

Vebjørn Sundli Alseth
Thomas Faack Bjørnstad

Deteksjon av fisk i video

Bacheloroppgave i Bachelor ingeniør - Data
Veileder: Marius Pedersen

Mai 2020

Vebjørn Sundli Alseth
Thomas Faack Bjørnstad

Deteksjon av fisk i video

Bacheloroppgave i Bachelor ingeniør - Data
Veileder: Marius Pedersen
Mai 2020

Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for datateknologi og informatikk



Kunnskap for en bedre verden

Forord

Prosjektgruppen ønsker å takke Norsk institutt for naturforskning (NINA) for en spennende og utfordrende oppgave. Prosjektet har vært veldig lærerikt for prosjektgruppen som sitter igjen med en utrolig god opplevelse. Prosjektgruppen håper at arbeidet som gruppen har utført kan være til nytte for NINA.

Prosjektgruppen vil også takke professor Marius Pedersen for god veiledning og assistanse gjennom prosjektet. Veiledningen har bidratt til at gruppen har innhentet mye ny kunnskap og opparbeidet en stor interesse for datasyn.

Innhold

Forord	iii
Figurer	viii
Kodelister	ix
Terminologi	x
Akronymer	xii
1 Innledning	1
1.1 Bakgrunn	1
1.1.1 Oppgavebeskrivelse fra oppdragsgiver	1
1.1.2 Grunnlag for oppgaven	2
1.2 Formål	4
1.2.1 Avgrensning	4
1.3 Prosjektmål	4
1.3.1 Effektmål	4
1.3.2 Resultatmål	4
1.3.3 Prosessmål	5
1.4 Roller	5
1.5 Rapportens struktur	6
2 Kravspesifikasjon	7
2.1 Funksjonelle krav	7
2.2 Ikke-funksjonelle krav	7
2.2.1 Plattform	7
2.2.2 Ytelse	7
2.2.3 Brukervennlighet	8
2.3 Aktivitetsdiagram	9
3 Metode	10
3.1 Målgruppe	10
3.1.1 For rapporten	10
3.1.2 For oppgaven	10
3.2 Egen bakgrunn og kompetanse	11
3.2.1 Hva vi kunne	11
3.2.2 Hva vi måtte lære	11
3.3 Rammer	11
3.3.1 Prosjektorganisering	11
3.3.2 Fremdriftsplan	14

3.3.3	Gjennomføring	15
3.4	Retningslinjer	15
3.4.1	Krav til dokumentasjon	15
3.4.2	Krav til kvalitetsgjennomganger	16
3.4.3	Krav til standarder og verktøy	17
3.4.4	Utviklingsmiljø og språk	17
3.4.5	Endringshåndtering	19
4	Bildebehandlings-bakgrunn	20
4.1	Eksisterende teknologi	20
4.1.1	Datasyn	20
4.1.2	Maskinlæring	20
4.1.3	Vårt valg	21
4.2	Bilde og video	21
4.2.1	Hvordan et digital kamera fungerer	22
4.2.2	Hva er et digitalt bilde	23
4.2.3	Hva er en video	24
4.3	Preprosesserings-teknikker	25
4.3.1	Nedskalering	25
4.3.2	Filtrering	26
4.3.3	Bakgrunnssubtraksjon	27
4.3.4	Morfologiske transformasjoner	29
4.3.5	Tilkoblede komponenter	30
5	Datsett	31
5.1	Datsett	31
5.1.1	Eksempler på forhold	34
5.1.2	Utvelging fra datsett	35
5.2	Etablering av 'ground truth'	37
5.2.1	Annotering med CVAT	37
5.2.2	Retningslinjer for annotasjon	38
5.2.3	Annotering av test- og utviklingssett	38
5.2.4	Filformat MOT ZIP 1.1	39
6	Testprosedyre	41
6.1	Produksjon av resultatfiler	41
6.1.1	Egenutviklet csv-format	41
6.1.2	Utskrift av resultatfiler	42
6.2	Resultatanalyse	44
6.3	Systematisering og analyse av testresultater	48
6.3.1	Boksplot av Jaccard index	48
6.3.2	Opprettelse av forvirringsmatriser	48
7	Deteksjonsalgoritme	50
7.1	Pipeline	50
7.1.1	Innlesing av bilde	50
7.1.2	Nedskalering	51
7.1.3	Bakgrunnssubtraksjon	51

7.1.4	Morfologiske operasjoner	51
7.1.5	Sammenkoblede komponenter	52
7.1.6	Konturer	52
7.2	Testresultater	52
7.2.1	Hurtigmodus	52
7.2.2	Saktemodus	53
7.2.3	Ekstern maskinlæringsalgoritme	55
7.3	Sammenligning av recall rates	57
7.4	Helhetlig vurdering, F1 score	61
8	GUI	64
8.1	Utvikling av design	64
8.1.1	Storyboards	64
8.1.2	Low-Fidelity Prototype	65
8.1.3	High-Fidelity Prototype	66
8.1.4	Colour Contrast Analyzer	67
8.2	Utvikling av grensesnitt	68
8.2.1	Kjøring gjennom IDE (PyCharm)	68
8.2.2	Kjørbar fil	68
8.2.3	Installasjonsfil	69
8.2.4	Valg underveis	69
8.3	Funksjonaliteter i GUI	69
8.3.1	Deteksjon	70
8.3.2	Testing	70
8.3.3	Utviklingsmodus	71
8.4	Brukertesting	71
8.4.1	Forbedringer av GUI	72
8.4.2	Konklusjon	73
9	Diskusjon	74
9.1	Oppfyllelse av resultatmål	74
9.2	Lisens	75
9.3	Forslag til videre arbeid	76
9.4	Kritikk av oppgaven	77
9.5	Evalueringsav gruppens arbeid	77
10	Konklusjon	79
A	Prosjektavtale	84
B	Gruppregler	88
C	Forprosjekt	90
D	Fisketelling Høyegga	124
E	Utførelse	140
F	Brukertesting Referat	150
G	Brukerguide	153
H	Møtereferat	179

Figurer

1.1	Fisketrapp med spalteåpning hentet fra CEDREN [5]. Foto: Jon Mutheth.	3
1.2	Eksempel på bilde med fisk fra videomateriale.	3
2.1	Figuren viser flyten ved kjøring av deteksjon på filer valgt av brukeren.	9
3.1	Gantt-diagram over prosjektet.	14
3.2	Demonstrasjon av hvordan verktøyet CCA [13] kan sjekke for kontraster mellom tekst og bakgrunn.	17
3.3	PyCharm sin <i>Project Interpreter</i> hvor gruppen har installert pakke- ne som prosjektet krever.	18
4.1	Illustrasjon av enkelte viktige komponenter i digitale kameraer, hen- tet fra GadgetReview [24].	22
4.2	Skjerm bilde av Fisk med original oppløsning på 1920 x 1080 piksler	25
4.3	Skjerm bilde av Fisk med nedskalert oppløsning på 640 x 360 piksler	26
4.4	Ved å zoome inn så er det mulig å se hvordan det nedskalerte bildet har mistet en del data og ser mer uskarpt ut enn det originale bildet.	26
4.5	Sammenligning av ulike filtreringsmetoder gjort på video <i>DevGod1.m4v</i> .	27
4.6	Skjerm bilde av hvordan funksjonen for bakgrunnsuttrekk fungerer i et bilde fra videomaterialet hvor 150 rammer benyttes for bakgrunnsmodelleringen.	28
4.7	Skjerm bilde av hvordan funksjonen for morfologisk transformasjon fungerer i et bilde fra videomaterialet med tilpassede parametere. . .	29
4.8	I et bilde har hver piksel åtte nabopiksler. Inspirert av [40].	30
4.9	Skjerm bilde av hvordan funksjonen for tilkoblede komponenter fun- gerer i et bilde fra videomaterialet med tilpassede parametere. . . .	30
5.1	Illustrasjon over fordelingen av videoer over de ulike mappene og diskene.	32
5.2	Illustrasjon over fordeling mellom dag og natt i videomaterialet. . .	32
5.3	Illustrasjon over fordeling av sikt i de ulike mappene og diskene. . .	33
5.4	Illustrasjon over klassifiseringen som er gjort av NINA på disk 1. . .	33

5.5	Eksempler på skjermbilder av fisker på ulike tider av døgnet i forskjellige forhold.	34
7.1	Boksplokk av Jaccard Index fra testsett, modus hurtig	53
7.2	Boksplokk av Jaccard Index fra testsett, modus sakte	55
7.3	Barplot av recall score fra de tre ulike algoritmene kjørt på KlarSiktMyeTurb 1-4	57
7.4	Barplot av recall score fra de tre ulike algoritmene kjørt på OkSiktMiddelsTurb 1-4	59
7.5	Barplot av recall score fra de tre ulike algoritmene kjørt på BrukbarSiktLiteTurb 1-4	60
7.6	Barplot av recall score fra de tre ulike algoritmene kjørt på Dårlig 1-2, Elendig 1-2	60
7.7	Barplot av recall score fra de tre ulike algoritmene kjørt på NattGodSikt 1-2, NattOkSikt 1-2	61
7.8	Barplot av F1-score fra de tre ulike algoritmene kjørt på KlarSiktMyeTurb 1-4	61
7.9	Barplot av F1-score fra de tre ulike algoritmene kjørt på OkSiktMiddelsTurb 1-4	62
7.10	Barplot av F1-score fra de tre ulike algoritmene kjørt på BrukbarSiktLiteTurb 1-4	62
7.11	Barplot av F1-score fra de tre ulike algoritmene kjørt på Dårlig 1-2, Elendig 1-2	63
7.12	Barplot av F1-score fra de tre ulike algoritmene kjørt på NattGodSikt 1-2, NattOkSikt 1-2	63
8.1	Prototype på ark for brukergrensesnittet.	65
8.2	Digital prototype i Adobe XD for brukergrensesnittet.	66
8.3	Ulike kontrastnivåer i det grafiske brukergrensesnittet.	67
8.4	Skjermbilde av Vindu 1 og 2 før og etter forbedringer fra brukertestene.	73

Kodelister

5.1	DevNatt1.csv Utdrag	39
6.1	TestDeteksjonGUI.py oppretting av mappe for resultatfiler	42
6.2	Fra TestDeteksjonGUI.py, henting av koordinater for detekterte rektangel	42
6.3	Fra TestDeteksjonGUI.py, opprettelse og lagring av CSV-linje	43
6.4	TestDeteksjonGUI.py skriving av resultatlinjer til fil	43
6.5	TestDeteksjonGUI.py start av timer for deteksjon	43
6.6	Fra TestDeteksjonGUI.py, skriving av metadata til fil	44
6.7	DevPoor1.csv utdrag	44
6.8	funksjoner.py multifilResultatAnalyseUtvikling	45
6.9	Funksjon resultatAnalyse, opprettelse av CSV-lesere og henting av metadata	45
6.10	Funksjon resultatAnalyse, opprettelse av lister og lesing av fasitfil	46
6.11	funksjoner.py resultatAnalyse	46
6.12	funksjoner.py jaccardIndexPolygons	47
6.13	Produksjon av boksplokk for testresultater	48
6.14	Opprettelse av forvirringsmatriser, skriving til fil	49
6.15	Confusion_matrix.txt etter å ha kjørt BrukbarSiktLiteTurb1 og BrukbarSiktLiteTurb2 fra testsett med modus sakte	49
7.1	Hardkodete argument for deteksjonsalgoritme, hurtig modus	50
7.2	Innlesing av bilde fra videofil	51
7.3	Nedskalering av bilde	51
7.4	Bakgrunnssubtraksjon	51
7.5	Morfologisk åpning av bilde	52
7.6	Fjerning av ikke-koblede komponenter	52
7.7	Konturdeteksjon i rammen	52
7.8	Hardkodete verdier for deteksjonsmodus hurtig	52
7.9	Hardkodete verdier for deteksjonsmodus sakte	53

Terminologi

C++ er et mye benyttet høynivå programmeringsspråk, da C++ bygger på programmeringsspråket C og det er implementert for de fleste operativsystemene, men har en mer komplisert syntaks enn programmeringsspråket Python.

Colour Contrast Analyzer er et analyseverktøy som analyserer kontrasten i gruppens løsning. Verktøyet er enkelt å ta i bruk og gir brukeren informasjon om kontrasten mellom elementer i applikasjonen.

CVAT (Computer Vision Annotation Tool) er et gratis, åpen kildekode, web-basert bilde og video annoterings-verktøy som brukes til annotering av videofiler.

GANTT er en type søylediagram som illustrere prosjektet tidsplan. GANTT-skjemaet illustrer start- og sluttidspunkter for ulike oppgaver, men også milepæler for når en oppgave skal være oppnådd.

LaTeX er et typesettingssystem for dokumentproduksjon. Blir ofte tatt i bruk ved vitenskapelige rapporter da systemet gjør referering og generering av innholdsfortegnelser, lister med tabeller, kryssreferering med mer, enkelt.

MATLAB (Matrix Laboratory) er et matematikkprogram med et helt eget script-språk basert på C. MATLAB kan utvides med flere verktøykasser og ettersom det tilbyr mye funksjonalitet enkelt, så har det blitt populært blant ingeniører.

Numpy er et bibliotek for programmeringsspråket Python, og støtter store flerdimensjonale matriser, sammen med en stor samling matematiske funksjoner på høyt nivå for å operere på disse matrisene.

NSIS (Nullsoft Scriptable Install System) er et skriptdrevet installasjonsforfatter-verktøy for Microsoft Windows. Verktøyet benyttes for å skape installasjonsfil av en *zip*et mappe og er kompatibelt med GPL-lisensen.

OpenCV (Open Source Computer Vision Library) er bibliotek med funksjoner hovedsakelig rettet mot datasyn. Biblioteket er gratis for bruk under åpen kildekode BSD-lisens.

PyCharm er et integrert utviklingsmiljø (IDE) som brukes i programmering av datamaskiner, spesielt for Python. Programmet er utviklet av det tsjekkiske selskapet JetBrains.

PyInstaller er et verktøy som gjør det mulig å lage frittstående kjørbare filer ut ifra Python-applikasjoner. Verktøyet sparer brukere fra å laste ned tilhørende bi-

blioteker som Python-filen ellers ville hatt behov av for å kjøre.

Python er et objektorientert programmeringsspråk som er utviklet med hensyn på lettlesthet og en klar syntaks.

Scrumban er en smidig styringsmetodikk som er en hybrid av Scrum og Kanban. Scrumban ble opprinnelig designet som en måte å overføre fra Scrum til Kanban.

Toggl er en tidssporingsapp som drives av Toggl OÜ. Toggl tilbyr enkel tidssporing og rapporteringstjenester på nettet, eller ved bruk av mobile og stasjonære applikasjoner.

Akronymer

CCA - Color Contrast Analyzer

CLAHE - Contrast-Limited Adaptive Histogram Equalization

CSV - Comma-separated values

CVAT - Computer Vision Annotation Tool

GUI - Graphical User Interface

Hi-fi - High-fidelity

IDE - Integrated Development Environment

IDI - Institutt for datateknologi og informatikk

Lo-fi - Low-fidelity

MB - Megabyte

NINA - Norsk Institutt for Naturforskning

NSIS - Nullsoft Scriptable Install System

NTNU - Norsk Institutt for Teknologi og Naturvitenskap

PNG - Portable Network Graphics

TB - Terrabyte

UI - User Interface

WCAG - Web Content Accessibility Guidelines

Innledning

1.1 Bakgrunn

Bacheloroppgaven avslutter studieprogrammet for Bachelor i ingeniørfag data hos NTNU, og skal integrere viktige deler av programmets faglige innhold [1]. Oppgaven skal derfor ta for seg reelle problemstillinger og bidra til innføring i vitenskapsteori og metode. Oppgaven skal resultere i en rapport som på en strukturert måte dokumenterer arbeidet som er gjennomført. Prosjektgruppen har blitt tildelt en oppgave som skal gjennomføres på vegne av Norsk Institutt for Naturforskning (NINA), å utvikle et verktøy for deteksjon av fisk i video. NINA er en uavhengig stiftelse som forsker på natur og samspillet natur – samfunn [2].

I starten av gjennomføringen av bacheloroppgaven gjøres det et forprosjekt, hvor gruppene planlegger hvordan utførelsen av selve bacheloroppgaven skal gjøres og hvilke rammer og mål oppgaven skal ha. Dette dokumentet har som mål å belyse disse punktene på en oversiktlig og enkel måte.

1.1.1 Oppgavebeskrivelse fra oppdragsgiver

Tilstanden i norske elver og innsjøer dokumenteres blant annet gjennom overvåking av fiskebestandene. Overvåkingsmetodene er mange, og valget av en bestemt metode avhenger av formålet med overvåkingen og de fysiske forholdene i vannmiljøet. Et viktig prinsipp er at overvåkingsmetodikken skal være rasjonell og at den ikke skal medføre unødig lidelse hos organismene som overvåkes. Norge henter mesteparten av sin kraftproduksjon fra vannkraft. I elvekraftverk medfører kraftverksdammer en barriere for fisk og andre vannlevende organismer som er avhengige av å bruke ulike deler av elva gjennom sine livsløp. Som et avbøtende tiltak er mange dammer utstyrt med fisketrapper som skal tillate oppstrøms passasje. Et viktig formål er derfor å overvåke om fisketrappene faktisk fungerer som de skal. Figur 1.1 viser hvordan en fisketrapp med en spalteåpning kan se ut.

I mange elver står det såkalte VAKI fisketellere som teller og tar opp video av all forbipasserende fisk. Disse tellerne er et godt verktøy for overvåking, men de har også sine begrensninger. De krever blant annet stor plass for montering og de kan

trolig være vandringshindere for fisk ettersom de må svømme gjennom selve VAKI maskinen. Et lite videokamera montert i fisketrappen er potensielt et godt verktøy for å overvåke aktiviteten i og rundt fisketrapper. Figur 1.2 viser et eksempel på aktivitet som kameraet kan fange opp. Utfordringen er imidlertid å prosessere videomaterialet. Dette har til nå blitt gjort manuelt, men er så tidkrevende at det i fremtiden vil være behov for avansert programvare for at metodikken skal være kostnadseffektiv og brukervennlig.

NINA ønsker å få laget et verktøy/programvare som automatisk kan detektere, helst i sanntid, når det kommer fisk inn i bildet. Gjenkjenningen skal være robust og fungere under alle slags forhold (forskjellig lysforhold, farge på vannet, uavhengig av mengde luftbobler, etc.). Det er også ønskelig å kunne telle antall fisk som passerer, artsbestemmelse (f.eks. om det er ørret, harr eller sik), og hvilken retning fisken har (oppover eller nedover elva). Det er også ønskelig med et brukergrensesnitt til verktøyet.

1.1.2 Grunnlag for oppgaven

Høyegga dam er en 175 meter lang betongdam med en maksimal høyde på 10 meter. For at fisk skal kunne passere den store demningen så er det blitt bygd en fisketrapp som skal gjøre det mulig for å fisk å fortsette oppover Glomma. Fiske-trappen var av typen klassisk kulpetrapp, noe som forutsatte at fisken måtte klare å hoppe fra kulp til kulp [3]. Da det er andre arter enn ørret og harr som også skal oppover elven, men som ikke besitter de samme ferdighetene til å hoppe oppover i fisketrappen, så ble det besluttet at Høyegga, som en del av SAFEPASS-prosjektet [4], skulle bygge om fisketrappen [5].

Kraftverkene kunne også bygd en helt ny fisketrapp som ville hatt spesifikasjoner som ville tilfredsstilt ferdighetene til alle artene, men da en vanlig fisketrapp fort kan koste i størrelsesorden 10 til 100 millioner kroner, så vil en mindre modifikasjon av allerede eksisterende trapp være mye mer aktuelt.

Det er i dag ønskelig å gjøre så små grep i naturen som mulig og det er på bakgrunn av dette at NINA har utplassert et kamera rettet mot den ene spalteåpningen for å kunne føre statistikk på fisk som passerer oppover elven. Om statistikken gir inntrykk for at modifikasjonene gjort med fisketrappen fungerer, så vil eierne av kraftverkene slippe å lage mange nye fisketrapper, men heller kunne gjøre liknende modifikasjoner på allerede eksisterende fisketrapper.

Arbeidet med å holde kameraet gående har vært relativt enkelt og siden kameraet står på døgnet rundt, så har oppgaven med å samle inn materiale vært utført passivt. I dag samles videomateriale inn for at personer skal sette seg ned å gå gjennom hvert eneste sekund med innhentet videomateriale, for å så føre statistikk på hva som passerer, hvilken retning fisken har og andre eventuelle hendel-



Figur 1.1: Fisketrapp med spalteåpning hentet fra CEDREN [5]. Foto: Jon Museseth.

ser som skulle forekomme. Ved å overvåke fiskevandring med bruk av kamera så produseres det veldig mange timer med videomateriale. Innsamling av videomaterialet er billig og enkelt, men ikke særlig mye brukt da det krever mye ressurser for å bearbeide materialet. Denne oppgaven er derfor veldig tidkrevende og setter krav til at brukeren som ser gjennom videomaterialet og fører statistikken må sitte kontinuerlig foran skjermen.



Figur 1.2: Eksempel på bilde med fisk fra videomateriale.

1.2 Formål

Formålet med oppgaven er å utvikle et verktøy som skal avlaste arbeidsmengden hos ansatte ved NINA ved å automatisere prosessen med å finne forekomster av fisk i videomaterialet produsert i laksetrappen i Høyegga. Dette innebærer at det skal utvikles en algoritme for fiskedeteksjon som implementeres i et format som gjør det praktisk mulig for ansatte hos NINA å bruke dette regelmessig, med en treffsikkerhet som er god nok til å benyttes for reell forskning.

1.2.1 Avgrensning

I møte med NINA i slutten av januar så ble ønsker for funksjonalitet presentert for gruppen. Prosjektgruppen ble i etterkant av møtet med NINA redusert til to medlemmer slik at prosjektoppgaven måtte avgrenses i forhold til dette.

Prosjektgruppen skal utvikle et verktøy som skal gjennomføre deteksjon av fisk i video i henhold til oppgavebeskrivelsen, men ikke gjøre artsbestemmelse eller gjenkjenning av enkeltindivider. Videomaterialet som gruppen skal arbeide med er kun fra ett område, hvor kun ett kamera har vært benyttet. Det grafiske brukergrensesnittet (GUI) skal være en del av den endelige løsningen, hvor brukervennlighet vil bli mer vektlagt enn designet av GUIen. Verktøyet skal i sanntid, ikke nødvendigvis hurtigere, gjennomføre deteksjon av fisk, før output blir eksportert i et .CSV-støttet format. Løsningen skal overleveres NINA, men gruppen skal ikke gjennomføre opplæring av NINAs ansatte.

1.3 Prosjektmål

1.3.1 Effektmål

- Eliminere behov for at personale eller innleid hjelp behøver å lete gjennom videomateriale fra start til slutt for å finne fisk
- Redusere den totale aktive tiden som brukes av ansatte hos NINA til innhenting av statistisk materiale fra fiskevideo med minst 80% i forhold til i dag.

1.3.2 Resultatmål

Utarbeide et verktøy for NINA som:

- Gir output i form av en .CSV-fil som kan importeres direkte inn i Microsoft Excel
- Ved gjennomkjøring av en tilfeldig videofil fra utlevert datamateriale fra 2017 i gjennomsnitt produserer tidsstempel som peker til minst 95% av fiskene registrert av manuell gjennomgang [6]

- Skriver ut tidsstempel som i 95% av tilfellene viser til et punkt i videomaterialet som har minst én fisk i et tidsvindu på +- 5 (fem) sekunder.
- Ved kjøring maksimalt bruker 1:1 total prosesseringstid per tid video ('sann-tid' prosessering)
 - Gitt at verktøyet kjører på hardware som er minst like kraftig som laptopene NINA bruker til daglig [7]
 - Gitt at original video er i formatet 1920 x 1080 med 25 bilder i sekundet
- Har et brukergrensesnitt som tilfredsstillt kravet om AA-sertifisering i henhold til kravet om universell utforming - WCAG 2.1 [8].
- Kun bruker videoformat (både som input og eventuell output) som støttes av VLC Media Player [9]
- Kan kjøres på dagens mest brukte operativsystemer
 - Windows 7 og nyere
 - MacOS Catalina og nyere

1.3.3 Prosessmål

- Gruppens medlemmer skal utvikle samarbeids- og kommunikasjonsevne til et profesjonelt nivå
- Gruppens medlemmer skal få økt kunnskap og erfaring innen datasyn og utvikling av brukergrensesnitt
- Gruppen skal få en karakter alle gruppens medlemmer er fornøyd med

1.4 Roller

Etter gjennomføring av forprosjektet så ble gruppen redusert til kun to medlemmer. Som en konsekvens av redueringen har den tidligere mer bestemte rollefordelingen, blitt mer flytende. Gruppen valgte å ha klare roller for å kunne differensiere arbeidet mellom grupped medlemmene og fortsatt opprettholde strukturen i gruppen.

Interne roller:

- Gruppeleder: Thomas Faack Bjørnstad
- Dokumentasjonsansvarlig: Vebjørn Sundli Alseth
- Ansvarlig for brukergrensesnitt og design: Thomas Faack Bjørnstad
- Utviklingsansvarlig: Vebjørn Sundli Alseth

Eksterne roller:

- Oppdragsgivers kontaktperson: Knut Marius Myrvold
- Veileder: Marius Pedersen

1.5 Rapportens struktur

Kapittel 1: Innledning

Kapittel 1 er en introduksjon til oppgaven og rapporten. Inneholder motivasjonen bak å gjennomføre oppgaven, hvem oppgaven er ment for og noe om kravene som er satt av arbeidsgiver.

Kapittel 2: Kravspesifikasjon

Kapittel 2 beskriver egenskapene til produktet og hvordan produktet skal oppføre seg under utførelse av ulike oppgaver.

Kapittel 3: Metode

Kapittel 3 beskriver hvilken fremgangsmetode prosjektgruppen har valgt å ta i bruk for gjennomføring av prosjektet.

Kapittel 4: Bildebehandlings-bakgrunn

Kapittel 4 beskriver teoretisk hvilke fagfelt som er relevante for å løse oppgaven, hva et digitalt bilde er og hvordan bildebehandling fungerer i praksis.

Kapittel 5: Datasett

Kapittel 5 beskriver innholdet i videomaterialet som er blitt utlevert og hvordan prosjektgruppen har gjort utvalg fra datasettet til test- og utviklingssett. Prosessen med å annotere datasettene og filformatet for dette blir også omtalt.

Kapittel 6: Testprosedyre

Kapittel 6 inneholder informasjon om testprosedyren til gruppen; en beskrivelse av selvutviklet filformat, hvordan hente ut resultater fra deteksjonsalgoritmen i et kompatibelt format og få sammenlignet disse med fasitfilene.

Kapittel 7: Deteksjonsalgoritme

Kapittel 7 beskriver hvordan deteksjonsalgoritmen fungerer, resultater fra testing på algoritmen og noe analyse rundt dette.

Kapittel 8: Grafisk brukergrensesnitt

Kapittel 8 beskriver utviklingen av det grafiske brukergrensesnittet, både utviklingen av designet og kodingen av GUIen, og viktigheten av brukertesting.

Kapittel 9: Diskusjon

Kapittel 9 inneholder diskusjon av oppgaven, gruppens arbeid og hvordan oppgaven kan arbeides med videre.

Kapittel 10: Konklusjon Kapittel 10 inneholder konklusjon av prosjektet i sin helhet.

Kravspesifikasjon

2.1 Funksjonelle krav

Gruppen har i samarbeid med NINA kommet fram til følgende funksjonelle krav for programvaren:

- Løsningen skal ha et verktøy som detekterer fisk i video
- Løsningen skal ha et grafisk brukergrensesnitt for deteksjonsverktøyet.
- Løsningen skal produsere filer med resultater fra deteksjonen
- Løsningen skal benytte formater som støttes av VLC [9].

Brukeren skal gjennom programvaren kunne kjøre deteksjon av fisk på videofiler som brukeren velger ved oppstart av programmet. Deteksjonen skal kunne kjøres med ulike parametere for å tilrettelegge for ulike forhold i videofilene. Programmet skal gi brukeren en fil med resultater når deteksjonen er fullført. Deteksjonen skal være implementert inn i et grafisk brukergrensesnitt for å gjøre programvaren mer intuitiv og enklere å ta i bruk.

2.2 Ikke-funksjonelle krav

2.2.1 Plattform

Programvaren skal kjøre på datamaskiner med følgende (eller nyere) operativsystemer:

- Microsoft Windows 7
- Apple MacOS Catalina

2.2.2 Ytelse

På datamaskinene med støttede versjoner skal programvaren oppfylle følgende krav:

- Det er forventet at programvaren skal ha oppstartstid på under 30 sekunder, hvor det ikke skal overstige 1 (ett) minutt under noen omstendigheter.

- Ved generell bruk skal alle knapper respondere i løpet av samme sekund de blir aktivert.
- Innlastingstider for filer skal holdes innenfor 2 (to) sekunder per ekstra fil.
- Den totale størrelsen på programvaren skal ikke overskride 150 MB.
 - Dette inkluderer ikke eksterne datasett som installeres eller filer som brukeren legger inn manuelt etter installasjon.
- Deteksjonsalgoritmen skal ikke bruke mer tid enn den totale lengde på videofilene som brukeren har lastet inn.
-

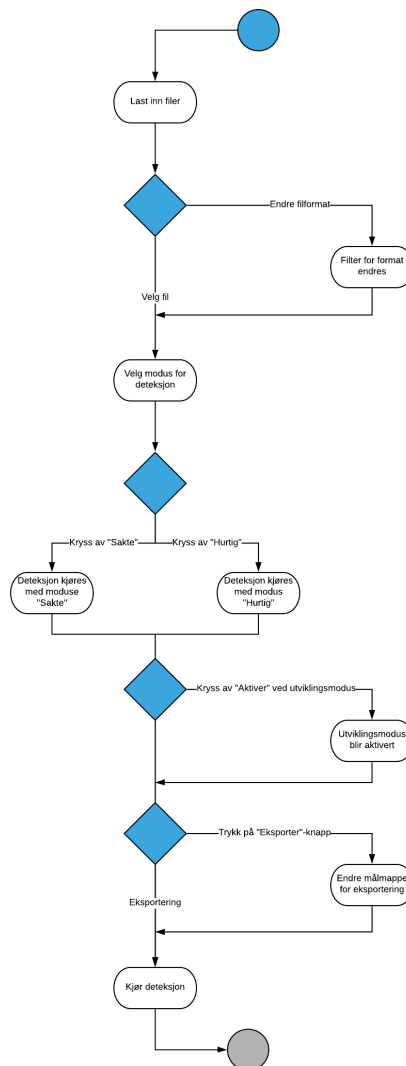
2.2.3 Brukervennlighet

Gruppen har valgt å fokusere på at det grafiske brukergrensesnittet skal følge Direktoratet for forvaltning og IKT (Difi) sine retningslinjer for universell utforming [10] og har derfor vektlagt nytten av en brukervennlig programvare. Gruppen har derfor valgt å fokusere på følgende krav:

- Programvaren skal ved oppstart ha en brukerguide tilgjengelig for brukeren.
- Programvaren skal inneholde et minimalistisk design.
- Programvaren skal gjøre det åpenbart hva brukeren skal gjøre.
- Programvaren skal være konsistent i utformingen av designet for å ikke skape forvirring hos brukeren.
- Programvaren skal opprettholde krav for kontraster.
- På bakgrunn av målgruppen så legger gruppen til rette for at sluttbrukeren kan navigere et enkelt brukergrensesnitt ved hjelp av brukerguiden.

2.3 Aktivitetsdiagram

Et aktivitetsdiagram gir en grafisk representasjon av hendelsesflyten i en løsning. Aktivitetsdiagrammet har en start node, aksjoner som sier hva som skal gjøres, piler som viser flyt av kontroll fra en aksjon til en annen, beslutningspunkter og en sluttnode. Gruppen har valgt å ta i bruk et aktivitetsdiagram for å representere flyten i løsningen. Figur 2.1 er en grafisk representasjon av flyten til løsningen når brukeren skal gjennomføre deteksjon av fisk fra videomateriale.



Figur 2.1: Figuren viser flyten ved kjøring av deteksjon på filer valgt av brukeren.

Metode

Prosjektgruppen måtte ta et valg om hvordan arbeidet med prosjektet skulle utføres. Dette kapitlet beskriver fremgangsmetoden prosjektgruppen valgte å ta i bruk for prosjektet.

3.1 Målgruppe

3.1.1 For rapporten

Målgruppe for rapporten vil hovedsakelig være studenter og ansatte som, slik som gruppen, har tilhørighet til Institutt for Datateknologi og Informatikk (IDI) ved NTNU.

Rapporten vil være et godt fundament for studenter som potensielt vil arbeide videre på produktet eller andre prosjekter som er relatert til bildebehandling og datasyn. Rapporten vil også være meget aktuell for andre interessenter som ikke nødvendigvis har tilhørighet til IDI, men som har bakgrunn eller kjennskap til bildebehandling og datasyn.

3.1.2 For oppgaven

Målgruppe for oppgaven vil hovedsakelig være de ansatte hos NINA som ønsker å ta i bruk produktet for å automatisk kunne få ut informasjon om hvor det befinner seg fisk i datamaterialet deres, slik at de selv slipper å se gjennom unødvendig materiale.

Produktet kan med mindre modifikasjoner benyttes for deteksjon av andre arter eller objekter, så produktet vil potensielt kunne tas i bruk av mange ulike aktører som ønsker å begrense det manuelle arbeidet.

3.2 Egen bakgrunn og kompetanse

3.2.1 Hva vi kunne

Gruppens medlemmer studerer dataingeniør og har dermed relativt lik fagbakgrunn. Gruppen har sett et stort behov for tidligere tilegnet kompetanse fra de foregående semestrene. Hovedsakelig har emner som, IMT2243 - Systemutvikling, IMT3881 - Vitenskapelig programmering, IMT1002 - Introduksjon til ingeniørfag - data, TØL1011 - Ingeniørfaglig systememne og de andre programmeringsemnene, vært til stor hjelp for å kunne gjennomføre prosjektet.

Det gruppen derimot mangler er bakgrunn i datasyn, noe gruppen kunne fått gjennom emnet, IMT3017 - Computer Vision. Dette emnet kan sees på som en videreføring av IMT3881 - Vitenskapelig programmering og går dypere inn på bildebehandling, og hvordan datasyn kan benyttes i ulike applikasjoner.

3.2.2 Hva vi måtte lære

Prosjektet krever innhenting av mye kunnskap, da det meste som er direkte relevant for å kunne løse oppgavene er helt nytt for hele gruppen. Oppgaven setter krav for forståelse av datasyn og medførte at gruppen brukte tid på å sette seg inn i emnet, IMT3017 - Computer Vision, og annen faglitteratur. Gruppen fikk god veiledning på relevant faglitteratur fra veileder om datasyn og ulike bildebehandlingsteknikker.

3.3 Rammer

3.3.1 Prosjektorganisering

Grunnlag for valg av utviklingsmodell

Dette prosjektet har store usikkerhetsmomenter i vanskelighetsgrad og tidsbehov for de ulike funksjonalitetene som skal og kan implementeres. Det er viktig å sikre at gruppen rekker å utvikle grunnfunksjonaliteten til verktøyet (resultatmål som er satt som krav). Beslutninger om hvor mye arbeid utover grunnfunksjonaliteten som skal implementeres må derfor gjøres underveis i prosjektet, etter hvert som det avdekkes hvor krevende de ulike resultatmålene viser seg å være. Her vil også innspill og prioriteringer fra oppdragsgiver kunne endre seg i takt med at verktøyet blir utviklet.

Prosjektgruppen er forholdsvis liten, og det er naturlig å anta at det kommer til å være et stort behov for fleksibilitet underveis i prosjektet, etterhvert som oppgaver kan vise seg å være mer, eller mindre, krevende enn originalt anslått.

Scrumban

På grunnlag av disse behovene har gruppen kommet frem til at det er viktig å velge en smidig utviklingsmodell. Gruppen er kjent med en rekke typer smidige utviklingsmodeller fra systemutviklings-faget [11], men har mest erfaring med scrum for litt mindre prosjekter med omtrentlig like store grupper som dette prosjektet. Oppdelingen av tid og arbeidsoppgaver utover sprinter som scrum tilbyr, virket utmerket for å tilrettelegge for smidige overganger etter beslutningspunkter i prosjektet. Begrensningene i hvor lett man kan bytte eller overta oppgaver var det største problemet gruppen så ved en ren scrum-metode, og valgte derfor å kombinere dette med Kanban-modellen.

Kanban

Kanban er en simpel modell som kan kombineres med andre utviklingsmodeller. Kanban er et system hvor ulike oppgaver deles opp og organiseres under ulike stadier på en tavle. Ulike begrensninger på hvor mange kort/oppgaver som kan være i et gitt stadiet på tavla tilbyr en enkel metode for å forhindre at gruppemedlemmer tar på seg for mange oppgaver om gangen. Kanban-tavlen gir en god visualisering av arbeidsoppgavene, viser tydelig fremgang for alle gruppens medlemmer og kan bidra til økt motivasjon ved å tydelig dele opp store arbeidsoppgaver til mindre fragmenter. For å få god utbytte av Kanban er det viktig at det brukes gjennom hele prosjektet, og bruken må derfor følges opp av gruppeleder underveis. Trello [12] har et tilbud som tilsvarer en Kanban-tavle gratis på nett, og det er denne implementasjonen av Kanban som gruppen kommer til å benytte i prosjektet.

Oppdeling i sprinter

Gruppen har valgt å dele opp hovedprosjektet utelukkende i én-ukers sprinter, hvor hver sprint startes på samme dag som det ukentlige veiledningsmøtet. Dette gir gruppa tid i forkant av veiledningsmøtet til å planlegge ukas sprint, og deretter få veiledning til arbeidet som skal gjøres i løpet av sprinten.

Sprint planning meeting

Hver sprint skal starte med et planleggingsmøte (sprint planning meeting), hvor en eventuell sprint backlog fra forrige sprint innarbeides i neste sprint, sammen med nye oppgaver. Oppgavene deles opp i mindre biter, kategoriseres og legges på Trello. Det blir skrevet referat av hvert planleggingsmøte.

Sprint review meeting

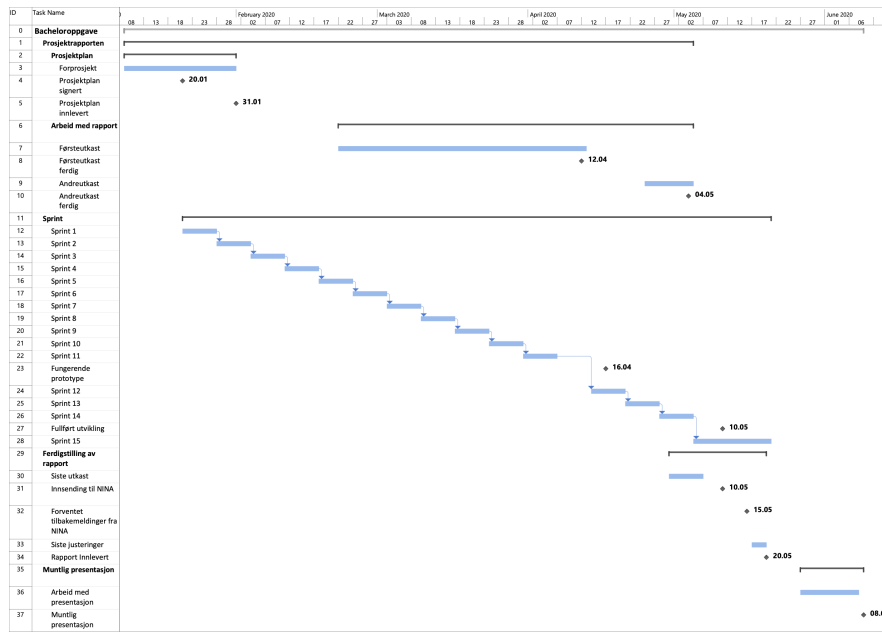
Ved slutten av hver sprint møtes gruppen for et sprintrefleksjons-møte (sprint review meeting). Her skal gruppa gjøre en vurdering på hvordan fremgangen er

i forhold til forventet, og hva som skal gjøres i neste sprint. Under dette møtet blir det vurdert om gruppen står i fare for å bli forsinket, og eventuelle tiltak mot dette diskuteres og iverksettes. Oppgaver som ikke ble fullført samles i en sprint backlog av dokumentasjonsansvarlig.

3.3.2 Fremdriftsplan

GANTT

Figur 3.1 viser en oversikt over hvordan prosjektgruppen har planlagt å dele inn perioden frem til innlevering.



Figur 3.1: Gantt-diagram over prosjektet.

Milepæler

Forprosjekt (01.02.2020)

Forprosjekt inkluderer prosjektplan, prosjektavtale og gruppe-regler.

Fungerende prototype (23.03.2020)

Ha ferdigstilt en fungerende prototype som skal kunne utføre den grunnleggende funksjonaliteten avtalt med NINA.

Førsteutkast rapport (12.04.2020)

Førsteutkast av prosjektrapport til påsken for å sikre at vi ligger godt an.

Andreutkast rapport (04.05.2020)

Andreutkast vil også bli levert til NINA for å få tilbakemeldinger før endelig innlevering.

Endelig utvikling av verktøy (04.05.2020)

Utvikling skal være fullført og alle resultatmål være oppfylt.

Innlevering av rapport (20.05.2020)

Frist for innlevering av rapporten.

Presentasjon (06.06.2020)

Muntlig presentasjon av prosjektet.

3.3.3 Gjennomføring

Andre rammer

Prosjektet innleveres som en rapport innen 20. mai og vil derfor ha en tidsramme fra 1. januar til 20. mai.

NINA har ikke satt noen konkrete rammer for teknologier og programvarer som kan benyttes for å løse oppgaven. Det eneste kravet gruppen har for leveransen er at ved ferdigstilling må verktøyet kunne generere CSV-filer som peker til detektere fisk i videomaterialet i sanntid med et grensesnitt som oppleves som tilstrekkelig godt av NINA.

Arbeidstider

Gruppen har ved tidligere anledning blitt enige om at hver og én i gruppen skal gjennomføre 22,5 timer med effektivt arbeid for hver sprint. Det vil si at kun arbeid logges og ikke tid som spising og prating.

Gruppen valgte å kun logge ren arbeidstid for å bedre få representert antall timer som er lagt ned. Det eneste som er uheldig med denne måten å regne timer på er at tid hvor for eksempel én gruppemedlem tenker på hvordan et problem kan løses ikke vil komme frem.

Gruppen ble ved begynnelsen enige om at gruppen skal samles på Campus hver eneste ukedag og satte opp en timetabell for hvilke tider gruppen skal sitte samlet og arbeide. Etter coronaviruset (COVID-19) slo til i mars har gruppen vært nødt til å jobbe hver for seg, og tidene har derfor blitt mer flytende. Gruppen ble derfor enige om at så lenge hvert gruppemedlem legger ned avtalt antall timer per sprint, så er det godt nok. For å sørge for at gruppemedlemmene kan opprettholde kontakten, så er fastsatte møter fortsatt holdt til avtalt tid.

3.4 Retningslinjer

3.4.1 Krav til dokumentasjon

Etttersom gruppemedlemmene etter prosjektets slutt ikke kommer til å være tilstrekkelig tilgjengelige for bistand til vedlikehold, oppdateringer og annen generell hjelp i forbindelse med verktøyet, må det i løpet av prosjektet utvikles dokumentasjon som gjør det så enkelt som mulig for andre å overta denne oppgaven. Dette kommer til å løses på to måter; utforming av en kortfattet og enkel brukermanual for bruk av det utviklede verktøyet og et mer teknisk dokument for

eventuelle utviklere som skal vedlikeholde og videreutvikle verktøyet i senere tid.

Denne brukermanualen skal gi en kortfattet beskrivelse av hva verktøyet tilbyr med hensyn på ulike funksjonaliteter. Deretter en litt mer detaljert forklaring av hvordan denne funksjonaliteten benyttes i praksis, med henvisninger til brukergrensesnittet og forklaringer med minimalt rom for feiltolkning.

Dokumentet for utviklere må inneholde en mer teknisk dyptgående beskrivelse av hvordan programvaren fungerer. Dokumentet skal, sammen med god kommentering i kildekoden, kunne gi en person med utdanning og erfaring tilsvarende gruppens medlemmer et godt overblikk over hvordan programvaren og brukergrensesnittet er satt sammen.

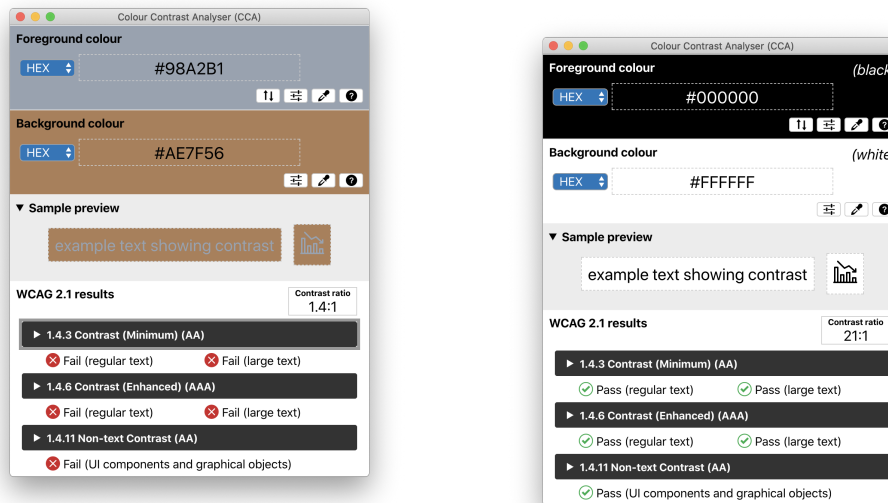
3.4.2 Krav til kvalitetsgjennomgang

For å verifisere at resultatmålene om andel fisk som detekteres og falske positive overholdes vil gruppen benytte utlevert datamateriale fra 2017 som testmateriale, hvor Excel-dokument [6] med resultater fra manuelt gjennomsyn av NINA benyttes som fasit. For at resultatmålene skal anses som oppfylt av gruppen, behøver testingen å vise at det statistisk sett kan vises at kravene overholdes med et signifikansnivå $p = 0,05$.

For verifisering av at resultatmålet på minst sanntidsprosessering overholdes skal gruppen benytte et testoppsett bestående av en MacBook Pro 2018 som kjører en virtuell maskin som simulerer spesifikasjonene til laptopene [7] NINA benytter til daglig.

For å kontrollere at GUI-en tilfredsstiller WCAG 2.1 så kommer gruppen til å benytte verktøyet Color Contrast Analyzer (CCA) [13]. CCA er en gratis programvare som Digitaliseringsdirektoratet anbefaler for å sammenligne kontrast mellom tekst og bakgrunn [10]. Figur ?? viser et eksempel på sammenligning mellom kontraster i CCA.

Verktøyet tilbyr muligheten til å velge en tekstfarge i brukergrensesnittet og sammenligne denne fargen opp mot bakgrunnen. Man ser så etter om verktøyet godkjenner kontrasten i henhold til AA-standarden.



(a) Eksempel på kontraster som ikke godkjennes.

(b) Eksempel på kontraster som godkjennes.

Figur 3.2: Demonstrasjon av hvordan verktøyet CCA [13] kan sjekke for kontraster mellom tekst og bakgrunn.

3.4.3 Krav til standarder og verktøy

- Prosjektrapport skal benytte NTNUs mal [14] for generell utforming av rapporten.
- Rapporten skal skrives i \LaTeX , på norsk bokmål, hvor gruppen skal benytte Overleaf [15] for sanntidssynkronisering.
- Gruppen skal benytte Git [16] og Bitbucket [17] for versjonskontroll av utviklingen av verktøyet.
- Gruppen skal benytte PyCharm til utviklingen av brukergrensesnittet.
- Gruppen skal benytte PyQt til utviklingen av backend for verktøyet.
- Gruppen skal benytte Colour Contrast Analyser [13] til kvalitetssjekk på GUI.

