åpnet 28. april 2022).

[38] «SRP JST SH 4-pin Connector 2mm Pitch (pack of 2) : Amazon.in: Industrial & Scientific». <https://www.amazon.in/SRP-4-pin-Connector-Pitch-pack/dp/B091Q1F3DP> (åpnet 27. april 2022).

[39] «Home — Spyder IDE». <https://www.spyder-ide.org/> (åpnet 19. mai 2022).

[40] *Installation: how to install DeepLabCut*. DeepLabCut, 2022. Åpnet: 5. mai 2022. [Online]. Tilgjengelig på: <https://github.com/DeepLabCut/DeepLabCut>

[41] «CPU vs. GPU: What's the Difference?», *Intel*. <https://www.intel.com/content/www/us/en/products/docs/processors/cpu-vs-gpu.html> (åpnet 6. mai 2022).

[42] «DMK 33UP5000 - USB 3.0 monochrome industrial camera». <https://www.theimagingsource.com/products/industrial-cameras/usb-3.0-monochrome/dmk33up5000/> (åpnet 6. mai 2022).

[43] «Home - Docker», 28. januar 2016. <https://www.docker.com/> (åpnet 6. mai 2022).

[44] «OpenCV: Introduction». <https://docs.opencv.org/4.x/d1/dfb/intro.html> (åpnet 19. mai 2022).

[45] «NumPy documentation — NumPy v1.22 Manual». <https://numpy.org/doc/stable/> (åpnet 19. mai 2022).

[46] «pandas documentation — pandas 1.4.2 documentation». <https://pandas.pydata.org/docs/> (åpnet 19. mai 2022).

[47] «PyQt5 Reference Guide — PyQt v5.15 Reference Guide». <https://www.riverbankcomputing.com/static/Docs/PyQt5/index.html> (åpnet 19. mai 2022).

[48] «Introduction — pyqtgraph 0.12.4.dev0 documentation». <https://pyqtgraph.readthedocs.io/en/latest/introduction.html> (åpnet 19. mai 2022).

[49] «pySerial — pySerial 3.4 documentation». <https://pyserial.readthedocs.io/en/latest/pyserial.html> (åpnet 19. mai 2022).

[50] «socket — Low-level networking interface — Python 3.10.4 documentation». <https://docs.python.org/3/library/socket.html> (åpnet 19. mai 2022).

Appendiks A Forkortelser og ordforklaringer

Liste over brukte forkortelser i alfabetisk rekkefølge.

CI	Ciona Intestinalis, Grønnsekkdyr i larvestadiet
CPU	Central Processing Unit, Prosessor
DLC	DeepLabCut
GPU	Graphics Processing Unit, grafisk prosessor
GUI	Grafisk Brukergrensesnitt
HW	Hardware, maskinvare
IDE	Integrated Development Environment
IR	Infra-red, Infrarød
LED	Lysemitterende Diode
PLA	Polylactic Acid (polymelkesyre)
PWM	Pulse Width Modulation, pulsbreddemodulasjon
Repo	Repository, oppbevaringssted (på GitHub)
S13	Chatzigeorgiou Group, forskningsgruppe 13 ved Sarsenteret
SW	Software, programvare

Appendiks B Prosjektledelse og styring

B.1 Prosjektorganisasjon

For organisering av gruppearbeidet delte vi opp oppgaven hvor alle hadde hvert sitt ansvarsområde. Hele gruppen samarbeidet likevel med tanker og gjennomføring av alle delene av prosjektet. Dennis hadde hovedansvaret for design av kretsen og lodding av denne. Håvard har fungert som gruppeleder og jobbet avslutningsvis med DLC, men underveis har det gått mye i rapportskrivning og design av enkelte figurer i SketchUp og montering av oppsett. Andreas sitt ansvarsområde havnet på 3D-modellering og designing, utskrift og lage til de nødvendige spesialtilpassede delene for prosjektet. Dette sikret god flyt og jevn fremgang på prosjektet. Ved siden av selve jobbingen med den fysiske løsningen, har alle skrevet i rapporten og dokumentert godt underveis. Innad i gruppen har kommunikasjonen gjennom hele prosjektet vært god. Kommunikasjon mellom veiledere og gruppen har foregått via e-post og har vært jevn gjennom hele prosjektet.