3.4.4 Utviklingsmiljø og språk

På bakgrunn av tidligere erfaring med prosjekter hvor bildebehandling skal gjennomføres, så stod det mellom Python og MATLAB for valg av programmeringsspråk. Begge språkene er godt kjente og etablerte språk innenfor datasyn og maskinlæring. En annen stor likhet er utvalget av pakker og verktøykasser som kan benyttes sammen med de miljøene. Det som skiller Python fra MATLAB er kostnaden, Python er opensource og kan derfor benyttes gratis. MATLAB derimot er dyrt og om en enkeltperson skal anskaffe en lisens, så vil totalsummen fort bikke titusener av kroner om verktøykasser skal være inkludert. Python er gratis med

pakker som gjør det mulig å utføre svært avanserte jobber.

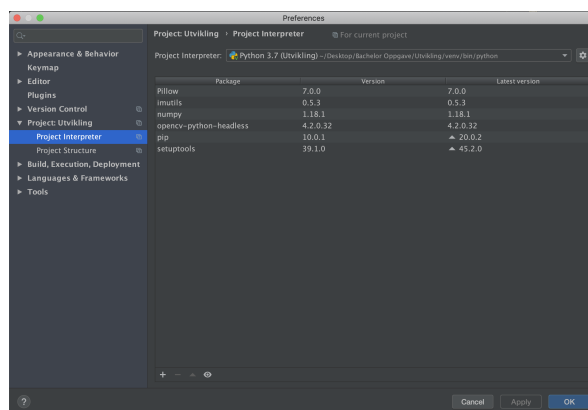
Ettersom det skal utvikles et grafisk grensesnitt for verktøyet ble det også sett på hvor lett dette kom til å være for de to språkene. Det naturlige valget falt da på Python for utviklingen. Gruppen ble likevel rådet til å benytte seg av MATLAB da NTNU har lisens på denne programvaren og er dermed gratis for studentene deres. MATLAB har mange gode programmer som er ferdiglaget og var derfor meget aktuelt for gruppen for å teste ulike funksjonaliteter først i MATLAB, før det ble tatt en vurdering på om dette skulle implementeres inn i Python.

PyCharm

PyCharm er et integrert utviklings miljø (IDE) som blir brukt til utvikling av programvare, spesielt for programvare som benytter språket *Python*. PyCharm er utviklet av det Tsjekiske selskapet *JetBrains* [18] og er kun ett av mange programmer utviklet av selskapet. PyCharm er i stand til å utføre kodeanalyse, har en grafisk debugger, en integrert enhetstester, integrasjon med versjonskontrollsystemer som *Git* og mer. JetBrains tilbyr en utdanningslisens til studenter hos de fleste høyskoler og universitet, inklusivt NTNU.

Ettersom gruppen hadde kjennskap til PyCharm [19] fra tidligere prosjekter, og hadde vært fornøyd med det, falt valget på PyCharm som det primære utviklingsmiljøet. PyCharm har en egen *Scientific Mode* som er en modus personer med lisens kan aktivere. Denne modusen gjør det enkelt å studere informasjon om bildene og dermed kunne gjøre justeringer for å få et bedre resultat.

PyCharm har en egen *Project Interpreter* som gjør det enkelt for å installere ønskede pakker og utvidelser. Pakkene som gruppen har tatt i bruk under prosjektet er; *NumPy* og *OpenCV*. Figur 3.3 viser PyCharm sin *Project Interpreter*.



Figur 3.3: PyCharm sin *Project Interpreter* hvor gruppen har installert pakkene som prosjektet krever.

3.4.5 Endringshåndtering

Det vil være naturlig at det i løpet av prosjektet vil komme ønsker om endringer i verktøyet fra oppdragsgiver. For å formalisere prosessen og sørge for at den er dokumentert og gjennomføres systematisk, kommer gruppen til å benytte seg av følgende prosedyre for endringshåndtering [14]:

1. Dokumentere endringens innhold
2. Analysere konsekvensene for prosjektet
3. Beregne eventuell kost/nytt
4. Godkjenning og aksept
5. Logg endringen
6. Juster planene
7. Informer interessentene
8. Gjennomfør endringen

Bildebehandlings-bakgrunn

Prosjektgruppen ble nødt til å ta et valg mellom fagfeltene maskinlæring og datasyn for å kunne løse oppgaven. Dette kapitlet vil ta for seg differansen mellom maskinlæring og datasyn, hva et digitalt bilde er og hvordan preprosesserings-teknikker kan benyttes for å manipulere bilder.

4.1 Eksisterende teknologi

Det er spesielt to fagfelt som kommer til å være relevant for å løse backend-delen av verktøyet som utvikles; datasyn ("computer vision") og maskinlæring.

4.1.1 Datasyn

Datasyn er vitenskapen og teknologien om hvordan maskiner kan 'se', hvor det å se her betyr å kunne plukke ut ønsket informasjon fra et bilde [20]. Deteksjon av objekter er det første og viktigste steget i å kunne gjenkjenne objekter. Tradisjonelt så kan spesielle utvalgte objekter bli oppdaget og gjenkjent ved å se om en mal og objektet stemmer overens, men hastigheten på gjenkjenningen har alltid vært et problem.

Datasyn er en passiv teknologi og i et typisk scenario så vil kameraet være fastmontert. Kameraet vil ha oversikt over det aktuelle området og ta i bruk algoritmer for å kunne prosessere utbyttet fra bildene. Selv om det i utgangspunktet kan virke enkelt, så er det ikke en enkel oppgave da både statiske og dynamiske objekter vil være tilstede i det aktuelle området. I tillegg vil flere faktorer som ikke er mulige å kontrollere som bevegelse av solen, overganger mellom dag og natt, skygger, kamera-artefakter fra objekter som fester seg på linsen, vibrasjoner i kameraet og ulike værelementer dukke opp. Alle disse faktorene spiller inn når algoritmene skal arbeide med å analysere bildene som kameraet fanger opp.

4.1.2 Maskinlæring

Klassisk maskinlæring og dataanalyse kan bli delt opp i flere kategorier. I den første kategorien så benytter vi datamaskiner for å utføre 'klassisk' dataanalyse-

metoder som for eksempel minste kvadraters metode, polynomisk interpolasjon og dataanalyse [21].

Maskinlærings-protokoller kan være veiledet eller ikke-veiledet. I veiledet læring deles data i navngitte kategorier, som eksempler med håndskrevne tall sammen med det nummeret det håndskrevne tallet faktisk skal representere. Oppgaven til maskinen blir å lære seg hvordan den kan sette navn på data utenfor treningssettet. Ikke-veiledet læring benytter seg av data som ikke er navngitt og målet for maskinen blir å finne naturlige kategorier som settet med treningsdata hører til. En tredje type er maskinlæringsoppgaver som å spille det kinesiske spillet “Go”, som innebærer kombinasjoner av veiledet og ikke-veiledet læring, sammen med treningssett som kan være generert av maskinen selv.

4.1.3 Vårt valg

Oppdragsgiver har etter innledende møte gitt gruppen mulighet til å velge mellom datasyn og maskinlæring. Programmets viktigste mål er å kunne gjennomføre deteksjon av fisk i video.

For å kunne gjennomføre deteksjon av fisk, så vil datasyn kreve at gruppen klarer å eliminere støy i videomaterialet og klarer å hente ut den nødvendige informasjonen for å kunne si om det er en fisk som kameraet ser, eller om det er noe annet som flyter forbi. Maskinlæring ville vært det primære valget om gruppen skulle gjennomført artsbestemmelse og hatt hovedfokus på dette, men ettersom primæroppgaven er deteksjon av fisk, så har gruppen valgt å ikke benytte seg av maskinlæring.

Gruppen har konkludert med at datasyn vil være mest egnet, både med tanke på at gruppen ønsker å lære seg noe nytt, men også fordi datasyn vil kunne være lettere å videreutvikle.

Gruppen har erfaring fra arbeidsoppgaver i Python og ettersom Python er et av de mest brukte programmeringspråkene for datasyn [22], så vil gruppen ta i bruk dette. Flere gode og godt dokumenterte bibliotek for datasyn er allerede i bruk, og språket generelt vil være enklere for brukermassen hos NINA å sette seg inn i enn syntaktisk kompliserte språk som for eksempel C++.

4.2 Bilde og video

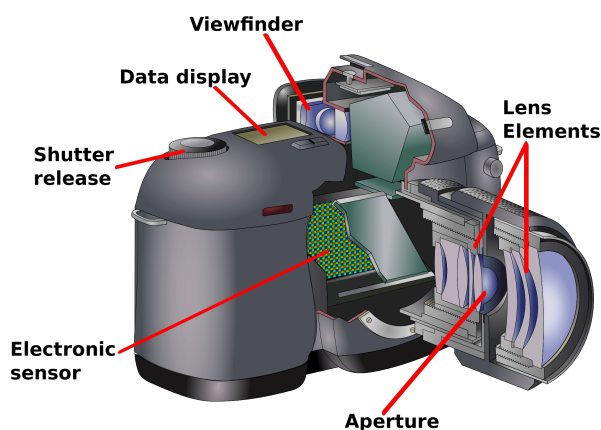
Datamaterialet som gruppen har blitt utdelt fra NINA er blitt filmet på et digitalt kamera under vannoverflaten. Kameraet som er blitt benyttet av NINA er en Marshall Electronics CV502-WPM [23], hvor videomaterialet har blitt filmet med en oppløsning på 1080p (1920 x 1080 piksler) og med en hastighet på 25 bilder i se-

kundet. Kameraet ble plassert inni et vanntett kammer med glassfront. Synsfeltet for kameraet er på 72 grader, som dekker tilnærmet lik hele kulpen.

4.2.1 Hvordan et digitalt kamera fungerer

Digitale kameraer er bygd opp med tilnærmet like komponenter. Figur 4.1 illustrerer enkelte viktige komponenter i et digitalt kamera. Marshall CV502-WPM er et kringkastingskamera. Dette kringkastingskameraet tilbyr god oppløsning og et stort antall bilder i sekundet. Marshall CV502-WPM er bygget på de samme prinsippene som digitale kameraer.

Vi kan dele et digitalt kamera opp i 4 hovedkomponenter; bildesensor, digital omformer, kretskort og en skjerm for å vise bildene. Bildesensoren i et digitalt kamera inneholder millioner av lyssensitive piksler, også kalt "arrays", som individuelt måler lyset som treffer dem [24]. På toppen av bildesensoren sitter et fargefilter som bare lar enkelte piksler måle visse farger på lysbølger. En digital omformer sørger for omforming av verdiene som hver enkelt piksel samler inn. Dataene som hver piksel samler inn blir da omformet til et digitalt signal. Signalene som omformes av den digitale omformeren blir ført videre ved hjelp av kretskortet i kameraet. Kretskortet inneholder alle datamaskinbrikkene som kameraet bruker for å registrere data. Kretsene på kortet fører dataene fra bildesensoren og andre brikker til minnekortet hvor dataene lagres. Digitalkameraer har som regel en skjerm. Skjermen benyttes som regel av brukeren for å gjøre justeringer på kameraets innstillinger, men også for å kunne se gjennom hvordan de fotograferte bildene ble. Enkelte kameraer har en søker for å kunne se hvordan bildet kan se ut før det blir tatt, men kameraer med skjerm gjør det mulig å gjøre det samme med bare skjermen.



Figur 4.1: Illustrasjon av enkelte viktige komponenter i digitale kameraer, hentet fra GadgetReview [24].

Komponentene nevnt ovenfor tar alle part i prosessen om å ta et bilde. Selve prosessen kan deles opp i 4 hovedhandlinger. Først må lyset måles av det digitale kameraet. For å kunne måle lyset må utløserknappen trykkes for å åpne lukkeren, som lar lyset bevege seg gjennom linsen og treffe bildesensoren. De individuelle pikslene på bildesensoren måler deretter intensiteten til lyset på millioner av forskjellige punkter på bildesensoren, og skaper nøyaktige målinger om lyset i omgivelsene. Samtidig som lyset fra omgivelsene rundt kameraet beveger seg gjennom lisen, så må kameraet sørge for at lyset fokuseres nøyaktig på bildesensoren. Kameraet har forskjellige glasselementer i lisen som kan roteres for å endre fokuset og dermed hvordan lyset treffer bildesensoren. Et digitalt kamera vil gjøre denne justeringen automatisk, men en god del kameraer har også en ring som gjør det mulig å justere kameraets glasselementer manuelt.

Hver piksel konverterer det målte lyset som passerer gjennom linsen til elektroner. En piksel som måler et lysere lys vil inneholde flere elektroner, noe som resulterer i en større akkumulert ladning. For å få en digital verdi som kan lagres har kameraet en ADC (analog-to-digital converter), som er en brikke som omformer lyssignalet ved hver piksel til en digital verdi. Etter at lyset fra omgivelsene har blitt omformet til en digital verdi, kan kameraets kretskort flytte dataene som binære bits gjennom kretsløpet. Dataene skrives til kameraets minnekort ved hjelp av fastvaren til det digitale kameraet.

4.2.2 Hva er et digitalt bilde

Et digitalt bilde er en representasjon av et ekte bilde, men hvor bildet beskrives ved digitale verdier som kan lagres og håndteres av en datamaskin. For å kunne oversette et bilde ut ifra digitale verdier så blir verdiene delt opp i mindre områder som vi kaller piksler (bildelementer). For hver eneste piksel så lagrer kameraet en liten bit av den totale informasjonen. Informasjon som kan være lagret i en enkelt piksel sier noe om egenskapene til akkurat denne pikselen, som kan være lysintensitet eller fargen. Verdiene som kameraet fanger opp blir posisjonert i et array med rader og kolonner som korresponderer til den vertikale og horisontale posisjonen til pikslene i bildet. [25]

Et digitalt bilde kan være av flere ulike typer. Et fargebilde består av tre ulike kanaler, som normalt sett vil være av typen RGB-bilde. Et RGB-bilde vil si at vi har én rød kanal (R), én grønn kanal (G) og én blå kanal (B). RGB-bilder benyttes som regel på skjermer, men i fargeprintere så benyttes det ofte fire fargekanaler, CMYK (Cyan, Magenta, Yellow, Black). Alle bilder er ikke fargebilder, og svart-hvitt bilder vil kun benytte intensiteten av lyset som treffer pikslene og lage et bilde ut ifra dette. Et normalt bilde, uavhengig av farge eller ikke, benytter 8 bits-verdier som gir 256 mulige verdier (0-255). Et svart-hvitt bilde har kun én kanal og vil dermed kun ha 256 forskjellige toner. Et fargebilde med 8-bits per farge, vil ha 24 bits per piksel. Siden hver bit kan representere to mulige farger, så gir det en

total på 16 777 216 mulige farger [25]. I underkapitlet om “Preprosessering” ser vi nærmere på ulike fargekanaler og eksempler på bilder som illustrerer de ulike fargekanalene.

4.2.3 Hva er en video

Digital video er bygd opp av en gruppe med individuelle bilder, hvor hvert enkelt bilde representerer en liten del. Filmer er ofte bygd opp av 24 bilder i sekundet, mens datamaterialet som brukes i denne oppgaven stammer fra et kringkastingskamera som filmer 25 bilder i sekundet. Økende antall piksler i et bilde vil naturligvis kreve stadig mer plass å lagre, som fort kan føre til en betraktelig økning i total lagringsplass som kreves. Om filmene fra kringkastingskameraet som NINA benyttet ikke hadde vært komprimert, ville ett minutt med video i 1920x1080 piksler oppløsning tatt opptil 8.69 GB plass. På den andre siden ville en oppløsning på 1280x720 bare krevd 3.86 GB med plass.

Kompresjon

Ukomprimert digital video krever store mengder med lagring. Det er på bakgrunn av dette at det er utviklet komprimeringsordninger som Motion Picture Experts Group (MPEG) [26], som er avgjørende for å håndtere digital video på dagens datamaskiner.

MPEG-4 filer består av både bilde og lyd, og er ofte i formatet **.mp4**, men kan også forekomme i andre formater som **.m4a**, **.m4v**, og **.m4r**. MPEG-4 er basert på **.mov**, som er filformatet til QuickTime [27]. Kompresjonsalgoritmen som MPEG-4 formatet benytter er ikke tapsløs, som vil si at ved kompresjons så blir lyd- og video-kvaliteten dårligere, men det er også mulig å opprette virkelig tapsløse kodede regioner i taps-kodede bilder eller å støtte sjeldne tilfeller hvor hele kodingen er tapsfri [28].

Om man tar ett minutt med video filmet med en oppløsning på 1920x 1080, som ville krevd 8.69 GB med lagringsplass ukomprimert, og komprimert med MPEG-4 (H.264) så ville dette kun krevd 617.22 MB med lagring. Dette er en reduksjon på nesten 93%, noe som gjør det mulig å lagre lange videofiler med god oppløsning på dagens datamaskiner.

4.3 Preprosesseringsteknikker

For at deteksjonsalgoritmen skal kunne ha en reell mulighet for å plukke ut objektene vi ønsker og filtrere ut andre uønskede objekter, er det behov for at det gjøres endringer på bildet. Datasettet skal benyttes til datasyn og for at maskinen skal kunne “se” fiskene i bildet og samtidig ikke bruke mer tid enn nødvendig. For å kunne gjennomføre slike operasjoner er det behov for preprosesseringsteknikker. Gruppen har sett på liknende problemstillinger i andre applikasjoner [29] og derfor valgt å se nærmere på de følgende preprosesseringsteknikkene:

4.3.1 Nedskalering

Videofiler med gode oppløsninger inneholder store mengder med data og som vi har sett i de foregående kapitlene, så vil dette kreve mye av datamaskinens prosessering å håndtere så mye data på en gang. I programmer hvor det er viktig med god ytelse og flyt i programmet kan en løsning være å redusere antall piksler som datamaskinen må jobbe med. Dette kan gjøres gjennom en nedskalering. Resize er en funksjon som er implementert i OpenCV-biblioteket. Funksjonen gjør det mulig å gjøre bilder både mindre og større ved å endre dimensjonene til bildet, men OpenCV anbefaler da å tilpasse interpolering etter bruksområde for å få best mulig resultat [30].

Ved å kjøre en nedskalering på et bilde så vil også andre prosesseringsteknikker som kjøres på det nedskalerte bilde få utbytte av den økte hastigheten, som vil resultere i at programmet eksempelvis vil ha en bedre flyt eller få forbedret ytelse. En konsekvens av å nedskalere bildet er at mindre informasjon vil resultere i mindre detaljer, slik som vi kan se i figur ?? hvor det nedskalerte bildet oppfattes som mer pikselert. I enkelte tilfeller kan mindre mengder detaljer bidra til en enklere deteksjon av et objekt om det er mye støy tilstede. Figur 4.2 viser et eksempel på et bilde med original oppløsning 1920 x 1080 og figur 4.3 viser et nedskalert bilde 640 x 360 av figur 4.2.



Figur 4.2: Skjermbilde av Fisk med **original** oppløsning på 1920 x 1080 piksler



Figur 4.3: Skjerm bilde av Fisk med **nedskalert** oppløsning på 640 x 360 piksler



(a) Originale bilde zoomet inn på hodet til fisken.



(b) Nedskalert bilde zoomet inn på hodet til fisken.

Figur 4.4: Ved å zoome inn så er det mulig å se hvordan det nedskalerte bildet har mistet en del data og ser mer uskarpt ut enn det originale bildet.

4.3.2 Filtrering

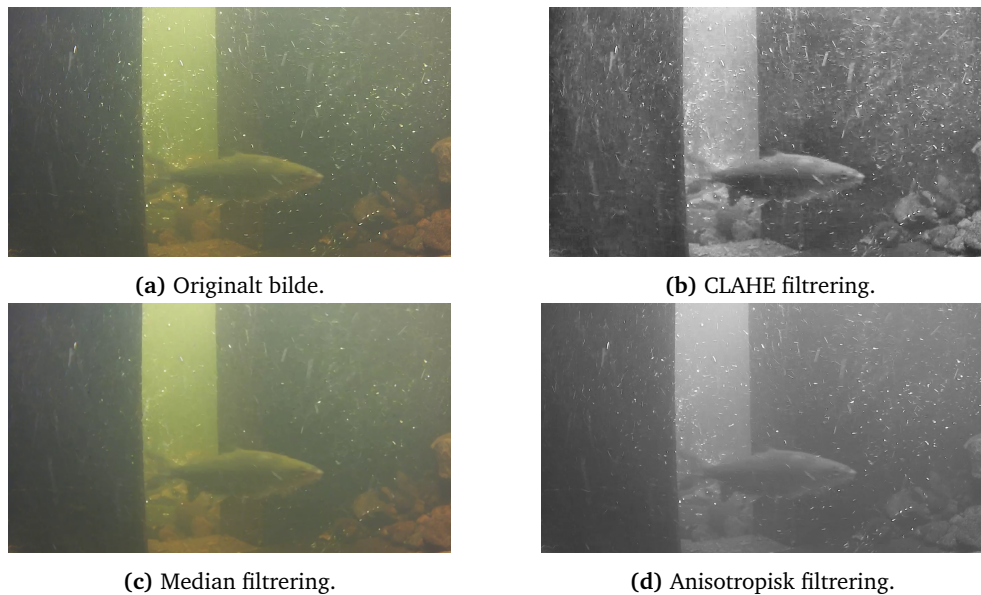
Gruppen har i gjennomgang av videomaterialet observert mye støy i videoene. For å kunne gjøre en bedre deteksjon så vil det potensielt hjelpe å fjerne støy i videomaterialet gjennom filtrering. Gruppen har derfor valgt å finne et utvalg filtreringsmetoder som er benyttet i liknende applikasjoner [29]:

- Anisotropisk filtrering [31]
- Lineær filtrering [32]
- Fourier filtrering [33]
- CLAHE [34]
- Median filtrering [35]
- Gaussian filtrering [35]

Gruppen ønsker samtidig å bevare kantene til fisken slik at deteksjonen fortsatt kan oppdage fisken etter at en filtrering er gjennomført. Vanlige filtreringsmetoder har en tendens til å gjøre hele bildet glatt og dermed ødelegge bildestrukturene som er visuelt viktige. En kantbevarende filtreringsmetode endrer utjevnsad-

ferden adaptivt avhengig av den lokale bildestrukturen. Generelt utføres maksimal utjevning “flate” (ensartede) bilderegioner, mens glatting reduseres nær eller på tvers av kantlignende strukturer [36].

For å kunne bevare mest mulig av kantene i bildet, så har gruppen valgt å se videre på *anisotropisk filtrering* og *median filtrering* på bakgrunn av hvordan filtreringen bevarer kantene på en god måte. I tillegg har gruppen valgt å se nærmere på *CLAHE* ettersom at denne filtreringen har gitt gode resultater i liknende applikasjoner under vann [34]. Figur ?? sammenligner et originalt bilde mot bilder som det er gjennomført filtrering på.



Figur 4.5: Sammenligning av ulike filtreringsmetoder gjort på video *Dev-God1.m4v*.

4.3.3 Bakgrunnssubtraksjon

Background subtraction (BS) eller bakgrunnssubtraksjon er en viktig preprocessingsteknikk i mange datasyn-baserte applikasjoner. Noen av de mest brukte tilfellene hvor BS benyttes er i tilfeller hvor antall besøkende som kommer inn eller forlater et rom skal måles, eller et trafikkamera som henter ut informasjon om kjøretøyene. I disse tilfellene er det behov for å trekke ut personen eller kjøretøyene alene. Rent teknisk kan man si at den bevegelige forgrunnen trekkes ut fra den statiske bakgrunnen [37].

I tilfeller hvor et bilde av bare bakgrunnen alene, som et bilde av et rom uten besøkende eller et bilde av veien uten kjøretøy, så er det en forholdsvis enkel oppgave å trekke forgrunnen fra den statiske bakgrunnen. Da vil det kun være behov

for å trekke det nye bildet fra bakgrunnen for å få objektet i forgrunnen alene. I de fleste tilfeller er et slik bilde ofte tilgjengelig og må derfor trekke ut bakgrunnen fra de bildene som er tilgjengelig. Oppgaven blir fort mer utfordrende når det er skygge av kjøretøyene, siden skyggen også beveger seg, så vil en enkelt subtraksjon også markere skyggen som en del av forgrunnen.

På bakgrunn av at det er ulike faktorer som kan spille inn i ulike applikasjoner så har OpenCV implementert en del ulike varianter av BS som gjør det mulig å endre hvordan forgrunnen skal trekkes ut fra bakgrunnen.

BackgroundSubtractorMOG2

Denne algoritmen er en Gaussisk blandingsbasert bakgrunns- / forgrunns-segenteringsalgoritme. En viktig funksjon i denne algoritmen er at den velger riktig antall gaussiske fordelinger for hver piksel. Dette gir en god tilpasningsevne til forskjellige omgivelser på grunn av endringer i belysning eller liknende [37].

BackgroundSubtractorGMG Denne algoritmen kombinerer statistisk estimering av bakgrunnsbilde og Bayesiansk segmentering per piksel. Algoritmen bruker de første (120 som standard) rammer for bakgrunnsmodellering. Den benytter en probabilistisk forgrunns-segenteringsalgoritme som identifiserer mulige forgrunnsobjekter ved hjelp av Bayesiansk inferens. Anslagene er tilpasningsdyktige; nyere observasjoner er tyngre vektet enn eldre observasjoner for å imøtekomme variabel belysning.

BackgroundSubtractorKNN

Denne algoritmen er basert på K-nærmeste naboer (KNN), som vil da gi en verdi ved å skape et gjennomsnitt til alle verdiene til de k-nærmeste naboene. Algoritmen er rimelig simpel, men er veldig effektiv hvis antallet piksler i forgrunnen er lav. Figur 4.6 viser hvordan *BackgroundSubtractorKNN* skiller bakgrunnen og forgrunnen.



Figur 4.6: Skjerm bilde av hvordan funksjonen for bakgrunnsuttrek fungerer i et bilde fra videomaterialet hvor 150 rammer benyttes for bakgrunnsmodelleringen.

4.3.4 Morfologiske transformasjoner

Morfologiske transformasjoner er enkle matriseoperasjoner basert på bildefor- men. Det utføres normalt på binære bilder. Algoritmen trenger to innganger, det ene er det originale bildet, det andre kalles struktureringselement eller kjerne som bestemmer operasjonens art. To grunnleggende morfologiske operatører er ero- sjon og utvidelse. Dette leder videre inn på variantformer som åpning, lukking, gradient osv. inn i bildet [38].

Erosjon

Den grunnleggende ideen om erosjon er som jorderosjon, operasjonen eroderer grensene for forgrunnsobjektet. Så hva gjør det? Kjernen glir gjennom bildet. En piksel i det originale bildet (enten 1 eller 0) vil bare bli betraktet som 1 hvis alle pikslene under kjernen er 1, ellers er den erodert (gjort til null). Så alle piksler nær grensen vil bli kastet avhengig av størrelsen på kjernen. Så tykkelsen eller størrelsen på forgrunnsobjektet avtar eller bare det hvite området avtar i bildet. Teknikken er veldig nyttig for å fjerne støy i bilder.

Utvidelse

Utvidelse er motsatt av erosjon. Her er et pikselelement '1' hvis minst en piksel under kjernen er '1'. Så det øker størrelsen på forgrunnen i bildet, som vil si at størrelsen på objektet øker. I tilfeller som støyfjerning følges normalt erosjon av utvidelse. Fordi erosjon fjerner støy, men den krymper også objektet vårt. Så vi utvider det. Siden støyen er borte, vil de ikke komme tilbake, men objektområdet vårt øker. Det er også nyttig når deler av objektet vårt blir ødelagte som en kon- sekvens av erosjon.

Åpning og Lukking

Åpning og Lukking er kun to av flere ulike varianter som benytter seg av erosjon og utvidelse på ulike måter. Åpning er bare et annet navn på erosjon etterfulgt av utvidelse. Det er nyttig for å fjerne støy, men sørger samtidig for at objektet vårt ikke avtar i størrelse. Lukking er motsatt av Åpning, utvidelse etterfulgt av ero- sjon. Det er nyttig når du lukker små hull i forgrunnen, eller små svarte punkter på objektet. Figur 4.7 demonstrere effekten av Åpning og Lukking på Figur 4.6.



Figur 4.7: Skjerm bilde av hvordan funksjonen for morfologisk transformasjon fungerer i et bilde fra videomaterialet med tilpassede parametere.

4.3.5 Tilkoblede komponenter

Tilkoblede komponenter er en algoritme som benytter seg av de to mulige verdiene binære bilder kan ha, og så knytter sammen områder med ikke-nullverdier. Hvert element i hver tilkoblet komponent er omgitt av minst ett annet element fra den samme komponenten og forskjellige komponenter berører ikke hverandre, det er nuller rundt hver og en [39]. Den tilkoblede komponenten er en iterativ algoritme med formålet om å merke et bilde ved å bruke åtte eller fire tilkoblingspikslers. Figur 4.8 viser en piksel med 8 tilkoblingspikslers. To pikslers er tilkoblet hvis de har samme verdi og er naboer [40].

1	2	3
4		5
6	7	8

Figur 4.8: I et bilde har hver piksel åtte nabopikslers. Inspirert av [40].

Komponentanalyse for å koble sammen elementer kan ofte være en viktig del av bildebehandling, da arbeidet med å finne alle konturer vil ta lengre tid [39]. Ved å benytte en komponentanalyse så går det raskt å ekskludere alle irrelevante deler av bildet i henhold til funksjonens trekk. I tilfeller hvor omgivelsene er preget av mye støy som dekker mindre områder, så kan et funksjons-trekk være størrelsen på elementene for at komponentanalysene skal regne området som en komponent. Figur 4.9 viser hvordan deler av fisken og noe støy blir regnet som komponenter.



Figur 4.9: Skjermbilde av hvordan funksjonen for tilkoblede komponenter fungerer i et bilde fra videomaterialet med tilpassede parametere.

Datasett

Datamaterialet gruppen fikk utlevert av NINA var av betydelig størrelse. Dette kapittelet vil ta for seg kategoriserings- og utvelgingsprosessene gruppen utførte på datamaterialet, samt en seksjon om hvordan deler av dette materialet ble gjennomgått for fisk, delt opp i utviklings- og testsett som ble annotert til bruk senere i testing.

5.1 Datasett

Datasettet utlevert av NINA var meget stort og gruppen ser det derfor som hensiktsmessig å benytte en seksjon for å forklare hvordan datasettet har blitt bearbejdet. Det første datasettet som gruppen ble tildelt av NINA var på 1,6TB og besto av 255,5 timer med videomateriale. Etter møtet med NINA den 29.01.2020 ble gruppen tildelt enda mer videomateriale fra NINA, dette materialet var på 850GB, og inneholdt 135,5 timer med videomateriell. I tillegg til selve videomaterialet, hadde NINA også inkludert et dokument [D](#).

Datasettet kan presenteres på følgende måte:

- Antall disketter med materiale: 6
- Antall videoer totalt: 782
- Antall timer med video: 391
- Total mengde materiell: 2,46TB
- Varighet på alle videoene, med enkelte få unntak, er på 30min
- Oppløsningen er på 1920 x 1080, som vil tilsi *1080p*
- Alle filene er i .mp4-format

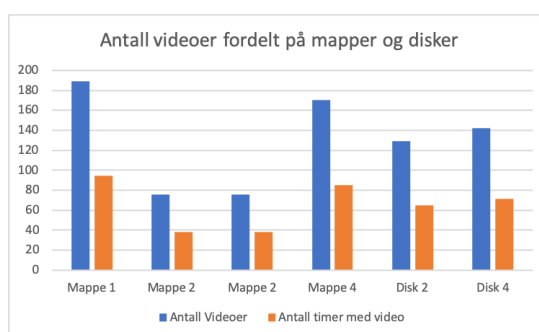
Videre er det mulig å dele datasettet opp i mer spesifikke kjennetegn ved de ulike videoene:

- Klart vann / mye turbulens: 101
- Ok klart vann / mye turbulens: 268
- Ok klart vann / mindre turbulens: 242
- Grumsete vann / nesten ingen turbulens: 171
- Natt: 170
- Dag: 617

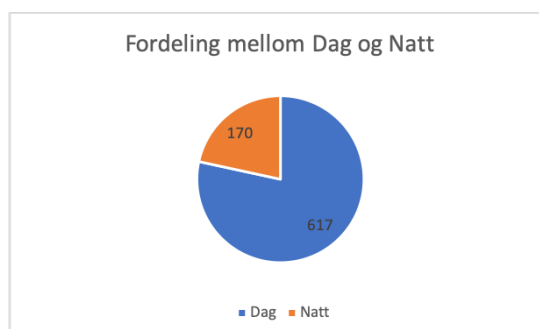
For klassifisering så er dette kun gjort på disk 1, som består av mappe 1, 2, 3 og 4. Dette skyldes at NINA har kun sett gjennom videomaterialet på disk 1 og ikke disk 2 og 4. Informasjon under vil derfor kun gjelde disk 1:

- Klassifisert som God kvalitet: 125
- Klassifisert som OK kvalitet: 352
- Klassifisert som Dårlig kvalitet: 37

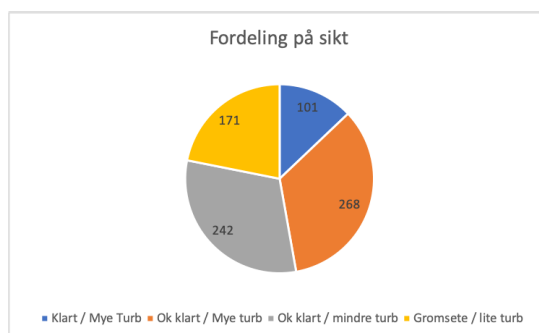
Gruppen har ut ifra presentasjonen av datasettet over lagd figur 5.1, 5.2, 5.3 og 5.4 for å illustrere datasettet på en bedre måte.



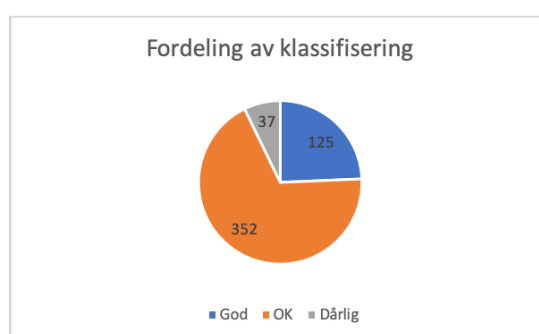
Figur 5.1: Illustrasjon over **fordelingen** av videoer over de ulike mappene og diskene.



Figur 5.2: Illustrasjon over fordeling mellom **dag** og **natt** i videomaterialet.



Figur 5.3: Illustrasjon over fordeling av **sikt** i de ulike mappene og diskene.



Figur 5.4: Illustrasjon over **klassifiseringen** som er gjort av NINA på disk 1.

Ut ifra tallene over har gruppen konkludert med at det er en sterk korrelasjon mellom kvaliteten på videoen og henholdsvis turbulens og grums. Normal vannføring fører til en del turbulens og vannet vil derfor ikke stå mye stille. Turbulens i videomaterialet vil være innslag av luftbobler. Når vannet er i kontinuerlig bevegelse med en del hastighet, så vil grums derfor ikke dannes og videoene vil oppfattes klare, men være preget av en del støy fra luftboblene som passerer kameraet.

Ved høy vannføring er det gjerne mer grums. Høy vannføring vil oppleves som mer stillestående vann i videobildet fordi vannstanden nedenfor kameraet er høyt (og dermed strømmes ikke vannet like fort forbi kameraet). Så lenge vannet ikke blir veldig grumsete, så vil videomateriale med litt grums, men tilnærmet null turbulens føre til mer nøyaktige deteksjoner.

Det er forøvrig verdt å nevne at det opprinnelige datasettet ville vært større, men etter gjennomgang av det første videomaterialet som gruppen fikk utdelt fra NINA, så valgte NINA å fjerne deler av videomaterialet grunnet helt ubrukelige forhold. De siste diskene, som er navngitt *disk 2* og *disk 4*, ble utdelt på møtet 29.01.2020. Videomaterialet fra disse diskene har ikke vært gjennomgått av NINA, så her er alt videomaterialet bevart.

5.1.1 Eksempler på forhold

Figur 5.5 a-f viser eksempler på fisk tydelig i bildet under forskjellige sikt- og turbulensforhold. Det er verdt å merke at dette er skjermbilder hvor fiskene er avbildet idet de passerer igjennom eller foran spalten i bildet. Spalten tilbyr en veldig tydelig kontrast mellom fisk og de statiske elementene i bildet. I figur 4.5e er det vanskelig å se de delene av fisken som ikke er foran spalteåpningen, som viser hvor utfordrende det kan være å finne fisk i grumsete vann dersom bakgrunnen i bildet ikke er opplyst.



(a) Eksempel på ørret om dagen i klart vann med mye turbulens.



(b) Eksempel på ørret om natten i klart vann med mye turbulens.



(c) Eksempel på flere harr som passerer om dagen i ok klart vann.



(d) Eksempel på ørret om natten i ok klart vann med en del turbulens.



(e) Eksempel på harr om dagen i grumsete vann med nesten ingen turbulens.



(f) Eksempel på harr om natten i grumsete vann med nesten ingen turbulens.

Figur 5.5: Eksempler på skjermbilder av fisker på ulike tider av døgnet i forskjellige forhold.

5.1.2 Utvelging fra datasett

Hovedsakelig har gruppen benyttet NINA sin oversikt over videomaterialet. Gruppen ser det som veldig heldig at NINA allerede hadde hatt en ansatt til å se gjennom alt videomateriale på den første disken og hadde laget en oversikt over art, retning på fisken og kvaliteten på videoen (forhold). Det andre datasettet som gruppen fikk utlevert under møtet med NINA var ikke gjennomgått av NINA på forhånd, og dette måtte eventuelt gjøres av prosjektgruppen. Prosessen med å se gjennom materialet på disk 2 og 4 manuelt ga gruppen raskt et klart inntrykk av hvor tidkrevende det faktisk ville vært om gruppen skulle brukt tid på å se gjennom alt videomaterialet på den første disken som ble utlevert av NINA. Ettersom den første disken allerede inneholdt en betydelig mengde data ble dette ikke gjennomført.

Det ble tidlig i prosjektet klart at det ble nødvendig å velge ut visse filer som skulle brukes for statistisk testing og utvikling. Ettersom det er fisk i bildet i en relativt lav andel av videomaterialet var det interessant å velge ut videofiler som hadde en viss fiskeforekomst slik at man faktisk fikk mulighet til å teste om deteksjonsalgoritmen kunne finne fiskene. Samtidig var det gunstig om fordelingen av videofilene var et representativt utvalg av den faktiske videomassen. I fordelingen mellom de ulike sikt- og turbulensforholdene ble det sett bort ifra alt videomaterialet som på forhånd hadde blitt kategorisert som "ubrukelig" av NINA, da oppgaven var å avlaste arbeidsmengden som gikk med til fisketitting, ikke ekspandere hvor mye av det innhentede videomaterialet som skulle brukes. Gruppen besluttet å dele det utvalgte datamaterialet opp i to grupper; utviklingssett og testsett.

Utviklingssett

Formålet med utviklingssettet var å ha en kort og variert sammensetning av videofiler som kunne brukes til å gi en indikasjon på hva slags ytelse deteksjonsalgoritmen ga med forskjellige parametere. Ettersom dette datasettet potensielt skulle kjøres igjennom veldig mange ganger var det viktig å finne balansepunktet mellom å ha et stort nok grunnlag til å kunne trekke korte konklusjoner fra, samtidig som det ikke skulle ta så lang tid å kjøre igjennom at det gikk utover hvor mange ulike parametere gruppen fikk tid til å prøve ut.

Gruppen hadde tidlig i utviklingsfasen besluttet at det var nødvendig å bruke bakgrunnssubtraksjon i deteksjonsalgoritmen. For å tilrettelegge for at bakgrunnssubtraksjonen skulle få mange nok rammer til å utforme et skikkelig bakgrunnsbilde, ble det alltid lagt inn minst 150 rammer før det kom en fisk inn i bildet i utviklingssettet. Gruppen besluttet at utviklingssettet skulle maksimalt havne på 4 minutter til sammen.

Som figur 5.2 og 5.3 indikerer, kan en grei tilnærming til et representativt utvalg være 75% dag og 25% natt i utviklingssettet.

Beskrivelse av videofiler i utviklingssettet:

- DevGod1.m4v: Totalt 875 rammer, 185 rammer som inneholder fisk. Hele klippet er preget av høy turbulens, ørret svømmer igjennom spalten og ut av bildet. Samme ørret svømmer inn i bilde igjen i en kort stund.
- DevGod2.m4v: Totalt 850 rammer, 394 rammer som inneholder fisk. Hele klippet er preget av moderat turbulens, med litt dårligere siktforhold enn DevGod1.m4v. Tre harr kommer inn fra høyresiden i bildet, nokså nært hverandre. Alle tre svømmer ut av bildet på høyre side igjen.
- DevNatt1.m4v: Totalt 675 rammer, 36 rammer som inneholder fisk. Hele klippet er preget av meget høy turbulens, forholdsvis lyst bilde i forhold til å være nattvideo. <art> kommer svømmende fra høyre siden av bildet og ut igjennom spalteåpningen i høy hastighet.
- DevNatt2.m4v: Totalt 650 rammer, 237 rammer som inneholder fisk. Klippet er preget av lite turbulens og veldig mørkt bilde. Ørret kommer inn fra høyre i bildet, blir værende i samme område av bildet resten av klippet.
- DevOK1.m4v: Totalt 625 rammer, 105 rammer som inneholder fisk. Klippet er preget av veldig høy turbulens. Skitt på linsa som beveger seg litt i flere seksjoner av bildet. Objekt kommer flytende forbi bildet fra høyre i høy hastighet. To fisk kommer svømmende inn spalteåpning og ut av bildet.
- DevOK2.m4v: Totalt 650 rammer, 194 rammer som inneholder fisk. Klippet er preget av moderat turbulens. Fisk kommer svømmende gjennom spalteåpning og ut av bildet.
- DevPoor1.m4v: Totalt 700 rammer, 59 rammer som inneholder fisk. Klippet er preget av lite turbulens og veldig dårlig sikt, vanskelig å se stort annet enn konturen av spalteåpningen. Fisk kommer svømmende igjennom spalteåpningen og ut av bildet på høyre side.
- DevPoor2.m4v: Totalt 750 rammer, 86 rammer som inneholder fisk. Klipp er preget av lite turbulens og dårlig sikt. Fisk kommer svømmende igjennom spalteåpningen og ut av bildet på høyre side.

Totalt er utviklingssettet på 5775 rammer. Ved 25 rammer per sekund tilsvarer dette 231 sekunder, eller i underkant av 4 minutter.

Testsett Formålet med testsettet var å danne en solid, representativ base man kunne gjøre betydelige statistiske beregninger på. Gruppen besluttet at testsettet totalt skulle bli på omtrent 5 timer med video. Ved å dele dette opp i 20 klipp, hvert på 15 minutter, og forsøke å få fordelingen til å stemme nogenlunde med figur 5.2 og 5.3 har gruppen endt opp med følgende filer i testsettet:

- Fire videoklipp, BrukBarSiktLiteTurb 1 - 4
- OkSiktMiddelsTurb 1 - 4
- KlarSiktMyeTurb 1 - 4
- Dårlig 1 - 2
- Elendig 1 - 2

- NattOkSikt 1 - 2
- NattGodSikt 1 - 2

Det er her benyttet selvbeskrivende navn for å forenkle prosessen med å finne ut hvilke filer som tilhører hvilken kategori ved senere behov.

5.2 Etablering av 'ground truth'

For å tilrettelegge for testing senere i prosjektet var det behov for å etablere en 'ground truth' av et utvalg av datasettet. For å kunne gjøre en ren objektiv evaluering av kvaliteten på deteksjonene verktøyet utfører, var det behov for å både vite hvilke rammer i en gitt videofil som inneholder fisk, og hvor i disse rammene fisken(e) befinner seg til enhver tid. Dette behovet er veldig vanlig for maskinlæring, og en naturlig konsekvens av dette er at det også finnes en lang rekke verktøy for å hjelpe til med denne annoteringsprosessen.

5.2.1 Annotering med CVAT

CVAT [41] (Computer Vision Annotation Tool) er et gratis, åpent web-basert verktøy utviklet av Intel, laget for annotering av bilder og video. Verktøyet er lisensert under MIT-lisensen[42]. Gruppen ble tipset om verktøyet av veileder, som allerede hadde gode erfaringer med bruk av det ved flere tidligere anledninger. CVAT har mange muligheter for ulike formater man kan laste opp fullførte annotasjoner i, bla. XML, YOLO, LabelMe og MOT. Ettersom utviklingen av hovedprogrammet og design av hvordan testingen skulle automatiseres skjedde parallelt med utarbeiding av 'ground truth'-settet, ga dette mye fleksibilitet ved at filformatet til 'ground truth'-settet ikke måtte fastsettes tidlig i prosessen.

Utviklerne oppgir selv fire svakheter ved verktøyet per 1. Mars 2019[43];

- Sterkt begrenset støtte for ulike nettlesere (Verktøyet testes kun i Google Chrome, fungerer kanskje på ulike Chromium-baserte nettlesere)
- Begrenset automatisert testing
- Begrenset dokumentasjon av kildekode. Dette vil eventuelt gjøre det spesielt utfordrende å skulle tilpasse eller videreutvikle verktøyet.
- Utfordringer med ytelse ved store mengder annotert data

Det å være låst til å måtte bruke visse nettlesere ble ikke sett på som noe problem for gruppen. Ettersom verktøyet allerede hadde funksjonalitetene det var behov for til annoteringen, var det lite sannsynlig at gruppen hadde behov for å gjøre noen endringer i selve verktøyet. På grunn av dette var både mangelen på automatisert testing og begrenset dokumentasjon ikke forventet å være noe problem.

Utfordringer med ytelse ved store mengder annotert data var punktet som kunne være et potensielt problem. Etter å ha gjort en enkel test på annotering av videofiler på 15 minutter oppstod det et problem ved opplasting av filene. Maksimal filstørrelse så ut til å ligge rundt 1 gigabyte, men de originale videofilene

var i overkant av 1.5 gigabyte. Ved å nedskalere fra den originale oppløsningen på 1920x1080 i .mp4-format til 1280x720 i .m4v-format med kvalitet satt til 23 i HandBrake ble filene typisk et sted mellom 600 og 950 megabyte.

CVAT er et web-basert verktøy som er utviklet slik at det behøver å kjøres fra en container, typisk ved hjelp av Docker[44]. Når containeren kjøres vil man kunne få tilgang til selve verktøyet via lokalnettverket, typisk ved å gå til `localhost:<port>` i Google Chrome (CVAT støtter i utgangspunktet kun Google Chrome og Chromium-baserte browsere). Ettersom det var en del arbeid å få containeren opp og kjøre, og at spesielt oppsett med proxy/VPN for å gi eksterne maskiner tilgang til verktøyet var mer krevende, besluttet gruppen at kun ett gruppemedlem skulle ta seg av annoteringsprosessen. Dette tillot at all annotering kunne gjøres lokalt, og minimerte den forventede tiden prosessen skulle ta ved at aktiviteter som opplasting av nye videofiler, spoling i video og henting av ferdigannotert materiale skulle være så enkelt som mulig.

5.2.2 Retningslinjer for annotasjon

For å sikre at all annotering ble gjort likt gjennom hele testsettet ble det nødvendig å utvikle noen retningslinjer for annotasjon.

- Det skal kun benyttes én annoteringsramme for hver fisk
- Annoteringsrammen skal alltid dekke alt som er synlig av fisken, inkludert finner/ekstremiteter. Utover dette skal boksen dekke så lite areal som mulig, en såkalt 'minimum bounding box'.
- Hvis det er okkluderende objekter i bildet skal kun det av fisken som er synlig annoteres. Dersom den samme fisken er synlig på begge sider av det okkluderende objekter skal det benyttes én annoteringsramme for å lage en 'minimum bounding box' for alt som er synlig av fisken.
- Dersom fisken som skal annoteres er for lik bakgrunnen til at man klarer å se nøyaktig hvor den er for hver ramme, men det likevel er helt sikkert at den fortsatt er i bildet, skal en kombinasjon av menneskelig intuisjon og interpolering brukes til å få til den beste tilnærmingen som mulig.

5.2.3 Annotering av test- og utviklingssett

Før selve annoteringsprosessen ble satt igang ble filene som skulle annoteres gjennomgått for alle forekomster av fisk. Start- og sluttider for forekomstene ble notert ned og konvertert til rammenummer fra videofilens start. Alle videofilene ble konvertert til en oppløsning på 1280x720 i .m4v-format i HandBrake[45] med kvalitet satt til 23, og deretter lastet opp til lokalt kjørende instans av CVAT som separate oppgaver. All annotering gjort av gruppa foregikk i CVAT 0.6.1.

Annoteringen foregikk over flere sprints. Prosessen viste seg å være langt mer krevende enn originalt forventet; det var veldig mye materiale med tidvis ekstremt høy forekomst av fisk. Dette kombinert med at oppgaven var veldig repetitiv gjorde det utfordrende å jobbe med i lengre tidsrom sammenhengende.

5.2.4 Filformat MOT ZIP 1.1

Av de ulike formatene CVAT kunne tilby for nedlasting av annoteringsmateriale, virket MOT ZIP 1.1 umiddelbart som den som var lettest å bruke. Formatet er veldig direkte, har kommaseparerte verdier og ser veldig likt ut det gruppen allerede hadde tenkt på å lage.

Etter nedlasting av annotasjonsfil fra CVAT består hver linje i filen av følgende kommaseparerte verdier:

1. Rammenummer i videofilen
2. Nummer på hvilken “annotasjonsgruppe” den hører til
3. Horisontal avstand fra venstrekant av bildet til annotert rektangel
4. Vertikal avstand fra øvre kant av bildet til annotert rektangel
5. Bredde på detektert rektangel, målt i antall piksler
6. Høyde på detektert rektangel, målt i antall piksler
7. Konfidens på deteksjon (Ved gruppens bruk alltid 1)
8. Navn på annotasjon (f.eks. “fisk”)

Utdrag av 10 første linjer i fasitfilen for DevNatt1 fra utviklingssettet vises i liste 5.1:

Kodeliste 5.1: DevNatt1.csv Utdrag

```
675
413,1,1246.326171875,274.48046875,33.673828125,41.315940856933594,1,1,1.0
414,1,1220.1356026785713,275.35468750000007,59.86439732142867,54.35159933907636,1,1,1.0
415,1,1193.945033482143,276.2289062500001,86.05496651785711,67.38725782121924,1,1,1.0
416,1,1167.7544642857142,277.10312500000015,112.24553571428578,80.422916303362,1,1,1.0
417,1,1141.5638950892858,277.9773437500002,138.43610491071422,93.45857478550482,1,1,1.0
418,1,1115.373325892857,278.8515625000003,164.6266741071429,106.49423326764759,1,1,1.0
419,1,1089.1827566964284,279.7257812500003,190.81724330357156,119.52989174979041,1,1,1.0
420,1,1062.9921875,280.60000000000036,217.0078125,132.56555023193323,1,1,1.0
421,1,1044.5709635416667,293.76666666666705,227.52903645833294,128.17069346109986,1,1,1.0
```

Det er kun behov for rammenummer, X1, Y1, bredde og høyde fra disse filene (verdier 1, 3, 4, 5 og 6) i resultatanalysen som har blitt utarbeidet i oppgaven.

Tap av ferdigannotert materiale

Ved et tidspunkt under sprint 8 fikk gruppe medlemmet som hadde ansvaret for annotering et problem med den lokale instansen av CVAT. Containeren mistet responsivitet, og etter en maskinstart klarte ikke Docker lenger å starte den i hele tatt. Dette problemet viste seg å være vanskelig å løse, ettersom CVAT er basert på et filsystem som ikke er kompatibelt med Windows, og det i tillegg er vanskelig å finne dokumentasjon på hvordan filene er organisert i dette systemet.

Ved dette punktet hadde 17 av 20 filer i testsettet blitt ferdigannotert, men bare 12 av filene hadde blitt opplastet fra CVAT. Forsøk på gjenoppretting av filene ble ikke vellykket, og det ble ikke investert for mye i tid i dette, da det som ble lagt inn av innsats ikke så ut til å gi noen fremgang. For at det skulle være mulig å holde seg til skjema og ikke bli satt tilbake for mye av dette problemet, fikk gruppen tilbud om å få inn assistanse fra en forskningsassistent som jobber med lignende oppgaver til å gjøre det resterende annotasjonsarbeid, inkludert det som hadde gått tapt.

For å kvalitetssikre annoteringene som hadde blitt outsourcet, og også eget tidligere annotert materiale, ble det utviklet en enkel Python-fil¹. Filen åpner en videofil, nedskalert til 1280x720 piksler, og en fasitfil i MOT 1.0/1.1-format og spiller av videofilen ramme for ramme. For hver ramme sjekker den om fasitfilen inneholder en annotasjon for den gitte rammen. Dersom den finner en annotasjon tegnes denne i videorammen som et grønt rektangel. Dette gjør at videofilen med annotasjoner spilles av i en hastighet som er litt raskere enn sanntid

¹annotationChecker.py fra kildekode

Testprosedyre

Dette kapitlet omhandler prosessen med å utvikle prosedyrer for å få uthentet resultater fra deteksjonsalgoritmen, å få sammenlignet resultatene med de annoterte datasettene, og til slutt få satt disse resultatene i system.

For å kunne måle om resultatmålene som har blitt satt faktisk oppnås var det et behov for å gjennomføre testing av deteksjonsalgoritmen. Tilgang på et test-system virket også som et veldig nyttig verktøy for å kunne kontinuerlig forbedre deteksjonsalgoritmen. Ved å se på de individuelle resultatene fra målinger av de veldig varierte vannforholdene kunne gruppen forhåpentligvis finne sterke og svake områder av deteksjonsalgoritmen, og prioritere fokusområder basert på disse resultatene. Det er mange likheter mellom utskriften av det endelige resultatet og utskriften av resultatfiler som skulle brukes i testing. Dette kapitlet tar kun for seg produksjon og analyse av resultatfiler brukt til testing, endelige resultat omtales i eget kapittel.

6.1 Produksjon av resultatfiler

6.1.1 Egenutviklet csv-format

Ettersom gruppen var fri på hvordan det egenutviklede filformatet skulle se ut, ble dette høyst minimalistisk og enkelt utformet. CSV-formatet ble beholdt, da det ble ansett som enklere å lage et format som var så likt som MOT ZIP 1.1-formatet som mulig. Det genereres en separat resultatfil for hver videofil det kjøres deteksjon på.

I starten av dokumentet ble det lagt til tre linjer for metadata fra deteksjonen som var kjørt:

1. Tid for gjennomkjøring av deteksjon
2. Oppløsning deteksjonen ble gjennomført i
3. Tall på hvor mange rammer som ble benyttet for bakgrunnssubtraksjon

Alle resterende linjer i filen tilsvarende en positiv deteksjon i en gitt ramme. Hver linje følger samme format:

1. Rammenummer i videofilen

2. Horisontal avstand fra starten på bildet til detektert rektangel, målt i antall piksler
3. Vertikal avstand fra starten på bildet til detektert rektangel, målt i antall piksler
4. Bredde på detektert rektangel, målt i antall piksler
5. Høyde på detektert rektangel, målt i antall piksler

Hver av disse verdiene er kommaseparerte. Merk at koordinatene og bredde/høyde til rektangelet er basert på den eventuelt nedskalerte videoen, og kan dermed ikke nødvendigvis sammenlignes direkte med de manuelt annoterte rammene fra datasettene.

6.1.2 Utskrift av resultatfiler

For å få skrevet resultater fra deteksjonen til fil var det to utfordringer som måtte løses; å få lagret de nødvendige dataene kontinuerlig ettersom det ble gjort deteksjoner, og deretter få skrevet disse dataene til riktig fil i riktig format.

Det er behov for å opprette en mappe alle resultatfilene skal legges i før deteksjonsalgoritmen settes igang. For at mappenavnet skal være garantert unikt blir navnet generert basert på dato og tid, ved hjelp av Pythons innebygde datetimestil bibliotek [46]. For at mappene skal være lette å sortere basert på navn, blir funksjonen `datetime.datetime.now().isoformat()` benyttet. Denne funksjonen returnerer henholdsvis år, måned, dag, time, minutt og sekunder. Disse verdiene er kolonseparerte, som kan skape problemer på noen operativsystemer dersom det brukes som en filsti. Dette løses ved å erstatte alle forekomster av kolon (:) med understrek (_). Mappen opprettes ved å kalle funksjonen `os.mkdir` med en relativ filsti som argument. Denne mappen vil bli benyttet til resultatfiler gjennom resten av deteksjonen som er startet. Kodeliste 6.1 viser hvordan dette skjer i kildekoden.

Kodeliste 6.1: TestDeteksjonGUI.py oppretting av mappe for resultatfiler

```

1  datetime = str(datetime.datetime.now().isoformat())
2  datetimePath = datetime.replace(":", "_")
3  os.mkdir(str("Test/" + datetimePath))

```

De fire verdiene for rektangelet (nummer 2-5 fra liste) som beskriver hvor en deteksjon forekom blir alle hentet ut fra `openCV`-funksjonene `findContours` [47] og `boundingRect` [48], som vil bli omtalt nærmere i kapittel om deteksjonsalgoritmen. Se kodeliste ?? for implementasjon.

Kodeliste 6.2: Fra TestDeteksjonGUI.py, henting av koordinater for detekterte rektangel

```

1  contours, hierarchy = cv.findContours(frame, cv.RETR_EXTERNAL, cv.CHAIN_APPROX_NONE)
2  (x, y, w, h) = cv.boundingRect(contour)

```

Rammenummeret for deteksjonen blir hentet ut ved å kopiere `CAP_PROP_POS_FRAMES`-verdien, som er en del av `VideoCapture`-objektet som benyttes for å iterere gjennom videofilene. Verdien kopieres over til variabelen `detected_frame_number`.

Det opprettes en string, `found_frame_string`, ved å manuelt legge inn og komma-separere verdiene fra henholdsvis `detected_frame_number`, og `x`, `y`, `w`, `h` fra `boundingRect`. Denne stringen appendes til `frameList`, en liste som da vil inneholde ferdigformaterte deteksjonslinjer som er klare til å bli skrevet til fil. Se kodeliste 6.3 for implementasjon.

Kodeliste 6.3: Fra `TestDeteksjonGUI.py`, opprettelse og lagring av CSV-linje

```

1 detected_frame_number = cap.get(cv.CAP_PROP_POS_FRAMES)
2 found_frame_string = str(str(int(detected_frame_number)) + "," + str(x) + ","
3 + str(y) + "," + str(w) + "," + str(h))
4 frameList.append(found_frame_string)

```

Etter deteksjonen er ferdigkjørt på en videofil vil programmet hoppe til (kodeblokk under). For å få skrevet til fil blir det opprettet et filobjekt ved hjelp av den innebygde `open`-funksjonen i Python. Filstien er basert på den tidligere opprettede `datetimePath`, selve filnavnet hentes fra siste entry i listen som resulterer av å kjøre en string split på den relative pathen videofilen det ble kjørt deteksjon på stammer fra.

For å skrive deteksjonslinjene opprettes en `for`-løkke som itererer igjennom hele `frameList`, se kodeliste 6.4. For å unngå en tom linje på slutten av filen legges det inn linjeskift ("`\n`") før selve deteksjonslinjen skrives ut. Da det uansett skal skrives inn metadata i starten av filen ved et senere tidspunkt er det ikke et problem at det opprettes en tom linje i starten av dokumentet.

Kodeliste 6.4: `TestDeteksjonGUI.py` skrivning av resultatlinjer til fil

```

1 fileStringSplit = file.split('\\')
2 file = str('./Test/' + datetimePath + '/' + fileStringSplit[-1])
3 file = file.replace('.m4v', '.csv') # Fjerner .mp4 ending
4 print(file)
5 statFil = open(file, "w+")
6
7 for frame in frameList:
8     statFil.write("\n")
9     statFil.write(str(frame))

```

For å få målt hvor lang tid det tar å kjøre deteksjonsalgoritmen på en gitt videofil, benyttes `perf_counter`-funksjonen fra `Time`-biblioteket[49] til Python. Verdien som returneres fra funksjonen har ikke noe skikkelig referansepunkt, men er likevel nyttig for å nøyaktig måle tiden mellom to hendelser. Verdien som returneres fra `perf_counter` ved filstart lagres i verdien `file_start`, se kodeliste 6.5.

Kodeliste 6.5: `TestDeteksjonGUI.py` start av timer for deteksjon

```

1 for file in files:
2     file_start = perf_counter()

```

Etter deteksjon er ferdigkjørt og detekterte rammer har blitt skrevet til fil stoppes tidtageren, se kodeliste 6.6. Dette gjøres så sent som mulig for å få en så nøyaktig tidtagning som mulig. Hvor lang tid som har gått beregnes ved å ta tiden returnert fra `perf_counter()`-funksjonen, trekke ifra tiden som ble tidligere lagret i `file_start`-variabelen og lagre resultatet i variabelen `time_passed_file`.

Ved dette punktet er alle deteksjonslinjene skrevet til fil, og filobjektet peker derfor helt til slutten av filen. For å få lagt inn metadata i starten av filen benyttes filobjektet sin seek-funksjon. Variablene Res og BGShistory er parametre for deteksjonsalgoritmen som er definert ved oppstart av programmet. Merk her at første linje som skrives inn ikke behøver en ny linje, da denne allerede ble opprettet i kodeliste ??.

Kodeliste 6.6: Fra TestDeteksjonGUI.py, skriving av metadata til fil

```

1  time_passed_file = perf_counter() - file_start # Stops test timer after writing to file
2  statFil.seek(0, 0)
3  statFil.write(str(time_passed_file))
4  statFil.write("\n")
5  statFil.write(str(Res))
6  statFil.write("\n")
7  statFil.write(str(BGShistory))
8  statFil.write("\n")
9  statFil.close()

```

Utdrag av de 10 første linjene i en resultatfil generert etter gjennomkjøring av deteksjon på DevPoor1 fra utviklingssettet:

Kodeliste 6.7: DevPoor1.csv utdrag

```

13.025328800000011
(640, 480)
150
7,395,324,77,116
27,74,317,68,80
27,423,293,119,187
27,554,273,86,110
27,0,218,108,80
27,0,92,74,180
27,90,0,550,359

```

Merk her at alle koordinatene består av heltall, i motsetning til det tidligere omtalte MOT ZIP 1.1-formatet hvor alle koordinater er kommatall.

6.2 Resultatanalyse

Prosessen med å analysere resultatfilene opp mot fasitfilene kan grovt sett deles opp i to biter; en sammenligning av hvilke rammer resultatfilene og fasitfilene indikerer at inneholder fisk, og en vurdering av hvor godt deteksjonsalgoritmen treffer i de tilfellene der både resultat- og fasitfil indikerer at det finnes en fisk i rammen.

Etter fullført testdeteksjon kalles funksjonen multifilResultatAnalyseUtvikling eller multifilResultatAnalyseTest, avhengig av hvilket datasett deteksjonen er utført på. Funksjonene står for kall av funksjoner som både står for analysering av resultatene og systematisering av disse. Dette delkapittelet vil bare ta for seg analyserings-biten av funksjonene.

Funksjon 6.8 mottar den unike strengen som ble opprettet i 6.1 og bruker denne til å opprette en relativ sti til mappen som inneholder CSV-filene som ble

produsert av deteksjonen. Listen `resFiles` opprettes ved å gjøre et kall til `readFromPath`¹-funksjonen, som iterer igjennom alle filene i mappen. Filsti til de korresponderende fasitfilene lages ved å gjøre noen enkle strengoperasjoner, og funksjonen `resultatAnalyse` kalles med disse to strengene.

Kodeliste 6.8: funksjoner.py multifilResultatAnalyseUtvikling

```

1 def multifilResultatAnalyseUtvikling(folderName):
2     folderPath = str("./Test/" + folderName + "/")
3     fasitPathStub = "./Fasitfiler/DevSet/"
4     plotList = []
5
6     resFiles = readFromPath(folderPath, ".csv")
7     for resFile in resFiles:
8         resSplit = resFile.split('/')
9         fasitPath = str(fasitPathStub + resSplit[3])
10        bestJacFit, referenceList, testList = resultatAnalyse(fasitPath, resFile)
11        plotList.append(bestJacFit)

```

For å få finne ut hvor godt de polygonene som skrives til resultatfilen stemmer overens med de annoterte områdene i fasitfilen, gjøres det en rekke operasjoner som til slutt produserer en liste med jaccard indexer for hver deteksjon som er gjort. For å få lest inn filene, benyttes modulen `reader` fra pakken `CSV`[50]. Dette gjør det enkelt å iterere over linjer og få tak i de ønskede data fra lister som returneres. I funksjonen `resultatAnalyse`, kodeliste 6.9, opprettes det to CSV-lesere for fasit- og resultatfil med filstier som sendes til funksjonen som parametere. Linjene med metadata på starten av filen leses inn og lagres i variabler. X- og Y-axisratio er her forholdstall som beregnes og lagres til senere bruk. Merk her at det hardkodes verdier fra oppløsningen annotasjonene ble gjort i.

Kodeliste 6.9: Funksjon `resultatAnalyse`, opprettelse av CSV-lesere og henting av metadata

```

1 def resultatAnalyse(fasitPath, resPath):
2     fasitFil = open(fasitPath)
3     fasitReader = csv.reader(fasitFil)
4
5     frameCountTotalLine = next(fasitReader)
6     frameCountTotal = int(frameCountTotalLine[0])
7
8     resultatFil = open(resPath)
9     resultatReader = csv.reader(resultatFil)
10    resPathSplit = resPath.split('/')
11    timePassed = next(resultatReader)
12
13    resultResolutionRow = next(resultatReader)
14    resultResolutionY = resultResolutionRow[1]
15    resultResolutionY = resultResolutionY.replace(' ', '\u0020')
16    resultResolutionY = int(resultResolutionY)
17    resultResolutionX = resultResolutionRow[0]
18    resultResolutionX = resultResolutionX.replace(' ', '\u0020')
19    resultResolutionX = int(resultResolutionX)
20    XAxisRatio = resultResolutionX / 1280
21    YAxisRatio = resultResolutionY / 720

```

¹Direkte kopi av <https://mkyong.com/python/python-how-to-list-all-files-in-a-directory/>

```

22
23     backgroundSubtractionFrames = next(resultatReader)
24     next(resultatReader)

```

For å få lest inn verdiene fra fasitfil på en lett brukelig metode, opprettes det en liste analyseList, som består av lister hvor indeksen til en gitt liste korresponderer med rammenummeret i videofilen det har blitt kjørt deteksjon på. Deretter populeres listene ved å iterere igjennom hele fasitfilen. Hver linje som inneholder annotasjonsdata appendes til analyseList sin liste med indeks som samsvarer med rammenummeret for linjen. Dette betyr at en gitt liste i analyseList vil inneholde modulus 4 av antall entries annotasjonspolygoner. ReferenceList er en liste som inneholder binære verdier (0 eller 1) på om en gitt ramme inneholder en fisk eller ikke. Indeksen som korresponderer med rammenummeret i en fasitlinje blir satt til 1, for å markere at rammen inneholder fisk. Se kodeliste 6.10.

Kodeliste 6.10: Funksjon resultatAnalyse, opprettelse av lister og lesing av fasitfil

```

1     referenceList = []
2     testList = []
3     for i in range(frameCountTotal + 1):
4         referenceList.append(0)
5         testList.append(0)
6     analyseList = [[] for i in range(frameCountTotal + 1)]
7         for fasitLinje in fasitReader:
8             frameNr = int(fasitLinje[0])
9             referenceList[frameNr] = 1
10            analyseList[frameNr].append(int(float(fasitLinje[2])))
11            analyseList[frameNr].append(int(float(fasitLinje[3])))
12            analyseList[frameNr].append(int(float(fasitLinje[4])))
13            analyseList[frameNr].append(int(float(fasitLinje[5])))

```

Koden for å få lest inn resultatLinjene og sammenlignet disse med data fra fasitfilene består av en lang for-løkke, se kodeliste 6.11. Hver iterasjon leser inn en linje fra resultatfilen. Listen testList populeres med samme metode som referenceList i kodeliste 6.10, men basert på linjene fra resultatfilen. Rammenummeret i resultatlinjen må bestå en rekke sjekker; at den ikke er 0, er innenfor intervallet med rammer i fasitfilen og ikke er en del av de rammene som benyttes til bakgrunnssubstraksjon i starten av hver videofil. Dersom den består sjekkene, blir resten av resultatlinjen sendt til funksjonen jaccardIndexPolygons for hver fasitlinje som har blitt lagt i den samsvarende indeksen for analyseList. Den høyeste jaccard scoren fra denne løkken blir lagt til i listen jaccardIndexList.

Kodeliste 6.11: funksjoner.py resultatAnalyse

```

1     for resultatLinje in resultatReader:
2         frameNr = int(resultatLinje[0])
3         if frameNr > frameCountTotal:
4             print("Rammetall_ " + str(frameNr) +
5                   " i resultatfilen_ikke_ dekket_ av_ fasitfilen.")
6             break
7         testList[frameNr] = 1
8         if referenceList[frameNr] != 0 and frameNr > int(backgroundSubtractionFrames[0]):
9             nrOfFrames = int(len(analyseList[frameNr]))
10            i = 0

```

```

11     bestJaccardScore = 0
12     while i < nrOfFrames:
13         jaccardScore = jaccardIndexPolygons(analyseList[frameNr][i], analyseList[frameNr][i + 1],
14         analyseList[frameNr][i + 2], analyseList[frameNr][i + 3], resultatLinje[1], resultatLinje[2],
15         resultatLinje[3], resultatLinje[4], XAxisRatio, YAxisRatio)
16         if jaccardScore > bestJaccardScore:
17             bestJaccardScore = jaccardScore
18         i += 4
19     jaccardIndexList.append(bestJaccardScore)

```

For å simulere polygoner og enkelt gjøre beregninger på areal og overlapp blir modulen Polygon fra pakken shapely.geometry benyttet [51]. Funksjonen jaccardIndexPolygons tar imot fire parametere for hvert av de to polygonene, og et forholdstall i X- og Y-aksene som ble vist i (ref til uthenting av ratios). Det må gjøres en rekke konverteringer fra strings til og mellom inter og floater for å få gjennomført de nødvendige beregningene. Merk spesielt at parameterne for x2- og y2-koordinatene for Polygon A er henholdsvis bredde og høyde på rektangelet, og ikke absolutte koordinatpunkter. Ettersom Polygon-klassen fra shapely.geometry-pakken behøver absolutte koordinater, beregnes dette lett ved å addere sammen parameterne Ax1 med Ax2, og Ay1 med Ay2.

For å finne Jaccard Indexen av de to polygonene blir de to Polygon-objektene først opprettet. Ved å kalle intersection-metoden fra et av Polygon-objektene med det andre polygonen som argument returneres det overlappende arealet mellom de to rektanglene. De resterende matematiske operasjonene blir forholdsvis trivielle. Funksjonen returnerer jaccard Indexen mellom de to rektanglene.

Kodeliste 6.12: funksjoner.py jaccardIndexPolygons

```

1     def jaccardIndexPolygons(Ax1, Ay1, Ax2, Ay2, Bx1, By1, Bx2, By2, Xratio, Yratio):
2         Ax1 = int(Ax1)
3         Ay1 = int(Ay1)
4         Ax2 = int(Ax2 + Ax1)
5         Ay2 = int(Ay2 + Ay1)
6         polygonA = Polygon([(Ax1, Ay1), (Ax2, Ay1), (Ax2, Ay2), (Ax1, Ay2)])
7
8         Bx1 = int(float(Bx1) / Xratio)
9         By1 = int(float(By1) / Yratio)
10        Bx2 = int((float(Bx2) / Xratio) + float(Bx1))
11        By2 = int((float(By2) / Yratio) + float(By1))
12        polygonB = Polygon([(Bx1, By1), (Bx2, By1), (Bx2, By2), (Bx1, By2)])
13
14        intersectionArea = polygonA.intersection(polygonB)
15        if (polygonA.area + polygonB.area - intersectionArea.area) > 0.0:
16            jaccardIndex = (intersectionArea.area) / (polygonA.area + polygonB.area
17            - intersectionArea.area)
18        else:
19            jaccardIndex = 0
20
21        return jaccardIndex

```

6.3 Systematisering og analyse av testresultater

Proessen med systematisering og analyse av testresultatene kan deles opp i to biter; fremstilling av boksploTT for listen med Jaccard Index-verdier og opprettelse av forvirringsmatriser med medfølgende statistiske interessante verdier. For å forenkle utviklingen av denne prosessen er det i kildekoden opprettet to separate funksjoner, en for utviklingssett og en for testsett.

6.3.1 BoksploTT av Jaccard index

For å fremstille de potensielt lange listene med jaccard indekser i et mer leselig format blir det opprettet boksploTT for hver liste. For å slippe å måtte se på en lang rekke bildefiler separat, blir alle de genererte boksploTTene lagt i et samlet bilde ved hjelp av Matplotlib.pyplot[52]. Koden for å gjennomføre dette, presentert i kodeliste 6.13, behøver kun listen plotList. Listens første skuff, 0, inneholder en streng med filnavnet på resultatfilen. Alle resterende skuffer i listen er verdier på beste jaccard index for en uspesifisert ramme. Disse verdiene stammer fra et tidligere funksjonskall til resultatanalyse-funksjonen, som benytter funksjonen fra kodeliste 6.12 omtalt tidligere.

Kodeliste 6.13: Produksjon av boksploTT for testresultater

```

1 plotListIterator = iter(plotList)
2 figs, axes = plt.subplots(5, 4, sharey=True)
3
4 for i, axs in enumerate(axes):
5     for j, col in enumerate(axes):
6         pltList = next(plotListIterator)
7         pltListName = str(pltList[0]).split('.')
8         pltListName[0].replace("File1-", "")
9         axes[i, j].boxplot(pltList[1])
10        axes[i, j].set_xticks([])
11        axes[i, j].set_title(str(pltListName[0]), fontsize=5)
12
13 plt.subplots_adjust(hspace=0.25, wspace=0.25)
14 plt.savefig(folderPath + "statistics.png", bbox_inches='tight')
```

6.3.2 Opprettelse av forvirringsmatriser

Kodelistene 6.10 og 6.11 viser hvordan det ved dette punktet finnes to lister, referenceList og resultList, som er fylt med binære verdier for om en gitt indeks sin korresponderende ramme betegnes som å inneholde fisk av henholdsvis fasitfilen og resultatfilen. For å få satt dette i et system som er mer leselig fra et statistisk synspunkt blir det opprettet forvirringsmatriser og deriverte metaverdier ved hjelp av pakken sklearn.metrics[53].

Liste 6.14 viser hvordan disse forvirringsmatrisene opprettes. Ved å lese inn en filsti blir hele mappen med resultatfiler loopet igjennom. Hver fil kjøres igjennom funksjonen resultatAnalyse tidligere omtalt. To av listene som returneres, referenceList og predictedList, er alt som trengs for å generere forvirringsmatrisene. Ved

å kjøre et funksjonskall til `confusion_matrix` med disse listene som argument, blir det returnert en ordinær forvirringsmatrise. To lignende funksjonskall kjøres også, og de tre variablene blir skrevet til fil som strenger.

Kodeliste 6.14: Opprettelse av forvirringsmatriser, skriving til fil

```

1 confusion_matrix_file = open(str(folderPath + "confusion_matrix.csv"), "w+")
2 resFiles = readFromPath(folderPath, ".csv")
3
4 for resFile in resFiles:
5     resSplit = resFile.split('/')
6     fasitPath = str(fasitPathStub + resSplit[3])
7     bestJacFit, referenceList, predictedList = resultatAnalyse(fasitPath, resFile)
8     confusion_matrixes = confusion_matrix(referenceList, predictedList)
9     confusion_matrixes_normalized = confusion_matrix(referenceList,
10    predictedList, normalize='true')
11    confusion_matrixes_report = (classification_report(referenceList, predictedList))
12    confusion_matrix_file.write(resFile + "\n" +
13    str(confusion_matrixes) + "\n" + str(confusion_matrixes_normalized)
14    + "\n" + str(confusion_matrixes_report) + "\n")
15    plotList.append(bestJacFit)
16 confusion_matrix_file.close()

```

Liste 6.15 viser et eksempel på hvordan en fil kjørt på to filer fra testsettet vil se ut etter funksjonen er ferdig kjørt. Det vises her henholdsvis en forvirringsmatrise, en normalisert forvirringsmatrise og diverse beregnede verdier

Kodeliste 6.15: `Confusion_matrix.txt` etter å ha kjørt `BrukbarSiktLiteTurb1` og `BrukBarSiktLiteTurb2` fra testsett med modus sakte

```

1 ./Test/2020-04-01T14_07_12.728930/BrukbarSiktLiteTurb1.csv
2 [[16029 382]
3 [ 4412 1668]]
4 [[0.97672293 0.02327707]
5 [0.72565789 0.27434211]]
6           precision    recall  f1-score   support
7
8          0         0.78      0.98      0.87     16411
9          1         0.81      0.27      0.41     6080
10
11     accuracy                0.79     22491
12   macro avg         0.80      0.63      0.64     22491
13 weighted avg         0.79      0.79      0.75     22491
14
15 ./Test/2020-04-01T14_07_12.728930/BrukbarSiktLiteTurb2.csv
16 [[21088 330]
17 [ 657 429]]
18 [[0.9845924 0.0154076 ]
19 [0.60497238 0.39502762]]
20           precision    recall  f1-score   support
21
22          0         0.97      0.98      0.98     21418
23          1         0.57      0.40      0.47      1086
24
25     accuracy                0.96     22504
26   macro avg         0.77      0.69      0.72     22504
27 weighted avg         0.95      0.96      0.95     22504

```

Deteksjonsalgoritme

Dette kapitlet vil beskrive hvordan deteksjonsalgoritmen gruppen har utviklet fungerer, og avslutningsvis vise hvordan denne algoritmen presterer i forhold til fasitfilene ved å benytte testprosedyre som har blitt beskrevet.

7.1 Pipeline

Deteksjonsalgoritmen jobber separat på ett og ett bilde om gangen, med et unntak for bakgrunnssubtraksjon. Bildet blir sendt igjennom en rekke operasjoner, før det til slutt blir utført en sjekk på om det inneholder det som kvalifiserer til en positiv deteksjon. Merk at kodelister som presenteres i dette delkapitlet tidvis vil være kompilerte versjoner av den originale koden, ved bruk av funksjoner utenfor deteksjon.py er funksjoner definert i starten av listen. Det vil i denne seksjonen bli benyttet det samme bildet for alle eksempler, ramme X fra video y.

Ved initialisering av deteksjonsalgoritmen vil det være valgt en modus som inneholder hardkodete verdier for de ulike prosesseringsteknikkene som blir benyttet, se liste 7.1.

Kodeliste 7.1: Hardkodete argument for deteksjonsalgoritme, hurtig modus

```
1 if modus == "Hurtig": #Hurtig modus er valgt
2   Res = fu.getResolution(360)
3   interpolation = cv.INTER_LANCZOS4
4
5   BGShistory = 100
6   BGSubtractor = cv.createBackgroundSubtractorKNN(BGShistory, 30.0, detectShadows=False)
7
8   kernel = cv.getStructuringElement(cv.MORPH_OPEN, (3, 3))
9
10  minSize = 1000
11  maxSize = float('inf')
12  connectivity = 8
```

7.1.1 Innlesing av bilde

For å få lest inn rammer fra en videofil opprettes det et VideoCapture[54]-objekt fra openCV med en filsti til videofilen som argument, se kodeliste 7.2. Ved å bruke

VideoCapture-objektets read-metode i en evig while-løkke får deteksjonsalgoritmen kjørt det gitte bildet igjennom hele prosesseringslinjen før neste bilde hentes inn, og unngår slik en potensiell race condition. Dette er while-løkken resten av algoritmen opererer innenfor.

Kodeliste 7.2: Innlesing av bilde fra videofil

```

1 cap = cv.VideoCapture(file)
2 while True:
3     ret, frame = cap.read()
4     if frame is None:
5         break

```

7.1.2 Nedskalering

For å få deteksjonsalgoritmen til å operere tilfredsstillende raskt er det behov for å nedskalere bildet den jobber på. I tillegg til økt prosesseringshastighet, vil dette også utføre en velkommen støyreduksjon på bildet. Dette gjennomføres ved å bruke openCV-funksjonen `resize`[55]. Merk her at oppløsning og interpolasjonsmetode som benyttes er bestemt i kodeliste 7.1. Se kodeliste ?? for implementasjon.

Kodeliste 7.3: Nedskalering av bilde

```

1 def resize(img, resolution, interpolation):
2     img = cv.resize(img, resolution, interpolation=interpolation)
3     return img
4 frame = resize(frame, Res, interpolation)

```

7.1.3 Bakgrunnssubtraksjon

Etter å ha sett et tilstrekkelig antall bilder, som standard 120, har man bygd opp stor nok historikk til å kunne benytte bakgrunnssubtraksjon med god effekt. Bakgrunnssubtraksjons-objektet som ble opprettet i kodeliste 7.1 kjøres på bildet, og returnerer et binært bilde hvor alle statiske elementer har blitt fjernet, se kodeliste 7.4 for implementasjon.. Etersom støy i form av turbulens og bobler i bildet ikke vil bli betraktet som statiske elementer, kan man ved kjøring i utviklingsmodus se at dette blir spesielt fremtredende etter denne operasjonen.

Kodeliste 7.4: Bakgrunnssubtraksjon

```

1 frame = BGSubtractor.apply(frame)

```

7.1.4 Morfologiske operasjoner

Det er etter konverteringen til binært bilde et betydelig behov for å fjerne så mye støy som mulig. Ved å benytte den morfologiske operasjonen åpning, essensielt en erosjon fulgt av en dilasjon, vil de fleste små grupper med hvite piksler bli eliminert uten at det i for høy grad går utover forekomsten av store grupperinger.

Operasjonen gjennomføres ved å benytte openCVs funksjon `morphologyEx`[56] med argumentet `CV_MORPH_OPEN` på bildet, som vist i kodeliste 7.5.

Kodeliste 7.5: Morfologisk åpning av bilde

```
1 frame = cv.morphologyEx(frame, cv.MORPH_OPEN, kernel, iterations=1)
```

7.1.5 Sammenkoblede komponenter

For å ytterligere redusere støy i bildet, brukes funksjonen `connectedComponents` fra openCV. Funksjonen leter etter sammenhengende biter med piksler, og fjerner piksler som ikke oppfyller kriteriene. Se kodeliste 7.4.

Kodeliste 7.6: Fjerning av ikke-koblede komponenter

```
1 frame = fu.connectedComponents(frame, connectivity, minSize, maxSize)
2 frame = np.uint8(frame)
```

7.1.6 Konturer

Som den avsluttende prosesseringen, som også kan betraktes som selve deteksjonen i algoritmen, gjennomføres et søk etter konturer, som vist i liste 7.7. Dersom det oppdages en eller flere konturer i rammen her, vil det bli ansett som en eller flere positive deteksjoner. Verdiene for konturene som oppdages skrives til fil, som vist tidligere i testprosedyre. Ved dette punktet er deteksjonsalgoritmen ferdig med denne rammen, og vil hoppe til start med neste ramme.

Kodeliste 7.7: Konturdeteksjon i rammen

```
1 contours, hierarchy = cv.findContours(frame, cv.RETR_EXTERNAL, cv.CHAIN_APPROX_NONE)
2
3     frame = cv.cvtColor(frame, cv.COLOR_GRAY2RGB)
4
5     for contour in contours:
6         if 200 <= cv.contourArea(contour) <= 100000:
```

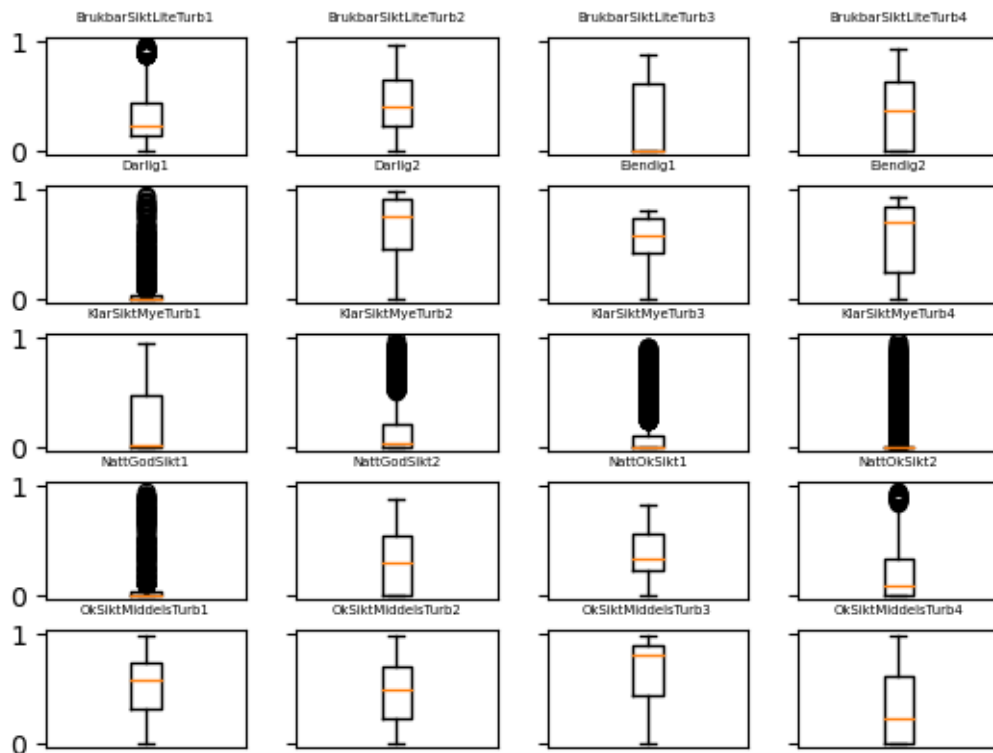
7.2 Testresultater

7.2.1 Hurtigmodus

Ved testing av hurtigmodus på testsett ble parametere vist i kodeliste 7.8 benyttet. Testprosedyre ble automatisk fulgt, som resulterte i boksplokk 7.1 og metaverdier fra forvirringsmatriser som ble produsert er vist i tabell 7.1.

Kodeliste 7.8: Hardkodete verdier for deteksjonsmodus hurtig

```
1 if modus == "Hurtig": #Hurtig modus er valgt
2     Res = fu.getResolution(360)
3     interpolation = cv.INTER_LANCZOS4
4
```

Figur 7.1: Boksploott av Jaccard Index fra testsett, modus hurtig

```

5   BGSHistory = 100
6   BGSubtractor = fu.backgroundSubtractorKNN(BGSHistory, 30.0, False)
7
8   kernel = cv.getStructuringElement(cv.MORPH_OPEN, (3, 3))
9
10  minSize = 1000
11  maxSize = float('inf')
12  connectivity = 8

```

7.2.2 Saktemodus

Ved testing av saktemodus ble parametere vist i kodeliste 7.9 benyttet. Testprosedyre ble automatisk fulgt, som resulterte i boksploott 7.2 og metaverdier fra forvirringsmatriser som ble produsert vist i tabell 7.2.

Kodeliste 7.9: Hardkodete verdier for deteksjonsmodus sakte

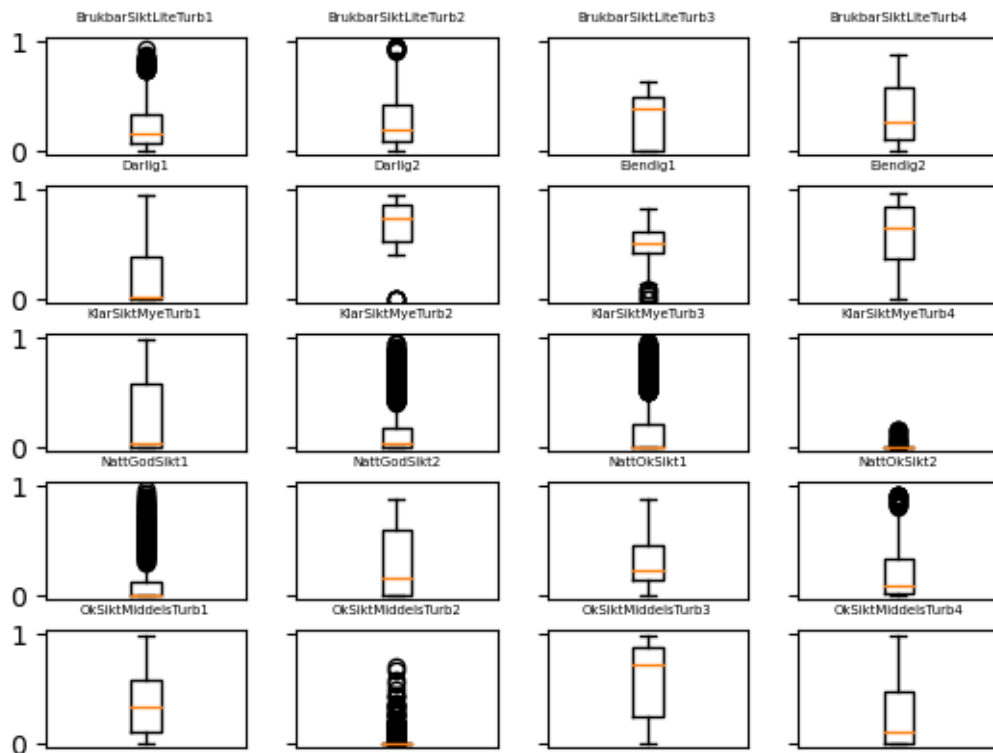
```

1 elif modus == "Sakte": #Sakte modus er valgt
2
3     Res = fu.getResolution(480)
4     interpolation = cv.INTER_LANCZOS4
5
6     BGSHistory = 150
7     BGSubtractor = fu.backgroundSubtractorKNN(BGSHistory, 50.0, False)

```

Modus hurtig, testsett				
Fisk / ikke fisk	Presisjon	Recall	F1-score	Support
BrukbarSiktLiteTurb1.csv				
0	0.88	0.61	0.72	16411
1	0.43	0.78	0.55	6080
BrukbarSiktLiteTurb2.csv				
0	0.97	0.99	0.98	21418
1	0.76	0.49	0.60	1086
BrukbarSiktLiteTurb3.csv				
0	0.96	0.74	0.83	21593
1	0.03	0.21	0.06	908
BrukbarSiktLiteTurb4.csv				
0	0.86	0.81	0.84	19302
1	0.16	0.22	0.19	3199
Darlig1.csv				
0	1.00	0.01	0.01	22327
1	0.01	1.00	0.02	174
Darlig2.csv				
0	0.99	0.99	0.99	22334
1	0.26	0.30	0.28	167
Elendig1.csv				
0	1.00	0.72	0.83	22420
1	0.01	0.70	0.02	81
Elendig2.csv				
0	1.00	0.84	0.91	22298
1	0.04	0.70	0.07	203
KlarSiktMyeTurb1.csv				
0	1.00	0.17	0.29	22097
1	0.02	0.96	0.04	404
KlarSiktMyeTurb2.csv				
0	0.98	0.10	0.18	21088
1	0.07	0.97	0.13	1413
KlarSiktMyeTurb3.csv				
0	0.89	0.00	0.01	22092
1	0.02	0.97	0.03	411
KlarSiktMyeTurb4.csv				
0	0.97	0.01	0.01	21955
1	0.02	0.99	0.05	547
NattGodSikt1.csv				
0	1.00	0.24	0.39	22289
1	0.01	1.00	0.02	212
NattGodSikt2.csv				
0	1.00	0.16	0.27	22468
1	0.00	0.97	0.00	33
NattOkSikt1.csv				
0	0.99	0.99	0.99	22213
1	0.37	0.41	0.39	279
NattOkSikt2.csv				
0	0.99	0.45	0.62	22081
1	0.03	0.86	0.05	415
OkSiktMiddelsTurb1.csv				
0	0.92	0.84	0.88	17300
1	0.59	0.77	0.67	5192
OkSiktMiddelsTurb2.csv				
0	0.86	0.92	0.89	17623
1	0.61	0.48	0.54	4882
OkSiktMiddelsTurb3.csv				
0	0.98	0.91	0.94	19868
1	0.55	0.85	0.67	2633
OkSiktMiddelsTurb4.csv				
0	0.93	0.74	0.83	20087
1	0.21	0.57	0.30	2414

Tabell 7.1: Testresultater fra kjøring av deteksjonsalgoritme på testsett, hurtig modus



Figur 7.2: Boksploott av Jaccard Index fra testsett, modus sakte

```

8   kernel = cv.getStructuringElement(cv.MORPH_OPEN, (5, 5))
9
10
11   minSize = 1000
12   maxSize = float('inf')
13   connectivity = 8

```

7.2.3 Ekstern maskinlæringsalgoritme

For å kunne ha et sammenligningsgrunnlag som ikke bare er basert på deteksjonsalgoritmen gruppen selv har utviklet, har veileder for oppgaven, Marius Pedersen, utviklet en egen deteksjonsalgoritme basert på maskinlærings. Han beskriver selv algoritmen slik: Dyplæringsalgoritmen er basert på et forhåndstrenet Alexnet, hvor de tre siste lagene er fine-tuned for å klassifisere bilder i 3 klasser (hel fisk, delvis fisk, ingen fisk). Dette er gjort på et datasett bestående av omtrent 500 000 bilder, hvor bildene av fisk hovedsakelig er fra siden (sidebilde av fisken)."

For å få ut resultater fra dyplæringsalgoritmen som kan sammenlignes med deteksjonsalgoritmen utviklet i denne oppgaven har veileder selv formet verktøyet til å produsere resultatfiler som er kompatible med testprosedyren gruppen har utviklet. Ettersom algoritmen fungerer på et fundamentalt forskjellig plan, er det

Modus sakte, testsett				
Fisk / ikke fisk	Presisjon	Recall	F1-score	Support
BrukbarSiktLiteTurb1.csv				
0	0.78	0.98	0.87	16411
1	0.81	0.27	0.41	6080
BrukbarSiktLiteTurb2.csv				
0	0.97	0.98	0.98	21418
1	0.57	0.40	0.47	1086
BrukbarSiktLiteTurb3.csv				
0	0.96	0.80	0.87	21593
1	0.04	0.17	0.06	908
BrukbarSiktLiteTurb4.csv				
0	0.87	0.87	0.87	19302
1	0.20	0.20	0.20	3199
Darlig1.csv				
0	1.00	0.43	0.60	22327
1	0.01	0.99	0.03	174
Darlig2.csv				
0	0.99	0.99	0.99	22334
1	0.14	0.30	0.19	167
Elendig1.csv				
0	1.00	0.79	0.88	22420
1	0.01	0.54	0.02	81
Elendig2.csv				
0	1.00	0.88	0.94	22298
1	0.04	0.60	0.08	203
KlarSiktMyeTurb1.csv				
0	1.00	0.52	0.69	22097
1	0.04	0.96	0.07	404
KlarSiktMyeTurb2.csv				
0	0.99	0.34	0.50	21088
1	0.09	0.95	0.16	1413
KlarSiktMyeTurb3.csv				
0	1.00	0.09	0.17	22092
1	0.02	1.00	0.04	411
KlarSiktMyeTurb4.csv				
0	0.98	0.86	0.91	21955
1	0.05	0.31	0.09	547
NattGodSikt1.csv				
0	1.00	0.39	0.56	22289
1	0.02	0.99	0.03	212
NattGodSikt2.csv				
0	1.00	0.40	0.57	22468
1	0.00	0.94	0.00	33
NattOkSikt1.csv				
0	0.99	0.99	0.99	22213
1	0.23	0.36	0.28	279
NattOkSikt2.csv				
0	0.99	0.64	0.78	22081
1	0.04	0.81	0.08	415
OkSiktMiddelsTurb1.csv				
0	0.89	0.84	0.87	17300
1	0.55	0.65	0.60	5192
OkSiktMiddelsTurb2.csv				
0	0.80	0.14	0.24	17623
1	0.22	0.87	0.35	4882
OkSiktMiddelsTurb3.csv				
0	0.98	0.92	0.95	19868
1	0.59	0.87	0.70	2633
OkSiktMiddelsTurb4.csv				
0	0.93	0.75	0.83	20087
1	0.20	0.50	0.28	2414

Tabell 7.2: Testresultater fra kjøring av deteksjonsalgoritme på testsett, sakte modus

her ikke relevant å utføre analyse av hvor godt algoritmen treffer i hver ramme, men heller kun utføre en binær analyse på hvilke rammer som algoritmen mener inneholder fisk.