B.2 Risikoliste

Innledningsvis i prosjektet analyserte vi konsekvenser av ulike risikoer knyttet til prosjektet. Dette ga en indikasjon på hvilke tiltak som trengtes om uforutsette situasjoner oppstod.

	Risiko	Sannsynlighet	Konsekvens	Mulig løsning
1	Sykdom	Høy	Forsinkelse i arbeidet, andre gruppe-medlemmer smittes	Jobbe hjemmefra, munnbind der det jobbes tett.
2	Leveringstid på deler og komponenter	Middels	Forsinkelse i arbeidet, mangler i sluttresultatet.	Benytte komponenter som er lettere tilgjengelig, kjøpe i butikk og ikke på nett.
3	Skade på krets ved lodding	Middels	Forsinkelse i arbeidet, feil på krets, skader på komponenter.	Prøvelodde før det loddes på kretsen, bestille et spesialtilpasset kretskort.
4	Skade på oppsett ved montering	Lav	Forsinkelse i arbeidet, skader på komponenter utstyr og verktøy.	Lage de forskjellige delene av oppsettet hardføre, benytte rett verktøy for montering.
5	Få tilgang til Sarscenteret sent i prosjektet.	Lav	Får ikke startet arbeidet i tide, forsinkelser for hele prosjektet.	Tett dialog med Sarscenteret for å få korttilgang.

Tabell 7-1 Risikoliste.

B.3 Fremdriftsplan

Fremdriftsplanen som ble laget i forbindelse med forstudiet til prosjektet har blitt fulgt uten store avvik fra planen. Det var kun noe som tok lenger tid enn først antatt, og noe som kunne begynnes tidligere enn antatt. Det som tok lengst tid var å ferdigstille programvaren for det nye oppsettet, men dette hadde med at det var vanskelig å teste programvare skikkelig uten at systemet var ferdig. Med totalt

fire uker forsinkelse var dette det største avviket fra tidsplanen, men montering av oppsettet og dyplæringsmodellen tok kortere tid. Totalt gjorde dette at den ferdige løsningen ikke gikk ut over den totale tiden brukt i prosjektet. Hovedgrunnen for forsinkelsene i prosjektet har vært sykdom, da både Covid-19 og influensa har slått ut flere gruppe-medlemmer på ulike tidspunkter underveis i prosjektet.

Det ble lagt til en ekstra milepæl etter den trente modellen var ferdig i den faktiske tidsplanen. Dette var for å markere at hovedarbeidet med prosjektet var ferdig, kun mindre justeringer ble gjort etter dette. Se Tabell 7-2 og Tabell 7-3 for den planlagte og den faktiske tidsplanen.

Tidsplass Bachelor, BO22EB-O4		2022																										
Nr.	Aktivitet	Dato start	Dato slutt	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	Kommentarer
Uke	Man.	Fre.	7.1.	14.1.	21.1.	28.1.	4.2.	11.2.	18.2.	25.2.	4.3.	11.3.	18.3.	25.3.	1.4.	8.4.	15.4.	22.4.	29.4.	6.5.	13.5.	20.5.	27.5.	3.6.	10.6.	17.6.		
1	Undervisning - Metode	14.01.22	25.02.22																									
2	Forstudiearbeid	14.01.22	07.02.22																									
3	Forstudie - innlevering		07.02.22																									
	Bacheloroppgavearbeid																											
4	Kartlegge komponenter og programvare	24.01.22	04.02.22																									
5	Design rigg, i forhold til krav	31.01.22	04.02.22																									
6	Design kretser og det tekniske.	31.01.22	11.02.22																									
7	Plassering av kamera, base, sensorer og lys	07.04.22	18.02.22																									
8	Design/utvikle programvare	17.04.22	01.04.22																									
9	Måle/kutte/skaffe - spesialdel/deler	28.02.22	09.04.22																									
10	3D-printe spesialdel	07.03.22	18.03.22																									
11	Midtveispresentasjon	uke 12	uke 13																									
12	Ferdigstille programvare	14.05.22	01.04.22																									
13	Montere kamera, base, kretser og lys	28.04.22	05.04.22																									
14	Ferdigstille rigg	08.04.22	08.04.22																									
15	Teste fullstendig rigg	08.04.22	22.04.22																									
16	Sette opp DeepLabCut	25.04.22	06.05.22																									
17	Samle data	02.05.22	13.05.22																									
18	Trene modell av samlet data.	13.05.22	20.05.22																									
	Eksamensperiode	02.05.22	15.06.22																									
19	Ferdigstille rapport	13.05.22	27.05.22																									
20	Bacheloroppgave - innlevering		27.05.22																									
21	Bacheloroppgave - presentasjon		uke 22																									
22	EXPO		uke 24																									
Totalt Antall Timer pr. Student																												