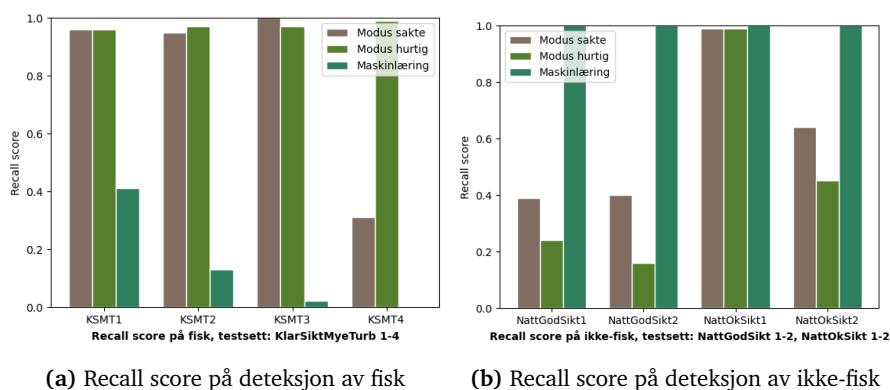
For å få kjørt funksjonen for å analysere resultatfilene ble filene lagt i en egen mappe og funksjonen ble kjørt i et lite script med et funksjonskall til `multifilResultatAnalyseTest` med filstien til mappen som argument. Metaverdiene fra forvirringsmatrisene som ble produsert er vist i tabell 7.3:

7.3 Sammenligning av recall rates

Ved å se på de ulike recall rate-verdiene til algoritmene vil man få et overblikk over hvor stor andel av ikke-fisk/fisk-rammer de klarer å markere korrekt. Ettersom verdiene er normaliserte er det stor variasjon i hvor signifikante disse verdiene vil være, da noen av testsett-filene ikke inneholder så høye antall med rammer med fisk. Det blir i denne seksjonen mest fokus på å se hvor stor andel av fiskerammene algoritmene klarer å detektere korrekt. Hver gang boksploott blir omtalt, refererer dette til figur 7.2 og 7.1.

Klar sikt, mye turbulens

Figur 7.3 viser plott av recall rates på fisk/ikke-fisk i testsett-video under kategorien 'Klar sikt, mye turbulens'. Det er en veldig betydelig forskjell i recall rates på de to datasyn-baserte algoritmene og maskinlæring. Da det er ekstremt mye støy i bildet på disse filene, kan det antas at mange av de positive deteksjonene egentlig er ansamlinger av bobler i rammene. Ved å se på boksploottene for datasyn-algoritmene blir dette bekreftet, da alle middelverdiene for de fire korresponderende plottene ser ut til å ligge helt nede ved 0. Dette bekreftes ytterligere av at recall rates på ikke-fisk på video 1, 2 og til en viss grad 4 stort sett ligger rundt 0.5 eller lavere.



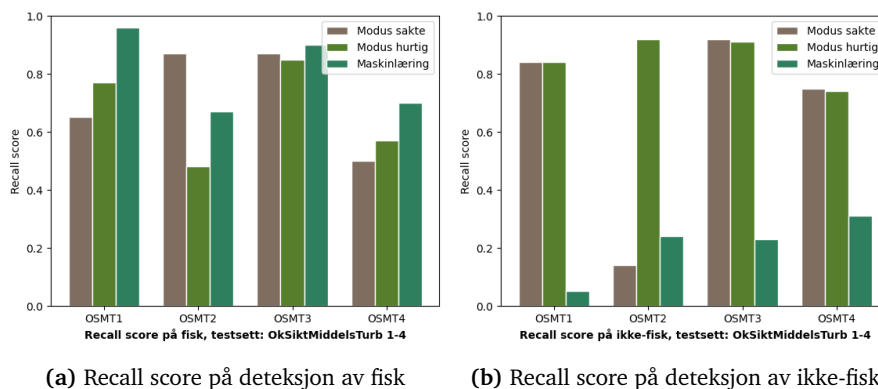
Figur 7.3: Barplot av recall score fra de tre ulike algoritmene kjørt på KlarSikt-MyeTurb 1-4

OK sikt, middels turbulens

Hel fisk p=0.50, halv fisk, p=0.75, testsett				
Fisk / ikke fisk	Presisjon	Recall	F1-score	Support
BrukbarSiktLiteTurb1.csv				
0	0.73	0.20	0.31	16411
1	0.27	0.80	0.41	6080
BrukbarSiktLiteTurb2.csv				
0	0.97	0.76	0.85	21418
1	0.09	0.47	0.15	1086
BrukbarSiktLiteTurb3.csv				
0	1.00	0.05	0.10	21593
1	0.04	1.00	0.08	908
BrukbarSiktLiteTurb4.csv				
0	1.00	0.29	0.44	19302
1	0.19	1.00	0.32	3199
Darlig1.csv				
0	0.99	0.95	0.97	22327
1	0.06	0.39	0.10	174
Darlig2.csv				
0	0.99	1.00	1.00	22334
1	0.00	0.00	0.00	167
Elendig1.csv				
0	1.00	0.87	0.93	22420
1	0.00	0.15	0.01	81
Elendig2.csv				
0	1.00	0.98	0.99	22298
1	0.20	0.51	0.29	203
KlarSiktMyeTurb1.csv				
0	0.99	1.00	0.99	22097
1	0.83	0.41	0.55	404
KlarSiktMyeTurb2.csv				
0	0.95	1.00	0.97	21088
1	0.96	0.13	0.23	1413
KlarSiktMyeTurb3.csv				
0	0.98	1.00	0.99	22092
1	0.08	0.02	0.03	411
KlarSiktMyeTurb4.csv				
0	0.98	1.00	0.99	21955
1	0.02	0.00	0.00	547
NattGodSikt1.csv				
0	0.99	1.00	0.99	22289
1	0.32	0.08	0.13	212
NattGodSikt2.csv				
0	1.00	1.00	1.00	22468
1	0.00	0.00	0.00	33
NattOkSikt1.csv				
0	0.99	1.00	0.99	22213
1	0.25	0.00	0.01	279
NattOkSikt2.csv				
0	0.98	1.00	0.99	22081
1	0.22	0.00	0.01	415
OkSiktMiddelsTurb1.csv				
0	0.81	0.05	0.10	17300
1	0.23	0.96	0.37	5192
OkSiktMiddelsTurb2.csv				
0	0.73	0.24	0.37	17623
1	0.20	0.67	0.31	4882
OkSiktMiddelsTurb3.csv				
0	0.95	0.23	0.37	19868
1	0.13	0.90	0.23	2633
OkSiktMiddelsTurb4.csv				
0	0.90	0.31	0.46	20087
1	0.11	0.70	0.19	2414

Tabell 7.3: Testresultater fra kjøring av maskinlæring på testsett, hel fisk p=0.50, halv fisk p=0.75

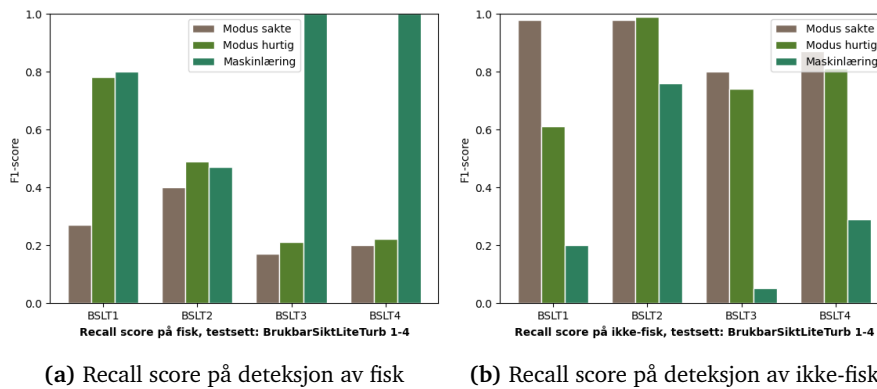
Figur 7.4 viser plott av recall rates på fisk/ikke-fisk i testsett-video under kategorien 'Klar sikt, mye turbulens'. Alle algoritmene ser ut til å detektere gode mengder fisk, spesielt maskinlæringsalgoritmen. På den andre siden har maskinlæringen oppsiktsvekkende lave recall rates på rammer med ikke-fisk; det kan se ut til at maskinlæringsalgoritmen stort sett mener det er fisk i de fleste rammene rundt dette sikt- og turbulensnivået. Den store forskjellen på deteksjoner her kan bekreftes ytterligere av å se på boksplottene for de to datasynalgoritmene, hvor det ser ut til at de stort sett treffer i rammene de mener inneholder fisk. Det er her et unntak for hurtig modus, som treffer veldig dårlig på video nummer 2. Dette gjenspeiles også i den spesielt lave recall raten.



Figur 7.4: Barplot av recall score fra de tre ulike algoritmene kjørt på OkSiktMiddelsTurb 1-4

Brukbar sikt, lite turbulens

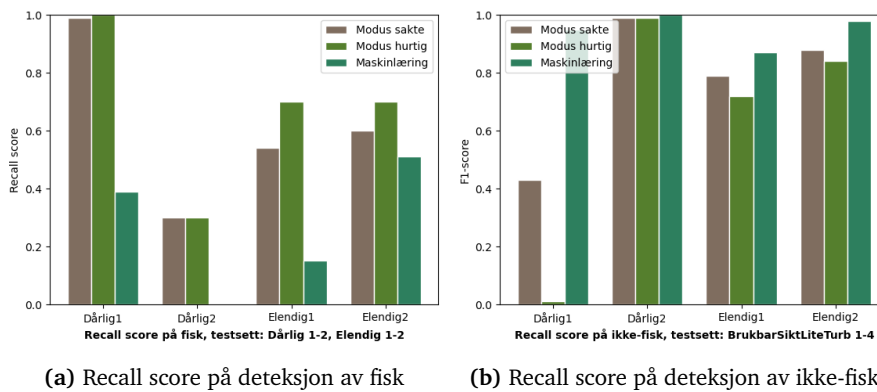
Figur 7.5 viser plott av recall rates på fisk/ikke-fisk i testsett-video under kategorien 'Brukbar sikt, lite turbulens'. I recall rates på fisk ser man igjen en oppsiktsvekkende høy verdi på maskinlæring, da spesielt i video 3 og 4. Dette settes igjen i sammenheng med tilsvarende lave verdier i recall rate på de samme videofilene. Ifølge resultatene for datasynalgoritmene ser det ut til at hurtig modus detekterer litt større andel fisk, som også støttes opp av at boksplottene ser forholdsvis like ut.



Figur 7.5: Barplot av recall score fra de tre ulike algoritmene kj rt p  BrukbarSiktLiteTurb 1-4

D rlig og elendig

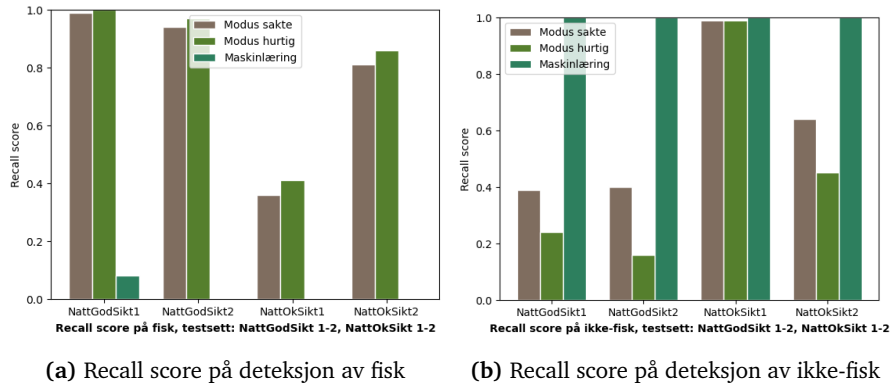
Figur 7.6 viser plott av recall rates p  fisk/ikke-fisk i testsett-video under kategoriene 'D rlig' og 'Elendig'. Datasynsalgortimene ser ut til   jevnt over finne flere fisk, spesielt i video D rlig1. Det b r her p pekes at hurtig-algoritmen her har tiln rmet null recall rate p  ikke-fisk, og naturligvis da f r en utmerket score p  fiskedeteksjon. Boksplottene for D rlig1 har for begge datasynsalgortimene middelverdier som ligger helt nede p  0, som st tter opp at disse tallene ikke er like imponerende som det kan se ut ved f rste  yekast.



Figur 7.6: Barplot av recall score fra de tre ulike algoritmene kj rt p  D rlig 1-2, Elendig 1-2

Natt

Figur 7.7 viser plott av recall rates p  fisk/ikke-fisk i testsett-video under kategoriene 'NattGodSikt' og 'NattOkSikt'. Maskinl ringsalgoritmen ser ikke ut til   v re i stand til   detektere fisk i nattbilder. Datasynsalgortimene ser ut til   stort sett score greit p  nattvideo, spesielt p  video fra kategorien NattOkSikt.



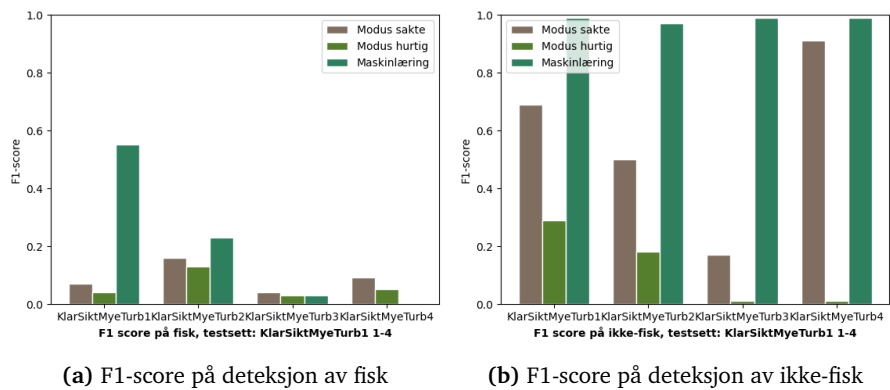
Figur 7.7: Barplot av recall score fra de tre ulike algoritmene kj rt p  NattGodSikt 1-2, NattOkSikt 1-2

7.4 Helhetlig vurdering, F1 score

For   gj re en helhetlig vurdering av deteksjonsalgoritmene vil det her benyttes F1-score. For at dette skal v re et godt m l p  hvor effektivt verkt yet er, forutsetter det at NINA anser det   unng  falske positive som like mye verd som   finne sanne positive. Ettersom det er en direkte matematisk kobling mellom F1-score for fisk og ikke-fisk vil det her fokuseres prim rt p  F1-score for deteksjon av fisk.

Klar sikt, mye turbulens

Figur 7.8 viser plott av F1-scores p  fisk/ikke-fisk i testsett-video under kategorien 'Klar sikt, mye turbulens'. Ingen av algoritmene imponerer spesielt, og antas   v re forholdsvis ubrukelige til denne typen vannforhold.

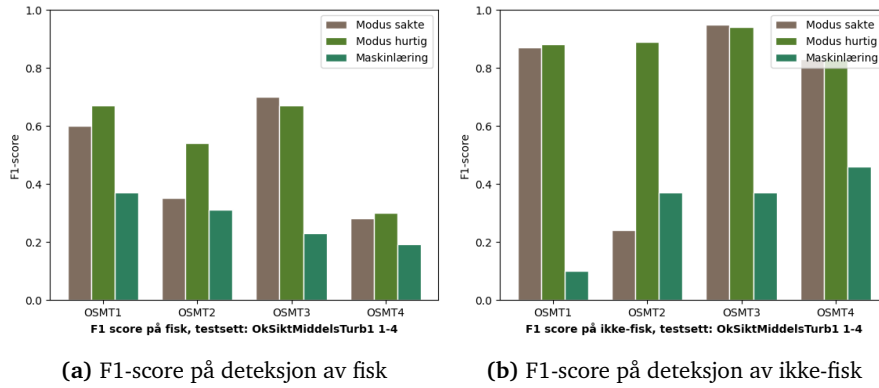


Figur 7.8: Barplot av F1-score fra de tre ulike algoritmene kj rt p  KlarSiktMye-Turb 1-4

OK sikt, middels turbulens

Figur 7.9 viser plott av F1-scores p  fisk/ikke-fisk i testsett-video under kategorien

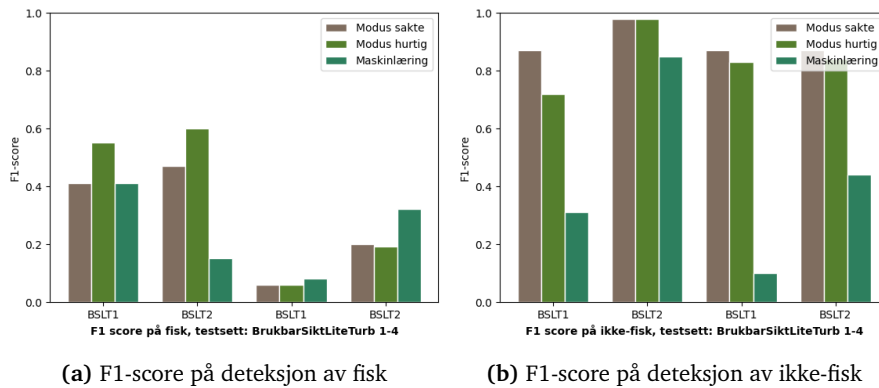
'Ok sikt, middels turbulens'. Det er her en stor forbedring i ytelsen hos alle algoritmene i forhold til 'Klar sikt, mye turbulens', men spesielt hos de datasyn-baserte.



Figur 7.9: Barplot av F1-score fra de tre ulike algoritmene kj rt p  OkSiktMiddelsTurb 1-4

Brukbar sikt, lite turbulens

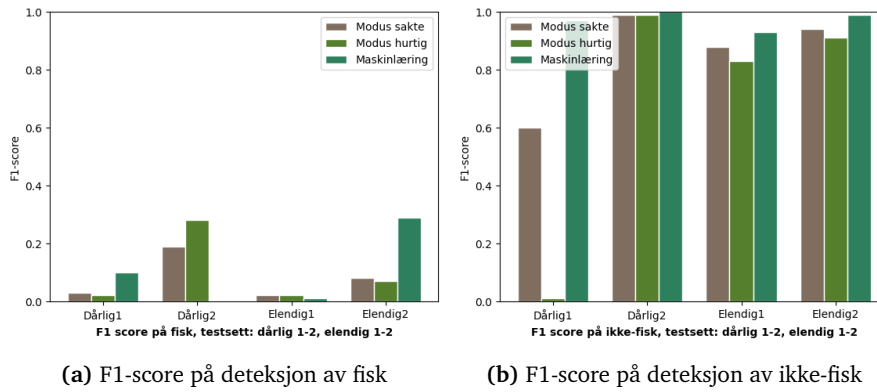
Figur 7.10 viser plott av F1-scores p  fisk/ikke-fisk i testsett-video under kategorien 'Brukbar sikt, lite turbulens'. Resultatene er jevnt over noe d rligere enn 'Ok sikt, middels turbulens', ogs  her ser de datasynsbaserte algoritmene til   v re noe bedre egnet.



Figur 7.10: Barplot av F1-score fra de tre ulike algoritmene kj rt p  BrukbarSikt-LiteTurb 1-4

D rlig og elendig

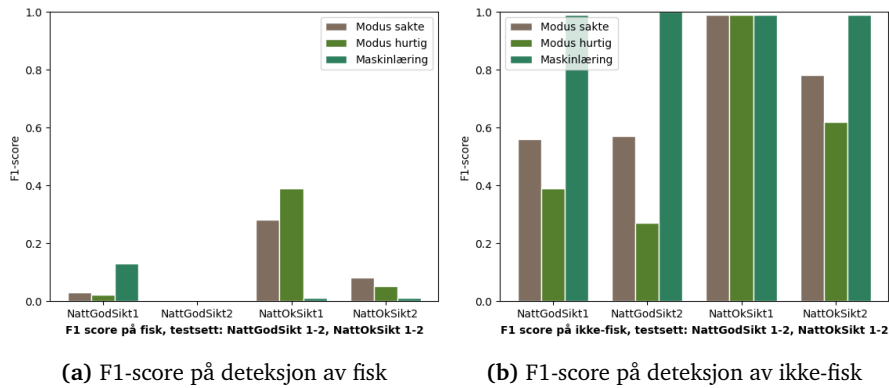
Figur 7.11 viser plott av F1-scores p  fisk/ikke-fisk i testsett-video under kategoriene 'D rlig' og 'Elendig'. Alle tre algoritmene ser ut til   v re noks  ubrukelige ved f rste  yekast. BoksploTTene indikerer derimot at deteksjonsalgoritmen klarer   treffe godt i perioder av videofilene.



Figur 7.11: Barplot av F1-score fra de tre ulike algoritmene kj rt p  D rlig 1-2, Elendig 1-2

Natt

Figur 7.12 viser plott av F1-scores p  fisk/ikke-fisk i testsett-video under kategoriene 'NattGodSikt' og 'NattOkSikt'. Ingen av deteksjonsalgoritmene ser ut til   v re spesielt nyttige for nattvideo, kanskje med unntak av datasynsalgoritmene. Boksplottene indikerer at de i perioder ser ut til   detektere fisk veldig godt, men dette druknes trolig ut av store mengder falske positive i F1-scorene.



Figur 7.12: Barplot av F1-score fra de tre ulike algoritmene kj rt p  NattGodSikt 1-2, NattOkSikt 1-2

GUI

Gruppen har valgt å lage et grafisk brukergrensesnitt (GUI), for å øke brukervennligheten, ved å ha en programvare som er enkel å ta i bruk for alle. Uavhengig av teknologisk bakgrunn, så vil programvaren kunne bli benyttet av flere personer og derigjennom gjøre det enkelt for NINA sine forskere å ta i bruk programvaren. Brukertesting vil benyttes for å kontrollere om prosjektgruppen har vært vellykket med utviklingen av det grafiske brukergrensesnittet og for å gjøre revideringer basert på tilbakemeldinger fra testpersonene.

Om programvaren hadde hatt en målgruppe med personer som har en mer datakyndig bakgrunn, så kunne gruppen klart seg med at programmet kjører i kommandolinje. Ved bruk av ren kommandolinje ville gruppen spart en del tid, som kunne vært brukt for å videreutvikle deteksjonsalgoritmen.

8.1 Utvikling av design

For å kunne utvikle en GUI som vil være enkel å ta i bruk, har gruppen valgt å lage enkle skisser av ønsket prototype, før det ble gjennomført prototyping på ark og digitalt. For å ikke utelukke muligheter, har gruppen skissert flere ulike versjoner for å se hvordan det grafiske brukergrensesnittet kan være utformet.

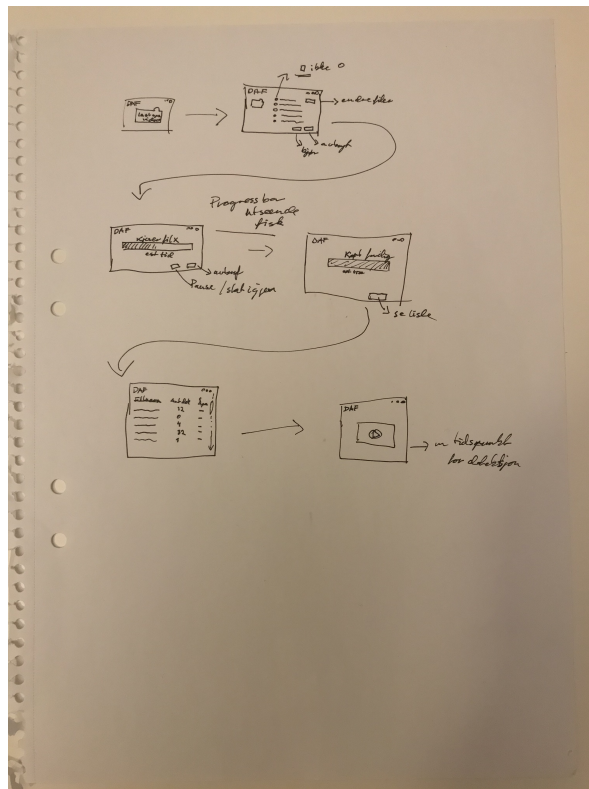
8.1.1 Storyboards

Før gruppen satte i gang med skissering av en low-fidelity (lo-fi) prototype, ble det laget flere utkast på storyboard. Disse representerer flyten i systemet og gir dermed en enkel representasjon av hvordan programmet skal fungere. Storyboardene skulle være enkle, lette å skissere og ta for seg flest mulig ulike fremgangsmetoder. Målet med å lage storyboard er for å kunne identifisere spesifikasjoner, og for å få en oversikt over prosessen. Resultatet fra storyboardene gir utvikleren et bilde av hvordan systemet skal fungerer og gjør det lettere å begynne på en lo-fi prototype.

8.1.2 Low-Fidelity Prototype

En lo-fi prototype, kan være på ark. En lo-fi er nyttig å lage, da det er enkelt å gjøre endringer og modellere hvordan en brukers fremgang i grensesnittet kan se ut. En annen fordel med ark er at dette ikke tar lang tid eller mange ressurser å lage. Gruppen har fokusert på at prototypen skal være minimalistisk, og derfor kun inkludere nødvendig funksjonalitet.

Den endelige versjonen på papir kan ses i figur 8.1.



Figur 8.1: Prototype på ark for brukergrensesnittet.

8.1.3 High-Fidelity Prototype

Etter å ha lagd flere lo-fi prototyper på ark, var neste steg å lage en digital prototype. Dette ble gjort ved å benytte verktøyet *Adobe XD*. Adobe XD benyttes av designere til digital prototyping, og designstudentene på NTNU i Gjøvik benytter dette verktøyet. Gruppen valgte derfor å benytte seg av denne programvaren.

På Adobe XD ble det lett skissert en enkel prototype, i henhold til den siste prototypen på ark. Den første digitale prototypen ble presentert til veileder, og tilbakemeldingene fra veileder medførte revidering. Dette ga bakgrunn for å bruke mer tid på å lage en detaljert og digital prototype, også kalt en high-fidelity (hi-fi) prototype. På denne måten ble den digitale lo-fi prototypen utformet videre til en hi-fi prototype. Ved bruk av en hi-fi prototype er det å lettere kunne visualisere seg hvordan det endelige systemet kan se ut. En hi-fi prototype vil også hjelpe utvikleren av brukergrensesnittet når selve systemet skal utvikles, ved å gi klare retningslinjer for hvordan utseende og funksjonalitet skal representeres i systemet.

Den endelige digitale versjonen av prototypen er vist i figur 8.2.

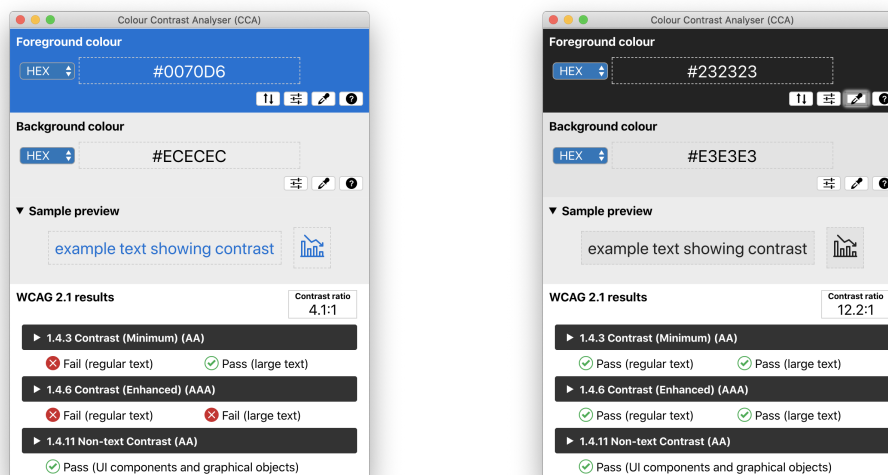


Figur 8.2: Digital prototype i Adobe XD for brukergrensesnittet.

8.1.4 Colour Contrast Analyzer

Det er forsøkt å følge i WCAG 2.1 [8]. WCAG 2.1 er retningslinjer for universell utforming av IKT-løsninger. Her er det blant annet fokus på kontrastforhold. Kontrastforholdet på all tekst skal være på minst 4.5:1 i følge AA standarden. Det er derfor vært fokus på å holde seg til farger som følger dette. Ved å ha fokus på kontrastforhold, vil også brukere som er svaksynte og fargeblinde ha mulighet til å tyde innholdet og få en god brukeropplevelse.

Det er brukt et verktøy for å teste kontrastnivået på applikasjonen. Målingene fra verktøyet, Colour Contrast Analyzer [13], forteller brukeren om kontrastnivået mellom for- og bakgrunn. Ut ifra dette har gruppen målt de ulike delene av brukergrensesnittet og kommet fram til at det dårligste kontrastnivået oppnås når det ene UI-elementet og en av tekstene i brukergrensesnittet måles mot sine bakgrunner. Alle deler av applikasjonen tilfredsstiller kravet til WCAG 2.1. Det eneste som kommer nært å bryte kravet er mappen for innlasting av filer, og selv denne opprettholder et kontrastnivå på 4.1:1, som Figur 8.3a viser at er godkjent kontrastnivå for brukergrensesnitt-komponenter og grafiske objekter.



(a) Kontrast mellom UI-objekt og bakgrunn.

(b) Kontrast mellom tekst og bakgrunn.

Figur 8.3: Ulike kontrastnivåer i det grafiske brukergrensesnittet.

8.2 Utvikling av grensesnitt

Etter at gruppen har gjennomført prosessen med å designe det ønskede systemet så må løsningen utvikles. Gruppen har begrenset med erfaring med å utvikle grafiske brukergrensesnitt og har derfor måtte benytte seg av dokumentasjon tilgjengelig på internett for å tilegne seg kunnskapen som det er behov for.

PyQt [57] er et sett med Python-bindinger for applikasjonsrammeverket; Qt [58]. Det naturlige valget ble å skrive brukergrensesnittet i Python slik som deteksjon, og dermed falt automatisk valget på PyQt. Brukergrensesnittet er derfor skrevet i PyQt5 som er siste versjonen av applikasjonsrammeverket til Qt. Qt tilbyr mange ulike muligheter for å designe og utvikle brukergrensesnitt, men gruppen har valgt å ikke benytte seg av eksterne applikasjoner som tilbys for å sørge for at læringsutbyttet blir så stort som mulig. Blant verktøyene gruppen kunne valgt å ta i bruk er *Qt Designer* som gjør det mulig å dra ønskede elementer direkte inn i en ramme og dermed lage brukergrensesnittet ut ifra ferdiglagde objekter. Disse objektene måtte derfor gruppen tilegne seg kunnskap om hvordan fungerer og samhandler, før kode for å implementere objektene ble skrevet.

8.2.1 Kjøring gjennom IDE (PyCharm)

Utviklingen av grensesnittet har blitt gjort i PyCharm [19] slik som utviklingen av deteksjonsalgoritmen. Dette er for å holde alt på ett sted og fordi det er i dette programmet at brukergrensesnitt har blitt utviklet i tidligere prosjekter. Strømmen på utvikling har bestått av tilegning av kunnskap om de ulike komponentene som prototypen skal bestå av, testing av hvordan komponentene kan implementeres enkeltvis før de alle implementeres sammen og samkjøres med andre deler av systemet.

8.2.2 Kjørbar fil

Etter at utviklingen av selve grensesnittet har blitt gjort i *PyCharm*. Så har gruppen gjort programmet om til en kjørbare fil ved hjelp av *PyInstaller*. *PyInstaller* er en funksjon som er blitt kjørt i terminalen til en av gruppemedlemmene hvor programmet henter alle importerte filer og funksjoner fra biblioteker og laster ned disse, før programmet blir samlet i en kjørbare fil.

Grunnen til at gruppen har valgt å lage en kjørbare fil er fordi gruppen ser det som utfordrende for personer med manglende erfaring med IDE'er å selv kjøre programmet i en IDE som *PyCharm*. En kjørbare fil vil kun kreve at brukeren starter filen og følger applikasjonens instruksjoner.

8.2.3 Installasjonsfil

Gruppen har i diskusjon med NINA konkludert med at det er ønskelig at programvaren skal være lett tilgjengelig og ikke kreve at de må installere annen programvare for å få programmet til å kjøre.

Det har derfor vært viktig å ha en installasjonsfil som forenkler prosessen med å ta i bruk programvaren for NINA. For å kunne lage en installasjonsfil så må output fra PyInstaller og andre tilhørende filer samles i en egen mappe. Gruppen har valgt å ta i bruk et verktøy som heter Nullsoft Scriptable Install System (NSIS) for å lage installasjonsfilen. Verktøyet gjør det enkelt å bare ta en *zip*pet mappe med de ønskede filene og gjør denne om til én installasjonsfil.

Installasjonsfilen installerer programvaren som en mappe ved ønsket destinasjon på PCen til brukeren. Det gjør det enkelt for NINA å selv velge hvor de ønsker å ha programvaren. Ved å ha en installasjonsfil så er det betraktelig enklere å distribuere programmet.

8.2.4 Valg underveis

Under utvikling så opplevde gruppen problemer med at implementering av en videospiller ble mer utfordrende enn forventet, da implementering var basert på en eldre versjon av VLC og skapte feilmeldinger under kjøring. Det ble dermed besluttet at det ikke var behov for en videospiller for å løse oppgaven.

Gruppen ønsket at alle knapper skulle ha en klar funksjonalitet og at det skulle være så få knapper som mulig for å unngå forvirring for brukerne. I tillegg ble det bestemt at programmet skulle kunne kjøre i 2 ulike moduser eller gjøre det mulig for brukeren å selv bestemme parametere for deteksjonen. Modusene skulle være en hurtig versjon og en ordinær, eller sakte versjon.

Da programmet risikerer å måtte ta i mot mange filer så var det ønskelig å implementere en tabell med alle filnavnene for å gjøre det enklere å kunne holde kontroll over alle de valgte filene. I tillegg ble det lagt til mer informasjon om filene i tabellen, informasjon om lengde på videofilen og størrelse målt i MB (mega-byte).

8.3 Funksjonaliteter i GUI

Sluttproduktet til prosjektet kan utføre flere ulike oppgaver og har derfor funksjonalitet delt opp i ulike funksjoner. Primært vil en bruker gjennomføre deteksjon av fisk i videomateriale som brukeren har innhentet selv. Brukeren kan ta i bruk datasettene som gruppen har annotert og laget fasitfiler. Datasettene består av videomateriale som NINA har samlet inn. De inkluderte datasettene gjør det mulig

å gjennomføre testing av løsningen og om ønskelig kan brukeren gjøre endringer på parameterne til deteksjonsalgoritmen for å så kontrollere resultatet gjennom testing på de inkluderte datasettene. Funksjonalitetene under er også beskrevet av gruppen i brukerguiden, Appendix G, som er inkludert i det grafiske brukergrensesnittet.

8.3.1 Deteksjon

Deteksjonsalgoritmen utfører deteksjon på videomateriale som brukeren laster inn, men kan også utføre deteksjon på vedlagte datasett. For å kunne gjennomføre deteksjon i varierende forhold, så har gruppen implementert to ulike moduser som skal imøtekomme problematikken de varierende forholdene skaper. Brukeren kan ved bruk av radioknapper velge mellom de ulike deteksjonsmodusene før løsningen kjører deteksjon på videomaterialet. Figur 8.4d viser de to modusene brukeren kan velge mellom.

Modus: Hurtig

Modusen “Hurtig” er en deteksjonsmodus som skal utføre deteksjon hurtigere enn modusen “Sakte”. Modusen har parametere som utfører mindre preprosessering på videomaterialet som lastes inn. Reduksjonen i mengde preprosessering fører til at løsningen bruker kortere tid på å bli ferdig med videomaterialet. Modusen vil fjerne mindre støy og kan i tilfeller hvor det støy er fraværende, fungere veldig bra, men i videomateriale med mye støy så kan deteksjonen bli betraktelig svakere.

Modus: Sakte

Modusen “Sakte” er en deteksjonsmodus som skal utføre deteksjon saktere enn modusen “Hurtig”. Modusen har parametere som utfører mer preprosessering på videomaterialet som lastes inn. Økningen i mengde preprosessering fører til at løsningen bruker lengre tid på å bli ferdig med videomaterialet. Modusen vil fjerne mer støy og vil gjennomføre mer presise deteksjoner i tilfellene hvor videomaterialet har en del støy. Denne modusen kan derfor betraktes som mer presis, men på bekostning av en økning i tiden som løsningen trenger for å bli ferdig med videomaterialet.

8.3.2 Testing

Gruppen har utført annotasjon av deler av videomaterialet som gruppen ble tildelt av NINA. Arbeidet med annotasjon av videomaterialet var tidkrevende, men til gjengjeld gjorde det mulig for gruppen å utføre testing av deteksjonsalgoritmen. Gruppen plukket ut enkelte videoer og på bakgrunn av lengde på videoene, så ble det opprettet to datasett som gruppen annoterte. Figur 8.4d viser de to datasettene brukeren kan velge mellom.

Utviklingssett

Utviklingssettet består av 8 videofiler, hvor av hver video er på 25 til 35 sekunder. Utviklingssettet skulle gjøre det enkelt for gruppen å teste deteksjonsalgoritmen på ulike forhold hvor gruppen visste at én eller flere fisk kom til å passere kameraet. Utviklingssettet har gjort det mulig for å gruppen å eksperimentere med mange ulike parametere over en kort tidsperiode før testingen ble utprøvd på lengre videofiler.

Testsett

Testsettet består av 20 videofiler, hvor av hver video er på 30 minutter. Testsettet skulle gi gruppen en mer virkelighetsnær testing for deteksjonsalgoritmen. Videomaterialet utdelt av NINA var oppdelt i 30 minutter lange videoer med enkelte unntak. Ved å teste deteksjonsalgoritmen på like lange videoer så kan gruppen samle inn informasjon om deteksjonsalgoritmen presterer godt nok i mer virkelighetsnære tilfeller og om kravet om sanntid blir opprettholdt.

8.3.3 Utviklingsmodus

Løsningen er utviklet med fokus på at resultatfiler fra deteksjonen skal produseres. Resultatfilene forteller brukeren om hvor i programmet en deteksjon er gjort, men viser ikke brukeren hvordan deteksjonen ble gjort. Ved å aktivere utviklingsmodus så kan brukeren observere hvordan de ulike preprosesseringsteknikken gjør endringer på videomaterialet som lastes inn. Om utviklingsmodus er **aktivert** så vil løsningen åpne flere vinduer hvor preprosesseringsteknikken som vises er beskrevet i vinduets tittel. Ved å aktivere denne funksjonen så kan brukeren oppleve en reduksjon i hastighet på deteksjonen, dette skyldes at modusen krever mer av datamaskinene og en del av datamaskinens ressurser vil benyttes for å vise brukeren preprosesseringsteknikkene. Utviklingsmodus kan være interessant for brukeren å aktivere om brukeren ønsker å se hva som foregår, men kan være til hjelp for brukere som ønsker å se hvordan justeringer av løsningens parametere vil påvirke deteksjonen.

8.4 Brukertesting

For å sikre at det grafiske brukergrensesnittet som gruppen utviklet skulle bli så enkelt som mulig å ta i bruk, så valgte gruppen å gjennomføre brukertesting. Gruppen konkluderte tidlig i prosjektet at brukertesting skulle gjennomføres for å sikre at applikasjonen ikke bare bærer preg av hva gruppens medlemmer ser for seg at skal fungere. Det viktigste for gruppen var at applikasjonen skulle være enkel å ta i bruk for NINAs forskere. Brukertesting brukes for å sikre at nye løsninger fungerer, og er derfor et behov for at gruppen skal kunne beslutte om det grafiske brukergrensesnittet ble vellykket eller ikke [59]. I Appendix F ligger fullstendig referat fra brukertesting.

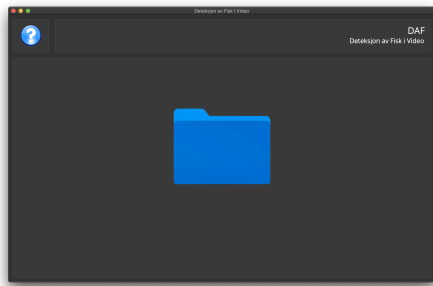
Brukertesting ble gjennomført i 2 omganger, først ble eksterne personer som ikke er tilknyttet NINA testet. Etter at tilbakemeldinger fra den første gruppen var tatt i betraktning av gruppen, så ble en revidering av det grafiske brukergrensesnittet gjort på bakgrunn av tilbakemeldingene. I etterkant av revideringen så ble den andre gruppen som bestod av forskere ved NINA brukertestet.

Testperson 1 og 2

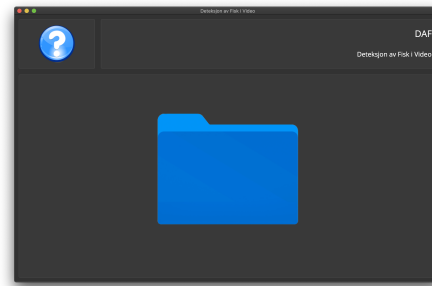
Testperson 1 og 2 tilhører den første gruppen med personer som ble brukertestet. Testpersonene i denne gruppen tok i bruk *versjon 1.5* av applikasjonen. Testpersonene i denne gruppen var primært fornøyd med utseendet i GUIen, men hadde et par forbedringspunkter. Begge oppfattet mappen for innlasting av filer som intuitiv, men som de resterende ikonene i brukergrensesnittet, så var mappen på grensen til å være for liten. Dette gjaldt spesielt spørsmålsteget som er tilknyttet brukerguiden. Knappene ble oppfattet som litt små og i referatet fra brukertesting, Appendix F, så ble det ytret et ønske om “farge på knappen som skal være primærhandling”. Til tross for en del ønsker om forbedringer og enkelte utfordringer vedrørende navigasjon til brukerguiden, så oppfattet de 2 første testpersonene at GUIen var enkel å ta i bruk og intuitiv når større deler av applikasjonen hadde blitt utforsket.

8.4.1 Forbedringer av GUI

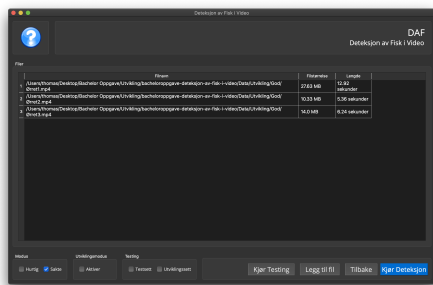
Etter at testperson 1 og 2 hadde gjennomført brukertesting på *versjon 1.5*, så tok gruppen grep og implementerte en del av ønskene til testpersonene. Gruppen konkluderte med at de heller kunne fjerne enkelte funksjoner om de nye funksjonene skulle få negative tilbakemeldinger hos de andre testpersonene, men at ved ingen negative tilbakemeldinger, så kunne endringene forbli. Gruppen tok først tak i problemet om ulike skjermforhold og gjorde applikasjonen mer dynamisk for å støtte ulike skjermforhold. Videre sa gruppen seg meget enig i testperson 1 sin mening om å benytte radioknapper for modus, slik at kun én modus er mulig å velge om gangen. Standard valg for modus ble også fjernet, da gruppen oppfattet det som naturlig at brukeren selv skal ta valget uten påvirkning fra applikasjonen. Da både testperson 1 og 2 ønsket å endre utseende til knappene, så ble tekstene til knappene gjort tydeligere og «Kjør deteksjon»-knappen, som er å regne som primærhandling, fikk en ny farge for å lettere kunne differensieres fra de andre knappene. Etersom begge brukerne opplevde problemer med å oppdage brukerguiden, så ble ikonet betraktelig forstørret for å se om de fremtidige brukerne ville opplevde det enklere å navigere frem til brukerguiden. Meldingsboksene ble også forstørret for å gjøre det enklere å lese hva applikasjonen informerer brukeren om. Endringene ble så implementert og en ny versjon, *versjon 1.6*, av applikasjonen ble gjort klar for videre brukertesting. Figur 8.4 viser endringene som ble gjort mellom *versjon 1.5* og *versjon 1.6* for å imøtekomme tilbakemeldingene fra testperson 1 og 2.



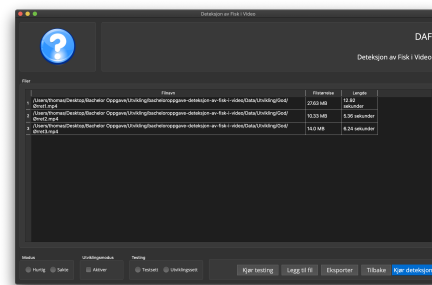
(a) Vindu 1 før forbedringer.



(b) Vindu 1 etter forbedringer.



(c) Vindu 2 før forbedringer.



(d) Vindu 2 etter forbedringer.

Figur 8.4: Skjerm bilde av Vindu 1 og 2 før og etter forbedringer fra brukertestene.

Testperson 3 og 4

Testperson 3 og 4 tilhører den andre gruppen med personer som ble brukertestet. Testpersonene i denne gruppen tok i bruk *versjon 1.6* av applikasjonen. Testpersonene i denne gruppen navigerte seg forholdsvis enkelt gjennom applikasjonen, uten særlige forsinkelser. Testperson 3 benyttet litt lenger tid på å velge mappen ved oppstart og kom frem til at en beskrivende tekst i tillegg til ikonet ville være til stor hjelp. Bruken av farge for å indikere primærhandling ble trukket fram som et moment i applikasjonen som gjorde det enkelt å differensiere knappen fra de andre knappene. Begge brukerne trakk frem et ønske om at status for deteksjonen skulle være synlig i samme vindu som applikasjonen, “slik at status til programmet ville vært tydelig til enhver tid” Appendix F.

8.4.2 Konklusjon

Gruppen opplevde store forskjeller i tid som brukerne brukte for å navigere frem til brukerguiden. Etter endringen som ble gjort i *versjon 1.6* med å forstørre ikonet til brukerguiden, så brukte testperson 3 og 4 i gjennomsnitt 6 ganger så kort tid på å finne brukerguiden. I tillegg ble generell tid for gjennomføring av oppgaver betraktelig redusert etter at skjermforhold i applikasjonen hadde blitt forbedret og tydeliggjøring av knapper. Gruppen opplevde at en fremheving av primærhandling bidro betraktelig til å senke den totale tiden som testpersonen brukte på å gjennomføre oppgavene.

Diskusjon

9.1 Oppfyllelse av resultatmål

Det ble i forprosjektet laget en rekke resultatmål som her vil bli vurdert om er blitt oppfylt.

1. **Gir output i form av en .CSV-fil som kan importeres direkte inn i Microsoft Excel**

Resultatfilene som produseres kan fint importeres inn i Microsoft Excel. Det resultatmålet derimot ikke sier noe om er hvor ferdigformatert filen skal være for å være så nyttig som mulig for NINA. Selv om resultatmålet teknisk sett er oppnådd, tar gruppen selvkritikk på hvordan dette er løst i verktøyet som leveres inn.

2. **Ved gjennomkjøring av en tilfeldig videofil fra utlevert datamateriale fra 2017 i gjennomsnitt produserer tidsstempel som peker til minst 95% av fiskene registrert av manuell gjennomgang.**

Dette resultatmålet kan tolkes til å være det samme som å ha en gjennomsnittlig recall rate på minst 0.95, så lenge man sier at 'peker til minst 95% av fiskene' betyr 'peker til minst 95% av rammer som inneholder fisk'. Selv om man i resultatene kan finne enkeltfiler hvor deteksjonsalgoritmen har scoret over 0.95, er dette langt fra gjennomsnittet. Basert på dette vil gruppen ikke betegne dette resultatmålet som oppfylt.

3. **Skriver ut tidsstempel som i 95% av tilfellene viser til et punkt i videomaterialet som har minst én fisk i et tidsvindu på+- 5 (fem) sekunder.**

Ettersom det ikke har blitt implementert noe funksjonalitet som interpolerer resultater, kan det her ikke gjøres en direkte vurdering på om resultatmålet er nådd. Det virker likevel rimelig å anta at dette resultatmålet kan tilnærmes å være likt en målt presisjon på gjennomsnittlig 0.95. Selv om det, som med recall rate, finnes noen spesifikke tilfeller hvor dette er oppfylt, er den gjennomsnittlige verdien ikke i nærheten av dette, og gruppen må si at resultatmålet ikke er blitt oppfylt.