Tabell 7-2 Planlagt fremdriftsplan.

Faktisk Tidsplan Bachelor, BO22EB-O4		UkeNr:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24				
Nr.	Aktivitet:	Dato start	2t		2t		2t		2t		2t		2t		2t		2t		2t		2t		2t		2t					
		Dato slutt																												
1	Undervisning - Metode	14.01.22	25.02.22																											
2	Forstudiearbeid	14.01.22	07.02.22																											
3	Forstudie - innlevering Bacheloroppgavearbeid		07.02.22																											
4	Kartlegge komponenter og programvare	26.02.22	02.02.22																											
5	Designe rigg, i forhold til krav	02.02.22	04.02.22																											
6	Designe kretser og det tekniske.	26.01.22	18.02.22																											
7	Plassering av kamera, base, sensorer og lys	07.02.22	18.02.22																											
8	Designe/utvikle programvare	25.02.22	22.04.22																											
9	Måle/kutte/skaffe - spesialdeler/deler																													
10	3D-printe spesialdeler																													
11	Midtveispresentasjon	uke 12	uke 13																											
12	Ferdigstille programvare	22.04.22	03.05.22																											
13	Montere kamera, base, kretser og lys	14.03.22	01.04.22																											
14	Ferdigstille rigg	01.04.22	05.04.22																											
15	Teste fullstendig rigg	06.04.22	28.04.22																											
16	Sette opp DeepLabCut	26.04.22	26.04.22																											
17	Samle data	29.04.22	29.04.22																											
18	Trene modell av samlet data.	02.05.22	10.05.22																											
	Eksamensperiode																													
19	Ferdigstille rapport	10.05.22	23.05.22																											
20	Bacheloroppgave - innlevering																													
21	Bacheloroppgave - presentasjon	uke 22	uke 22																											
22	EXPO	uke 24	uke 24																											
Totalt Antall Timer pr. Student																														

Tabell 7-3 Faktisk fremdriftsplan.

Appendiks C Brukerdokumentasjon

Systemet kan spore bevegelser, tilføre forskjellige former for stimuli som vibrasjoner og lys i alle mulige farger. Programmet skal også kunne gjenkjenne og estimere bevegelsene som dyrene gjør. Det skal også kunne lagre all dataen lokalt på datamaskinen.

C.1 Brukerinstruksjoner

Under beskrives oppstart og bruk av systemet.

1. Start datamaskinen.
2. Etter oppstart, åpne programmet ImMobilize som er plassert på skrivebordet på datamaskinen.
 - a. Denne vil så starte IDE-en Spyder3 hvor ImMobilize kjøres fra.
3. Start ImMobilize ved å trykke på den grønne pilen «run».
 - a. GUI-en «Stimuli & Trigger Delivery» vil da starte.
4. Fra «Camera and Serial Connection» kobles kamera og mikrokontroller til.
 - a. Et kamera har en tilhørende mikrokontroller. Dette er beskrevet i listen under.

Kamera: → Seriell Port:	
Kamera 1 →	USB-SERIAL CH340 (COM24)
Kamera 2 →	USB-SERIAL CH340 (COM16)
Kamera 3 →	USB-SERIAL CH340 (COM19)
Kamera 4 →	USB-SERIAL CH340 (COM11)
 - b. Trykk «Connect» for både kamera og mikrokontroller.
5. Systemet skal da være klart til opptak. Tabell 7-4 beskriver de forskjellige funksjonene programmet har.
6. Etter alt av innstillinger og stimuli er valgt under fanen «Experiment Settings», startes et forsøk ved å trykke «Start Experiment».
7. Avslutt imMobilize ved å krysse ut GUI-en.

Tabell 7-4 Lise over funksjoner i ImMobilize.

RGB-Stimuli	For å tilføre lys-stimuli med farger, juster på innstillingene som er vist på Figur 4-60, og deretter lagre det i tidslinjen for med å trykke på «Add stim» knappen. Det er også mulig å sette en starttid og varighet på lyset.
-------------	--

Hvitt Lys	Hvitt-lys-funksjonen har ingen styrkejustering og trenger bare å aktiveres ved å benytte seg av bryteren i Figur 4-60, som bytter mellom «LED stim» og «White Light». Deretter legges til med «Add to Stims»-knappen.
CameraControls	IR-lysene reguleres under «Camera Controls»-seksjonen Figur 4-57 og justeres direkte i sanntid med det fysiske systemet. Hvis IR-lysene settes til 0 utgir de ingen form for lys, men settes de til 255 utgir de sin fulle styrke. Under denne seksjonen er det også mulighet for å forandre bildefrekvensen til kameraet, eksponeringen, lysstyrke og gamma. «Preview»-knappen åpner opp et nytt vindu som viser hva kameraet ser. «reset»-knappen setter innstillingene i «Camera Controls»-seksjonen til standard innstillinger.
Vibrasjonsstimuli	Motoren har akkurat som RGB-lysene, en mulighet for å sette en starttid og varighet på hvor lenge funksjonen skal vare. Dette justeres under «VibrationStimuli»-seksjonen fra Figur 4-59. Det er også mulig å sette ønsket frekvens som motoren skal vibrere med. Deretter legger man stimuli til historien med å trykk på «Add to Stims»-knappen under samme seksjon.
MetaData	Metadata-seksjonen er hovedsakelig for å plassere lagringsplass for den innsamlede dataen samt sette visse verdier og navn på forskjellige deler av eksperimentet Figur 4-63. Øverst er det mulig å velge mellom auto-generert navn for eksperiment filen, eller med et eget definert navn. Rett under er det mulig å sette antall minutter og sekunder selve eksperimentet skal vare og dokumentere hvor mange dyr det blir plassert i hver brønn. Kan også dokumentere for klemme-tiden og om de er «dechorinated», samt er det mulig å sette egne navn for «genotype» og «drug name» inni tekstboksene.
ExperimentView	I eksperiment vinduet (Figur 4-61) viser oversikten over de forskjellige stimuliene som er lagt til i tidslinjen på eksperimentet. Etter en har lagt til ønsket stimuli lagrer man de ved å bruke «Save Stimuli Profile» knappen. Og det kan lett finnes på nytt igjen ved å bruke «Load Stimuli Profile» knappen. Hvis det er lagt til feil stimuli, kan dette fjernes ved å markere ønsket stimuli-søyle og bruke «Delete Stim»-knappen for å slette valgte stimuli eller «Delete All Stim»-knappen for å slette alle fra et eksperiment.
Serial-Connection	På nedtrekkmenyen i Figur 4-56 er det forskjellige valgene av hvilken Arduino-en kan koble seg til. Velg og trykk «Connect» knappen for å koble til. Når en er tilkoblet bytter samme knappen navn til «Disconnect», ved å presse denne knappen kobler programmet seg fra mikrokontrolleren. «Scan» knappen under «Serial Connection» seksjonen oppdaterer og leter etter nye com-porter i tilfelle programmet ikke oppdager tilkoblingen av et av kameraene. I tekstboksen under «scan» knappen i samme seksjon er de mulig å skrive inn kommandoer for å overstyre/teste forskjellige funksjoner. EKS: i255 setter IR-lyset til maksimum og i0 setter IR-lyset til 0. Avmerkingsboksen «Read Serial Live» aktiverer tilgangen til direktdata fra temperatursensorene, og viser dette på en graf.

LazyMode	I lazyMode-funksjonen setter man antall ganger programmet skal gjennomføre eksperimentet etter hverandre. Så skriv in tall og press «LazyMode» knappen.
Connect to Camera	Denne seksjonen fungerer ganske likt «serial-connection», men kobler til kameraer i stedet for Arduino. Og den har ikke tekstboks funksjonen.
VideoControls	Funksjonene under denne seksjonen er det samme Metadata bare uten klekkesid-funksjonen og seksjonene for navngiving av avmerking av «genetics» og «drugs».