4. **Ved kjøring maksimalt bruker 1:1 total prosesseringstid per tid video ('sann-tid' prosessering)**

Diverse kjøring av deteksjonsalgoritmer har foregått på mange ulike data-

maskiner. Ingen av disse har noen gang stått i fare for å ikke oppfylle kravet med prosessering i sanntid, og anses derfor som oppnådd.

5. **Har et brukergrensesnitt som tilfredsstillt kravet om AA-sertifisering i henhold til kravet om universell utforming - WCAG 2.1**

Dette resultatmålet er bevist oppfylt i kapittel om GUI, se figur ??.

6. **Kun bruker videoformat (både som input og eventuell output) som støttes av VLC Media Player**

Det har kun blitt benyttet .mp4 og eventuelt .m4v i løpet av prosjektet, som begge er støttede videoformater av VLC.

7. **Kan kjøres på dagens mest brukte operativsystemer**

- Windows 7 og nyere
- MacOS Catalina og nyere

Det har gjennom hele prosjektet blitt utviklet parallelt på Mac og Windows, med medfølgende testing. Gruppen vil anse dette resultatmålet som absolutt oppnådd.

9.2 Lisens

For å kunne utgi et program eller en programvare så må brukte biblioteker tas i betraktning før utgivelse. En programvare vil som regel ha en lisens knyttet til programvaren som sier noe om hvordan programvaren kan brukes, endres og distribueres. Programvarene som gruppen har benyttet har åpen kildekode, men ulike retningslinjer for lisensiering. Det er hovedsakelig to typer lisenser som gruppen har benyttet; den snille varianten, MIT og BSD, og den litt mer strikte varianten, GNU GPL. MIT/BSD tillater at endringer av kildekoden og distribusjon ettersom hva den enkelte ønsker, men setter krav for at den må opplyses hvem den opprinnelige opprettshaveren er. Siden programvare gitt ut under BSD-lisens ikke krever redistribusjon under den samme lisensen, så er det derfor ikke krav om at den nye programvaren må være fritt tilgjengelig. I motsetning så krever GNU GPL at programvare ved redistribusjon skal være fritt tilgjengelig slik som den opprinnelige programvaren [60]. Dette er for å sikre brukernes frihet ved å sørge for at forbedringer og redistribusjoner forblir åpent tilgjengelig for alle [61].

Programvaren gruppen har utviklet benytter funksjoner fra flere biblioteker. OpenCV og PyQt5 er bibliotekene som er benyttet mest, men ved avgjørelse av lisens som programvaren kan gis ut på, så må alle biblioteker tas i betraktning og den "strengeste" lisensen må opprettholdes. PyQt5 er gitt ut under GNU GPL lisensen [62] og krever derfor at ved distribusjon av gruppens programvare, så må programvaren være fritt tilgjengelig. Gruppen ser på det som positivt med en videreføring av GNU GPL-lisensen som krever at nye versjoner vil bli utgitt under de samme kriteriene, som sikrer at forbedringer av programvaren forblir fritt tilgjengelig. Programvaren vil derfor følge **GNU General Public License versjon 3**.

9.3 Forslag til videre arbeid

Gruppen ble i møtet med NINA i slutten av januar enige om en del funksjonaliteter som NINA kunne ønske seg implementert. Etersom gruppen ble redusert til to medlemmer i begynnelsen av februar så måtte gruppen dessverre redusere mengden funksjonalitet løsningen skulle inneha.

Gruppen har utarbeidet en løsning som i utgangspunktet ikke burde være begrenset til å jobbe med videomateriale av fisk eller under vann. Det kunne derfor vært interessant å teste løsningen på andre typer video med statisk bakgrunn, hvor det er ønskelig å detektere objekter i bevegelse. Dette kan være detektering av personer, overvåkning av biler som passerer på en vei eller deteksjon av andre dyr. Løsningen krever kun små justeringer på hvordan deteksjonsalgoritmen skal arbeide ut ifra de andre omgivelsene, men burde ha grunnlaget for å fungerer i andre applikasjoner.

NINA ytret et stort ønske om å gjennomføre artsbestemmelse og individgjenkjenning, men også informasjon om retningen objektet som passere i videomaterialet har. Gruppen hadde på bakgrunn av redueringen av antall medlemmer ikke muligheten til å arbeide på artsbestemmelse, individgjenkjenning og retningsbestemmelse, men ser verdien av informasjonen som NINA kan innhente ved implementering av funksjonalitetene.

Gruppen har følgende forslag til videre arbeid:

- Artsbestemmelse av detektert fisk
- Individgjenkjenning av detektert fisk
- Spesifisering på hvilken retning en fisk svømmer igjennom spalten (opp eller ned)
- Generell forbedring av deteksjonsalgoritmen. Dette burde være spesielt lett å komme i gang med, da det er utviklet utviklings- og testsett med medfølgende automatisert testing og statistikk fra disse.
- Implementasjon av flere moduser, gjerne skreddersydd for spesifikke sikt- og turbulensforhold
 - En forlengelse av dette kunne vært en adaptiv algoritme, som forsøker å automatisk velge beste modus
- Å gi brukeren mulighet til å endre parametere selv som del av det grafiske brukergrensesnittet
- En statusbar i det grafiske brukergrensesnittet, som viser progresjonen i en deteksjonskjøring. Denne kan også inneholde et tidsestimat på hvor lang tid som er igjen.
- Finne et bedre mål enn F1-score på helhetlig resultat til bruk i testprosedyre. En F-score som er vektet i retning recall rate, for å sørge for at verktøyet ikke skal gå glipp av fisk.

- Produksjon av skikkelige filer ved kjøring av ordinær deteksjon. Dette bør inneholde en interpolasjon av detekterte rammer, og utskrift av ranges av rammer i stedet for enkeltrammer.
- Forbedring av forhold på steder hvor det gjøres opptak som skal analyseres av verktøyet. Dette kan for eksempel være farging av bakgrunn for å gi en sterkere kontrast mellom fisk og betongvegg.

9.4 Kritikk av oppgaven

Prosjektgruppen erfarte at oppgaven generelt var vanskelig å vurdere hvor “løsbar” den var, uavhengig av valg av teknikk (maskinlæring eller datasyn). Mangelen på klare forutsetninger til hva prosjektgruppen skulle oppnå førte til at prosjektgruppen satte i overkant ambisiøse mål. Prosjektgruppen sitter etter gjennomføring av prosjektet igjen med følelsen av at målene som ble satt ikke er egnet for en gruppe på bare to personer.

Opgaven ble begrenset i den forstand at datamaterialet allerede var hentet inn, og dermed ikke muliggjorde at prosjektgruppen kunne påvirke hvordan det hentes inn.

Prosjektet har vært en god læringsprosess for prosjektgruppens medlemmer og gitt erfaring om vurdering av arbeidsmengde, men sluttproduktet kunne potensielt vært av bedre kvalitet om klare forutsetninger for oppgaven hadde blitt presentert for prosjektgruppen.

9.5 Evaluering av gruppens arbeid

Prosjektgruppen hadde ikke fått klare forutsetninger om sluttproduktet og erfarte derfor at målene prosjektgruppen satte ble for ambisiøse. Til tross for at prosjektgruppen ble redusert til to medlemmer i begynnelsen av februar så valgte gruppen å avgrense oppgaven så lite som mulig, og satt derfor igjen med at både deteksjonsalgoritme, datasett og brukergrensesnitt skulle utvikles. Dette medførte at gruppen ikke fikk tilstrekkelig med tid til å arbeide på selve deteksjonsalgoritmen, som i utgangspunktet egentlig var essensen i oppgaven.

Gruppen sitter igjen med følelsen av at sluttproduktet kun er et slags “skall” for verktøyet. Alle elementer som har dukket opp rundt verktøyet er blitt utviklet eller tilnærmet utviklet, med unntak av selve kjernen; deteksjonsalgoritmen.

Det grafiske brukergrensesnittet kunne vært sløyfet med fordel for deteksjonsalgoritmen og det resterende arbeidet. Et annet alternativ kunne vært å kutte ned på omfanget til testsettet.

Prosjektgruppen synes i ettertid at de samarbeidet godt. Dette gjelder både kommunikasjon, oppfølging og assistering av hverandre, og delegering av arbeidsoppgaver for å samsvare med interesseområder.

Konklusjon

Gruppen har i rapporten beskrevet prosessen med å selekere og utvikle et test- og utviklingssett, en automatisert testprosedyre, en datasynsbasert deteksjonsalgoritme med to moduser, og et grafisk brukergrensesnitt som syr dette sammen med ulike funksjonaliteter.

Deteksjonsalgoritmene har blitt vist å være sterkere enn en maskinlæringsbasert algoritme i mange sikt- og turbulensforhold. Totalt sett har gruppen ikke lyktes i å utvikle en deteksjonsalgoritme som yter godt nok til at den pålitelig kan tas i bruk av oppdragsgiver umiddelbart.

Et solid testsystem og forslag til en rekke forbedringer og videre arbeid bør gjøre dette til et attraktivt verktøy å jobbe videre med.

Bibliografi

- [1] *BIDAT39 - Bacheloroppgave - Dataingeniør*, <https://www.ntnu.no/studier/emner/BIDAT39>, besøkt 24.01.2020.
- [2] *Norsk institutt for naturforskning*, <https://www.nina.no/>, besøkt 24.01.2020.
- [3] *Laksetrapp*, <https://snl.no/laksetrapp>, besøkt 30.01.2020.
- [4] *SafePass*, <https://www.cedren.no/Prosjekter/SafePass>, besøkt 30.01.2020.
- [5] *Smart ombygging av fisketrapp skal gjøre det lettere for å fisken å vandre i Glomma*, <https://www.cedren.no/Nyheter/Article/ArticleId/4128/Smart-ombygging-av-fisketrapp-skal-gjore-det-lettere-for-a-fisken-a-vandre-i-Glomma>, besøkt 30.01.2020.
- [6] *Fisketelling_Hoyegga_2017.xlsx*, Dokument utlevert av NINA under møte 29.01.2020, inneholder informasjon fra manuell telling av alt videomateriale fra 2017.
- [7] *Spesifikasjoner på laptop hos NINA*, Informasjon overlevert muntlig under møte 29.01.2020.
- [8] *WCAG 2.0-standarden*, <https://uu.difi.no/krav-og-regelverk/wcag-20-standarden>, besøkt 27.01.2020.
- [9] *VLG: Official Site*, <https://www.videolan.org/>, besøkt 17.01.2020.
- [10] *Hvordan teste universell utforming av ditt nettsted*, <https://uu.difi.no/krav-og-regelverk/kom-i-gang/hvordan-teste-universell-utforming-av-ditt-nettsted>, besøkt 31.01.2020.
- [11] *IMT2243 - Systemutvikling*, <https://www.ntnu.no/studier/emner/IMT2243/2016#tab=omEmnet>, besøkt 27.01.2020.
- [12] *Trello*, <https://trello.com/>, besøkt 17.01.2020.
- [13] *Colour Contrast Analyser*, <https://developer.paciellogroup.com/resources/contrastanalyser/>, besøkt 30.01.2020.
- [14] *Maler og standarder - NTNU*, <http://aitel.hist.no/fag/maler-standarder/fossefall/forstudiet.html>, besøkt 20.01.2020.
- [15] *Overleaf, online LaTeX editor*, <https://www.overleaf.com/>, besøkt 17.01.2020.
- [16] *Git*, <https://git-scm.com/>, besøkt 31.01.2020.

- [17] *BitBucket*, <https://bitbucket.org/>, besøkt 17.01.2020.
- [18] *JetBrains: Developer Tools for Professionals and Teams*, <https://www.jetbrains.com/>, besøkt 18.02.2020.
- [19] *PyCharm - The Python IDE for Professional Developers*, <https://www.jetbrains.com/pycharm/>, besøkt 18.02.2020.
- [20] S. R. Yoshida, *Computer Vision*. Hauppauge, N.Y.: Nova Science Publishers, Inc., 2011.
- [21] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost og N. Wiebe, *Quantum machine learning*. London: Nature Publishing Group, 2017.
- [22] K. Srinath, «Python—The Fastest Growing Programming Language», *International Research Journal of Engineering and Technology (IRJET)*, årg. 4, nr. 12, s. 354–357, 2017.
- [23] *Marshall Electronics CV-502*, <http://www.marshall-usa.com/discontinued/cameras/CV502-WP.php>, besøkt 04.05.2020.
- [24] *How Does a Digital Camera Work?*, <https://www.gadgetreview.com/how-does-a-digital-camera-work>, besøkt 29.04.2020.
- [25] *Digital Images*, <https://www.encyclopedia.com/computing/news-wires-white-papers-and-books/digital-images>, besøkt 29.04.2020.
- [26] *Moving Picture Experts Group*, https://en.wikipedia.org/wiki/Moving_Picture_Experts_Group, besøkt 29.04.2020.
- [27] *What Is an MP4 File (and How Do I Open One)?*, <https://www.howtogeek.com/365037/what-is-an-mp4-file-and-how-do-i-open-one/>, besøkt 29.04.2020.
- [28] *Advanced Video Coding*, https://en.wikipedia.org/wiki/Advanced_Video_Coding, besøkt 03.05.2020.
- [29] Saran K B og Sreelekha G, «Traffic video surveillance: Vehicle detection and classification», i *2015 International Conference on Control Communication Computing India (ICCC)*, 2015, s. 516–521.
- [30] *Geometric Image Transformations*, https://docs.opencv.org/2.4/modules/imgproc/doc/geometric_transformations.html, besøkt 02.05.2020.
- [31] J. Yuan og G. He, «Application of an Anisotropic Diffusion Based Preprocessing Filtering Algorithm for High Resolution Remote Sensing Image Segmentation», i *2008 Congress on Image and Signal Processing*, bd. 3, 2008, s. 629–633.
- [32] *Image Filtering*, <https://docs.opencv.org/2.4/modules/imgproc/doc/filtering.html>, besøkt 04.05.2020.
- [33] *Fourier Transform*, https://docs.opencv.org/master/de/dbc/tutorial_py_fourier_transform.html, besøkt 04.05.2020.

- [34] R. Rai, P. Gour og B. Singh, «Underwater Image Segmentation using CLAHE Enhancement and Thresholding», *International Journal of Emerging Technology and Advanced Engineering*, årg. 2, s. 118–123, jan. 2012.
- [35] *Smoothing Images*, https://docs.opencv.org/master/d4/d13/tutorial_py_filtering.html, besøkt 04.05.2020.
- [36] Wilhelm B og Mark J B, *Edge-Preserving Smoothing Filters*. 2016, s. 413–451.
- [37] *Background Subtraction*, https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_video/py_bg_subtraction/py_bg_subtraction.html, besøkt 02.05.2020.
- [38] *Morphological Transformations*, https://docs.opencv.org/trunk/d9/d61/tutorial_py_morphological_ops.html, besøkt 02.05.2020.
- [39] *Extracting connected components from a binary image*, https://subscription.packtpub.com/book/application_development/9781788474443/3/ch03lvl1sec47/extracting-connected-components-from-a-binary-image, besøkt 04.05.2020.
- [40] *The connected components algorithm*, <https://www.oreilly.com/library/view/learn-opencv-4/9781789341225/3f4b3c3d-1390-4e98-bb5c-3f22d6d793c1.xhtml>, besøkt 04.05.2020.
- [41] *Computer Vision Annotation Tool - CVAT*, <https://github.com/opencv/cvat>, Sist besøkt 18.05.2020.
- [42] *The MIT License*, <https://opensource.org/licenses/MIT>, Sist besøkt 18.05.2020.
- [43] *Computer Vision Annotation Tool: A Universal Approach to Data Annotation*, <https://software.intel.com/en-us/articles/computer-vision-annotation-tool-a-universal-approach-to-data-annotation>, Sist besøkt 12.04.2020.
- [44] *Docker: Empowering App Development for Developers*, <https://www.docker.com/>, Sist besøkt 18.05.2020.
- [45] *HandBrake*, <https://handbrake.fr/>, Sist besøk 20.05.2020.
- [46] *datetime — Basic date and time types*, <https://docs.python.org/3/library/datetime.html>, Sist besøkt 03.05.2020.
- [47] *findContours*, https://docs.opencv.org/2.4/modules/imgproc/doc/structural_analysis_and_shape_descriptors.html#findcontours, Sist besøkt 03.05.2020.
- [48] *boundingRect*, https://docs.opencv.org/2.4/modules/imgproc/doc/structural_analysis_and_shape_descriptors.html#boundingrect, Sist besøkt 03.05.2020.
- [49] *time — Time access and conversions*, <https://docs.python.org/3/library/time.html>, Sist besøkt 03.05.2020.

- [50] *csv* — *CSV File Reading and Writing*, <https://docs.python.org/3/library/csv.html>, Sist besøkt 03.05.2020.
- [51] *The Shapely User Manual*, <https://shapely.readthedocs.io/en/latest/manual.html>, Sist besøkt 03.05.2020.
- [52] *matplotlib.pyplot*, https://matplotlib.org/api/pyplot_api.html, Sist besøkt 03.05.2020.
- [53] *sklearn.metrics*, <https://scikit-learn.org/stable/modules/classes.html#module-sklearn.metrics>, Sist besøkt 03.05.2020.
- [54] *cv::VideoCapture Class Reference*, https://docs.opencv.org/3.4/d8/dfe/classcv_1_1VideoCapture.html, Sist besøkt 03.05.2020.
- [55] *Geometric Image Transformations*, https://docs.opencv.org/2.4/modules/imgproc/doc/geometric_transformations.html#resize, Sist besøkt 03.05.2020.
- [56] *Image Filtering*, <https://docs.opencv.org/2.4/modules/imgproc/doc/filtering.html#morphologyex>, Sist besøkt 03.05.2020.
- [57] *What is PyQt?*, <https://riverbankcomputing.com/software/pyqt/intro>, besøkt 25.03.2020.
- [58] *Qt*, <https://www.qt.io/>, besøkt 25.03.2020.
- [59] *Brukertesting*, [https://www.difi.no/fagomrader-og-tjenester/tidstyver/tidstyvdatabasen-verktoy-og-metoder/brukertesting.](https://www.difi.no/fagomrader-og-tjenester/tidstyver/tidstyvdatabasen-verktoy-og-metoder/brukertesting/), besøkt 12.05.2020.
- [60] *Licenses & Standards*, <https://opensource.org/licenses>, besøkt 06.05.2020.
- [61] *Licenses*, <https://www.gnu.org/licenses/licenses.en.html>, besøkt 06.05.2020.
- [62] *License FAQ*, <https://www.riverbankcomputing.com/commercial/license-faq>, besøkt 06.05.2020.
- [63] *Toggl - Free Time Tracking Software*, <https://toggl.com>, besøkt 18.02.2020.

Prosjektavtale

Prosjektavtale

mellom NTNU Fakultet for informasjonsteknologi og elektroteknikk (IE) på Gjøvik (utdanningsinstitusjon), og

_____ (oppdragsgiver), og

_____ (student(er))

Avtalen angir avtalepartenes plikter vedrørende gjennomføring av prosjektet og rettigheter til anvendelse av de resultater som prosjektet frembringer:

1. Studenten(e) skal gjennomføre prosjektet i perioden fra _____ til _____ .

Studentene skal i denne perioden følge en oppsatt fremdriftsplan der NTNU IE på Gjøvik yter veiledning. Oppdragsgiver yter avtalt prosjektbistand til fastsatte tider. Oppdragsgiver stiller til rådighet kunnskap og materiale som er nødvendig for å få gjennomført prosjektet. Det forutsettes at de gitte problemstillinger det arbeides med er aktuelle og på et nivå tilpasset studentenes faglige kunnskaper. Oppdragsgiver plikter på forespørsel fra NTNU å gi en vurdering av prosjektet vederlagsfritt.

2. Kostnadene ved gjennomføringen av prosjektet dekkes på følgende måte:

- Oppdragsgiver dekker selv gjennomføring av prosjektet når det gjelder f.eks. materiell, telefon/fax, reiser og nødvendig overnatting på steder langt fra NTNU på Gjøvik. Studentene dekker utgifter for ferdigstilling av prosjektmateriell.
- Eiendomsretten til eventuell prototyp tilfaller den som har betalt komponenter og materiell mv. som er brukt til prototypen. Dersom det er nødvendig med større og/eller spesielle investeringer for å få gjennomført prosjektet, må det gjøres en egen avtale mellom partene om eventuell kostnadsfordeling og eiendomsrett.

3. NTNU IE på Gjøvik står ikke som garantist for at det oppdragsgiver har bestilt fungerer etter hensikten, ei heller at prosjektet blir fullført. Prosjektet må anses som en eksamensrelatert oppgave som blir bedømt av intern og ekstern sensor. Likevel er det en forpliktelse for utøverne av prosjektet å fullføre dette til avtalte spesifikasjoner, funksjonsnivå og tider.

4. Alle bacheloroppgaver som ikke er klausulert og hvor forfatteren(e) har gitt sitt samtykke til publisering, kan gjøres tilgjengelig via NTNUs institusjonelle arkiv hvis de har skriftlig karakter A, B eller C.

Tilgjengeliggjøring i det åpne arkivet forutsetter avtale om delvis overdragelse av opphavsrett, se «avtale om publisering» (jfr Lov om opphavsrett). Oppdragsgiver og veileder godtar slik offentliggjøring når de signerer denne prosjektavtalen, og må evt. gi skriftlig melding til studenter og instituttleder/fagenhetsleder om de i løpet av prosjektet endrer syn på slik offentliggjøring.

Den totale besvarelsen med tegninger, modeller og apparatur så vel som programlisting, kildekode mv. som inngår som del av eller vedlegg til besvarelsen, kan vederlagsfritt benyttes til undervisnings- og forskningsformål. Besvarelsen, eller vedlegg til den, må ikke nyttes av NTNU til andre formål, og ikke overlates til utenforstående uten etter avtale med de øvrige parter i denne avtalen. Dette gjelder også firmaer hvor ansatte ved NTNU og/eller studenter har interesser.

5. Besvarelsens spesifikasjoner og resultat kan anvendes i oppdragsgivers egen virksomhet. Gjør studenten(e) i sin besvarelse, eller under arbeidet med den, en patentbar oppfinnelse, gjelder i forholdet mellom oppdragsgiver og student(er) bestemmelsene i Lov om retten til oppfinnelser av 17. april 1970, §§ 4-10.
6. Ut over den offentliggjøring som er nevnt i punkt 4 har studenten(e) ikke rett til å publisere sin besvarelse, det være seg helt eller delvis eller som del i annet arbeide, uten samtykke fra oppdragsgiver. Tilsvarende samtykke må foreligge i forholdet mellom student(er) og faglærer/veileder for det materialet som faglærer/veileder stiller til disposisjon.
7. Studenten(e) leverer oppgavebesvarelsen med vedlegg (pdf) i NTNUs elektroniske eksamenssystem. I tillegg leveres ett eksemplar til oppdragsgiver.
8. Denne avtalen utferdiges med ett eksemplar til hver av partene. På vegne av NTNU, IE er det instituttleder/faggruppelider som godkjenner avtalen.
9. I det enkelte tilfelle kan det inngås egen avtale mellom oppdragsgiver, student(er) og NTNU som regulerer nærmere forhold vedrørende bl.a. eiendomsrett, videre bruk, konfidensialitet, kostnadsdekning og økonomisk utnyttelse av resultatene. Dersom oppdragsgiver og student(er) ønsker en videre eller ny avtale med oppdragsgiver, skjer dette uten NTNU som partner.
10. Når NTNU også opptrer som oppdragsgiver, trer NTNU inn i kontrakten både som utdanningsinstitusjon og som oppdragsgiver.
11. Eventuell uenighet vedrørende forståelse av denne avtale løses ved forhandlinger avtalepartene imellom. Dersom det ikke oppnås enighet, er partene enige om at tvisten løses av voldgift, etter bestemmelsene i tvistemålsloven av 13.8.1915 nr. 6, kapittel 32.

12. Deltakende personer ved prosjektgjennomføringen:

NTNUs veileder (navn): _____

Oppdragsgivers kontaktperson (navn): _____

Student(er) (signatur): _____ dato _____

_____ dato _____

_____ dato _____

_____ dato _____

Oppdragsgiver (signatur): _____ dato _____

Signert avtale leveres digitalt i Blackboard, rom for bacheloroppgaven.

Godkjennes digitalt av instituttleder/faggruppeleder.

Om papirversjon med signatur er ønskelig, må papirversjon leveres til instituttet i tillegg.

Plass for evt sign:

Instituttleder/faggruppeleder (signatur): _____ dato _____

Grupperegler

Grupperegler

- Hvert grupped medlem skal bidra med 30 timer totalt per uke. Hvis dette kravet ikke oppfylles, skal grupped medlemmet jobbe inn denne tiden senere i.a. prosjektet.
- Forventes at minimum 45 minutter av hver arbeidstid skal gå til ren effektiv jobbing.
- Forventes at hvert medlem møter til fastsatt tid, dette inkluderer da gruppemøter, møter med veileder og annet oppsatt arbeid. Skulle noe forekomme, så forventes det er medlemmet sier i fra til resten av gruppen så fort som mulig. Ikke meldt for-sent-komming bøtelegges med tiden som personen har kommet for sent, ganger 2. Denne tiden blir opptil vedkommende å fordele.
- Grupped medlemmer pålegges å stille med oppdatert statusinformasjon for ukens avsluttende sprintmøte.
- Ved interne uenigheter eller store beslutninger i gruppa kan det kalles inn til demokratisk avstemning. Dersom dette forekommer, er alle grupped medlemmer pålagt å stemme. Ved betydelige uenigheter tillates nedstemte grupped medlemmer å ta opp saken med veileder.
- Dersom et grupped medlem føler at han ikke ligger i rute til å bli ferdig med sprintens arbeidsoppgave(r) skal han informere resten av gruppa så tidlig som mulig, og be om assistanse dersom dette lar seg gjøre.
- Konsekvenser for regelbrudd vil i første omgang være en advarsel. Ved gjentatte regelbrudd vil saken bli tatt opp med veileder.
 - Ved mange gjentatte regelbrudd og samtale med veileder, så kan ekskludering finne sted. NB: Ingen ekskludering de 28 siste dagene før innleveringsfristen.

Underskrift: Alex Rogat sluy

Underskrift: Thomas Jøndal

Underskrift: Vehem Alseth

Forprosjekt



NTNU

Norwegian University of
Science and Technology

Deteksjon av fisk i video

Prosjektplan

Alex Rogatskiy
Vebjørn Sundli Alseth
Thomas Faack Bjørnstad

1. februar 2020

Innhold

1	INTRODUKSJON	1
1.1	Bakgrunn	1
2	BAKGRUNN FOR PROSJEKTET	2
2.1	Oppgavebeskrivelse	2
2.1.1	Oppgavebeskrivelse fra oppdragsgiver	2
2.1.2	Grunnlag for oppgaven	2
2.2	Dagens system og rutine	4
2.3	Eksisterende teknologi	4
2.3.1	Datasyn	4
2.3.2	Maskinl�ring	5
2.3.3	V�rt valg	5
2.4	Rammebetingelser	6
2.5	Ressursbehov	6
3	PROSJEKTM�L	7
3.1	Effektm�l	7
3.2	Resultatm�l	7
3.3	Prosessm�l	8
3.4	Prosjektets omfang	8
4	RISIKOANALYSE	9
4.1	Valg av risikomodell	9
4.2	FMEA (Failure Mode Effect Analysis)	9
4.3	FMEA - Risikoanalyse	10
4.4	Forebyggende tiltak og konsekvensh�ndtering av hendelser som g�r frem av sikkerhetsutredningen FMEA	12
5	RETNINGSLINJER	16
5.1	Krav til dokumentasjon	16
5.2	Krav til kvalitetsgjennomganger	16
5.3	Krav til standarder og verkt�y	18
5.4	Endringsh�ndtering	18
6	PROSJEKTORGANISERING	19
6.1	Grunnlag for valg av utviklingsmodell	19
6.2	Scrumban	19
6.2.1	Kanban	20
6.2.2	Oppdeling i sprinter	20

<i>Innhold</i>	2
6.2.3 Sprint planning meeting	20
6.2.4 Sprint review meeting	20
6.3 Grupperoller	21
6.4 Milepæler	22
7 VEDLEGG	25

Figurer

2.1	Fisketrapp med spalteåpning hentet fra CEDREN [5].	3
2.2	Eksempel på bilde med fisk fra videomateriale.	4
5.1	Demonstrasjon av hvordan verktøyet CCA [13] kan sjekke for kontraster mellom tekst og bakgrunn.	17
6.1	Gantt-diagram over prosjektet.	22

INTRODUKSJON

1.1 Bakgrunn

Bacheloroppgaven avslutter studieprogrammet for Bachelor i ingeniørfag data hos NTNU, og skal integrere viktige deler av programmets faglige innhold [1]. Oppgaven skal derfor ta for seg reelle problemstillinger og bidra til innføring i vitenskapsteori og metode. Oppgaven skal resultere i en rapport som på en strukturert måte dokumenterer arbeidet som er gjennomført. Gruppen har blitt tildelt en oppgave som skal gjennomføres på vegne av Norsk Institutt for Naturforskning (NINA). NINA er en uavhengig stiftelse som forsker på natur og samspillet natur – samfunn [2].

I starten av gjennomføringen av bacheloroppgaven gjøres det et forprosjekt, hvor gruppene planlegger hvordan utførelsen av selve bacheloroppgaven skal gjøres og hvilke rammer og mål oppgaven skal ha. Dette dokumentet har som mål å belyse disse punktene på en oversiktlig og enkel måte.

BAKGRUNN FOR PROSJEKTET

2.1 Oppgavebeskrivelse

2.1.1 Oppgavebeskrivelse fra oppdragsgiver

Tilstanden i norske elver og innsjøer dokumenteres blant annet gjennom overvåking av fiskebestandene. Norsk Institutt for Naturforskning (NINA) bruker i dag undervannskamera for dette arbeidet. I mange elver står det kamera å filmer døgnnet rundt i såkalte fisketrapp. Fra disse filmene gjør man fisketelling, bestemmelse av art, med mer. Figur 2.1 viser et skjermbilde fra en slik video av en ørret på vei gjennom en fisketrapp. Dette arbeidet i dag gjennomføres for det meste manuelt hvor en person ser gjennom videofilene. Dette er meget tidkrevende og man ønsker derfor å automatisere arbeidet. Oppgaven bidrar til å kunne gi en grundig statusbeskrivelse av fiskesamfunnet, og gjøre at forskere hos NINA kan bruke tiden bedre enn å se gjennom video.

NINA ønsker å få laget et verktøy/programvare som kan automatisk detektere, helst i sanntid, når det kommer fisk inn i bildet. Gjenkjenningen skal være robust og fungere under alle slags forhold (forskjellig lysforhold, farge på vannet, uavhengig av mengde luftbobler, etc.). Det er også ønskelig å kunne telle antall fisk som passerer, artsbestemmelse (f.eks. om det er ørret, harr eller sik), og hvilken retning fisken har (oppover eller nedover elva). Det er også høyst ønskelig med et brukergrensesnitt til verktøyet.

2.1.2 Grunnlag for oppgaven

Høyegga dam er en 175 meter lang betongdam med en maksimal høyde på 10 meter. For at fisk skal kunne passere den store demningen så er det blitt bygd en fisketrapp som skal gjøre det mulig for å fisk å fortsette oppover Glomma. Fiske-trappen var av typen klassisk kulpetrapp for laks, noe som forutsatte at fisken måtte klare å hoppe fra kulp til kulp [3]. Da det er andre arter som også skal oppover elven, men som ikke besitter de samme ferdighetene til å hoppe oppover i fisketrappen, så ble det besluttet at Høyegga, som en del av SAFEPASS-prosjektet [4], skulle bygge om fisketrappen [5].



Figur 2.1: Fisketrapp med spalteåpning hentet fra CEDREN [5].

NINA kunne også bygd en helt ny fisketrapp som ville hatt spesifikasjoner som ville tilfredsstilt ferdighetene til alle artene, men da en vanlig fisketrapp fort kan koste 100 millioner kroner, så vil en lettere modifikasjon av allerede eksisterende trapp være mye mer aktuelt.

Det er i dag ønskelig å gjøre så små grep inn i naturen som mulig og det er på bakgrunn av dette at NINA har utplassert et kamera rettet mot den ene spalteåpningen for å kunne føre statistikk på fisk som passerer oppover elven. Om statistikken skulle gi inntrykk for at modifikasjonene gjort med fisketrappen fungerer, så vil NINA slippe å lage mange nye fisketrapper, men heller kunne gjøre liknende modifikasjoner på allerede eksisterende fisketrapper.

Arbeidet med å holde kameraet gående har vært relativt enkelt og siden kameraet står på døgnet rundt, så har oppgaven med å samle inn materiale vært utført passivt. I dag samles videomateriale inn for at personer skal sette seg ned å gå gjennom hvert eneste sekund med innhentet videomateriale, for å så føre statistikk på hva som passerer, hvilken retning fisken har og andre eventuelle hendelser som skulle forekomme. Denne oppgaven er derfor veldig tidkrevende og setter krav til at brukeren som ser gjennom videomaterialet og fører statistikken må sitte foran skjermen hele tiden.



Figur 2.2: Eksempel på bilde med fisk fra videomateriale.

2.2 Dagens system og rutine

Det er NINA som står for innhenting av video ved å utplassere og hente kameraer i ulike elver. Per i dag så overleveres videomateriale til enkeltpersoner som sitter manuelt og ser gjennom hver enkelt video. Videomateriale som da tilsvarer 30 timer, vil i dette tilfellet ta minimum 30 timer med arbeid hvor en person sitter og notere ned når fisk passerer, hvilken art den tilhører og eventuelt om fisken gjør noe uventet.

2.3 Eksisterende teknologi

Det er spesielt to fagfelt som kommer til å være relevant for å løse backend-delen av verktøyet som utvikles; datasyn ("computer vision") og maskinlæring.

2.3.1 Datasyn

Datasyn er vitenskapen og teknologien om hvordan maskiner kan 'se', hvor det å se i dette tilfellet betyr å kunne plukke ut ønsket informasjon fra et bilde [6]. Deteksjon av objekter er det første og viktigste steget i å kunne gjenkjenne objekter. Tradisjonelt så kan spesielle utvalgte objekter bli oppdaget og gjenkjent ved å se om en mal og objektet stemmer overens, men hastigheten på gjenkjenningen har alltid vært et problem.

Datasyn er en passiv teknologi og i et typisk scenario så vil kameraet være fastmontert. Kameraet vil ha oversikt over det aktuelle området og ta i bruk algoritmer for å kunne prosessere utbyttet fra bildene. Selv om det i utgangspunktet kan virke enkelt, så er det ikke en enkel oppgave da både statiske og dynamiske objekter vil være tilstedet i det aktuelle området. I tillegg vil flere faktorer som ikke er mulige å kontrollere som bevegelse av solen, overganger mellom dag og natt, skygger,

kamera-artefakter fra objekter som fester seg på linsen, vibrasjoner i kameraet og ulike værelementer dukke opp. Alle disse faktorene spiller inn når algoritmene skal arbeide med å analysere bildene som kameraet fanger opp.

2.3.2 Maskinlæring

Klassisk maskinlæring og dataanalyse kan bli delt opp i flere kategorier. I den første kategorien så benytter vi datamaskiner for å utføre 'klassisk' dataanalyse-metoder som for eksempel minste kvadraters metode, polynomisk interpolasjon og dataanalyse [7].

Maskinlærings-protokoller kan være veiledet eller ikke-veiledet. I veiledet læring deles data i navngitte kategorier, som eksempler med håndskrevne tall sammen med det nummeret det håndskrevne tallet faktisk skal representere. Oppgaven til maskinen blir å lære seg hvordan den kan sette navn på data utenfor treningssettet. Ikke-veiledet læring benytter seg av data som ikke er navngitt og målet for maskinen blir å finne naturlige kategorier som settet med treningsdata hører til. En tredje type er maskinlæringsoppgaver som å spille det kinesiske spillet Go", som innebærer kombinasjoner av veiledet og ikke-veiledet læring, sammen med treningssett som kan være generert av maskinen selv.

2.3.3 Vårt valg

Oppdragsgiver har etter innledende møte gitt gruppen myndighet til å velge mellom datasyn og maskinlæring. Programmets viktigste mål er å kunne gjennomføre deteksjon av fisk i video.

For å kunne gjennomføre deteksjon av fisk, så vil datasyn kreve at gruppen klarer å eliminere støy i videomaterialet og klarer å hente ut den nødvendige informasjonen for å kunne si om det er en fisk som kameraet ser, eller om det er noe annet som flyter forbi. Maskinlæring ville vært det primære valget om gruppen skulle gjennomført artbestemmelse og hatt hovedfokus på dette, men ettersom primæroppgaven er deteksjon av fisk, så har gruppen valgt å ikke benytte seg av maskinlæring.

Gruppen har konkludert med at datasyn vil være mest egnet, både med tanke på at gruppen ønsker å lære seg noe nytt, men også fordi datasyn vil kunne være lettere å videreutvikle.

Gruppen har tidligere gjennomført mindre arbeidsoppgaver i Python og ettersom Python er et av de desidert mest brukte programmeringsspråkene for datasyn [8], så vil gruppe ta i bruk dette. Mange gode og godt dokumenterte bibliotek for datasyn er allerede i utstrakt bruk, og språket generelt vil være enklere for brukermassen hos NINA å sette seg inn i enn syntaktisk kompliserte språk som for eksempel C++.

2.4 Rammebetingelser

Prosjektet vil gjennomføres i perioden fra 01.01.2020 til 20.05.2020, hvor den endelige prosjektrapport skal leveres. En muntlig presentasjon av oppgaven vil foregå i løpet av tidsrommet 06.06.2020 - 08.06.2020.

2.5 Ressursbehov

Gruppen kommer trolig til å ha behov for betydelige mengder videomateriale fra NINA for å kunne gjennomføre tilstrekkelig testing av verktøyet, både til utvikling og kvalitetssikring. For å gjøre oppbevaring så enkelt som mulig anser gruppen det som best å ha dette videomaterialet på eksterne disketter og/eller opplastet i en sky-løsning.

Om gruppen får tid til å utvide programvaren til å kunne gjøre artbestemmelse, så kan det være behov for maskinlæring. For å kunne gjennomføre maskinlæring effektivt, så vil det være behov for en maskin med en dedikert GPU. Fargelaben på NTNU i Gjøvik har maskiner med spesifikasjoner som tilfredsstillende behovet for GPU, så tilgang til Fargelaben vil være ønskelig om gruppen ender opp med å bruke maskinlæring.

PROSJEKTMÅL

3.1 Effektmål

- Eliminere behov for at personale eller innleid hjelp behøver å lete gjennom videomateriale fra start til slutt for å finne fisk
- Redusere den totale aktive tiden som brukes av ansatte hos NINA til innhenting av statistisk materiale fra fiskevideo med minst 80% i forhold til i dag.

3.2 Resultatmål

Utarbeide et verktøy for NINA som:

- Gir output i form av en .CSV-fil som utformes etter NINAs ønsker
- Ved gjennomkjøring av en tilfeldig videofil fra utlevert datamateriale fra 2017 i gjennomsnitt produserer tidsstempel som peker til minst 95% av fiskene registrert av manuell gjennomgang[9]
- Skriver ut tidsstempel som i 95% av tilfellene viser til et punkt i videomaterialet som har minst én fisk i et tidsvindu på +- 5 sekunder.
- Ved kjøring maksimalt bruker 1:1 total prosesseringstid per tid video ('sann-tid' prosessering)
 - Gitt at verktøyet kjører på hardware som er minst like kraftig som laptopene NINA bruker til daglig[10]
 - Gitt at video er i formatet 1920 x 1080 med 30 bilder i sekundet
- Har et brukergrensesnitt som tilfredstiller kravet om AA-sertifisering i henhold til kravet om universiellutforming - WCAG 2.0 [11].
- Kun bruker videoformat (både som input og eventuell output) som støttes av VLC Media Player [12]
- Kan kjøres på dagens mest brukte operativsystemer
 - Windows 7 og nyere
 - MacOS Catalina og nyere

Utarbeide et verktøy for NINA som gjerne:

- Kan artsbestemme detektert fisk med minst 80% nøyaktighet
 - Målet på minst 80% ser bort ifra forsøk på artsbestemmelse ved falske positiver i det detekterte materialet
- Kan bestemme hvilken vei en detektert fisk passerte med minst 95% nøyaktighet
 - Målet på minst 95% ser bort ifra forsøk på retningsbestemmelse hos falske positiver i det detekterte materialet
- Kan gjenkjenne enkeltindivider over brukerdefinerte tidsrom.
- Beskriver usikkerhet i deteksjonen på bakgrunn av metadata fra videomaterialet som legges inn i en egen kolonne i CSV-outputen.
 - Outputen er en kommatall-verdi som er en kombinasjon av lysnivå og fargevarians

3.3 Prosessmål

- Gruppens medlemmer skal utvikle samarbeids- og kommunikasjonsevne til et profesjonelt nivå
- Gruppens medlemmer skal få økt kunnskap og erfaring innen datasyn og utvikling av brukergrensesnitt
- Gruppen skal få en karakter alle gruppens medlemmer er fornøyd med

3.4 Prosjektets omfang

- Gruppen skal ikke drive med opplæring av NINAs brukermasse
- Verktøyet skal ikke foreta statistiske beregninger, kun forenkle/automatisere uthenting av statistikkgrunnlaget

RISIKOANALYSE

4.1 Valg av risikomodell

Grunnen til at FMEA har blitt valgt som modell for risikoanalyse er fordi gruppen ikke har nok konkret empiri og erfaring til å gjennomføre en usikkerhetsanalyse. Denne typen analyse hadde vært aktuell dersom gruppen hadde basert utviklingen i prosjektet på for eksempel det å nå konkrete milepæler eller delt opp arbeidsmengden i konkrete delmål. I det tilfelle kunne hvert delmål blitt gjennomgått av gruppemedlemmer hvor det hadde blitt foretatt en estimat over hvor lang tid det hadde tatt å gjennomføre de ulike oppgavene. Ettersom gruppen jobber etter fastsatte sprinter som har en bestemt lengde, lar ikke dette seg utføre, dermed bortfaller også denne sikkerhetsanalysemetoden.

Samtidig har ikke valget falt på FTA ettersom gruppen må ha et konkret hendelsesforløp i systemet. Dette har ikke gruppen ettersom den jobber i ulike sprinter, hvor konkrete arbeidsoppgaver, samt total arbeidsmengde per sprint, er ukjent for oss for de kommende sprintene, og blir kun bestemt i sprint planleggingsmøte en uke om gangen.

4.2 FMEA (Failure Mode Effect Analysis)

Hensikten er å avdekke kritiske feil i systemet, i vårt tilfelle avdekke eventuelle svakheter og hendelser som kan påvirke leveransen av bacheloroppgaven/rapporten.

Gruppen tar for seg én aktivitet/komponent om gangen, alle øvrige antas å fungere som normalt. Denne modellen tar dermed ikke for seg ulike kritiske kombinasjoner av hendelser som kan eventuelt inntreffe samtidig eller i et hendelsesforløp. Analytikernes kompetanse om systemet er også avgjørende. I dette tilfellet er gruppen forholdsvis uerfarne med gjennomføring av slike prosjekter, dermed vil kvaliteten av analysen bli svekket som en direkte følge av dette.

4.3 FMEA - Risikoanalyse

Tabell 4.1: FMEA Risikoanalyse

<i>Starten av tabellen</i>							
ID	Hendelse	Feilmodi	Feilmodi - presisering	Konsekvenser	PO	PD	S
1	Sykdom	Sykdom blant gruppemedlemmer	Kortids sykdom	Stor backlog for neste sprint	7	10	5
2			Langtidssyksom / gruppemedlem forlater gruppen	Hele arbeidsmengden må fordeles mellom friske gruppemedlemmer. Kvalitetsreduksjon i rapportleveransen	3	10	10
3		Veileder blir sykmeldt	Korttidssykdom	Gruppen må fortsette uten veiledninger i en kort periode. Dårligere kvalitet på endelig rapportleveranse	4	10	3
4			Langtidssykdom	Gruppen blir tildelt ny veileder. Kan medføre dårligere kvalitet på veiledninger, samt redusert kvalitet på endelig rapportleveranse	2	10	5
5	Korrupsjon/tap av data/ nedetid av programvare	Tap av videomateriale	Lagringsmediumet blir ødelagt og / eller Onedrive-dataen forsvinner	Gruppen må etterspørre nødvendig videomateriale på nytt	2	8	3
6		Programvare blir korrumpert	Utviklingsverktøyet oppdateres og blir inkompatibelt med kildekoden	Rulling tilbake til siste fungerende versjon av programvaren. Større backlog for neste sprint	1	1	2
7		Biblioteker blir korrumperte	Biblioteker som benyttes av programvare blir korrumpert	Gruppen bruker tid til å samkjøre bibliotekversjoner. Større backlog for neste sprint	1	3	2
8		Tap av kildekode	Delvis tap av kildekode fra versjonskontrollplattformen Git	Gruppen må gjenopp- ta arbeidet fra den siste fungerende lokale versjonen av koden	1	10	6
9			Fullstendig tap av kildekode, som følge uforutsette omstendigheter hos gruppen eller verktøyet som benyttes	Gruppen må starte på nytt med utviklingen av programvaren. Redusert eller manglende leveranse av programvare	1	10	10
10		Nedetid	Nedetid av programvare som benyttes for å skrive rapport. I vårt tilfelle Overleaf	Gruppen forhindres med å arbeide med rapporten i planlagt/avsatt periode	5	7	4

Tabell 4.2: FMEA Risikoanalyse

<i>Fortsettelse av tabellen</i>							
ID	Hendelse	Feilmodi	Feilmodi - presisering	Konsekvenser	PO	PD	S
11	Tid	Sprintmål blir ikke oppnådd	Milepæler nås ikke innen fastsatt dato som følge av uforutsette hendelser	Større backlog for neste sprint	6	5	4
12		Fristen for innlevering av rapport blir ikke overholdt	Gruppen leverer ikke bacheloroppgave- rapporten innen fastsatt tid	Gruppen får ikke levere inn bachelorrapporten	2	2	10
13	Uenigheter	Uenighet i gruppa ved avgjørelser	Manglende evne til å inngå kompromisser blant gruppemedlemene	Betent gruppedynamikk	6	4	4
14		Uenighet med oppdragsgiver	Misforståelse av kravspesifikasjon gitt av oppdragsgiver	Oppdragsgiver blir mindre fornøyd med sluttprodukt	2	4	4
15		Uenighet med veileder	Uenighet om avgjørelser for prosjektet	Større usikkerhet knyttet opp mot de avgjørelsene gruppen foretar	5	2	3
<i>Tabell slutt</i>							

4.4 Forebyggende tiltak og konsekvenshåndtering av hendelser som går frem av sikkerhetsutredningen FMEA

Tabell 4.3: Konsekvenshåndtering

<i>Starten av tabellen</i>			
ID	Konsekvens	Konsekvenshåndtering	Forebyggende tiltak
1	Stor backlog for neste sprint	Den som er syk, har ansvaret for å kommunisere dette til de andre gruppemedlemmene. Slik at arbeidet kan fordeles mellom friske gruppemedlemmer. Den som er syk må også ta igjen den arbeidstiden som er tapt etter at vedkommende har blitt frisk igjen. De friske gruppemedlemmene skal også kommunisere jevnt med den som er syk, slik at vedkommende er oppdatert på hva som er blitt gjort og hva som må gjøres.	Alle gruppemedlemmene skal holde hverandre oppdatert til enhver tid, med hva de jobber med vha. ukentlige sprintmøter, men også på fremgangen av de individuelle oppgavene underveis i sprintene. Ved evt. sykdom vil dette tiltaket gjøre det enklere for den som er syk å ta igjen de andre.
2	Hele arbeidsmengden må fordeles mellom friske gruppemedlemmer. Kvalitetsreduksjon i rapportleveransen	Konsekvensen er også delvis et tiltak. Resterende gruppemedlemmer møtes sammen med veileder og avklarer hvordan dette blir håndtert videre. Evt. at hele arbeidsmengden må fordeles mellom friske gruppemedlemmer	Gruppemedlemmer må kommunisere med hverandre og holde hverandre oppdaterte på hva som foregår så langt det lar seg gjøre. Men dersom en slik hendelse inntreffer kommer det til å skje brått og uventet for alle gruppemedlemmene. Det vil ikke være mulig å forutse at det inntreffer. Dermed er det begrenset med hva slags forebyggende tiltak som kan lages for dette
3	Gruppen må fortsette uten veiledninger i en kort periode. Dårligere kvalitet på endelig rapportleveranse	Dersom veilederen blir syk, fortsetter gruppen å følge sine oppsatte arbeidssprinter som vanlig. Kommunikasjon med veileder kan foregå på mail. Eventuelt kan det opprinnelige møte bli flyttet til et senere tidspunkt	Gruppen kan ikke forebygge sykdom hos veileder, men jevnlig kommunikasjon mellom veileder og gruppen burde avdekke når denne hendelsen inntreffer relativt raskt slik at gruppen vet det på forhånd før en eventuell veiledningsmøte
4	Gruppen blir tildelt ny veileder. Kan medføre dårligere kvalitet på veiledninger, samt redusert kvalitet på endelig rapportleveranse	Gruppen må oppdatere den nye veilederen på fremgangen. Veileder må også sette seg inn i gruppens oppgave for å få til best mulig kvalitet på veiledninger	Gruppen kan ikke forutse langtidssykdom hos veileder. Men gruppen må føre logg/referat fra møter med veileder og oppdragsgiver, samtidig som å ha oversikt over fremgangen i prosjektet. Til sammen vil dette sørge for at ny veileder kan lettere sette seg inn i fremgangen hos gruppen

Tabell 4.4: Konsekvenshåndtering

<i>Fortsettelse av tabellen</i>			
ID	Konsekvens	Konsekvenshåndtering	Forebyggende tiltak
5	Gruppen må etterspørre nødvendig videomateriale på nytt	Gruppen møtes, eller kommuniserer sammen digitalt hvor det avtales hvem er det som får ansvaret for å ta kontakt med veileder eller oppdragsgiver for å få kopi av nødvendig videomateriale på nytt	Gruppen lager et redundant system for lagring av nødvendig videomateriale slik at den ikke ligger lagret bare på et sted. Dette avtales og gjennomføres på et gruppemøte i det videomateriale utleveres
6	Rulling tilbake til siste fungerende versjon av programvaren. Større backlog for neste sprint	Gruppen møtes, slik at skadeomfanget av hendelsen kan utredes i fellesskap. Etter at dette er kartlagt vil det være aktuelt å rulle programvaren som er berørt tilbake til siste fungerende versjon, samt samkjøre programvareversjonen hos samtlige gruppemedlemmer	Før utviklingen starter bør gruppen gå gjennom programvaren som benyttes og sørge for automatiske oppdateringer er slått av
7	Gruppen bruker tid til å samkjøre bibliotekversjoner. Større backlog for neste sprint	Gruppen møtes, slik at skadeomfanget av hendelsen kan utredes i fellesskap. Etter at dette er kartlagt vil det være aktuelt å samkjøre bibliotekversjonene hos samtlige gruppemedlemmer. Dersom det viser seg til å være svært tidskrevende, blir dette en del av backloggen for neste sprint.	Før utviklingen starter bør gruppen gå gjennom bibliotekene som gruppen vet skal benyttes. Dersom/når andre biblioteker blir lagt til, skal dette dokumenteres i commit - meldingen i versjonskontrollprogrammet Git ved opplastning
8	Gruppen må gjenoppta arbeidet fra den siste fungerende lokale versjonen av koden	Gruppen møtes, slik at skadeomfanget av hendelsen kan utredes i fellesskap. Etter at dette er kartlagt vil det være aktuelt å samkjøre koden slik at alle fortsetter utviklingen fra siste fungerende versjon av koden. Dersom det viser seg til å være svært tidskrevende, blir dette en del av backloggen for neste sprint. Gruppen må da legge inn mer arbeid/timeverk for å ta igjen	Gruppen møtes før utviklingen starter og etablerer som god praksis at alt kode som lastes opp/committes, skal være fungerende og kjørbare kode. Dessverre kan ikke gruppen forutse tap av kode som følge av at nettstedet har nedetid eller blir utsatt for et angrep.