C.2 Drifts- og vedlikeholdsdokumentasjon

For installering av programvare benytt dokumentasjon under filen «Readme.md» på GitHub:

<https://github.com/havardfi/BO22EB-04->

[UpgradingDataAcquisitionSystem/blob/main/Readme.md#software](https://github.com/havardfi/BO22EB-04-/blob/main/Readme.md#software)

I prosjektet blir det brukt forskjellige programvarer og digitale biblioteker. Alle nødvendige pakker som trenges for å kjøre ImMobilize er under filen «requirements.txt» som også er dokumentert GitHub-lagringsområdet i koblingen over. Under ligger en liste over denne filen.

Requirments.txt	Bruksområde
OpenCV [44]	Håndterer datasyn i sanntid.
NumPy [45]	Håndterer matematiske funksjoner.
Pandas [46]	Håndterer dataanalyse og datamanipulering.
PyQT5 [47]	Tillater å lage GUI kompatibel med flere operativsystemer.
PyQtGraph [48]	Brukes til vitenskapelig grafikk og GUI.
PySerial [49]	Sørger for støtte til serielle tilkoblinger.
Socket [50]	Muliggjør å sende informasjon mellom to enheter.

Tabell 7-5 Innhold i "Requirments.txt".

C.3 Vanlige Feil

Scan-knapp: Av og til når «scan»-knappen under «Connect to Camera» seksjonen ble aktivert, forsvinner noen av kameraene fra listen. For å få kameraene tilbake måtte man koble fra og til kameraet manuelt, ved å dra ut og inn kablet som er koblet mellom kameraet og PC-en.

Mangel på kameravalg-alternativ: Når det er mindre kamera-valgalternativ enn det faktisk er kamera tilkoblet oppstår det et problem med at kameraet sitt tilkoblingsvalg passer ikke med COM-portene som er standard fra tidligere. I den situasjonen gjelder dermed ikke koblingene som er vist i Tabell 4-3. Her også er det en ut og inn kobling av kameraet som må til.

Vibrasjonsstimuli: Siden input-strengen er 8 bits, men kun de fire siste bitsene blir brukt til motorens varighet, vil vibrasjonsstimuli ha en maksimal inputverdi på 9999 ms som tilsvarer rett i underkant av 10 sekunder varighet. Om ønsket lengre varighet, er det mulig å sette flere sekvenser etter hverandre.

Appendiks D Øvrige diagrammer og lister

D.1 Figurliste

Figur 1-1 Sars Logo (Foto/ill.: Sars Center)	9
Figur 1-2 Eksisterende løsning med tre oppsett. Tre kameraer filmer én CI hver. [6]	10
Figur 2-1 Petriskål [9] og reaksjonsbrett med seks brønner [10].	12
Figur 3-1 Skisse i 2D sett forfra.....	19
Figur 3-2 Skisse i 2D sett fra siden.....	19
Figur 3-3 Illustrasjon av kamera med filter	20
Figur 3-4 Skisse i 3D av oppsettet.	20
Figur 4-1 Sammenheng mellom programvare og maskinvare.	21
Figur 4-2 Skruer og mutter tilhørende rammen.	22
Figur 4-3 Hjørnebeslag.	22
Figur 4-4 Aluminiums-ramme.	22
Figur 4-5 Skisse i 2D forfra med mål.....	23
Figur 4-6 Skisse i 2D fra siden med mål.....	24
Figur 4-7 Ferdigmontert rammeverk i inkubator.	25
Figur 4-8 Rammeverk i inkubator. Venstre.	26
Figur 4-9 Rammeverk i inkubator. Høyre.	26
Figur 4-10 Oppvarmet agarose.....	27
Figur 4-11 Agaroseform.....	27
Figur 4-12 Ferdig agarose i reaksjonsbrett.....	28
Figur 4-13 Plate inni Basen.....	28
Figur 4-14 Base med hull på siden for IR-lys.	28
Figur 4-15 Øvre System - Fysisk.....	29
Figur 4-16 Fester til øvre rammeverk - 3D-Modell.....	29
Figur 4-17 Holder til IR-filter, underside.....	29
Figur 4-18 Holder til IR-filter, side.	29
Figur 4-19 Holder til lysbeskytter, og feste til Kamera.	30
Figur 4-20 Spektralsensitivitet til DMK-33UP1300.	31
Figur 4-21 DMK-33UP1300.....	31
Figur 4-22 MVL25M1 – 25 mm EFL, f/1.4.....	31
Figur 4-23 Arduino Nano.	32
Figur 4-24 Arduino i forhold til andre komponenter.....	33
Figur 4-25 MOSFET IRL510.	33
Figur 4-26 Krets av IR LED-er som festes rundt basen.	34
Figur 4-27 IR LED.	34
Figur 4-28 Krets for IR LED på koblingsbrett.	34
Figur 4-29 Lyspære i kontakt.	35
Figur 4-30 Krets for halogen lysstimuli.....	35
Figur 4-31 NeoPixel Ring 16.	35
Figur 4-32 Krets for RGB-lysstimuli.....	35
Figur 4-33 Motorer koblet.....	36
Figur 4-34 Vibrerende minimotor.	36
Figur 4-35 Krets for motorer.	36

Figur 4-36 Temperatursensor.....	37
Figur 4-37 Krets for temperatursensor.	37
Figur 4-38 Terminal.	38
Figur 4-39 Konverter.	38
Figur 4-40 Bryter.	38
Figur 4-41 Tre-pin kontakt.....	38
Figur 4-42 Krets terminal med bryter.....	38
Figur 4-43 Step Down Konverterer.....	39
Figur 4-44 Kretsboks for oppsettet.	39
Figur 4-45 Kretsboks uten lokk.....	39
Figur 4-46 Kretsboks underside.....	39
Figur 4-47 Fullstendig krets.	40
Figur 4-48 Plassering av kretsboks, side.....	41
Figur 4-49 Plassering av kretsboks.	41
Figur 4-50 Oppsett med kretsboks åpen. Alt tilkoblet.	42
Figur 4-51 Brukergrensesnitt "Camera and Serial Connection".	43
Figur 4-52 Brukergrensesnitt "Experiment Settings".	44
Figur 4-53 Programkode for å hente tilgjengelige kameraer.	44
Figur 4-54 Meny med tilgjengelige kameraer.	44
Figur 4-55 Programkode for å hente tilgjengelige serielle porter.	45
Figur 4-56 Meny med serielle porter.	45
Figur 4-57 Kamerainnstillinger og opptak.	46
Figur 4-58 Live Data.....	46
Figur 4-59 vibrasjonsstimuli.	46
Figur 4-60 Lysstimuli.....	46
Figur 4-61 ExperimentView.	47
Figur 4-62 Lazy Mode.	47
Figur 4-63 MetaData.	47
Figur 4-64 Eksempel på invertert bilde av CI.....	49
Figur 4-65 Én "frame" fra invertert video uten markeringer.	50
Figur 4-66 Forstørret "frame" uten markeringer.	51
Figur 4-67 Forstørret "frame" med markeringer.....	51
Figur 4-68 Sannsynlig plassering, kart 1.	52
Figur 4-69 Sannsynlig plassering, kart 3.	52
Figur 4-70 Sannsynlig plassering, kart 2.	52
Figur 5-1 Eksponeringsinnstillinger.	54
Figur 7-1 Tegning av krets.	75
Figur 7-2 Komponentliste med prisanslag.....	76

D.2 Tabelliste

Tabell 2-1 Krav og tilleggskrav.....	12
Tabell 4-1 Spesifikasjoner, Arduino Nano. [29]	32
Tabell 4-2 MOSFET IRL510, produktbeskrivelse.....	33
Tabell 4-3 Kamera/COM-port - tilkobling.....	45
Tabell 5-1 Kamerainnstillinger.....	54
Tabell 5-2 Innstillinger og resultater DLC modell 1.	55
Tabell 5-3 Innstillinger og resultater DLC modell 2.	56
Tabell 7-1 Risikoliste.....	64
Tabell 7-2 Planlagt fremdriftsplan.....	66
Tabell 7-3 Faktisk fremdriftsplan.....	67
Tabell 7-4 Lise over funksjoner i ImMobilize.....	68
Tabell 7-5 Innhold i "Requirments.txt"	70