Tabell 4.5: Konsekvenshåndtering

<i>Fortsettelse av tabellen</i>			
ID	Konsekvens	Konsekvenshåndtering	Forebyggende tiltak
9	Gruppen må starte på nytt med utviklingen av programvaren. Redusert eller manglende leveranse av programvare	Veileder må informeres umiddelbart. Gruppen møtes, slik at skadeomfanget av hendelsen kan utredes i fellesskap. Det vil være aktuelt å se på hva slags kode som er lagret lokalt, slik at dette kan arbeidet kan fortsette fra sist tilgjengelig iterasjon av koden. Etter at dette er kartlagt vil det være aktuelt å begynne å utvikle på nytt. Gruppemedlemmene må regne med å bruke all tilgjengelig som ikke er opprinnelig satt av til arbeid med oppgaven til utvikling.	Gruppemedlemmene pålegges å ta backup av kode hver uke etter fullført sprintmøte. Backupen lagres lokalt hos gruppemedlemmene. På denne måten, kan vi begrense den verst tenkelige scenario til å være at gruppen mister maksimalt 1 sprint med data.
10	Gruppen forhindres med å arbeide med rapporten i planlagt/avsatt periode	Dersom programvaren Overleaf er nede, kan gruppemedlemmene jobbe videre med andre oppsatte oppgaver i mellomtiden, samt jobbe lokalt på hver sin datamaskin, da det antas at nedetiden er midlertidig. Gruppemedlemmene må regne med å ta igjen den tiden som er tapt som følge av nedetiden for redusere backloggen for neste sprint.	Etter hvert sprint pålegges gruppemedlemmene ta backup av kildekoden på Overleaf slik at denne kan lagres lokalt. Samtidig er hvert enkelt gruppemedlem ansvar for å lagre en lokal kopi av alle figurer, bilder, tabeller og vedlegg som lastes opp på Overleaf.
11	Større backlog for neste sprint	Gruppen møtes eller kommuniserer med hverandre digitalt, hvor det avtales og planlegges hvordan arbeidsmengden skal håndteres videre. Gruppemedlemmene må regne med ekstra arbeid utover det som er planlagt på forhånd for å redusere backloggen for neste sprint.	I og med at gruppemedlemmene er veldig uerfarne når det kommer til prosjektarbeid, må gruppen dra lærdom underveis i gjennomføringen av sprintene. Når gruppen planlegger en ny sprint må gjennomføringen samt arbeidsmengden fra forrige sprint tas til vurdering, slik at neste sprint kan tilpasses tilsvarende.

Tabell 4.6: Konsekvenshåndtering

<i>Fortsettelse av tabellen</i>			
ID	Konsekvens	Konsekvenshåndtering	Forebyggende tiltak
12	Gruppen får ikke levere inn bachelorrapporten	Konsekvensen av dette er katastrofal. Det er dessverre ikke mulig å gjøre noe med det ettersom fristen for innlevering er endelig. Gruppen forplikter seg også å snakke med veileder umiddelbart.	Gruppen må planlegge å bli enig om å ikke levere rapporten sammen dagen som innleveringsfristen er, men heller et par dager før. På denne måten kan eventuelle feil ved innlevering løses i god tid før innlevering.
13	Betent gruppedynamikk	Gruppen møtes, problemstillingen tas opp i plenum. Gruppemedlemmene pålegges å begrunne argumentene saklig. I tilfeller hvor øvrig punkt ikke løser problemet, skal dette tas opp med veileder for åpen diskusjon der.	Gruppen møtes i begynnelsen av prosjektperioden, og utarbeider gruppereglement som alle gruppemedlemmene må skrive under på. Dette skal sørge for at hver av gruppemedlemmene forplikter seg til ta ansvar. Samtidig pålegges også gruppe-medlemmene til å behandle hverandre med respekt.
14	Oppdragsgiver blir mindre fornøyd med sluttproduktet	Gruppen møter oppdragsgiver i god tid før leveransefristen for å forsøke å komme til enighet. Veileder skal også kobles på, slik at det ikke oppstår en konflikt.	Gruppen pålegges å ha jevnlig dialog med oppdragsgiver via epost, og via personlige møter etter behov hvor kravspesifikasjonen gjennomgås. Gruppen pålegges også å presentere handlingsplanen for oppdragsgiver for gjennomgang og evt. tilbakemelding.
15	Større usikkerhet knyttet opp mot de avgjørelsene gruppen foretar	Gruppen møtes, og gjennomgår problemstillingen hvor gruppen, må ta uenigheten med veileder til vurdering og selv ta valg om ønsket beslutning.	Dersom/når uenigheten oppstår, skal gruppen ta denne til vurdering, møtes og gjennomgå problemstillingen, deretter foreta en beslutning.
<i>Tabell slutt</i>			

RETNINGSLINJER

5.1 Krav til dokumentasjon

Ettersom gruppe medlemmene etter prosjektets slutt ikke kommer til å være tilstrekkelig tilgjengelige for bistand til vedlikehold, oppdateringer og annen generell hjelp i forbindelse med verktøyet, må det i løpet av prosjektet utvikles dokumentasjon som gjør det så enkelt som mulig for andre å overta denne oppgaven. Dette kommer til å løses på to måter; utforming av en kortfattet og enkel brukermanual for bruk av det utviklede verktøyet og et mer teknisk dokument for eventuelle utviklere som skal vedlikeholde og videreutvikle verktøyet i senere tid.

Denne brukermanualen skal gi en kortfattet beskrivelse av hva verktøyet tilbyr med hensyn på ulike funksjonaliteter. Deretter en litt mer detaljert forklaring av hvordan denne funksjonaliteten benyttes i praksis, med henvisninger til brukergrensesnittet og forklaringer med minimalt rom for feiltolkning.

Dokumentet for utviklere må inneholde en mer teknisk dyptgående beskrivelse av hvordan programvaren fungerer. Dokumentet skal, sammen med god kommentering i kildekoden, kunne gi en person med utdanning og erfaring tilsvarende gruppens medlemmer et godt overblikk over hvordan programvaren og brukergrensesnittet er satt sammen.

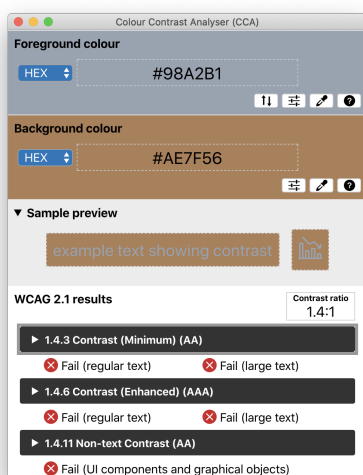
5.2 Krav til kvalitetsgjennomgang

For å verifisere at resultatmålene om andel fisk som detekteres og falske positive overholdes vil gruppen benytte utlevert datamateriale fra 2017 som testmateriale, hvor Excel-dokument [9] med resultater fra manuelt gjennomsyn av NINA benyttes som fasit. For at resultatmålene skal anses som oppfylt av gruppen, behøver testingen å vise at det statistisk sett kan vises at kravene overholdes med et signifikansnivå $p = 0,05$.

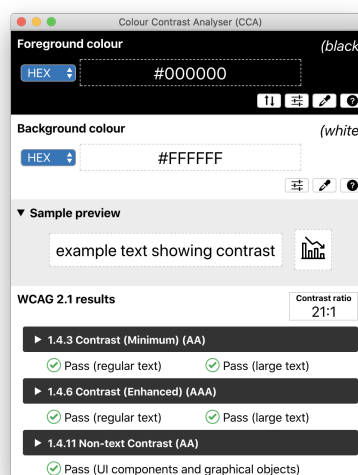
For verifisering av at resultatmålet på minst sanntidsprosessering overholdes skal gruppen benytte et testoppsett bestående av en MacBook Pro 2018 som kjører en virtuell maskin som simulerer spesifikasjonene til laptopene [10] NINA benytter til daglig.

For å kontrollere at GUI-en tilfredsstillende WCAG 2.0 så kommer gruppen til å benytte verktøyet Color Contrast Analyzer (CCA) [13]. CCA er en gratis programvare som Digitaliseringsdirektoratet anbefaler for å sjekke kontrast mellom tekst og bakgrunn [14].

Verktøyet tilbyr muligheten til å velge en tekstfarge i brukergrensesnittet og sjekke denne fargen opp mot bakgrunnen. Man ser så etter om verktøyet godkjenner kontrasten i henhold til AA-standarden.



(a) Eksempel på kontraster som ikke godkjennes.



(b) Eksempel på kontraster som godkjennes.

Figur 5.1: Demonstrasjon av hvordan verktøyet CCA [13] kan sjekke for kontraster mellom tekst og bakgrunn.

5.3 Krav til standarder og verktøy

- Prosjektrapport skal benytte NTNUs mal [15] for generell utforming av rapporten.
- Rapporten skal skrives i \LaTeX , på norsk bokmål, hvor gruppen skal benytte Overleaf [16] for sanntidssynkronisering.
- Gruppen skal benytte Git[17] og Bitbucket [18] for versjonskontroll av utviklingen av verktøyet.
- Gruppen skal benytte PyCharm til utviklingen av brukergrensesnittet.
- Gruppen skal benytte PyQt til utviklingen av backend for verktøyet.
- Gruppen skal benytte Colour Contrast Analyser [13] til kvalitetssjekk på GUI.

5.4 Endringshåndtering

Det er naturlig at det i løpet av prosjektet vil komme ønsker om endringer i verktøyet fra oppdragsgiver. For å formalisere prosessen og sørge for at den er dokumentert og gjennomføres ordentlig, kommer gruppen til å benytte seg av følgende prosedyre for endringshåndtering [15]:

1. Dokumenter endringens innhold
2. Analyser konsekvensene for prosjektet
3. Beregn eventuell kost/nytt
4. Godkjennelse og aksept
5. Logg endringen
6. Juster planene
7. Informer interessentene
8. Gjennomfør endringen

PROSJEKTORGANISERING

6.1 Grunnlag for valg av utviklingsmodell

Dette prosjektet har store usikkerhetsmoment i vanskelighetsgrad og tidsbehov for de ulike funksjonalitetene som skal og kan implementeres. Det er viktig å forsikre at gruppen rekker å utvikle grunnfunksjonaliteten til verktøyet (resultatmål som er satt som krav), samtidig som det er ønskelig å bruke tid som blir til overs til å videreutvikle verktøyet til å oppfylle ønskelige behov (resultatmål som er satt som 'gjerne'). Beslutninger om hvor mye arbeid utover grunnfunksjonaliteten som skal implementeres må derfor gjøres underveis i prosjektet, etter hvert som det avdekkes hvor krevende de ulike resultatmålene viser seg å være. Her vil også innspill og prioriteringer fra oppdragsgiver kunne endre seg i takt med at verktøyet blir utviklet.

Prosjektgruppen er forholdsvis liten, og det er naturlig å anta at det kommer til å være et stort behov for fleksibilitet underveis i prosjektet, etterhvert som oppgaver kan vise seg å være mer, eller mindre, krevende enn originalt anslått.

6.2 Scrumban

På grunnlag av disse behovene har gruppen kommet frem til at det er viktig å velge en smidig utviklingsmodell. Gruppen er kjent med en rekke typer smidige utviklingsmodeller fra systemutviklings-faget [19], men har mest erfaring med scrum for litt mindre prosjekter med omtrentlig like store grupper som dette prosjektet. Oppdelingen av tid og arbeidsoppgaver utover sprinter som scrum tilbyr virket utmerket for å tilrettelegge for smidige overganger etter beslutningspunkter i prosjektet. Begrensningene i hvor lett man kan bytte eller overta oppgaver var det største problemet gruppen så ved en ren scrum-metode, og valgte derfor å kombinere dette med Kanban-modellen.

6.2.1 Kanban

Kanban er en simpel modell som lett kan kombineres med andre utviklingsmodeller. I essens er Kanban et system hvor ulike oppgaver deles opp og organiseres under ulike stadier på en tavle. Ulike begrensninger på hvor mange kort/oppgaver som kan være i et gitt stadie på tavla tilbyr en enkel metode for å forhindre at gruppelemmer tar på seg for mange oppgaver om gangen. Kanban-tavlen gir en god visualisering av arbeidsoppgavene, viser tydelig fremgang for alle gruppens medlemmer og kan bidra til økt motivasjon ved å tydelig dele opp store arbeidsoppgaver til mindre fragmenter. For å få skikkelig utbytte av Kanban er det viktig at det brukes jevnlig gjennom hele prosjektet, og bruken må derfor følges opp av gruppeleder underveis. Trello [20] har et tilbud som tilsvarer en Kanban-tavle gratis på nett, og er implementasjonen av Kanban gruppen kommer til å benytte i prosjektet.

6.2.2 Oppdeling i sprinter

Gruppen har valgt å dele opp hovedprosjektet utelukkende i én-ukers sprinter, hvor hver sprint startes på samme dag som det ukentlige veiledningsmøtet. Dette gir gruppa tid i forkant av veiledningsmøtet til å planlegge ukas sprint, og deretter få veiledning til arbeidet som skal gjøres i løpet av sprinten.

6.2.3 Sprint planning meeting

Hver sprint skal starte med et planleggingsmøte (sprint planning meeting), hvor en eventuell sprint backlog fra forrige sprint innarbeides i neste sprint, sammen med nye oppgaver. Oppgavene deles opp i mindre biter, kategoriseres og legges på Trello. Det blir skrevet referat av hvert planleggingsmøte.

6.2.4 Sprint review meeting

Ved slutten av hver sprint møtes gruppen for et sprintrefleksjons-møte (sprint review meeting). Her skal gruppa gjøre en vurdering på hvordan fremgangen er i forhold til forventet, og hva som skal gjøres i neste sprint. Under dette møtet blir det vurdert om gruppen står i fare for å havne bak skjema, og eventuelle tiltak mot dette diskuteres og iverksettes. Oppgaver som ikke ble fullført samles i en sprint backlog av dokumentasjonsansvarlig.

6.3 Grupperoller

Til tross for at gruppen er liten, vurderes det som hensiktsmessig at ansvarsområder for ulike områder fordeles allerede før hovedprosjektet startes. Dette kommer til å gjøre fordeling av arbeidsoppgaver for hver sprint enklere, og avgjørelser som må tas ved ulike veiskiller får en naturlig beslutningstaker. Grovt sett kommer de som er ansvarlige for sine områder også til å ha mer fokus på å jobbe med disse, men skal også sørge for at man får hjelp med å oppnå sine ukentlige sprintmål fra andre gruppe-medlemmer dersom det er nødvendig.

Gruppeleder har ansvar for at gruppe-regler overholdes, eventuelt tiltak om de ikke gjør det, og tar seg av kommunikasjon med eksterne parter involvert i prosjektet.

Dokumentasjonsansvarlig har det generelle ansvaret for skriving av forprosjekt, prosjektrapport og annen dokumentasjon som utarbeides i løpet av prosjektperioden. Dette innebærer også å påse at nødvendig rapportering og dokumentasjon i løpet av utviklingsprosessen(e) gjennomføres.

Ansvarlig for brukergrensesnitt og design har ansvaret for utforming og utvikling av brukergrensesnittet. Denne personen får også ansvaret for å bringe oppdragsgiver inn for input og innvendinger underveis i utviklingsprosessen for å sikre et så brukervennlig godt brukergrensesnitt som mulig.

Utviklingsansvarlig skal stå for utviklingen av backend-delen av det endelige verktøyet. Ettersom det er denne delen av utviklingen som har største usikkerhetsmomenter er det spesielt viktig at denne personen sørger for å hente inn hjelp fra andre gruppe-medlemmer så snart det viser seg å være nødvendig.

Interne roller:

- Gruppeleder: Thomas Faack Bjørnstad
- Dokumentasjonsansvarlig: Vebjørn Sundli Alseth
- Ansvarlig for brukergrensesnitt og design: Alex Rogatskiy
- Utviklingsansvarlig: Thomas Faack Bjørnstad

Eksterne roller:

- Oppdragsgivers kontaktperson: Knut Marius Myrvold
- Veileder: Marius Pedersen

6.4 Milepæler

Forprosjekt (01.02.2020)

Forprosjekt inkluderer prosjektplan, prosjektavtale og gruppeeregler.

Fungerende prototype (23.03.2020)

Ha ferdigstilt en fungerende prototype som skal kunne utføre den grunnleggende funksjonaliteten avtalt med NINA.

Førsteutkast rapport (12.04.2020)

Førsteutkast av prosjektrapport til påsken for å sikre at vi ligger godt an.

Andreutkast rapport (04.05.2020)

Andreutkast vil også bli levert til NINA for å få tilbakemeldinger før endelig innlevering.

Endelig utvikling av verktøy (04.05.2020)

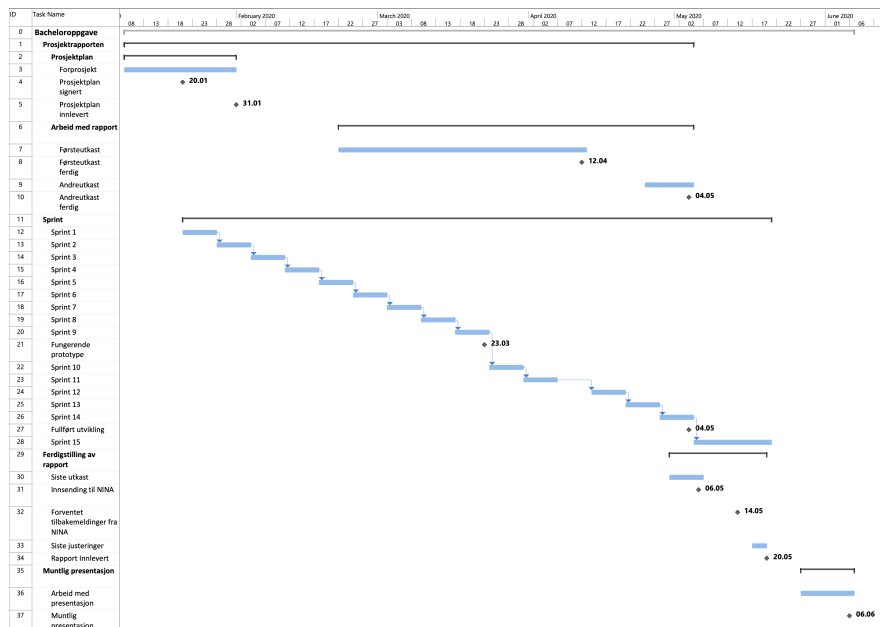
Utvikling skal være fullført og alle resultatmål være oppfylt.

Innlevering av rapport (20.05.2020)

Frist for innlevering av rapporten.

Presentasjon (06.06.2020)

Muntlig presentasjon av prosjektet.



Figur 6.1: Gantt-diagram over prosjektet.

Bibliografi

- [1] *BIDAT39 - Bacheloroppgave - Dataingeniør*, <https://www.ntnu.no/studier/emner/BIDAT39>, besøkt 24.01.2020.
- [2] *Norsk institutt for naturforskning*, <https://www.nina.no/>, besøkt 24.01.2020.
- [3] *Laksetrapp*, <https://snl.no/laksetrapp>, besøkt 30.01.2020.
- [4] *SafePass*, <https://www.cedren.no/Prosjekter/SafePass>, besøkt 30.01.2020.
- [5] *Smart ombygging av fisketrapp skal gjøre det lettere for å fisken å vandre i Glomma*, <https://www.cedren.no/Nyheter/Article/ArticleId/4128/Smart-ombygging-av-fisketrapp-skal-gjore-det-lettere-for-a-fisken-a-vandre-i-Glomma>, besøkt 30.01.2020.
- [6] S. R. Yoshida, *Computer Vision*. Hauppauge, N.Y.: Nova Science Publishers, Inc., 2011.
- [7] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost og N. Wiebe, *Quantum machine learning*. London: Nature Publishing Group, 2017.
- [8] K. Srinath, «Python–The Fastest Growing Programming Language», *International Research Journal of Engineering and Technology (IRJET)*, årg. 4, nr. 12, s. 354–357, 2017.
- [9] *Fisketelling_Hoyegga_2017.xlsx*, Dokument utlevert av NINA under møte 29.01.2020, inneholder informasjon fra manuell telling av alt videomateriale fra 2017.
- [10] *Spesifikasjoner på laptop hos NINA*, Informasjon overlevert muntlig under møte 29.01.2020.
- [11] *WCAG 2.0-standarden*, <https://uu.difi.no/krav-og-regelverk/wcag-20-standarden>, besøkt 27.01.2020.
- [12] *VLC: Official Site*, <https://www.videolan.org/>, besøkt 17.01.2020.
- [13] *Colour Contrast Analyser*, <https://developer.paciellogroup.com/resources/contrastanalyser/>, besøkt 30.01.2020.
- [14] *Hvordan teste universell utforming av ditt nettsted*, <https://uu.difi.no/krav-og-regelverk/kom-i-gang/hvordan-teste-universell-utforming-av-ditt-nettsted>, besøkt 31.01.2020.

- [15] *Maler og standarder - NTNU*, <http://aitel.hist.no/fag/maler-standarder/fossefall/forstudiet.html>, besøkt 20.01.2020.
- [16] *Overleaf, online LaTeX editor*, <https://www.overleaf.com/>, besøkt 17.01.2020.
- [17] *Git*, <https://git-scm.com/>, besøkt 31.01.2020.
- [18] *BitBucket*, <https://bitbucket.org/>, besøkt 17.01.2020.
- [19] *IMT2243 - Systemutvikling*, <https://www.ntnu.no/studier/emner/IMT2243/2016#tab=omEmnet>, besøkt 27.01.2020.
- [20] *Trello*, <https://trello.com/>, besøkt 17.01.2020.

VEDLEGG

Prosjektavtale

mellom NTNU Fakultet for informasjonsteknologi og elektroteknikk (IE) på Gjøvik (utdanningsinstitusjon), og

NORSK INSTITUTT FOR NATURFORSKNING

NINA

(oppdragsgiver), og

Alex Rogatskiy, Thomas Faack Bjørnstad,

Vebjørn Sundli Alseth

(student(er))

Avtalen angir avtalepartenes plikter vedrørende gjennomføring av prosjektet og rettigheter til anvendelse av de resultater som prosjektet frembringer:

1. Studenten(e) skal gjennomføre prosjektet i perioden fra 09/01/20 til 20/05/20.

Studentene skal i denne perioden følge en oppsatt fremdriftsplan der NTNU IE på Gjøvik yter veiledning. Oppdragsgiver yter avtalt prosjektbistand til fastsatte tider. Oppdragsgiver stiller til rådighet kunnskap og materiale som er nødvendig for å få gjennomført prosjektet. Det forutsettes at de gitte problemstillinger det arbeides med er aktuelle og på et nivå tilpasset studentenes faglige kunnskaper. Oppdragsgiver plikter på forespørsel fra NTNU å gi en vurdering av prosjektet vederlagsfritt.

2. Kostnadene ved gjennomføringen av prosjektet dekkes på følgende måte:
 - Oppdragsgiver dekker selv gjennomføring av prosjektet når det gjelder f.eks. materiell, telefon, reiser og nødvendig overnatting på steder langt fra NTNU i Gjøvik. Studentene dekker utgifter for ferdigstillelse av prosjektmateriell.
 - Eiendomsretten til eventuell prototyp tilfaller den som har betalt komponenter og materiell mv. som er brukt til prototypen. Dersom det er nødvendig med større og/eller spesielle investeringer for å få gjennomført prosjektet, må det gjøres en egen avtale mellom partene om eventuell kostnadsfordeling og eiendomsrett.
3. NTNU IE på Gjøvik står ikke som garantist for at det oppdragsgiver har bestilt fungerer etter hensikten, ei heller at prosjektet blir fullført. Prosjektet må anses som en eksamensrelatert oppgave som blir bedømt av intern og ekstern sensor. Likevel er det en forpliktelse for utøverne av prosjektet å fullføre dette til avtalte spesifikasjoner, funksjonsnivå og tider.

4. Alle bacheloroppgaver som ikke er klausulert og hvor forfatteren(e) har gitt sitt samtykke til publisering, kan gjøres tilgjengelig via NTNUs institusjonelle arkiv hvis de har skriftlig karakter A, B eller C.

Tilgjengeliggjøring i det åpne arkivet forutsetter avtale om delvis overdragelse av opphavsrett, se «avtale om publisering» (jfr Lov om opphavsrett). Oppdragsgiver og veileder godtar slik offentliggjøring når de signerer denne prosjektavtalen, og må evt. gi skriftlig melding til studenter og instituttleder/fagenhetsleder om de i løpet av prosjektet endrer syn på slik offentliggjøring.

Den totale besvarelsen med tegninger, modeller og apparatur så vel som programlisting, kildekode mv. som inngår som del av eller vedlegg til besvarelsen, kan vederlagsfritt benyttes til undervisnings- og forskningsformål. Besvarelsen, eller vedlegg til den, må ikke nyttes av NTNU til andre formål, og ikke overlates til utenforstående uten etter avtale med de øvrige parter i denne avtalen. Dette gjelder også firmaer hvor ansatte ved NTNU og/eller studenter har interesser.

5. Besvarelsens spesifikasjoner og resultat kan anvendes i oppdragsgivers egen virksomhet. Gjør studenten(e) i sin besvarelse, eller under arbeidet med den, en patentbar oppfinnelse, gjelder i forholdet mellom oppdragsgiver og student(er) bestemmelsene i Lov om retten til oppfinnelser av 17. april 1970, §§ 4-10.
6. Ut over den offentliggjøring som er nevnt i punkt 4 har studenten(e) ikke rett til å publisere sin besvarelse, det være seg helt eller delvis eller som del i annet arbeide, uten samtykke fra oppdragsgiver. Tilsvarende samtykke må foreligge i forholdet mellom student(er) og faglærer/veileder for det materialet som faglærer/veileder stiller til disposisjon.
7. Studenten(e) leverer oppgavebesvarelsen med vedlegg (pdf) i NTNUs elektroniske eksamenssystem. I tillegg leveres ett eksemplar til oppdragsgiver.
8. Denne avtalen utferdiges med ett eksemplar til hver av partene. På vegne av NTNU, IE er det instituttleder/faggruppeleder som godkjenner avtalen.
9. I det enkelte tilfelle kan det inngås egen avtale mellom oppdragsgiver, student(er) og NTNU som regulerer nærmere forhold vedrørende bl.a. eiendomsrett, videre bruk, konfidensialitet, kostnadsdekning og økonomisk utnyttelse av resultatene. Dersom oppdragsgiver og student(er) ønsker en videre eller ny avtale med oppdragsgiver, skjer dette uten NTNU som partner.
10. Når NTNU også opptrer som oppdragsgiver, trer NTNU inn i kontrakten både som utdanningsinstitusjon og som oppdragsgiver.
11. Eventuell uenighet vedrørende forståelse av denne avtale løses ved forhandlinger avtalepartene imellom. Dersom det ikke oppnås enighet, er partene enige om at tvisten løses av voldgift, etter bestemmelsene i tvistemålsloven av 13.8.1915 nr. 6, kapittel 32.

12. Deltakende personer ved prosjektgjennomføringen:

NTNUs veileder (navn): Marius Pedersen

Oppdragsgivers kontaktperson (navn): Knut Marius Myrvold

Student(er) (signatur): Alex Rozalskiy dato 09/01/20

Thomas Børstøl dato 09/01/20

Vehem Aketi dato 09/01/20

_____ dato _____

Oppdragsgiver (signatur): UM Myrvold dato 17/1-20

Signert avtale leveres digitalt i Blackboard, rom for bacheloroppgaven.

Godkjennes digitalt av instituttleder/faggruppeleder.

Om papirversjon med signatur er ønskelig, må papirversjon leveres til instituttet i tillegg.

Plass for evt sign:

Instituttleder/faggruppeleder (signatur): Man Pedersen dato 15/1/20

Grupperegler

- Hvert gruppemedlem skal bidra med 30 timer totalt per uke. Hvis dette kravet ikke oppfylles, skal gruppemedlemmet jobbe inn denne tiden senere i.a. prosjektet.
- Forventes at minimum 45 minutter av hver arbeidstime skal gå til ren effektiv jobbing.
- Forventes at hvert medlem møter til fastsatt tid, dette inkluderer da gruppemøter, møter med veileder og annet oppsatt arbeid. Skulle noe forekomme, så forventes det er medlemmet sier i fra til resten av gruppen så fort som mulig. Ikke meldt for-sent-komming bøtelegges med tiden som personen har kommet for sent, ganger 2. Denne tiden blir opptil vedkommende å fordele.
- Gruppemedlemmer pålegges å stille med oppdatert statusinformasjon for ukens avsluttende sprintmøte.
- Ved interne uenigheter eller store beslutninger i gruppa kan det kalles inn til demokratisk avstemning. Dersom dette forekommer, er alle gruppemedlemmer pålagt å stemme. Ved betydelige uenigheter tillates nedstemte gruppemedlemmer å ta opp saken med veileder.
- Dersom et gruppemedlem føler at han ikke ligger i rute til å bli ferdig med sprintens arbeidsoppgave(r) skal han informere resten av gruppa så tidlig som mulig, og be om assistanse dersom dette lar seg gjøre.
- Konsekvenser for regelbrudd vil i første omgang være en advarsel. Ved gjentatte regelbrudd vil saken bli tatt opp med veileder.
 - Ved mange gjentatte regelbrudd og samtale med veileder, så kan ekskludering finne sted. NB: Ingen ekskludering de 28 siste dagene før innleveringsfristen.

Underskrift: Alex Rogat sluy

Underskrift: Thomas Jøndal

Underskrift: Vehem Alseth

Fisketelling Høyegga

Observasjon	ID	Date	Time	Sortname	Direction	Disk	Filnavn	Første gang	Andre tidspunkt av samme individ	Sikt
1	Harr170505_1	05/05/2017	14:34 - 15:04	Harr	Opp	1	File1-[2017-05-04_13-34-27]-000	16:10		Dårlig
2	Harr170505_2	05/05/2017	15:04 - 15:34	Harr	Kam	1	File1-[2017-05-04_13-34-27]-001	01:44	01:49; 02:12; 02:20	Dårlig
3	Fisk170505_1	05/05/2017	15:04 - 15:34	Fish	Kam	1	File1-[2017-05-04_13-34-27]-001	07:04	08:37; 08:54	Dårlig
4	Harr170505_3	05/05/2017	15:04 - 15:34	Harr	Kam	1	File1-[2017-05-04_13-34-27]-001	09:00	09:24; 09:49; 10:01; 11:01; 11:26; 11:51	Dårlig
5	Harr170505_4	05/05/2017	15:04 - 15:34	Harr	Kam	1	File1-[2017-05-04_13-34-27]-001	09:00	09:24; 09:49; 10:01; 11:01; 11:26; 11:51	Dårlig
6	Fisk170505_2	05/05/2017	15:04 - 15:34	Fish	Kam	1	File1-[2017-05-04_13-34-27]-001	17:47		Dårlig
7	Harr170505_5	05/05/2017	15:04 - 15:34	Harr	Opp	1	File1-[2017-05-04_13-34-27]-001	23:23		Dårlig
8	Harr170505_5	05/05/2017	15:04 - 15:34	Harr	Kam	1	File1-[2017-05-04_13-34-27]-001	25:09		Dårlig
9	Harr170505_6	05/05/2017	15:34 - 16:04	Harr	Opp	1	File1-[2017-05-04_13-34-27]-002	00:39		Dårlig
10	Harr170505_7	05/05/2017	15:34 - 16:04	Harr	Kam	1	File1-[2017-05-04_13-34-27]-002	15:26		Dårlig
11	Harr170505_8	05/05/2017	15:34 - 16:04	Harr	Kam	1	File1-[2017-05-04_13-34-27]-002	18:25	18:39	Dårlig
12	Harr170505_9	05/05/2017	16:04 - 16:34	Harr	Opp	1	File1-[2017-05-04_13-34-27]-003	00:48		Dårlig
13	Harr170505_10	05/05/2017	16:04 - 16:34	Harr	Kam	1	File1-[2017-05-04_13-34-27]-003	02:27	02:39; 02:55	Dårlig
14	Harr170505_11	05/05/2017	16:04 - 16:34	Harr	Opp	1	File1-[2017-05-04_13-34-27]-003	09:12		Dårlig
15	Harr170508_1	08/05/2017	13:05 - 13:35	Harr	Kam	1	File1-[2017-05-04_13-34-27]-141	01:15	01:51; 02:03	Dårlig
16	Harr170508_2	08/05/2017	13:05 - 13:35	Harr	Kam	1	File1-[2017-05-04_13-34-27]-141	03:59		Dårlig
17	Harr170508_2	08/05/2017	13:05 - 13:35	Harr	Kam	1	File1-[2017-05-04_13-34-27]-141	05:47	06:45	Dårlig
18	Harr170508_3	08/05/2017	13:05 - 13:35	Harr	Ned	1	File1-[2017-05-04_13-34-27]-141	08:14		Dårlig
19	Harr170508_4	08/05/2017	13:05 - 13:35	Harr	Kam	1	File1-[2017-05-04_13-34-27]-141	09:39	10:10; 11:11;	Dårlig
20	Harr170508_4	08/05/2017	13:05 - 13:35	Harr	Kam	1	File1-[2017-05-04_13-34-27]-141	13:29	13:53	Dårlig
21	Harr170508_4	08/05/2017	13:05 - 13:35	Harr	Kam	1	File1-[2017-05-04_13-34-27]-141	18:15	18:34; 19:29; 20:34; 23:09	Dårlig
22	Harr170508_5	08/05/2017	13:05 - 13:35	Harr	Opp	1	File1-[2017-05-04_13-34-27]-141	23:42		Dårlig
23	Harr170508_5	08/05/2017	13:05 - 13:35	Harr	Kam	1	File1-[2017-05-04_13-34-27]-141	24:44	25:31; 26:01; 28:51	Dårlig
26	Harr170818_1	18/08/2017	12:47 - 13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-000	04:09		God
27	Harr170818_2	18/08/2017	12:47 - 13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-000	04:49	05:10	God
28	Harr170818_1	18/08/2017	12:47 - 13:17	Harr	Kam	1	File1-[2017-08-18_11-46-59]-000	05:21		God
29	Harr170818_2	18/08/2017	12:47 - 13:17	Harr	Kam	1	File1-[2017-08-18_11-46-59]-000	05:21		God
30	Ørret170818_1	18/08/2017	12:47 - 13:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-000	05:47	05:58	God
31	Ørret170818_1	18/08/2017	12:47 - 13:17	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-000	06:11		God
32	Harr170818_1	18/08/2017	12:47 - 13:17	Harr	Kam	1	File1-[2017-08-18_11-46-59]-000	06:18	06:30; 06:41	God
33	Harr170818_2	18/08/2017	12:47 - 13:17	Harr	Kam	1	File1-[2017-08-18_11-46-59]-000	06:18	06:30; 06:41	God
34	Ørret170818_2	18/08/2017	12:47 - 13:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-000	17:49	20:06	God
35	Ørret170818_3	18/08/2017	13:17 - 13:47	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-001	02:20		God
36	Sik170818_1	18/08/2017	13:17 - 13:47	Sik	Kam	1	File1-[2017-08-18_11-46-59]-001	07:29	08:13	God
37	Fisk170818_1	18/08/2017	13:47 - 14:17	Fish	Opp	1	File1-[2017-08-18_11-46-59]-002	03:26	03:29	God
38	Ørret170818_2	18/08/2017	13:47 - 14:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-002	13:21		God
39	Ørret170818_4	18/08/2017	14:17 - 14:47	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-003	11:10		God
40	Ørret170818_5	18/08/2017	14:17 - 14:47	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-003	12:25		God
41	Ørret170818_5	18/08/2017	14:17 - 14:47	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-003	12:52		God
42	Ørret170818_6	18/08/2017	14:17 - 14:47	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-003	16:38		God
43	Ørret170818_2	18/08/2017	14:17 - 14:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-003	22:50	23:13	God
44	Ørret170818_2	18/08/2017	14:47 - 15:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-004	13:13	21:44; 22:38; 25:52; 28:12; 29:25	God
45	Ørret170818_7	18/08/2017	15:17 - 15:47	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-005	03:33		God
46	Ørret170818_2	18/08/2017	15:17 - 15:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-005	05:28	09:05; 10:29; 12:49; 20:09; 26:53; 29:36	God
47	Ørret170818_8	18/08/2017	15:17 - 15:47	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-005	28:18		God
48	Ørret170818_2	18/08/2017	15:47 - 16:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-006	07:31	13:06; 14:29; 25:47; 28:38; 29:35	God
49	Ørret170818_9	18/08/2017	16:17 - 16:47	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-007	00:55		God
50	Notat_1	18/08/2017	16:17 - 16:47	Sik	Kam	1	File1-[2017-08-18_11-46-59]-007	12:24		God
51	Ørret170818_2	18/08/2017	16:17 - 16:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-007	15:32	19:32; 21:17; 24:29	God
52	Harr170818_3	18/08/2017	16:17 - 16:47	Harr	Opp	1	File1-[2017-08-18_11-46-59]-007	16:05		God
53	Ørret170818_2	18/08/2017	16:47 - 17:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-008	20:26		God
54	Sik170818_2	18/08/2017	16:47 - 17:17	Sik	Opp	1	File1-[2017-08-18_11-46-59]-008	24:26	25:42; 26:29;	God
55	Ørret170818_2	18/08/2017	17:17 - 17:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-009	08:34	19:38; 28:56;	God
56	Ørret170818_2	18/08/2017	17:47 - 18:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-010	11:27		God
57	Ørret170818_2	18/08/2017	18:17 - 18:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-011	16:35		God
58	Ørret170818_2	18/08/2017	18:47 - 19:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-012	06:11	11:06	God
59	Harr170818_4	18/08/2017	18:47 - 19:17	Harr	Kam	1	File1-[2017-08-18_11-46-59]-012	06:46		God
60	Harr170818_4	18/08/2017	18:47 - 19:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-012	06:57		God
61	Ørret170818_2	18/08/2017	19:17 - 19:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-013	00:59	06:46; 11:06; 13:27; 19:01	God
62	Ørret170818_10	18/08/2017	19:17 - 19:47	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-013	01:18	01:36	God
63	Ørret170818_2	18/08/2017	19:47 - 20:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-014	24:06	27:08	God
64	Ørret170818_2	18/08/2017	20:17 - 20:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-015	06:17	12:50; 20:28	Ok
65	Ørret170818_11	18/08/2017	20:47 - 21:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-016	03:53		Dårlig
66	Ørret170818_2	18/08/2017	20:47 - 21:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-016	04:04	13:42	Dårlig
67	Harr170818_5	18/08/2017	20:47 - 21:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-016	08:47		Dårlig
68	Fisk170819_1	19/08/2017	05:15 - 05:45	Fish	Kam	1	File1-[2017-08-18_11-46-59]-033	16:47		Dårlig
69	Ørret170819_1	19/08/2017	05:15 - 05:45	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-033	17:51		Dårlig
70	Lake170819_1	19/08/2017	05:15 - 05:45	Lake	Kam	1	File1-[2017-08-18_11-46-59]-033	20:08		Dårlig
71	Lake170819_1	19/08/2017	05:15 - 05:45	Lake	Ned	1	File1-[2017-08-18_11-46-59]-033	20:56		Dårlig
72	Lake170819_2	19/08/2017	05:47 - 06:17	Lake	Opp	1	File1-[2017-08-18_11-46-59]-034	06:00		Dårlig
73	Lake170819_3	19/08/2017	05:47 - 06:17	Lake	Opp	1	File1-[2017-08-18_11-46-59]-034	13:20		Dårlig
74	Lake170819_4	19/08/2017	06:17 - 06:47	Lake	Ned	1	File1-[2017-08-18_11-46-59]-035	04:51		Ok
75	Ørret170818_2	19/08/2017	07:47 - 08:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-038	28:08	29:12	God
76	Ørret170818_2	19/08/2017	08:17 - 08:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-039	04:13	14:03	God
77	Ørret170818_2	19/08/2017	08:47 - 09:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-040	13:51	15:12; 16:25; 24:56; 26:19	God
78	Ørret170819_2	19/08/2017	08:47 - 09:17	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-040	24:09		God
79	Ørret170818_2	19/08/2017	09:17 - 09:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-041	11:57	13:47; 24:35	God
80	Ørret170819_3	19/08/2017	09:47 - 10:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-042	19:06		God
81	Ørret170818_2	19/08/2017	09:47 - 10:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-042	20:22		God
82	Ørret170818_2	19/08/2017	10:17 - 10:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-043	02:54	05:10; 09:00; 26:20; 28:22;	God
83	Ørret170819_4	19/08/2017	10:17 - 10:47	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-043	21:05		God
84	Ørret170818_2	19/08/2017	10:47 - 11:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-044	06:16	16:26; 23:00	God
85	Ørret170819_5	19/08/2017	10:47 - 11:17	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-044	06:46		God
86	Ørret170818_2	19/08/2017	11:17 - 11:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-045	02:39	11:14; 27:36	God
87	Ørret170819_6	19/08/2017	11:17 - 11:47	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-045	08:01		God
88	Ørret170818_2	19/08/2017	11:47 - 12:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-046	10:11	11:41; 12:44; 23:33	God
89	Ørret170818_2	19/08/2017	12:17 - 12:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-047	03:19	09:55; 11:23	God
90	Ørret170818_2	19/08/2017	12:47 - 13:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-048	05:20	12:51; 14:12	God
91	Ørret170818_2	19/08/2017	13:17 - 13:47	Ørret	Kam	1	File1			