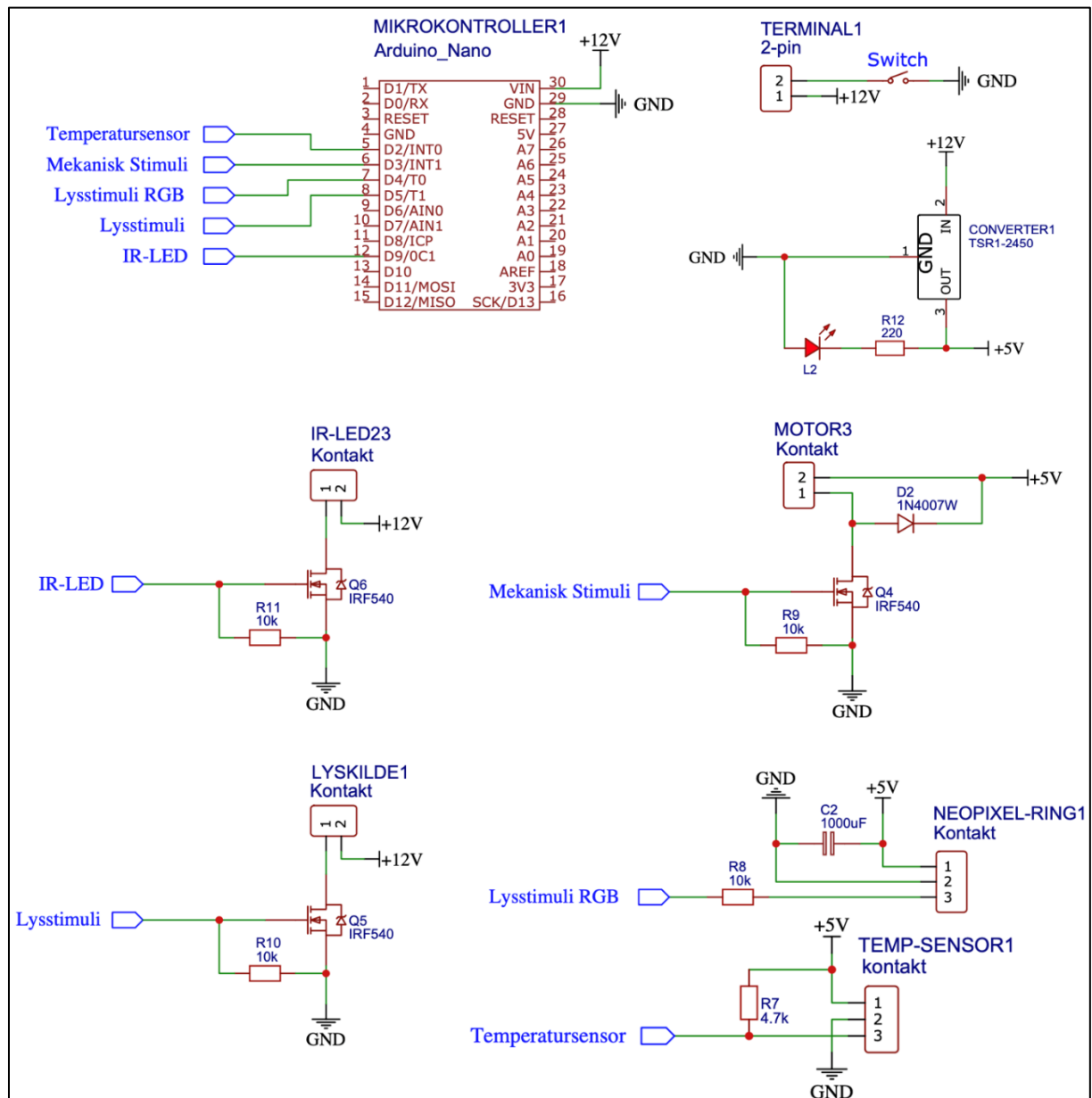
Appendiks E Kildekode, skjemedesign og komponentliste

E.1 Kildekode

Kobling til oppbevaringssted på GitHub:

<https://github.com/havardfj/BO22EB-04-UpgradingDataAcquisitionSystem>

E.2 Skjemadesign



Figur 7-1 Tegning av krets.