118	Ørret170818_2	19/08/2017 19:47-20:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-062	22:00	29:29	Ok
119	Ørret170818_2	19/08/2017 20:17-20:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-063	29:49		Dårlig
120	Sik170820_1	20/08/2017 08:17-08:47	Sik	Opp	1	File1-[2017-08-18_11-46-59]-087	04:24		Dårlig
121	Sik170820_1	20/08/2017 08:17-08:47	Sik	kam	1	File1-[2017-08-18_11-46-59]-087	05:28	06:49; 07:43; 11:26; 13:17;	Dårlig
122	Sik170820_2	20/08/2017 08:17-08:47	Sik	Opp	1	File1-[2017-08-18_11-46-59]-087	13:37	17:10	Dårlig
123	Ørret170820_1	20/08/2017 08:47-09:17	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-088	06:36		Dårlig
124	Harr170820_1	20/08/2017 08:47-09:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-088	09:11	09:45; 09:51; 12:41	Dårlig
125	Fish170820_1	20/08/2017 08:47-09:17	Fish	Opp	1	File1-[2017-08-18_11-46-59]-088	09:13		Dårlig
126	Ørret170820_2	20/08/2017 08:47-09:17	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-088	11:15		Dårlig
127	Harr170820_2	20/08/2017 08:47-09:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-088	11:19		Dårlig
128	Harr170820_1	20/08/2017 08:47-09:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-088	13:26		Dårlig
129	Harr170820_3	20/08/2017 08:47-09:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-088	14:32	15:09	Dårlig
130	Harr170820_4	20/08/2017 08:47-09:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-088	16:37		Dårlig
131	Sik170820_3	20/08/2017 09:17-09:47	Sik	Kam	1	File1-[2017-08-18_11-46-59]-089	02:41	26:09	Dårlig
132	Fish170820_2	20/08/2017 09:17-09:47	Fish	Opp	1	File1-[2017-08-18_11-46-59]-089	13:27		Dårlig
133	Ørret170818_2	20/08/2017 09:17-09:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-089	15:41	16:05; 16:58;	Dårlig
134	Ørret170818_2	20/08/2017 09:17-09:47	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-089	18:33		Dårlig
135	Sik170820_4	20/08/2017 09:47-10:17	Sik	Kam	1	File1-[2017-08-18_11-46-59]-090	2:29	4:13; 19:34	Dårlig
136	Harr170820_5	20/08/2017 10:47-11:17	Harr	Kam	1	File1-[2017-08-18_11-46-59]-092	26:46		Dårlig
137	Harr170820_5	20/08/2017 11:17-11:47	Harr	Kam	1	File1-[2017-08-18_11-46-59]-093	00:30	00:54; 1:03	Dårlig
138	Harr170820_5	20/08/2017 11:17-11:47	Harr	Ned	1	File1-[2017-08-18_11-46-59]-093	01:34		Dårlig
139	Harr170820_6	20/08/2017 11:17-11:47	Harr	Opp	1	File1-[2017-08-18_11-46-59]-093	02:51		Dårlig
140	Fish170820_3	20/08/2017 11:17-11:47	Fish	Opp	1	File1-[2017-08-18_11-46-59]-093	04:07		Dårlig
141	Harr170820_7	20/08/2017 11:17-11:47	Harr	Opp	1	File1-[2017-08-18_11-46-59]-093	12:27		Dårlig
142	Ørret170820_3	20/08/2017 11:17-11:47	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-093	16:46		Dårlig
143	Harr170820_8	20/08/2017 11:47-12:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-094	00:46		Dårlig
144	Ørret170820_3	20/08/2017 11:47-12:17	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-094	05:34		Dårlig
145	Harr170820_9	20/08/2017 11:47-12:17	Harr	Kam	1	File1-[2017-08-18_11-46-59]-094	27:32	28:06	Dårlig
146	Harr170820_9	20/08/2017 12:17-12:47	Harr	Ned	1	File1-[2017-08-18_11-46-59]-095	00:10		Dårlig
147	Harr170820_10	20/08/2017 12:17-12:47	Harr	Kam	1	File1-[2017-08-18_11-46-59]-095	07:28		Dårlig
148	Harr170820_11	20/08/2017 12:17-12:47	Harr	Kam	1	File1-[2017-08-18_11-46-59]-095	10:35		Dårlig
149	Harr170820_12	20/08/2017 12:17-12:47	Harr	Opp	1	File1-[2017-08-18_11-46-59]-095	17:54		Dårlig
150	Harr170820_13	20/08/2017 12:17-12:47	Harr	Opp	1	File1-[2017-08-18_11-46-59]-095	27:44	28:26	Dårlig
151	Harr170820_14	20/08/2017 12:47-13:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-096	01:29		Dårlig
152	Harr170820_15	20/08/2017 12:47-13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-096	16:22	17:19	Dårlig
153	Harr170820_15	20/08/2017 12:47-13:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-096	18:20		Dårlig
154	Harr170820_16	20/08/2017 12:47-13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-096	21:22	22:29	Dårlig
155	Harr170820_17	20/08/2017 12:47-13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-096	21:22	22:29; 22:50; 23:46	Dårlig
156	Harr170820_18	20/08/2017 12:47-13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-096	21:23	22:29; 22:50; 23:46	Dårlig
157	Harr170820_19	20/08/2017 12:47-13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-096	21:27	22:29; 22:50; 23:46	Dårlig
158	Sik170820_5	20/08/2017 12:47-13:17	Sik	Opp	1	File1-[2017-08-18_11-46-59]-096	23:05	23:48	Dårlig
159	Harr170820_16	20/08/2017 12:47-13:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-096	25:20		Dårlig
160	Harr170820_17	20/08/2017 12:47-13:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-096	25:20		Dårlig
161	Harr170820_18	20/08/2017 12:47-13:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-096	25:20		Dårlig
162	Harr170820_19	20/08/2017 12:47-13:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-096	25:20		Dårlig
163	Harr170820_16	20/08/2017 12:47-13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-096	26:06		Dårlig
164	Harr170820_17	20/08/2017 12:47-13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-096	26:06		Dårlig
165	Harr170820_18	20/08/2017 12:47-13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-096	26:06		Dårlig
166	Harr170820_19	20/08/2017 12:47-13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-096	26:06		Dårlig
167	Harr170820_16	20/08/2017 12:47-13:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-096	27:17		Dårlig
168	Harr170820_17	20/08/2017 12:47-13:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-096	27:17		Dårlig
169	Harr170820_18	20/08/2017 12:47-13:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-096	27:17		Dårlig
170	Harr170820_19	20/08/2017 12:47-13:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-096	27:17		Dårlig
171	Harr170820_17	20/08/2017 12:47-13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-096	28:43		Dårlig
172	Harr170820_18	20/08/2017 12:47-13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-096	28:43		Dårlig
173	Harr170820_19	20/08/2017 12:47-13:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-096	28:43		Dårlig
174	Harr170820_20	20/08/2017 13:17-13:47	Harr	Ned	1	File1-[2017-08-18_11-46-59]-097	07:54		Dårlig
175	Ørret170820_4	20/08/2017 13:17-13:47	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-097	09:21		Dårlig
176	Harr170820_21	20/08/2017 13:17-13:47	Harr	Opp	1	File1-[2017-08-18_11-46-59]-097	09:29		Dårlig
177	Harr170820_21	20/08/2017 13:17-13:47	Harr	Ned	1	File1-[2017-08-18_11-46-59]-097	09:54		Dårlig
178	Harr170820_22	20/08/2017 13:17-13:47	Harr	Opp	1	File1-[2017-08-18_11-46-59]-097	20:53	21:31; 21:50; 22:50	Dårlig
179	Harr170820_22	20/08/2017 13:17-13:47	Harr	Ned	1	File1-[2017-08-18_11-46-59]-097	24:35		Dårlig
180	Harr170820_23	20/08/2017 13:47-14:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-098	08:01		Dårlig
181	Harr170820_24	20/08/2017 13:47-14:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-098	08:07		Dårlig
182	Harr170820_25	20/08/2017 13:47-14:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-098	08:07		Dårlig
183	Harr170820_23	20/08/2017 13:47-14:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-098	09:07		Dårlig
184	Harr170820_24	20/08/2017 13:47-14:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-098	09:07		Dårlig
185	Harr170820_25	20/08/2017 13:47-14:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-098	09:07		Dårlig
186	Harr170820_23	20/08/2017 13:47-14:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-098	09:50		Dårlig
187	Harr170820_24	20/08/2017 13:47-14:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-098	09:50		Dårlig
188	Harr170820_25	20/08/2017 13:47-14:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-098	09:50		Dårlig
189	Harr170820_26	20/08/2017 13:47-14:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-098	14:21		Dårlig
190	Harr170820_27	20/08/2017 13:47-14:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-098	14:21		Dårlig
191	Harr170820_28	20/08/2017 13:47-14:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-098	14:21		Dårlig
192	Harr170820_29	20/08/2017 13:47-14:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-098	14:21		Dårlig
193	Harr170820_30	20/08/2017 13:47-14:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-098	20:51		Dårlig
194	Harr170820_31	20/08/2017 13:47-14:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-098	20:54		Dårlig
195	Harr170820_30	20/08/2017 13:47-14:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-098	21:44		Dårlig
196	Harr170820_31	20/08/2017 13:47-14:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-098	21:44		Dårlig
197	Harr170820_32	20/08/2017 13:47-14:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-098	21:44		Dårlig
198	Harr170820_33	20/08/2017 13:47-14:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-098	21:44		Dårlig
199	Harr170820_34	20/08/2017 13:47-14:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-098	22:27		Dårlig
200	Harr170820_35	20/08/2017 13:47-14:17	Harr	Opp	1	File1-[2017-08-18_11-46-59]-098	23:17		Dårlig
201	Harr170820_36	20/08/2017 13:47-14:17	Harr	Ned	1	File1-[2017-08-18_11-46-59]-098	25:17		Dårlig
202	Ørret170820_5	20/08/2017 14:17-14:47	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-099	22:35	24:55; 28:09; 28:52;	Dårlig
203	Ørret170820_6	20/08/2017 14:17-14:47	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-099	24:20		Dårlig
204	Harr170820_37	20/08/2017 14:17-14:47	Harr	Opp	1	File1-[2017-08-18_11-46-59]-099	27:34		Dårlig
205	Harr170820_38	20/08/2017 14:17-14:47	Harr	Opp	1	File1-[2017-08-18_11-46-59]-099	27:34		Dårlig
206	Harr170820_39	20/08/2017 14:17-14:47	Harr	Opp	1	File1-[2017-08-18_11-46-59]-099	27:40		Dårlig
207	Fish170820_4	20/08/2017 14:17-14:47	Fish	Ned	1	File1-[2017-08-18_11-46-59]-099	29:16		Dårlig
208	Ørret170820_5	20/08/2017 14:47-15:17	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-100	00:00	02:44; 07:17; 14:59; 15:48;	Dårlig
209	Harr170820_40	20/08/2017 14:47-15:17	Harr	Kam	1	File1-[2017-08-18_11-46-59]-100	00:09	00:30; 02:44	Dårlig
210	Harr170820_41	20/08/2017 14:47-15:17	Harr	Kam	1	File1-[2017-08-18_11-46-59]-100	00:30	02:44	Dårlig
211	Ørret170820_6	20/08/2017 14:47-15:17	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-100	05:06		Dårlig
212	Fish170820_5	20/08/2017 14:47-15:17	Fish	Ned	1	File1-[2017-08-			

234	Ørret170818_2	21/08/2017 09:18-09:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-137	06:49		Ok
235	Ørret170821_4	21/08/2017 09:18-09:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-137	10:53	16:24	Ok
236	Ørret170821_4	21/08/2017 09:18-09:48	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-137	16:45		Ok
237	Harr170821_6	21/08/2017 09:48-10:18	Harr	Opp	1	File1-[2017-08-18_11-46-59]-138	03:22		Ok
238	Ørret170818_2	21/08/2017 09:48-10:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-138	04:52	14:20; 26:57; 29:22	Ok
239	Ørret170821_5	21/08/2017 09:48-10:18	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-138	19:42		Ok
240	Ørret170821_5	21/08/2017 09:48-10:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-138	22:09		Ok
241	Lake170821_2	21/08/2017 09:48-10:18	Lake	Ned	1	File1-[2017-08-18_11-46-59]-138	25:09		Ok
242	Ørret170821_6	21/08/2017 09:48-10:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-138	27:50		Ok
243	Ørret170821_7	21/08/2017 10:18-10:48	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-139	06:47		Ok
244	Ørret170818_2	21/08/2017 10:18-10:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-139	10:01	20:24	Ok
245	Ørret170821_8	21/08/2017 10:18-10:48	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-139	22:24	23:38	Ok
246	Ørret170818_2	21/08/2017 10:48-11:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-140	02:16		Ok
247	Ørret170821_9	21/08/2017 10:48-11:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-140	02:39		Ok
248	Harr170821_7	21/08/2017 10:48-11:18	Harr	Opp	1	File1-[2017-08-18_11-46-59]-140	09:50	10:43	Ok
249	Ørret170821_8	21/08/2017 10:48-11:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-140	10:19		Ok
250	Harr170821_7	21/08/2017 10:48-11:18	Harr	Ned	1	File1-[2017-08-18_11-46-59]-140	11:01		Ok
251	Ørret170821_10	21/08/2017 10:48-11:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-140	18:57		Ok
252	Ørret170821_11	21/08/2017 10:48-11:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-140	21:07		Ok
253	Ørret170821_12	21/08/2017 10:48-11:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-140	24:25		Ok
254	Harr170821_8	21/08/2017 11:18-11:48	Harr	Kam	1	File1-[2017-08-18_11-46-59]-141	03:25	06:40	Ok
255	Harr170821_8	21/08/2017 11:18-11:48	Harr	Ned	1	File1-[2017-08-18_11-46-59]-141	06:52		Ok
256	Ørret170821_13	21/08/2017 11:18-11:48	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-141	08:57		Ok
257	Harr170821_9	21/08/2017 11:48-12:18	Harr	Opp	1	File1-[2017-08-18_11-46-59]-142	14:41		Ok
258	Harr170821_9	21/08/2017 11:48-12:18	Harr	Ned	1	File1-[2017-08-18_11-46-59]-142	15:17		Ok
259	Harr170821_9	21/08/2017 11:48-12:18	Harr	Opp	1	File1-[2017-08-18_11-46-59]-142	17:36		Ok
260	Ørret170821_14	21/08/2017 11:48-12:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-142	27:15		Ok
261	Ørret170818_2	21/08/2017 12:18-12:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-143	13:31	29:09	Ok
262	Harr170821_10	21/08/2017 12:18-12:48	Harr	Kam	1	File1-[2017-08-18_11-46-59]-143	14:50		Ok
263	Harr170821_10	21/08/2017 12:18-12:48	Harr	Ned	1	File1-[2017-08-18_11-46-59]-143	15:23		Ok
264	Harr170821_11	21/08/2017 12:18-12:48	Harr	Kam	1	File1-[2017-08-18_11-46-59]-143	16:37		Ok
265	Ørret170821_14	21/08/2017 12:18-12:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-143	19:54	22:40	Ok
266	Ørret170821_14	21/08/2017 12:48-13:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-144	15:17		Ok
267	Ørret170821_14	21/08/2017 12:48-13:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-144	17:14		Ok
268	Notat_4	21/08/2017 13:18-13:48	Ørret			File1-[2017-08-18_11-46-59]-145	02:49		Ok
269	Ørret170821_15	21/08/2017 13:18-13:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-145	06:08		Ok
270	Lake170821_3	21/08/2017 13:18-13:48	Lake	Opp	1	File1-[2017-08-18_11-46-59]-145	07:40	08:00	Ok
271	Ørret170818_2	21/08/2017 13:18-13:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-145	19:03	22:37; 26:07; 29:56	Ok
272	Notat_5	21/08/2017 13:18-13:48	Ørret			File1-[2017-08-18_11-46-59]-145	28:05		Ok
273	Harr170821_12	21/08/2017 13:18-13:48	Harr	Opp	1	File1-[2017-08-18_11-46-59]-145	25:30	25:54	Ok
274	Harr170821_13	21/08/2017 13:18-13:48	Harr	Opp	1	File1-[2017-08-18_11-46-59]-145	25:31	25:54	Ok
275	Notat_6	21/08/2017 13:18-13:48	Harr			File1-[2017-08-18_11-46-59]-145	26:35		Ok
276	Harr170821_13	21/08/2017 13:18-13:48	Harr	Ned	1	File1-[2017-08-18_11-46-59]-145	26:39		Ok
277	Harr170821_13	21/08/2017 13:18-13:48	Harr	Opp	1	File1-[2017-08-18_11-46-59]-145	29:59		Ok
278	Harr170821_13	21/08/2017 13:48-14:18	Harr	Ned	1	File1-[2017-08-18_11-46-59]-146	00:19		Ok
279	Harr170821_13	21/08/2017 13:48-14:18	Harr	Opp	1	File1-[2017-08-18_11-46-59]-146	01:43		Ok
280	Ørret170818_2	21/08/2017 13:48-14:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-146	02:06	18:48; 19:40; 20:22	Ok
281	Harr170821_14	21/08/2017 13:48-14:18	Harr	Opp	1	File1-[2017-08-18_11-46-59]-146	17:36		Ok
282	Harr170821_15	21/08/2017 13:48-14:18	Harr	Opp	1	File1-[2017-08-18_11-46-59]-146	17:39		Ok
283	Ørret170818_2	21/08/2017 13:48-14:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-146	21:48		Ok
284	Harr170821_16	21/08/2017 13:48-14:18	Harr	Kam	1	File1-[2017-08-18_11-46-59]-146	29:35		Ok
285	Ørret170821_16	21/08/2017 14:18-14:48	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-147	06:60	08:08; 10:35	Ok
286	Harr170821_17	21/08/2017 14:18-14:48	Harr	Opp	1	File1-[2017-08-18_11-46-59]-147	11:54		Ok
287	Harr170821_18	21/08/2017 14:18-14:48	Harr	Opp	1	File1-[2017-08-18_11-46-59]-147	11:56		Ok
288	Harr170821_19	21/08/2017 14:18-14:48	Harr	Opp	1	File1-[2017-08-18_11-46-59]-147	12:40	13:43	Ok
289	Ørret170818_2	21/08/2017 14:18-14:48	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-147	25:43	28:54; 29:48;	Ok
290	Ørret170818_2	21/08/2017 14:48-15:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-148	03:12	09:27; 20:01; 29:47	Ok
291	Harr170821_19	21/08/2017 14:48-15:18	Harr	Kam	1	File1-[2017-08-18_11-46-59]-148	10:08	11:02	Ok
292	Harr170821_19	21/08/2017 14:48-15:18	Harr	ned	1	File1-[2017-08-18_11-46-59]-148	11:22		Ok
293	Harr170821_19	21/08/2017 14:48-15:18	Harr	Opp	1	File1-[2017-08-18_11-46-59]-148	11:56		Ok
294	Harr170821_19	21/08/2017 14:48-15:18	Harr	Ned	1	File1-[2017-08-18_11-46-59]-148	12:22		Ok
295	Ørret170821_17	21/08/2017 14:48-15:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-148	15:45	18:47	Ok
296	Harr170821_20	21/08/2017 14:48-15:18	Harr	Opp	1	File1-[2017-08-18_11-46-59]-148	22:57	24:07; 24:15	Ok
297	Harr170821_21	21/08/2017 14:48-15:18	Harr	Kam	1	File1-[2017-08-18_11-46-59]-148	25:02	25:51	Ok
298	Harr170821_22	21/08/2017 14:48-15:18	Harr	Kam	1	File1-[2017-08-18_11-46-59]-148	25:02	25:51	Ok
299	Harr170821_23	21/08/2017 14:48-15:18	Harr	Kam	1	File1-[2017-08-18_11-46-59]-148	25:02	25:51	Ok
300	Harr170821_21	21/08/2017 14:48-15:18	Harr	Ned	1	File1-[2017-08-18_11-46-59]-148	25:02	25:51	Ok
301	Harr170821_22	21/08/2017 14:48-15:18	Harr	Ned	1	File1-[2017-08-18_11-46-59]-148	25:02	25:51	Ok
302	Harr170821_23	21/08/2017 14:48-15:18	Harr	Ned	1	File1-[2017-08-18_11-46-59]-148	25:02	25:51	Ok
303	Harr170821_24	21/08/2017 14:48-15:18	Harr	Kam	1	File1-[2017-08-18_11-46-59]-148	27:03		Ok
304	Fish170821_2	21/08/2017 14:48-15:18	Fish	Opp	1	File1-[2017-08-18_11-46-59]-148	28:57		Ok
305	Ørret170818_2	21/08/2017 15:18-15:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-149	18:50	29:46	Ok
306	Ørret170821_18	21/08/2017 15:18-15:48	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-149	24:52		Ok
307	Ørret170818_2	21/08/2017 15:48-16:18	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-150	01:44		Ok
308	Ørret170818_2	21/08/2017 15:48-16:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-150	02:00		Ok
309	Ørret170821_19	21/08/2017 15:48-16:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-150	05:22		Ok
310	Ørret170821_19	21/08/2017 15:48-16:18	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-150	09:10		Ok
311	Ørret170821_20	21/08/2017 15:48-16:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-150	26:19		Ok
312	Ørret170818_2	21/08/2017 15:48-16:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-150	26:26		Ok
313	Notat_7	21/08/2017 16:18-16:48				File1-[2017-08-18_11-46-59]-151	04:46		Ok
314	Ørret170818_2	21/08/2017 16:18-16:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-151	13:30	15:55; 21:14; 22:50; 27:41	Ok
315	Sik170821_2	21/08/2017 16:18-16:48	Sik	Kam	1	File1-[2017-08-18_11-46-59]-151	13:50	18:02	Ok
316	Harr170821_25	21/08/2017 16:18-16:48	Harr	Opp	1	File1-[2017-08-18_11-46-59]-151	20:36		Ok
317	Ørret170818_2	21/08/2017 16:48-17:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-152	00:12	06:12	Ok
318	Ørret170821_21	21/08/2017 16:48-17:18	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-152	03:25		Ok
319	Harr170821_26	21/08/2017 16:48-17:18	Harr	Kam	1	File1-[2017-08-18_11-46-59]-152	04:40		Ok
320	Ørret170818_2	21/08/2017 17:18-17:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-153	01:55	02:48; 03:47; 09:53; 14:58; 21:38	Ok
321	Notat_8	21/08/2017 17:18-17:48				File1-[2017-08-18_11-46-59]-153	05:42		Ok
322	Ørret170821_22	21/08/2017 17:18-17:48	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-153	09:26	09:52	Ok
323	Harr170821_27	21/08/2017 17:48-18:18	Harr	Kam	1	File1-[2017-08-18_11-46-59]-154	00:49		Ok
324	Harr170821_27	21/08/2017 17:48-18:18	Harr	Ned	1	File1-[2017-08-18_11-46-59]-154	01:06		Ok
325	Ørret170818_2	21/08/2017 17:48-18:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-154	07:59	13:26; 24:19; 28:14	Ok
326	Ørret170818_2	21/08/2017 18:18-18:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-155	04:44	11:30; 17:27	Ok
327	Ørret170821_23	21/08/2017 18:48-19:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-156	02:37		Ok
328	Ørret170821_24	21/08/2017 18:48-19:18	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-156	03:45		Ok

350	Ørret170818_2	22/08/2017 09:48-10:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-186	01:25	24:51	God
351	Ørret170818_2	22/08/2017 10:48-11:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-188	09:33	13:19	God
352	Ørret170822_3	22/08/2017 10:48-11:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-188	16:14		God
353	Ørret170818_2	22/08/2017 11:18-11:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-189	06:21	20:05	God
354	Ørret170822_3	22/08/2017 11:48-12:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-190	07:47	13:24	God
355	Ørret170822_3	22/08/2017 11:48-12:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-190	15:43		God
356	Ørret170822_4	22/08/2017 11:48-12:18	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-190	23:48		God
357	Ørret170818_2	22/08/2017 11:48-12:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-190	25:13	25:29;26:10	God
358	Ørret170822_5	22/08/2017 11:48-12:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-190	25:29		God
359	Ørret170822_6	22/08/2017 11:48-12:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-190	26:10		God
360	Ørret170818_2	22/08/2017 12:18-12:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-191	05:18	19:29;27:08;29:28	God
361	Harr170822_1	22/08/2017 12:18-12:48	Harr	Opp	1	File1-[2017-08-18_11-46-59]-191	22:22	23:46	God
362	Harr170822_2	22/08/2017 12:18-12:48	Harr	Opp	1	File1-[2017-08-18_11-46-59]-191	22:22	23:46	God
363	Harr170822_3	22/08/2017 12:18-12:48	Harr	Kam	1	File1-[2017-08-18_11-46-59]-191	23:48		God
364	Ørret170818_2	22/08/2017 12:48-13:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-192	01:46	26:11	God
365	Harr170822_4	22/08/2017 12:48-13:18	Harr	Opp	1	File1-[2017-08-18_11-46-59]-192	21:28	22:45;23:10	God
366	Harr170822_4	22/08/2017 12:48-13:18	Harr	Ned	1	File1-[2017-08-18_11-46-59]-192	23:12		God
367	Harr170822_4	22/08/2017 12:48-13:18	Harr	Opp	1	File1-[2017-08-18_11-46-59]-192	24:50		God
368	Ørret170818_2	22/08/2017 13:18-13:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-193	04:24	05:38;06:17;24:12	God
369	Notat170822_1	22/08/2017 13:18-13:48	Harr	Harr	1	File1-[2017-08-18_11-46-59]-193	05:54		God
370	Harr170822_5	22/08/2017 13:18-13:48	Harr	Opp	1	File1-[2017-08-18_11-46-59]-193	28:07		God
371	Ørret170818_2	22/08/2017 13:48-14:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-194	00:40	09:07;13:57;18:42	God
372	Ørret170822_7	22/08/2017 13:48-14:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-194	24:26		God
373	Ørret170822_8	22/08/2017 13:48-14:18	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-194	29:53		God
374	Ørret170822_8	22/08/2017 14:18-14:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-195	00:15		God
375	Ørret170818_2	22/08/2017 14:18-14:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-195	05:54		God
376	Ørret170822_9	22/08/2017 14:18-14:48	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-195	13:12		God
377	Harr170822_6	22/08/2017 14:18-14:48	Harr	Opp	1	File1-[2017-08-18_11-46-59]-195	13:23	14:43;15:12;17:22;18:02;19:05;19:27	God
378	Notat170822_2	22/08/2017 14:48-15:18	Sik		1	File1-[2017-08-18_11-46-59]-196	05:14		God
379	Sik170822_3	22/08/2017 14:48-15:18	Sik	Opp	1	File1-[2017-08-18_11-46-59]-196	06:24	07:54	God
380	Ørret170818_2	22/08/2017 14:48-15:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-196	13:50	17:00;25:50;29:17;	God
381	Ørret170818_2	22/08/2017 15:18-15:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-197	04:14	09:17;12:29	God
382	Harr170822_7	22/08/2017 15:18-15:48	Harr	Opp	1	File1-[2017-08-18_11-46-59]-197	13:19	16:57	God
383	Ørret170818_2	22/08/2017 15:48-16:18	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-198	00:14	02:18;07:28;11:02;27:29	God
384	Ørret170822_10	22/08/2017 15:48-16:18	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-198	16:12		God
385	Ørret170822_11	22/08/2017 15:48-16:18	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-198	16:14		God
386	Ørret170822_11	22/08/2017 15:48-16:18	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-198	16:48		God
387	Ørret170818_2	22/08/2017 16:18-16:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-199	08:31	13:37;16:06;19:57;23:24;29:37	God
388	Ørret170822_12	22/08/2017 16:18-16:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-199	14:07		God
389	Ørret170822_13	22/08/2017 16:18-16:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-199	14:07		God
390	Notat170822_3	22/08/2017 16:18-16:48	Sik	Kam	1	File1-[2017-08-18_11-46-59]-199	26:53		God
391	Sik170822_4	22/08/2017 16:18-16:48	Sik	Kam	1	File1-[2017-08-18_11-46-59]-199	28:18		God
392	Ørret170822_14	22/08/2017 16:18-16:48	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-199	28:45		God
393	Ørret170818_2	22/08/2017 16:49-17:19	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-200	00:08	00:59;13:08	God
394	Sik170822_4	22/08/2017 16:49-17:19	Sik	Kam	1	File1-[2017-08-18_11-46-59]-200	06:52	07:25;09:08;09:32;11:53;13:14	God
395	Harr170822_8	22/08/2017 16:49-17:19	Harr	Kam	1	File1-[2017-08-18_11-46-59]-200	07:25		God
396	Harr170822_9	22/08/2017 16:49-17:19	Harr	Kam	1	File1-[2017-08-18_11-46-59]-200	07:25		God
397	Harr170822_8	22/08/2017 16:49-17:19	Harr	Ned	1	File1-[2017-08-18_11-46-59]-200	07:41		God
398	Harr170822_9	22/08/2017 16:49-17:19	Harr	Ned	1	File1-[2017-08-18_11-46-59]-200	07:41		God
399	Ørret170818_2	22/08/2017 16:49-17:19	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-200	13:10		God
400	Sik170822_5	22/08/2017 16:49-17:19	Sik	Ned	1	File1-[2017-08-18_11-46-59]-200	23:24		God
401	Harr170822_10	22/08/2017 17:19-17:49	Harr	Opp	1	File1-[2017-08-18_11-46-59]-201	06:34		God
402	Notat170822_4	22/08/2017 17:49-18:19	Fish		1	File1-[2017-08-18_11-46-59]-202	08:11		God
403	Notat170822_5	22/08/2017 17:49-18:19	Harr		1	File1-[2017-08-18_11-46-59]-202	28:12		God
404	Sik170822_6	22/08/2017 18:19-18:49	Sik	Kam	1	File1-[2017-08-18_11-46-59]-203	01:42	03:21;05:17;06:09;08:39;09:56;10:55	God
405	Notat170822_6	22/08/2017 18:19-18:49	Elvenøy? Iggle?		1	File1-[2017-08-18_11-46-59]-203	07:10		God
406	Ørret170822_15	22/08/2017 18:19-18:49	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-203	20:11		God
407	Ørret170818_2	22/08/2017 18:19-18:49	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-203	23:56	27:10;28:02	God
408	Ørret170818_2	22/08/2017 18:49-19:19	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-204	04:29	08:27;11:33;25:59	God
409	Ørret170822_16	22/08/2017 18:49-19:19	Ørret	Opp	1	File1-[2017-08-18_11-46-59]-204	08:00	08:24	God
410	Harr170822_11	22/08/2017 19:19-19:49	Harr	Kam	1	File1-[2017-08-18_11-46-59]-205	07:08	07:49	God
411	Harr170822_11	22/08/2017 19:19-19:49	Harr	Ned	1	File1-[2017-08-18_11-46-59]-205	07:51		God
412	Harr170822_12	22/08/2017 19:19-19:49	Harr	Ned	1	File1-[2017-08-18_11-46-59]-205	09:12		God
413	Ørret170822_17	22/08/2017 19:19-19:49	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-205	09:47		God
414	Ørret170822_18	22/08/2017 19:19-19:49	Ørret	Ned	1	File1-[2017-08-18_11-46-59]-205	14:28		God
415	Ørret170818_2	22/08/2017 19:19-19:49	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-205	15:20		God
416	Ørret170818_2	22/08/2017 19:49-20:19	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-206	23:11	25:39	God
417	Ørret170818_2	22/08/2017 20:19-20:48	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-207	08:01	19:50;20:18	God
418	Sik170822_6	23/08/2017 05:19-05:49	Sik	Kam	1	File1-[2017-08-18_11-46-59]-225	00:37	04:47;05:38;8:11;18:01;18:53;19:22	Ok
419	Lake170823_1	23/08/2017 05:49-06:19	Lake	Opp	1	File1-[2017-08-18_11-46-59]-226	17:13		Ok
420	Lake170823_2	23/08/2017 05:49-06:19	Lake	Opp	1	File1-[2017-08-18_11-46-59]-226	20:39		Ok
421	Ørret170818_2	23/08/2017 06:19-06:49	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-227	00:21	07:35;25:07	Ok
422	Ørret170818_2	23/08/2017 06:49-07:19	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-228	09:06	24:26	Ok
423	Ørret170823_1	23/08/2017 06:49-07:19	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-228	29:12		Ok
424	Ørret170818_2	23/08/2017 07:19-07:49	Ørret	Kam	1	File1-[2017-08-18_11-46-59]-229	03:08		Ok
425	Ørret170818_2	28/08/2017 16:26-16:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-001	19:37	23:35	God
426	Ørret170828_1	28/08/2017 16:26-16:56	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-001	27:37		God
427	Ørret170818_2	28/08/2017 17:26-17:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-003	04:48	12:01;14:58;19:15;20:30;24:21;28:48	God
428	Ørret170828_2	28/08/2017 17:26-17:56	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-003	28:59		God
429	Sik170828_1	28/08/2017 17:56-18:26	Sik	Kam	1	File1-[2017-08-28_14-56-15]-004	00:08	01:42;04:38;10:41;12:21;15:20	God
430	Fish170828_1	28/08/2017 17:56-18:26	Fish	Opp	1	File1-[2017-08-28_14-56-15]-004	01:02		God
431	Ørret170818_2	28/08/2017 17:56-18:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-004	01:22		God
432	Ørret170828_3	28/08/2017 17:56-18:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-004	03:38		God
433	Ørret170828_4	28/08/2017 17:56-18:26	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-004	05:07		God
434	Sik170828_1	28/08/2017 18:26-18:56	Sik	Kam	1	File1-[2017-08-28_14-56-15]-005	19:26		God
435	Ørret170828_5	28/08/2017 18:56-19:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-006	05:24		God
436	Ørret170828_6	28/08/2017 18:56-19:26	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-006	05:34		God
437	Ørret170828_7	28/08/2017 19:26-19:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-007	14:41		God
438	Ørret170828_8	28/08/2017 19:26-19:56	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-007	22:00		God
439	Ørret170828_7	28/08/2017 19:26-19:56	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-007	25:05		God
440	Ørret170828_7	28/08/2017 19:26-19:56	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-007	25:05		God
441	Ørret170828_8	28/08/2017 19:56-20:26	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-008	07:18		God
442	Ørret170828_9	28/08/2017 19:56-20:26	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-008	08:28		God
443	Ørret1								

466	Ørret170829_7	29/08/2017	13:56-14:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-044	14:03		God
467	Ørret170818_2	29/08/2017	13:56-14:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-044	14:16	18:25; 21:13; 27:57	God
468	Ørret170818_2	29/08/2017	14:26-14:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-045	11:43	13:00	God
469	Ørret170818_2	29/08/2017	14:26-14:56	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-045	13:54		God
470	Ørret170828_7	29/08/2017	14:26-14:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-045	16:13	18:13; 20:06; 23:16; 24:27	God
471	Sik170829_2	29/08/2017	14:26-14:56	Sik	Kam	1	File1-[2017-08-28_14-56-15]-045	23:41	24:27; 27:45	God
472	Ørret170828_7	29/08/2017	14:26-14:56	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-045	24:45		God
473	Ørret170829_8	29/08/2017	14:26-14:56	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-045	25:29		God
474	Ørret170829_9	29/08/2017	14:26-14:56	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-045	25:37		God
475	Ørret170829_10	29/08/2017	14:56-15:26	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-046	01:18		God
476	Harr170829_1	29/08/2017	15:26-15:56	Harr	Opp	1	File1-[2017-08-28_14-56-15]-047	17:56		God
477	Harr170829_2	29/08/2017	15:26-15:56	Harr	Opp	1	File1-[2017-08-28_14-56-15]-047	17:57		God
478	Ørret170818_2	29/08/2017	15:26-15:56	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-047	25:33	28:13; 28:38	God
479	Ørret170829_11	29/08/2017	15:26-15:56	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-047	28:29		God
480	Ørret170818_2	29/08/2017	15:56-16:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-048	00:08	26:40	God
481	Ørret170818_2	29/08/2017	16:26-16:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-049	04:20	12:11; 27:14	God
482	Harr170829_3	29/08/2017	16:26-16:56	Harr	Opp	1	File1-[2017-08-28_14-56-15]-049	23:49		God
483	Harr170829_4	29/08/2017	16:56-17:26	Harr	Opp	1	File1-[2017-08-28_14-56-15]-050	00:49		God
484	Ørret170829_12	29/08/2017	16:56-17:26	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-050	22:49		God
485	Ørret170829_12	29/08/2017	16:56-17:26	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-050	25:16		God
486	Ørret170829_12	29/08/2017	16:56-17:26	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-050	25:41		God
487	Ørret170829_13	29/08/2017	17:26-17:56	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-051	11:24	12:56; 22:35	God
488	Ørret170829_13	29/08/2017	17:56-18:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-052	01:32	02:33; 10:41; 12:26; 13:13	God
489	Notat170829_1	29/08/2017	17:56-18:26	Sik			File1-[2017-08-28_14-56-15]-052	12:47		God
490	Ørret170829_14	29/08/2017	17:56-18:26	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-052	25:55		God
491	Ørret170829_13	29/08/2017	18:26-18:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-053	00:58	13:32; 18:29	God
492	Ørret170829_14	29/08/2017	18:26-18:56	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-053	02:47		God
493	Sik170829_3	29/08/2017	18:26-18:56	Sik	Kam	1	File1-[2017-08-28_14-56-15]-053	13:45		God
494	Ørret170829_13	29/08/2017	18:56-19:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-054	00:17		God
495	Ørret170829_13	29/08/2017	19:26-19:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-055	06:53	09:00; 15:36	God
496	Ørret170829_14	29/08/2017	19:56-20:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-056	07:39		Ok
497	Ørret170829_13	29/08/2017	19:56-20:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-056	12:05		Ok
498	Ørret170829_14	29/08/2017	19:56-20:26	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-056	14:10		Ok
499	Ørret170829_15	29/08/2017	19:56-20:26	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-056	26:00		Ok
500	Ørret170830_1	30/08/2017	05:56-06:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-076	17:19		Ok
501	Ørret170818_2	30/08/2017	05:56-06:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-076	28:01		Ok
502	Ørret170818_2	30/08/2017	06:26-06:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-077	05:48	11:46; 17:36	Ok
503	Ørret170830_2	30/08/2017	06:56-07:26	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-078	07:19		Ok
504	Ørret170818_2	30/08/2017	06:56-07:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-078	09:45	15:48; 17:10	Ok
505	Ørret170830_3	30/08/2017	07:26-07:56	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-079	09:09	10:00	Ok
506	Ørret170818_2	30/08/2017	07:26-07:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-079	13:15	27:47	Ok
507	Ørret170818_2	30/08/2017	07:56-08:26	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-080	03:40		God
508	Ørret170828_7	30/08/2017	08:26-08:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-081	19:48		God
509	Ørret170830_4	30/08/2017	08:26-08:56	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-081	25:26		God
510	Ørret170830_5	30/08/2017	08:56-09:26	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-082	21:18		God
511	Ørret170830_6	30/08/2017	09:26-09:56	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-083	06:08		God
512	Ørret170830_7	30/08/2017	09:26-09:56	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-083	06:50		God
513	Ørret170830_8	30/08/2017	09:26-09:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-083	14:18		God
514	Ørret170830_9	30/08/2017	09:26-09:56	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-083	20:54		God
515	Ørret170830_10	30/08/2017	09:26-09:56	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-083	21:04		God
516	Ørret170829_13	30/08/2017	09:26-09:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-083	22:47	24:58	God
517	Ørret170830_11	30/08/2017	09:56-10:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-084	04:07		God
518	Ørret170830_12	30/08/2017	09:56-10:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-084	17:03		God
519	Ørret170829_13	30/08/2017	09:56-10:26	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-084	29:55		God
520	Ørret170830_13	30/08/2017	10:26-10:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-085	00:31		God
521	Ørret170830_14	30/08/2017	10:56-11:26	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-086	24:46	25:45; 26:17; 28:25	God
522	Ørret170830_14	30/08/2017	11:26-11:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-087	00:04	01:03	God
523	Ørret170830_14	30/08/2017	11:26-11:56	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-087	01:13		God
524	Fish170830_1	30/08/2017	11:26-11:56	Fish	Opp	1	File1-[2017-08-28_14-56-15]-087	05:30		God
525	Ørret170830_15	30/08/2017	11:26-11:56	Ørret	Opp	1	File1-[2017-08-28_14-56-15]-087	06:05		God
526	Ørret170818_2	30/08/2017	11:26-11:56	Ørret	Kam	1	File1-[2017-08-28_14-56-15]-087	14:53	16:09	God
527	Ørret170830_16	30/08/2017	11:26-11:56	Ørret	Ned	1	File1-[2017-08-28_14-56-15]-087	18:15		God
528	Lake170830	30/08/2017	12:56-13:26	Lake	Ned	1	File1-[2017-08-28_14-56-15]-090	17:57		God
529	Harr170830_1	30/08/2017	13:26-13:56	Harr	Opp	1	File1-[2017-08-28_14-56-15]-091	17:37		God
530	Ørret170915_1	15/09/2017	07:31-08:01	Ørret	Kam	2	File1-[2017-09-14_18-00-50]-025	20:44	28:10	Ok
531	Ørret170818_2	15/09/2017	07:31-08:01	Ørret	Kam	2	File1-[2017-09-14_18-00-50]-025	29:40		Ok
532	Harr170915_1	15/09/2017	08:01-08:31	Harr	Kam	2	File1-[2017-09-14_18-00-50]-026	17:59		Ok
533	Sik170915_1	15/09/2017	08:31-09:01	Sik	Kam	2	File1-[2017-09-14_18-00-50]-027	09:45	15:12	Ok
534	Sik170915_2	15/09/2017	10:01-10:31	Sik	Kam	2	File1-[2017-09-14_18-00-50]-030	17:57	18:22; 22:04; 23:35; 25:27	Ok
535	Harr170915_2	15/09/2017	10:01-10:31	Harr	Opp	2	File1-[2017-09-14_18-00-50]-030	29:32		Ok
536	Sik170915_2	15/09/2017	10:31-11:01	Sik	Kam	2	File1-[2017-09-14_18-00-50]-031	00:00	02:27; 04:45; 06:28; 07:41	Ok
537	Harr170915_3	15/09/2017	10:31-11:01	Harr	Kam	2	File1-[2017-09-14_18-00-50]-031	06:56	07:41; 14:49	Ok
538	Ørret170915_2	15/09/2017	10:31-11:01	Ørret	Ned	2	File1-[2017-09-14_18-00-50]-031	22:41		Ok
539	Sik170915_2	15/09/2017	11:31-12:01	Sik	Kam	2	File1-[2017-09-14_18-00-50]-033	10:23	25:50; 27:24	Ok
540	Ørret170915_3	15/09/2017	11:31-12:01	Ørret	Opp	2	File1-[2017-09-14_18-00-50]-033	12:04		Ok
541	Ørret170915_3	15/09/2017	11:31-12:01	Ørret	Ned	2	File1-[2017-09-14_18-00-50]-033	13:01		Ok
542	Ørret170915_3	15/09/2017	11:31-12:01	Ørret	Opp	2	File1-[2017-09-14_18-00-50]-033	13:55	14:12; 15:14	Ok
543	Harr170915_4	15/09/2017	12:01-12:31	Harr	Kam	2	File1-[2017-09-14_18-00-50]-034	00:48		Ok
544	Sik170915_2	15/09/2017	12:31-13:01	Sik	Kam	2	File1-[2017-09-14_18-00-50]-034	23:42		Ok
545	Sik170915_2	15/09/2017	13:01-13:31	Sik	Kam	2	File1-[2017-09-14_18-00-50]-035	22:04		Ok
546	Fish170915_1	15/09/2017	13:01-13:31	Fish	Opp	2	File1-[2017-09-14_18-00-50]-035	23:40	24:46	Ok
547	Harr170915_5	15/09/2017	13:01-13:31	Harr	Kam	2	File1-[2017-09-14_18-00-50]-035	23:52		Ok
548	Harr170915_6	15/09/2017	13:01-13:31	Harr	Kam	2	File1-[2017-09-14_18-00-50]-035	24:54		Ok
549	Harr170915_7	15/09/2017	13:31-14:01	Harr	Kam	2	File1-[2017-09-14_18-00-50]-036	03:14	04:56; 05:07; 05:38	Ok
550	Harr170915_8	15/09/2017	13:31-14:01	Harr	Kam	2	File1-[2017-09-14_18-00-50]-036	05:07	05:38; 09:50	Ok
551	Sik170915_2	15/09/2017	14:01-14:31	Sik	Kam	2	File1-[2017-09-14_18-00-50]-037	13:04	14:38; 16:34; 18:01	Ok
552	Harr170915_9	15/09/2017	14:01-14:31	Harr	Kam	2	File1-[2017-09-14_18-00-50]-037	14:38	18:44; 22:20; 23:03; 24:13	Ok
553	Fish170915_2	15/09/2017	14:01-14:31	Fish	Ned	2	File1-[2017-09-14_18-00-50]-037	22:34		Ok
554	Sik170915_2	15/09/2017	14:31-15:01	Sik	Kam	2	File1-[2017-09-14_18-00-50]-038	22:06	25:57	Ok
555	Sik170915_2	15/09/2017	15:01-15:31	Sik	Kam	2	File1-[2017-09-14_18-00-50]-039	00:18		Ok
556	Ørret170818_2	15/09/2017	15:01-15:31	Ørret	Kam	2	File1-[2017-09-14_1			