E.3 Komponentliste

Under finnes en fullstendig liste med alle komponenter brukt i prosjektet. Ettersom prosjektet ikke hadde noe fastsatt budsjett, er det kun gjort et anslag på hva de forskjellige komponentene og materialene ville ha kostet.

Komponenter Til Prosjekt				
Objekt	Spesifikt	Antall	Pris	Sum
Infrarødt Kamera	DMK 33UP1300	1	kr 5 582,00	kr 5 582,00
Infrarød Filter	69mm i diameter, 3mm høyde	1	kr 500,00	kr 500,00
Avstandsstykke	Thorlabs c-mount LensTube 5mm CML05	1	kr 250,00	kr 250,00
Linse 25mm	25mm c-mount 1/2 inch	1	kr 2 894,00	kr 2 894,00
USB-3 Kabel		1	kr 79,00	kr 79,00
Arduino nano		1	kr 100,00	kr 100,00
Temperatur Sensor	VMA324 DS18B20	2	kr 66,00	kr 132,00
IR Led-lys	1,5V 850nm	22	kr 14,00	kr 308,00
Motor	3V-5V Vibratino motor	2	kr 14,00	kr 28,00
Socket til halogen lyspære	Ceramic MR16 MR11 GU5.3 Socket	1	kr 7,00	kr 7,00
Halogen lyspære	12V 20W	1	kr 20,00	kr 20,00
Led- ring	NeoPixel 16 LED Ring	1	kr 89,00	kr 89,00
MOSFET	IRL510 MOSFET	3	kr 22,00	kr 66,00
Buck Converter	Travo Power TSR 1-2450	1	kr 60,00	kr 60,00
Diode	1N4007W	1	kr 1,00	kr 1,00
Terminal	PCB 2 plug termial	1	kr 30,00	kr 30,00
LED	grønn	1	kr 0,50	kr 0,50
Kondensator	1000µF	1	kr 11,00	kr 11,00
Koblingsbrett	PCB dobbelsidig 6cmx8cm	1	kr 10,00	kr 10,00
Små ledninger	ledninger til krets	1	kr 33,00	kr 33,00
Av- på knapp	Må tåle 12V	1	kr 30,00	kr 30,00
Strømtilkobling	PCB 12V Plug Jack	1	kr 3,00	kr 3,00
Strømledningn	12V 2A	1	kr 139,00	kr 139,00
USB min B		1	kr 40,00	kr 40,00
Aluminiums ramme	Ratrig Startpakke , 12 aluminumbeslag	1	kr 999,00	kr 999,00
Hjørnebeslag	følger med startpakken		kr -	kr -
Ledninger	RS pro, Multicore industrial cable [kr/m]	4	kr 23,80	kr 95,20
Motstand	10kΩ	3	kr 6,00	kr 18,00
	4.7kΩ	1	kr 1,00	kr 1,00
	330Ω	1	kr 1,00	kr 1,00
	250Ω	1	kr 1,00	kr 1,00
	220Ω	1	kr 1,00	kr 1,00
	50Ω	2	kr 1,00	kr 2,00
Pin kontakt	2way 1row straight pin header	3	kr 2,00	kr 6,00
	3way 1row straight pin header	2	kr 2,00	kr 4,00
Skruer	Diverse	10	kr 7,00	kr 70,00
3D-printet deler	en kg rull med 3d printer fillament	1	kr 260,00	kr 260,00
SUM				kr 11 870,70

Figur 7-2 Komponentliste med prisanslag.

Appendiks F Linker til opptak

F.1 Modell 1

Modell på video treningssettet er hentet fra: https://youtu.be/gSMTXIPcY_s

Modell på video av brønn 1: <https://youtu.be/gtIlg6ShP6D4>

F.2 Modell 2

Modell på video 1 treningssettet er hentet fra: <https://youtu.be/Gh1rCeDvJ-0>

Modell på video 2 treningssettet er hentet fra: <https://youtu.be/12jcuNM0jnA>

Modell på usett video: <https://youtu.be/4o6pCyBZruw>

Modell på video av brønn 1: <https://youtu.be/W2hpHFjYLO8>

Modell på video av brønn 6: <https://youtu.be/s5GdsWpnc5I>