582	Ørret170916_3	16/09/2017 10:01-10:31	Ørret	Kam	2	File1-[2017-09-14_18-00-50]-078	07:43		Ok
583	Sik170916_1	16/09/2017 10:01-10:31	Sik	Kam	2	File1-[2017-09-14_18-00-50]-078	09:01	28:49	Ok
584	Sik170915_2	16/09/2017 10:31-11:01	Sik	Kam	2	File1-[2017-09-14_18-00-50]-079	00:50	02:08; 03:42; 09:58; 11:17; 18:55; 22:19	Ok
585	Ørret170916_4	16/09/2017 10:31-11:01	Ørret	Kam	2	File1-[2017-09-14_18-00-50]-079	15:36		Ok
586	Ørret170916_5	16/09/2017 10:31-11:01	Ørret	Ned	2	File1-[2017-09-14_18-00-50]-079	16:45		Ok
587	Harr170916_1	16/09/2017 12:01-12:31	Harr	Kam	2	File1-[2017-09-14_18-00-50]-082	29:55		Ok
588	Harr170916_2	16/09/2017 14:01-14:31	Harr	Kam	2	File1-[2017-09-14_18-00-50]-086	12:15	13:07	Ok
589	Harr170916_3	16/09/2017 16:01-16:31	Harr	Kam	2	File1-[2017-09-14_18-00-50]-090	06:24		Ok
590	Harr170916_3	16/09/2017 16:01-16:31	Harr	Ned	2	File1-[2017-09-14_18-00-50]-090	06:35		Ok
591	Harr170916_4	16/09/2017 18:31-19:01	Harr	Kam	2	File1-[2017-09-14_18-00-50]-095	00:56		Ok
592	Harr170916_4	16/09/2017 18:31-19:01	Harr	Ned	2	File1-[2017-09-14_18-00-50]-095	01:15		Ok
593	Ørret170917_1	17/09/2017 07:01-07:31	Ørret	Opp	2	File1-[2017-09-14_18-00-50]-120	26:07		Dårlig
594	Ørret170917_1	17/09/2017 07:01-07:31	Ørret	Ned	2	File1-[2017-09-14_18-00-50]-120	27:31		Dårlig
595	Ørret170917_2	17/09/2017 07:31-08:01	Ørret	Ned	2	File1-[2017-09-14_18-00-50]-121	10:36		Ok
596	Ørret170917_3	17/09/2017 07:31-08:01	Ørret	Kam	2	File1-[2017-09-14_18-00-50]-121	11:32		Ok
597	Ørret170917_4	17/09/2017 08:01-08:31	Ørret	Kam	2	File1-[2017-09-14_18-00-50]-122	10:46	12:13	Ok
598	Ørret170917_5	17/09/2017 09:01-09:31	Ørret	Kam	2	File1-[2017-09-14_18-00-50]-124	03:25		Ok
599	Ørret170917_6	17/09/2017 09:31-10:01	Ørret	Opp	2	File1-[2017-09-14_18-00-50]-125	25:26		Ok
600	Ørret170818_2	17/09/2017 10:01-10:31	Ørret	Kam	2	File1-[2017-09-14_18-00-50]-126	29:05	29:25	Ok
601	Ørret170917_7	17/09/2017 11:31-12:01	Ørret	Opp	2	File1-[2017-09-14_18-00-50]-129	04:00	05:17	Ok
602	Ørret170818_2	17/09/2017 11:31-12:01	Ørret	Kam	2	File1-[2017-09-14_18-00-50]-129	10:14		Ok
603	Ørret170818_2	17/09/2017 11:31-12:01	Ørret	Ned	2	File1-[2017-09-14_18-00-50]-129	11:18		Ok
604	Ørret170818_2	17/09/2017 11:31-12:01	Ørret	Ned	2	File1-[2017-09-14_18-00-50]-129	17:26		Ok
605	Ørret170917_8	17/09/2017 11:31-12:01	Ørret	Opp	2	File1-[2017-09-14_18-00-50]-129	19:23		Ok
606	Ørret170917_9	17/09/2017 14:01-14:31	Ørret	Opp	2	File1-[2017-09-14_18-00-50]-134	01:35		Ok
607	Harr170917_1	17/09/2017 14:31-15:01	Harr	Opp	2	File1-[2017-09-14_18-00-50]-135	26:33	27:14	Ok
608	Ørret170917_10	17/09/2017 16:31-17:01	Ørret	Kam	2	File1-[2017-09-14_18-00-50]-139	03:55		Ok
609	Ørret170818_2	17/09/2017 19:01-19:31	Ørret	Kam	2	File1-[2017-09-14_18-00-50]-144	26:14	27:48	Ok
610	Ørret170917_11	17/09/2017 19:01-19:31	Ørret	Opp	2	File1-[2017-09-14_18-00-50]-144	26:34		Ok
611	Ørret170924_1	24/09/2017 16:24-16:54	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-000	03:02		Ok
612	Harr170924_1	24/09/2017 16:24-16:54	Harr	Kam	3	File1-[2017-09-24_15-24-14]-000	05:16	05:52; 08:55; 13:24; 17:50	Ok
613	Ørret170924_2	24/09/2017 16:24-16:54	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-000	12:56		Ok
614	Ørret170924_3	24/09/2017 16:54-17:24	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-001	10:28		Ok
615	Harr170924_2	24/09/2017 17:24-17:54	Harr	Kam	3	File1-[2017-09-24_15-24-14]-002	22:47	23:14	Ok
616	Harr170924_2	24/09/2017 17:24-17:54	Harr	Ned	3	File1-[2017-09-24_15-24-14]-002	23:14		Ok
617	Harr170924_3	24/09/2017 17:54-18:24	Harr	Ned	3	File1-[2017-09-24_15-24-14]-003	03:22		Ok
618	Harr170924_4	24/09/2017 17:54-18:24	Harr	Kam	3	File1-[2017-09-24_15-24-14]-003	22:11	22:46; 23:14	Ok
619	Harr170924_4	24/09/2017 17:54-18:24	Harr	Ned	3	File1-[2017-09-24_15-24-14]-003	23:34		Ok
620	Ørret170924_4	24/09/2017 17:54-18:24	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-003	28:50		Ok
621	Fish170925_1	25/09/2017 08:54-09:24	Fish	Kam	3	File1-[2017-09-24_15-24-14]-033	03:14		Ok
622	Ørret170925_1	25/09/2017 08:54-09:24	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-033	22:07		Ok
623	Ørret170925_2	25/09/2017 08:54-09:24	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-033	26:16		Ok
624	Ørret170925_3	25/09/2017 08:54-09:24	Ørret	Ned	3	File1-[2017-09-24_15-24-14]-033	28:22		Ok
625	Ørret170925_2	25/09/2017 08:54-09:24	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-035	12:20		Ok
626	Ørret170925_4	25/09/2017 09:24-10:54	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-036	04:09		Ok
627	Ørret170925_5	25/09/2017 09:24-10:54	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-036	04:09		Ok
628	Ørret170925_6	25/09/2017 11:54-12:24	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-039	23:32		Ok
629	Ørret170925_7	25/09/2017 11:54-12:24	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-039	27:32		Ok
630	Ørret170925_8	25/09/2017 12:54-13:24	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-041	07:27		Ok
631	Harr170925_1	25/09/2017 12:54-13:24	Harr	Kam	3	File1-[2017-09-24_15-24-14]-041	07:27		Ok
632	Ørret170925_9	25/09/2017 12:54-13:24	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-041	10:17		Ok
633	Harr170925_2	25/09/2017 13:24-13:54	Harr	Kam	3	File1-[2017-09-24_15-24-14]-042	03:49		Ok
634	Harr170925_2	25/09/2017 13:24-13:54	Harr	Ned	3	File1-[2017-09-24_15-24-14]-042	04:11		Ok
635	Harr170925_3	25/09/2017 13:24-13:54	Harr	Kam	3	File1-[2017-09-24_15-24-14]-042	09:59		Ok
636	Harr170925_4	25/09/2017 13:54-14:24	Harr	Kam	3	File1-[2017-09-24_15-24-14]-043	22:13	23:40; 24:35; 26:55	Ok
637	Ørret170925_10	25/09/2017 14:54-15:24	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-045	06:59		Ok
638	Harr170925_5	25/09/2017 14:54-15:24	Harr	Ned	3	File1-[2017-09-24_15-24-14]-045	22:54		Ok
639	Ørret170925_11	25/09/2017 16:24-16:54	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-048	22:26	28:15	Ok
640	Ørret170925_11	25/09/2017 16:24-16:54	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-048	29:59		Ok
641	Harr170925_6	25/09/2017 16:54-17:24	Harr	Kam	3	File1-[2017-09-24_15-24-14]-049	02:43	03:49	Ok
642	Harr170925_7	25/09/2017 16:54-17:24	Harr	Opp	3	File1-[2017-09-24_15-24-14]-049	15:06	16:29	Ok
643	Ørret170925_12	25/09/2017 17:54-18:24	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-051	07:16		Ok
644	Lake170926_1	26/09/2017 07:25-07:55	Lake	Kam	3	File1-[2017-09-24_15-24-14]-078	11:40		Ok
645	Lake170926_2	26/09/2017 07:55-08:25	Lake	Kam	3	File1-[2017-09-24_15-24-14]-079	01:01		Ok
646	Ørret170926_1	26/09/2017 07:55-08:25	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-079	03:59		Ok
647	Ørret170926_2	26/09/2017 07:55-08:25	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-079	09:01		Ok
648	Ørret170926_3	26/09/2017 07:55-08:25	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-079	17:30		Ok
649	Ørret170926_4	26/09/2017 07:55-08:25	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-079	26:06		Ok
650	Harr170926_1	26/09/2017 07:55-08:25	Harr	Opp	3	File1-[2017-09-24_15-24-14]-079	28:42		Ok
651	Sik170926_1	26/09/2017 08:25-08:55	Sik	Opp	3	File1-[2017-09-24_15-24-14]-080	00:52	01:25; 03:47; 04:33; 05:26; 09:06	God
652	Harr170926_2	26/09/2017 08:25-08:55	Harr	Opp	3	File1-[2017-09-24_15-24-14]-080	03:47		God
653	Harr170926_2	26/09/2017 08:25-08:55	Harr	Ned	3	File1-[2017-09-24_15-24-14]-080	04:54		God
654	Harr170926_3	26/09/2017 08:25-08:55	Harr	Kam	3	File1-[2017-09-24_15-24-14]-080	28:17		God
655	Harr170926_3	26/09/2017 08:55-09:25	Harr	Kam	3	File1-[2017-09-24_15-24-14]-081	00:02	00:42	God
656	Sik170926_1	26/09/2017 08:55-09:25	Sik	Kam	3	File1-[2017-09-24_15-24-14]-081	28:26		God
657	Sik170926_1	26/09/2017 09:25-09:55	Sik	Kam	3	File1-[2017-09-24_15-24-14]-082	00:09	03:57; 10:29; 16:33; 18:22; 23:09; 29:07	God
658	Sik170926_1	26/09/2017 09:55-10:25	Sik	Kam	3	File1-[2017-09-24_15-24-14]-083	00:04		God
659	Ørret170926_5	26/09/2017 10:55-11:25	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-085	28:46		God
660	Ørret170926_5	26/09/2017 11:25-11:55	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-086	14:47	23:51; 25:39	God
661	Ørret170926_6	26/09/2017 11:25-11:55	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-086	17:49		God
662	Sik170926_2	26/09/2017 11:25-11:55	Sik	Kam	3	File1-[2017-09-24_15-24-14]-086	25:39		God
663	Sik170926_1	26/09/2017 11:55-12:25	Sik	Kam	3	File1-[2017-09-24_15-24-14]-087	25:29		God
664	Harr170926_4	26/09/2017 13:25-13:55	Harr	Opp	3	File1-[2017-09-24_15-24-14]-090	06:12		God
665	Harr170926_5	26/09/2017 13:25-13:55	Harr	Opp	3	File1-[2017-09-24_15-24-14]-090	06:14		God
666	Harr170926_6	26/09/2017 13:25-13:55	Harr	Opp	3	File1-[2017-09-24_15-24-14]-090	14:02		Ok
667	Harr170926_7	26/09/2017 13:55-14:25	Harr	Opp	3	File1-[2017-09-24_15-24-14]-091	26:23	29:56	Ok
668	Ørret170926_7	26/09/2017 14:55-15:25	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-093	28:14		Ok
669	Harr170926_8	26/09/2017 15:55-16:25	Harr	Opp	3	File1-[2017-09-24_15-24-14]-095	29:29		Ok
670	Harr170926_9	26/09/2017 16:25-16:55	Harr	Ned	3	File1-[2017-09-24_15-24-14]-096	23:42		Ok
671	Ørret170926_8	26/09/2017 16:25-16:55	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-096	28:54		Ok
672	Ørret170926_9	26/09/2017 17:25-17:55	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-098	09:35		Ok
673	Ørret170926_10	26/09/2017 17:25-17:55	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-098	09:36		Ok
674	Ørret170926_11	26/09/2017 17:25-17:55	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-098	24:49		Ok
675	Ørret170926_12	26/09/2017 17:55-18:25	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-099	05:43	06:09; 11:04	Ok
676	Ørret170926_13	26/09/2017 17:55-18:25	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-099	05:43	06:09	Ok
677	Ørret170926_14	2							

698	Ørret170818_2	27/09/2017 13:55-14:25	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-139	26:43		Ok
699	Ørret170927_13	27/09/2017 14:25-14:55	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-140	13:30		Ok
700	Ørret170927_14	27/09/2017 14:25-14:55	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-140	26:00		Ok
701	Ørret170927_15	27/09/2017 14:55-15:25	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-141	08:53		Ok
702	Fish170927_1	27/09/2017 14:55-15:25	Fish	Kam	3	File1-[2017-09-24_15-24-14]-141	20:48		Ok
703	Fish170927_2	27/09/2017 15:25-15:55	Fish	Ned	3	File1-[2017-09-24_15-24-14]-142	01:14		Ok
704	Ørret170927_16	27/09/2017 15:25-15:55	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-142	07:53		Ok
705	Harr170927_3	27/09/2017 15:25-15:55	Harr	Opp	3	File1-[2017-09-24_15-24-14]-142	10:45	14:17	Ok
706	Harr170927_4	27/09/2017 15:55-16:25	Harr	Kam	3	File1-[2017-09-24_15-24-14]-143	00:02		Ok
707	Ørret170927_17	27/09/2017 15:55-16:25	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-143	08:14		Ok
708	Fish170927_3	27/09/2017 16:25-16:55	fish	Kam	3	File1-[2017-09-24_15-24-14]-144	22:33		Ok
709	Harr170927_5	27/09/2017 17:25-17:55	Harr	Kam	3	File1-[2017-09-24_15-24-14]-146	05:22	08:35	Ok
710	Harr170927_6	27/09/2017 17:25-17:55	Harr	Kam	3	File1-[2017-09-24_15-24-14]-146	08:35		Ok
711	Harr170927_7	27/09/2017 17:25-17:55	Harr	Opp	3	File1-[2017-09-24_15-24-14]-146	20:27		Ok
712	Harr170927_8	27/09/2017 18:25-18:55	Harr	Kam	3	File1-[2017-09-24_15-24-14]-148	04:58		Ok
713	Ørret170927_18	27/09/2017 18:25-18:55	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-148	09:26	10:28	Ok
714	Ørret170927_19	27/09/2017 18:25-18:55	Ørret	Ned	3	File1-[2017-09-24_15-24-14]-148	13:21		Ok
715	Ørret170928_1	28/09/2017 07:25-07:55	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-174	00:39	07:02	Ok
716	Ørret170928_2	28/09/2017 07:25-07:55	Ørret	Ned	3	File1-[2017-09-24_15-24-14]-174	15:44		Ok
717	Ørret170928_3	28/09/2017 07:55-08:25	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-175	22:06	24:38; 26:21	Ok
718	Ørret170928_4	28/09/2017 07:55-08:25	Ørret	Kam	3	File1-[2017-09-24_15-24-14]-175	26:21		Ok
719	Ørret170928_5	28/09/2017 07:55-08:25	Ørret	Ned	3	File1-[2017-09-24_15-24-14]-175	29:15		Ok
720	Ørret170928_6	28/09/2017 08:25-08:55	Ørret	Opp	3	File1-[2017-09-24_15-24-14]-176	00:23		Ok
721	Ørret170828_7	29/06/2017 08:43-09:13	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-000	00:16	02:30; [Sjekk opp Id mot reg på disk 1	Ok
722	Ørret170828_7	29/06/2017 08:43-09:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-000	05:20		Ok
723	Ørret170629_1	29/06/2017 08:43-09:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-000	05:23		Ok
724	Ørret170629_1	29/06/2017 08:43-09:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-000	05:26		Ok
725	Ørret170629_1	29/06/2017 08:43-09:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-000	05:55		Ok
726	Ørret170629_2	29/06/2017 08:43-09:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-000	06:19		Ok
727	Fish170629_1	29/06/2017 08:43-09:13	Fish	Ned	4	File1-[2017-06-29_07-43-38]-000	06:55		Ok
728	Ørret170629_3	29/06/2017 08:43-09:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-000	07:58		Ok
729	Ørret170629_3	29/06/2017 08:43-09:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-000	08:02		Ok
730	Ørret170629_4	29/06/2017 08:43-09:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-000	10:03	11:30; 15:12	Ok
731	Ørret170629_4	29/06/2017 08:43-09:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-000	18:10		Ok
732	Fish170629_2	29/06/2017 08:43-09:13	Fish	Opp	4	File1-[2017-06-29_07-43-38]-000	24:57		Ok
733	Fish170629_3	29/06/2017 08:43-09:13	Fish	Opp	4	File1-[2017-06-29_07-43-38]-000	26:30		Ok
734	Ørret170629_5	29/06/2017 09:13-09:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-001	10:58		Ok
735	Ørret170629_6	29/06/2017 09:13-09:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-001	12:13		Ok
736	Ørret170629_6	29/06/2017 09:13-09:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-001	12:58		Ok
737	Ørret170629_6	29/06/2017 09:13-09:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-001	13:30		Ok
738	Ørret170629_7	29/06/2017 09:13-09:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-001	16:03		Ok
739	Ørret170629_8	29/06/2017 09:13-09:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-001	17:22		Ok
740	Ørret170629_9	29/06/2017 09:13-09:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-001	26:34		Ok
741	Ørret170629_10	29/06/2017 09:13-09:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-001	26:51		Ok
742	Ørret170629_10	29/06/2017 09:13-09:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-001	28:26		Ok
743	Fish170629_4	29/06/2017 09:13-09:43	Fish	Opp	4	File1-[2017-06-29_07-43-38]-001	28:45		Ok
744	Ørret170629_11	29/06/2017 09:43-10:13	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-002	00:11		Ok
745	Ørret170629_12	29/06/2017 09:43-10:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-002	03:08		Ok
746	Ørret170629_12	29/06/2017 09:43-10:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-002	03:08		Ok
747	Notat170629_1	29/06/2017 09:43-10:13			4	File1-[2017-06-29_07-43-38]-002	11:27		Ubrukeelig
748	Notat170629_2	29/06/2017 09:43-10:13			4	File1-[2017-06-29_07-43-38]-002	19:33		Ok
749	Ørret170629_13	29/06/2017 09:43-10:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-002	29:24		Ok
750	Ørret170629_14	29/06/2017 10:13-10:43	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-003	11:02	17:22	Ok
751	Harr170629_1	29/06/2017 10:13-10:43	Harr	Opp	4	File1-[2017-06-29_07-43-38]-003	15:28		Ok
752	Ørret170629_15	29/06/2017 11:13-11:43	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-005	02:41	04:14; 05:50; 08:20; 18:53; 24:50; 28:42	Ok
753	Ørret170629_15	29/06/2017 11:43-12:13	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-006	00:28		Ok
754	Ørret170629_16	29/06/2017 11:43-12:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-006	00:41	01:34; 01:46; 01:47; 07:08; 18:45	Ok
755	Ørret170629_15	29/06/2017 11:43-12:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-006	01:27		Ok
756	Ørret170629_15	29/06/2017 11:43-12:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-006	01:46	04:18	Ok
757	Ørret170629_17	29/06/2017 11:43-12:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-006	04:43		Ok
758	Ørret170629_18	29/06/2017 11:43-12:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-006	10:00		Ok
759	Fish170629_5	29/06/2017 11:43-12:13	Fish	Ned	4	File1-[2017-06-29_07-43-38]-006	26:53		Ok
760	Fish170629_6	29/06/2017 11:43-12:13	Fish	Opp	4	File1-[2017-06-29_07-43-38]-006	27:05		Ok
761	Ørret170629_16	29/06/2017 12:13-12:43	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-007	29:56		Ok
762	Ørret170629_16	29/06/2017 12:43-13:13	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-008	00:26	04:08; 19:24	Ok
763	Ørret170629_19	29/06/2017 12:43-13:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-008	22:02		Ok
764	Ørret170629_19	29/06/2017 12:43-13:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-008	22:11		Ok
765	Ørret170629_19	29/06/2017 12:43-13:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-008	22:27		Ok
766	Fish170629_7	29/06/2017 12:43-13:13	Fish	Ned	4	File1-[2017-06-29_07-43-38]-008	22:57		Ok
767	Ørret170629_16	29/06/2017 13:13-13:43	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-009	00:02	02:02	Ok
768	Ørret170629_16	29/06/2017 13:43-14:13	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-010	00:28	23:22	Ok
769	Ørret170629_20	29/06/2017 13:43-14:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-010	06:36		Ok
770	Ørret170629_21	29/06/2017 13:43-14:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-010	25:35		Ok
771	Fish170629_8	29/06/2017 13:43-14:13	Fish	Ned	4	File1-[2017-06-29_07-43-38]-010	25:58		Ok
772	Ørret170629_16	29/06/2017 14:13-14:43	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-011	01:06	02:47; 04:31; 16:27; 20:00	Ok
773	Ørret170629_22	29/06/2017 14:13-14:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-011	01:34		Ok
774	Ørret170629_23	29/06/2017 14:13-14:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-011	01:37		Ok
775	Ørret170629_23	29/06/2017 14:13-14:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-011	01:39		Ok
776	Ørret170629_24	29/06/2017 14:13-14:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-011	12:28		Ok
777	Ørret170629_16	29/06/2017 14:43-15:13	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-012	03:02	08:11; 14:32; 21:22;	Ok
778	Ørret170629_25	29/06/2017 14:43-15:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-012	04:47		Ok
779	Ørret170629_26	29/06/2017 14:43-15:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-012	04:49		Ok
780	Ørret170629_26	29/06/2017 14:43-15:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-012	05:04		Ok
781	Ørret170629_27	29/06/2017 14:43-15:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-012	08:59		Ok
782	Ørret170629_28	29/06/2017 14:43-15:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-012	09:02	25:25	Ok
783	Ørret170629_29	29/06/2017 14:43-15:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-012	09:31		Ok
784	Ørret170629_30	29/06/2017 14:43-15:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-012	27:33		Ok
785	Ørret170629_16	29/06/2017 15:13-15:43	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-013	00:05	03:33; 12:48	Ok
786	Ørret170629_31	29/06/2017 15:13-15:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-013	24:56		Ok
787	Ørret170629_31	29/06/2017 15:13-15:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-013	27:19		Ok
788	Ørret170629_16	29/06/2017 15:43-16:13	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-014	02:03	07:28; 12:17;	Ok
789	Ørret170629_32	29/06/2017 15:43-16:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-014	28:27		Ok
790	Ørret170629_32	29/06/2017 15:43-16:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-014	29:05		Ok
791	Ørret170629_33	29/06/2017 15:43-16:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-014	29:21		Ok
792	Ørret170629_16	29/06/2017 16:13-16:43	Ør						

814	Ørret170629_48	29/06/2017 18:43 - 19:13	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-020	00:39	06:52	Ok
815	Ørret170629_49	29/06/2017 19:13 - 19:43	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-021	24:58	26:25	Ok
816	Ørret170629_50	29/06/2017 21:13 - 21:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-025	26:15	26:48; 27:43	Ok
817	Ørret170629_51	29/06/2017 21:13 - 21:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-025	26:15		Ok
818	Ørret170629_51	29/06/2017 21:13 - 21:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-025	26:23		Ok
819	Ørret170629_52	29/06/2017 21:13 - 21:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-025	27:06		Ok
820	Ørret170629_53	29/06/2017 21:13 - 21:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-025	29:42		Ok
821	Ørret170629_54	29/06/2017 21:43 - 22:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-026	00:02		Ok
822	Ørret170818_2	29/06/2017 21:43 - 22:13	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-026	00:32		Ok
823	Fish170629_9	29/06/2017 21:43 - 22:13	Fish	Ned	4	File1-[2017-06-29_07-43-38]-026	03:37		Ok
824	Harr170629_1	29/06/2017 21:43 - 22:13	Harr	Opp	4	File1-[2017-06-29_07-43-38]-026	04:53		Ok
825	Ørret170629_55	29/06/2017 21:43 - 22:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-026	05:53		Ok
826	Ørret170629_56	29/06/2017 21:43 - 22:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-026	26:53	27:49	Ok
827	Ørret170629_57	29/06/2017 21:43 - 22:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-026	26:53		Ok
828	Ørret170629_57	29/06/2017 21:43 - 22:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-026	26:53		Ok
829	Ørret170629_56	29/06/2017 21:43 - 22:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-026	29:13		Ok
830	Ørret170629_58	29/06/2017 21:43 - 22:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-026	29:13		Ok
831	Ørret170629_59	29/06/2017 21:43 - 22:13	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-026	29:25		Ok
832	Ørret170629_60	29/06/2017 22:13 - 22:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-027	07:49	09:47; 14:50; 16:35; 19:46	Ok
833	Ørret170629_61	29/06/2017 22:13 - 22:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-027	14:26		Ok
834	Ørret170629_62	29/06/2017 22:13 - 22:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-027	20:23		Ok
835	Ørret170629_62	29/06/2017 22:13 - 22:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-027	21:18		Ok
836	Ørret170629_63	29/06/2017 22:13 - 22:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-027	21:20		Ok
837	Ørret170629_64	29/06/2017 22:13 - 22:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-027	21:48		Ok
838	Ørret170629_65	29/06/2017 22:13 - 22:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-027	22:24	24:21; 27:39	Ok
839	Ørret170629_66	29/06/2017 22:13 - 22:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-027	27:54		Ok
840	Ørret170629_66	29/06/2017 22:13 - 22:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-027	27:57		Ok
841	Ørret170629_66	29/06/2017 22:13 - 22:43	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-027	28:26		Ok
842	Ørret170629_66	29/06/2017 22:13 - 22:43	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-027	29:03		Ok
843	Ørret170629_66	29/06/2017 22:43 - 23:13	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-028	00:00	01:57; 05:35; 08:27; 14:06; 20:30; 23:02	Dårlig
844	Ørret170629_67	29/06/2017 22:43 - 23:13	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-028	01:45		Ok
845	Ørret170630_1	30/06/2017 04:14 - 04:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-039	00:29	03:51; 06:01; 08:50; 12:27; 13:12; 22:59	Dårlig
846	Ørret170630_2	30/06/2017 04:14 - 04:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-039	24:01	27:34	Dårlig
847	Ørret170630_1	30/06/2017 04:44 - 05:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-040	00:02	06:25; 14:00; 27:58	Ok
848	Ørret170630_3	30/06/2017 04:44 - 05:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-040	06:40		Ok
849	Ørret170630_4	30/06/2017 04:44 - 05:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-040	07:23		Ok
850	Ørret170630_4	30/06/2017 04:44 - 05:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-040	24:16		Ok
851	Ørret170630_5	30/06/2017 04:44 - 05:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-040	28:50		Ok
852	Ørret170630_5	30/06/2017 05:14 - 05:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-041	00:00	01:26; 04:50; 08:13; 11:39; 18:46; 26:54	Ok
853	Ørret170630_6	30/06/2017 05:44 - 06:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-042	00:22	03:14	Ok
854	Ørret170630_7	30/06/2017 05:44 - 06:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-042	03:47		Ok
855	Ørret170629_16	30/06/2017 05:44 - 06:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-042	03:47	06:42; 09:46; 12:02; 16:00	Ok
856	Ørret170630_8	30/06/2017 05:44 - 06:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-042	06:28		Ok
857	Ørret170630_9	30/06/2017 05:44 - 06:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-042	08:19		Ok
858	Ørret170630_10	30/06/2017 05:44 - 06:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-042	08:19		Ok
859	Ørret170630_9	30/06/2017 05:44 - 06:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-042	09:05		Ok
860	Ørret170630_10	30/06/2017 05:44 - 06:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-042	09:05		Ok
861	Ørret170630_10	30/06/2017 05:44 - 06:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-042	09:12		Ok
862	Ørret170630_11	30/06/2017 05:44 - 06:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-042	12:02	24:49; 26:54; 27:43; 29:07	Ok
863	Ørret170630_12	30/06/2017 05:44 - 06:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-042	23:36		Ok
864	Ørret170630_11	30/06/2017 05:44 - 06:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-042	26:57		Ok
865	Ørret170630_11	30/06/2017 06:14 - 06:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-043	00:40	08:06; 12:02; 16:52; 22:15; 24:26	Ok
866	Fish170630_1	30/06/2017 06:14 - 06:44	Fish	Opp	4	File1-[2017-06-29_07-43-38]-043	01:57		Ok
867	Ørret170630_12	30/06/2017 06:14 - 06:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-043	13:48		Ok
868	Ørret170630_12	30/06/2017 06:14 - 06:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-043	17:03		Ok
869	Ørret170630_12	30/06/2017 06:14 - 06:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-043	18:58		Ok
870	Fish170630_2	30/06/2017 06:44 - 07:14	Fish	Opp	4	File1-[2017-06-29_07-43-38]-044	04:57		Ok
871	Ørret170630_13	30/06/2017 06:44 - 07:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-044	05:56		Ok
872	Ørret170629_16	30/06/2017 06:44 - 07:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-044	06:46	10:30; 28:08	Ok
873	Ørret170630_14	30/06/2017 07:14 - 07:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-045	02:40		Ok
874	Ørret170629_16	30/06/2017 07:14 - 07:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-045	02:43	05:27; 08:02; 14:17; 18:33; 28:38	Ok
875	Ørret170630_15	30/06/2017 07:14 - 07:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-045	03:29		Ok
876	Ørret170629_16	30/06/2017 07:44 - 08:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-046	00:06	01:21 ; 03:53; 04:22; 13:02; 16:19; 17:21	Ok
877	Ørret170630_16	30/06/2017 07:44 - 08:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-046	03:13		Ok
878	Ørret170630_17	30/06/2017 07:44 - 08:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-046	06:18		Ok
879	Ørret170630_18	30/06/2017 07:44 - 08:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-046	10:51		Ok
880	Ørret170630_19	30/06/2017 07:44 - 08:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-046	16:26		Ok
881	Ørret170630_20	30/06/2017 07:44 - 08:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-046	16:42		Ok
882	Ørret170630_21	30/06/2017 07:44 - 08:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-046	16:49		Ok
883	Ørret170630_21	30/06/2017 07:44 - 08:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-046	17:05		Ok
884	Ørret170630_21	30/06/2017 07:44 - 08:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-046	17:50		Ok
885	Ørret170630_22	30/06/2017 07:44 - 08:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-046	19:03		Ok
886	Ørret170630_23	30/06/2017 07:44 - 08:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-046	26:06		Ok
887	Ørret170630_23	30/06/2017 07:44 - 08:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-046	26:24		Ok
888	Ørret170629_16	30/06/2017 08:14 - 08:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-047	00:03	04:26; 06:30; 07:51; 14:28; 20:21; 26:05	Ok
889	Ørret170630_24	30/06/2017 08:14 - 08:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-047	15:40		Ok
890	Fish170630_2	30/06/2017 08:14 - 08:44	Fish	Opp	4	File1-[2017-06-29_07-43-38]-047	18:35		Ok
891	Ørret170629_16	30/06/2017 08:14 - 08:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-047	26:11		Ok
892	Ørret170630_25	30/06/2017 08:44 - 09:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-048	01:34		Ok
893	Ørret170630_26	30/06/2017 08:44 - 09:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-048	04:12		Ok
894	Ørret170630_26	30/06/2017 08:44 - 09:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-048	04:38		Ok
895	Ørret170629_16	30/06/2017 08:44 - 09:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-048	18:18		Ok
896	Ørret170630_27	30/06/2017 08:44 - 09:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-048	26:21	29:15	Ok
897	Ørret170630_28	30/06/2017 08:44 - 09:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-048	27:22		Ok
898	Ørret170630_27	30/06/2017 09:14 - 09:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-049	00:39	02:16; 03:22; 14:42; 19:38; 20:33	Ok
899	Ørret170629_16	30/06/2017 09:14 - 09:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-049	21:06	25:14	Ok
900	Ørret170630_29	30/06/2017 09:44 - 10:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-050	01:04		Ok
901	Ørret170629_16	30/06/2017 09:44 - 10:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-050	01:48	05:52	Ok
902	Ørret170630_30	30/06/2017 09:44 - 10:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-050	03:04		Ok
903	Ørret170630_31	30/06/2017 09:44 - 10:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-050	22:57	25:19; 26:52	Ok
904	Ørret170630_31	30/06/2017 09:44 - 10:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-050	28:19		Ok
905									

930	Ørret170630_42	30/06/2017	12:14 - 12:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-055	17:00		Ok
931	Ørret170630_43	30/06/2017	12:14 - 12:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-055	22:47		Ok
932	Ørret170629_16	30/06/2017	12:44 - 13:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-056	00:10	03:30; 09:16; 14:24; 17:03; 23:03; 29:30	Ok
933	Fish170630_3	30/06/2017	12:44 - 13:14	Fish	Opp	4	File1-[2017-06-29_07-43-38]-056	07:41		Ok
934	Ørret170629_44	30/06/2017	12:44 - 13:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-056	11:23		Ok
935	Ørret170629_16	30/06/2017	13:14 - 13:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-057	00:02	01:53; 03:13; 06:04; 10:38; 18:38; 28:56	Ok
936	Ørret170630_45	30/06/2017	13:14 - 13:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-057	06:02		Ok
937	Ørret170630_46	30/06/2017	13:14 - 13:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-057	06:05		Ok
938	Ørret170630_45	30/06/2017	13:14 - 13:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-057	06:20		Ok
939	Ørret170630_47	30/06/2017	13:14 - 13:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-057	15:23		Ok
940	Ørret170630_48	30/06/2017	13:14 - 13:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-057	15:23		Ok
941	Ørret170630_49	30/06/2017	13:14 - 13:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-057	15:57		Ok
942	Ørret170630_50	30/06/2017	13:14 - 13:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-057	16:05		Ok
943	Ørret170629_16	30/06/2017	13:44 - 14:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-058	00:28	06:25; 08:17; 09:27; 16:06; 20:11; 24:53	Ok
944	Harr170630_8	30/06/2017	14:14 - 14:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-059	01:28		Ok
945	Ørret170629_16	30/06/2017	14:14 - 14:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-059	01:38	06:02; 11:22; 13:45; 21:52; 25:55	Ok
946	Ørret170630_51	30/06/2017	14:14 - 14:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-059	20:20		Ok
947	Ørret170630_51	30/06/2017	14:14 - 14:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-059	22:14		Ok
948	Ørret170629_16	30/06/2017	14:44 - 15:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-060	00:24	02:13; 03:06; 06:05; 09:39; 14:24; 18:51	Ok
949	Ørret170630_52	30/06/2017	14:44 - 15:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-060	02:08		Ok
950	Ørret170630_53	30/06/2017	14:44 - 15:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-060	08:43		Ok
951	Ørret170630_54	30/06/2017	14:44 - 15:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-060	08:43		Ok
952	Ørret170630_55	30/06/2017	14:44 - 15:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-060	18:19		Ok
953	Ørret170630_56	30/06/2017	14:44 - 15:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-060	22:07		Ok
954	Ørret170629_16	30/06/2017	15:14 - 15:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-061	02:00		Ok
955	Ørret170630_57	30/06/2017	15:14 - 15:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-061	10:57		Ok
956	Ørret170630_58	30/06/2017	15:14 - 15:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-061	23:02		Ok
957	Ørret170630_59	30/06/2017	15:44 - 16:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-062	09:03		Ok
958	Ørret170629_15	30/06/2017	15:44 - 16:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-062	10:01		Ok
959	Ørret170630_60	30/06/2017	15:44 - 16:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-062	10:02		Ok
960	Ørret170630_61	30/06/2017	15:44 - 16:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-062	10:22		Ok
961	Ørret170630_62	30/06/2017	15:44 - 16:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-062	10:38		Ok
962	Ørret170629_15	30/06/2017	15:44 - 16:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-062	26:11	26:58	Ok
963	Ørret170630_63	30/06/2017	15:44 - 16:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-062	26:58		Ok
964	Fish170630_4	30/06/2017	15:44 - 16:14	Fish	Opp	4	File1-[2017-06-29_07-43-38]-062	29:28		Ok
965	Ørret170630_64	30/06/2017	16:14 - 16:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-063	00:16		Ok
966	Ørret170629_15	30/06/2017	16:14 - 16:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-063	02:58	03:38; 09:28; 10:44; 19:34; 29:53	Ok
967	Ørret170630_64	30/06/2017	16:14 - 16:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-063	02:59		Ok
968	Notatt170630_1	30/06/2017	16:14 - 16:44			4	File1-[2017-06-29_07-43-38]-063	08:43		Ok
969	Ørret170630_65	30/06/2017	16:14 - 16:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-063	08:56		Ok
970	Lake170630_1	30/06/2017	16:14 - 16:44	Lake	Opp	4	File1-[2017-06-29_07-43-38]-063	09:38		Ok
971	Ørret170629_15	30/06/2017	16:44 - 17:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-064	00:21	02:27	Ok
972	Ørret170629_15	30/06/2017	16:44 - 17:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-064	04:49		Ok
973	Ørret170629_15	30/06/2017	16:44 - 17:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-064	04:58		Ok
974	Ørret170630_65	30/06/2017	16:44 - 17:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-064	05:05		Ok
975	Ørret170629_15	30/06/2017	16:44 - 17:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-064	05:52		Ok
976	Ørret170629_15	30/06/2017	16:44 - 17:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-064	05:59	07:45; 12:13; 14:38; 15:15; 22:03; 23:46; 27:23	Ok
977	Ørret170629_15	30/06/2017	17:14 - 17:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-065	00:57	01:16; 04:24; 08:51; 12:23; 24:06; 29:36	Ok
978	Fish170630_5	30/06/2017	17:14 - 17:44	Fish	Ned	4	File1-[2017-06-29_07-43-38]-065	04:30		Ok
979	Ørret170629_15	30/06/2017	17:44 - 18:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-066	01:07	03:02; 04:30	Ok
980	Lake170630_1	30/06/2017	17:44 - 18:14	Lake	Ned	4	File1-[2017-06-29_07-43-38]-066	01:27		Ok
981	Lake170630_2	30/06/2017	17:44 - 18:14	Lake	Opp	4	File1-[2017-06-29_07-43-38]-066	03:02		Ok
982	Ørret170629_16	30/06/2017	17:44 - 18:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-066	04:30	05:26; 10:59; 14:59;	Ok
983	Ørret170630_66	30/06/2017	17:44 - 18:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-066	05:11		Ok
984	Lake170630_2	30/06/2017	17:44 - 18:14	Lake	Ned	4	File1-[2017-06-29_07-43-38]-066	11:54		Ok
985	Lake170630_2	30/06/2017	17:44 - 18:14	Lake	Opp	4	File1-[2017-06-29_07-43-38]-066	14:54	25:13	Ok
986	Ørret170630_67	30/06/2017	17:44 - 18:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-066	15:27		Ok
987	Ørret170630_68	30/06/2017	17:44 - 18:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-066	16:15		Ok
988	Ørret170629_16	30/06/2017	18:14 - 18:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-067	00:36	04:25; 06:26; 14:15	Ok
989	Sik170630_1	30/06/2017	18:14 - 18:44	Sik	Opp	4	File1-[2017-06-29_07-43-38]-067	15:05		Ok
990	Ørret170630_69	30/06/2017	18:14 - 18:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-067	15:09		Ok
991	Ørret170630_70	30/06/2017	18:14 - 18:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-067	26:49		Ok
992	Ørret170630_71	30/06/2017	18:44 - 19:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-068	14:58	18:02	Ok
993	Ørret170630_72	30/06/2017	18:44 - 19:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-068	18:39		Ok
994	Ørret170629_15	30/06/2017	19:14 - 19:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-069	27:04	29:37	Ok
995	Ørret170629_15	30/06/2017	19:44 - 20:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-070	00:02	02:34; 11:03; 15:30; 19:07; 26:05; 29:59	Ok
996	Ørret170629_15	30/06/2017	20:14 - 20:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-071	00:01	01:10; 03:16; 05:56	Ok
997	Sik170630_2	30/06/2017	20:14 - 20:44	Sik	Kam	4	File1-[2017-06-29_07-43-38]-071	04:04	06:12; 10:03;	Ok
998	Ørret170629_15	30/06/2017	20:14 - 20:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-071	06:12		Ok
999	Fish170630_6	30/06/2017	20:14 - 20:44	Fish	Ned	4	File1-[2017-06-29_07-43-38]-071	10:11		Ok
1000	Ørret170629_15	30/06/2017	21:14 - 21:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-073	16:02	17:40; 26:31	Ok
1001	Ørret170629_15	30/06/2017	21:44 - 22:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-074	02:24	03:04; 06:24; 25:39	Ok
1002	Ørret170630_73	30/06/2017	21:44 - 22:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-074	15:12		Ok
1003	Fish170630_7	30/06/2017	21:44 - 22:14	Fish	Ned	4	File1-[2017-06-29_07-43-38]-074	15:39		Ok
1004	Fish170630_8	30/06/2017	21:44 - 22:14	Fish	Ned	4	File1-[2017-06-29_07-43-38]-074	15:39		Ok
1005	Ørret170630_74	30/06/2017	21:44 - 22:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-074	18:29		Ok
1006	Ørret170630_75	30/06/2017	21:44 - 22:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-074	29:11		Ok
1007	Ørret170630_76	30/06/2017	21:44 - 22:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-074	29:59		Ok
1008	Ørret170629_15	30/06/2017	22:14 - 22:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-075	00:04	03:31; 10:15; 20:33; 22:59; 26:11; 29:52	Ok
1009	Ørret170629_15	01/07/2017	04:14 - 04:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-087	01:11	03:17; 06:32; 08:50; 11:21; 12:47; 17:03	Ok
1010	Ørret170701_1	01/07/2017	04:14 - 04:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-087	17:28		Ok
1011	Ørret170629_15	01/07/2017	04:44 - 05:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-088	00:11	00:45; 02:48; 10:00; 15:23	Ok
1012	Ørret170701_2	01/07/2017	05:14 - 05:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-089	19:23		Ok
1013	Ørret170701_3	01/07/2017	05:14 - 05:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-089	23:03		Ok
1014	Ørret170701_3	01/07/2017	05:14 - 05:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-089	23:25		Ok
1015	Ørret170701_4	01/07/2017	05:14 - 05:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-089	24:08		Ok
1016	Ørret170629_15	01/07/2017	05:14 - 05:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-089	28:10		Ok
1017	Fish170701_1	01/07/2017	05:44 - 06:14	Fish	Opp	4	File1-[2017-06-29_07-43-3			

1046	Ørret170701_10	01/07/2017 08:14 - 08:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-095	10:11		Ok
1047	Ørret170701_11	01/07/2017 08:14 - 08:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-095	10:49		Ok
1048	Ørret170629_15	01/07/2017 08:14 - 08:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-095	11:59		Ok
1049	Ørret170701_12	01/07/2017 08:14 - 08:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-095	18:03		Ok
1050	Ørret170701_13	01/07/2017 08:14 - 08:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-095	18:26		Ok
1051	Ørret170701_14	01/07/2017 08:14 - 08:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-095	18:58		Ok
1052	Ørret170629_16	01/07/2017 08:44 - 09:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-096	01:40	07:08; 09:31; 11:45; 14:42; 23:05; 29:54	Ok
1053	Ørret170818_2	01/07/2017 08:44 - 09:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-096	06:07	06:34	Ok
1054	Ørret170701_15	01/07/2017 08:44 - 09:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-096	08:25		Ok
1055	Ørret170629_16	01/07/2017 09:14 - 09:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-097	00:05	05:23; 07:21; 10:29; 11:35; 19:17; 20:02	Ok
1056	Harr170701_1	01/07/2017 09:14 - 09:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-097	04:29		Ok
1057	Ørret170701_15	01/07/2017 09:14 - 09:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-097	06:31		Ok
1058	Fish170701_5	01/07/2017 09:14 - 09:44	Fish	Opp	4	File1-[2017-06-29_07-43-38]-097	11:48		Ok
1059	Ørret170701_16	01/07/2017 09:14 - 09:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-097	14:06		Ok
1060	Ørret170701_16	01/07/2017 09:14 - 09:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-097	15:21		Ok
1061	Ørret170629_16	01/07/2017 09:44 - 10:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-098	00:28	00:53; 02:07; 04:15; 05:07; 05:54; 06:53	Ok
1062	Ørret170701_16	01/07/2017 09:44 - 10:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-098	02:14		Ok
1063	Ørret170701_17	01/07/2017 09:44 - 10:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-098	05:29		Ok
1064	Harr170701_2	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	06:34		Ok
1065	Harr170701_3	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	06:36		Ok
1066	Harr170701_4	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	06:36		Ok
1067	Harr170701_5	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	06:37		Ok
1068	Harr170701_6	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	06:38		Ok
1069	Harr170701_7	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	06:38		Ok
1070	Harr170701_8	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	06:39		Ok
1071	Harr170701_9	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	06:39		Ok
1072	Harr170701_10	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	06:40		Ok
1073	Harr170701_11	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	09:52		Ok
1074	Harr170701_12	01/07/2017 09:44 - 10:14	Harr	Ned	4	File1-[2017-06-29_07-43-38]-098	11:42		Ok
1075	Harr170701_12	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	11:45		Ok
1076	Harr170701_13	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	15:02		Ok
1077	Harr170701_14	01/07/2017 09:44 - 10:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-098	15:04		Ok
1078	Ørret170701_18	01/07/2017 09:44 - 10:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-098	16:10		Ok
1079	Ørret170818_2	01/07/2017 09:44 - 10:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-098	18:06	20:36	Ok
1080	Ørret170818_2	01/07/2017 09:44 - 10:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-098	21:02		Ok
1081	Ørret170629_15	01/07/2017 09:44 - 10:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-098	29:59		Ok
1082	Ørret170629_15	01/07/2017 10:14 - 10:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-099	00:02	04:43; 08:50; 15:24; 17:56; 29:24	Ok
1083	Harr170701_15	01/07/2017 10:14 - 10:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-099	01:17		Ok
1084	Ørret170701_19	01/07/2017 10:14 - 10:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-099	17:56		Ok
1085	Ørret170701_20	01/07/2017 10:14 - 10:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-099	18:46		Ok
1086	Fish170701_5	01/07/2017 10:14 - 10:44	Fish	Opp	4	File1-[2017-06-29_07-43-38]-099	22:31		Ok
1087	Ørret170701_21	01/07/2017 10:14 - 10:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-099	24:14		Ok
1088	Ørret170701_21	01/07/2017 10:14 - 10:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-099	24:25		Ok
1089	Ørret170701_21	01/07/2017 10:14 - 10:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-099	27:15		Ok
1090	Ørret170701_22	01/07/2017 10:14 - 10:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-099	28:30		Ok
1091	Harr170701_16	01/07/2017 10:14 - 10:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-099	28:47		Ok
1092	Harr170701_17	01/07/2017 10:14 - 10:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-099	28:49		Ok
1093	Harr170701_18	01/07/2017 10:14 - 10:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-099	28:50		Ok
1094	Harr170701_19	01/07/2017 10:14 - 10:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-099	28:52		Ok
1095	Ørret170629_15	01/07/2017 10:44 - 11:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-100	00:36	01:18; 09:03; 11:47; 17:08	Ok
1096	Ørret170701_23	01/07/2017 10:44 - 11:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-100	03:06		Ok
1097	Ørret170701_23	01/07/2017 10:44 - 11:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-100	03:22		Ok
1098	Ørret170701_24	01/07/2017 10:44 - 11:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-100	17:08		Ok
1099	Ørret170629_15	01/07/2017 10:44 - 11:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-100	17:20		Ok
1100	Ørret170701_25	01/07/2017 10:44 - 11:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-100	21:52		Ok
1101	Ørret170701_26	01/07/2017 10:44 - 11:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-100	24:13		Ok
1102	Ørret170701_27	01/07/2017 10:44 - 11:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-100	24:22		Ok
1103	Ørret170629_16	01/07/2017 10:44 - 11:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-100	25:04		Ok
1104	Harr170701_20	01/07/2017 10:44 - 11:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-100	29:23		Ok
1105	Ørret170629_16	01/07/2017 11:14 - 11:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-101	02:04	03:11; 04:19; 05:13; 06:53	Ok
1106	Ørret170701_28	01/07/2017 11:14 - 11:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-101	02:10		Ok
1107	Ørret170701_29	01/07/2017 11:14 - 11:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-101	07:15		Ok
1108	Ørret170629_16	01/07/2017 11:14 - 11:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-101	13:22	28:01	Ok
1109	Ørret170629_16	01/07/2017 11:14 - 11:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-101	18:15		Ok
1110	Ørret170629_16	01/07/2017 11:14 - 11:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-101	18:19		Ok
1111	Harr170701_21	01/07/2017 11:14 - 11:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-101	20:03		Ok
1112	Ørret170701_30	01/07/2017 11:14 - 11:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-101	21:16		Ok
1113	Harr170701_21	01/07/2017 11:14 - 11:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-101	23:30		Ok
1114	Harr170701_21	01/07/2017 11:14 - 11:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-101	23:30		Ok
1115	Ørret170701_30	01/07/2017 11:14 - 11:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-101	21:16		Ok
1116	Ørret170629_16	01/07/2017 11:44 - 12:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-102	00:02	02:57; 08:21; 12:22; 16:42; 19:49; 23:01	Ok
1117	Harr170701_22	01/07/2017 11:44 - 12:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-102	00:02		Ok
1118	Harr170701_23	01/07/2017 11:44 - 12:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-102	14:39		Ok
1119	Harr170701_24	01/07/2017 11:44 - 12:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-102	14:41		Ok
1120	Harr170701_25	01/07/2017 11:44 - 12:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-102	14:43		Ok
1121	Harr170701_26	01/07/2017 11:44 - 12:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-102	14:44		Ok
1122	Harr170701_27	01/07/2017 11:44 - 12:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-102	14:44		Ok
1123	Harr170701_28	01/07/2017 11:44 - 12:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-102	14:45		Ok
1124	Harr170701_29	01/07/2017 11:44 - 12:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-102	14:47		Ok
1125	Harr170701_30	01/07/2017 11:44 - 12:14	Harr	Kam	4	File1-[2017-06-29_07-43-38]-102	15:22		Ok
1126	Harr170701_31	01/07/2017 11:44 - 12:14	Harr	Kam	4	File1-[2017-06-29_07-43-38]-102	15:22		Ok
1127	Harr170701_32	01/07/2017 11:44 - 12:14	Harr	Kam	4	File1-[2017-06-29_07-43-38]-102	15:22		Ok
1128	Ørret170818_2	01/07/2017 11:44 - 12:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-102	20:08		Ok
1129	Harr170701_33	01/07/2017 11:44 - 12:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-102	26:11		Ok
1130	Ørret170629_16	01/07/2017 12:14 - 12:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-103	00:00	10:01; 14:05; 20:27; 26:59	Ok
1131	Ørret170701_30	01/07/2017 12:14 - 12:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-103	02:01		Ok
1132	Ørret170629_16	01/07/2017 12:44 - 13:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-104	00:52	02:45; 08:16; 10:16; 11:43; 17:47; 21:23	Ok
1133	Ørret170701_31	01/07/2017 12:44 - 13:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-104	04:41		Ok
1134	Fish170701_6	01/07/2017 12:44 - 13:14	Fish	Opp	4	File1-[2017-06-29_07-43-38]-104	05:33		Ok
1135	Fish170701_7	01/07/2017 12:44 - 13:14	Fish	Opp	4	File1-[2017-06-29_07-43-38]-104	05:33		Ok
1136	Ørret170701_32	01/07/2017 12:44 - 13:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-104	10:16		Ok
1137	Harr170701_34	01/07/2017 12:44 - 13:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-104	11:00		Ok
1138	Harr170701_35	01/07/2017 12:44 - 13:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-104	1		

1162	Ørret170629_16	01/07/2017	15:14 - 15:44	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-109	00:45	01:47; 04:30	Ok
1163	Ørret170701_36	01/07/2017	15:14 - 15:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-109	02:11		Ok
1164	Ørret170701_37	01/07/2017	15:14 - 15:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-109	07:04		Ok
1165	Ørret170701_38	01/07/2017	15:14 - 15:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-109	11:44		Ok
1166	Sik170701_3	01/07/2017	15:14 - 15:44	Sik	Kam	4	File1-[2017-06-29_07-43-38]-109	11:49		Ok
1167	Ørret170701_39	01/07/2017	15:14 - 15:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-109	12:03		Ok
1168	Ørret170701_40	01/07/2017	15:14 - 15:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-109	12:29		Ok
1169	Ørret170701_41	01/07/2017	15:14 - 15:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-109	13:07		Ok
1170	Ørret170701_41	01/07/2017	15:14 - 15:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-109	14:07		Ok
1171	Harr170701_51	01/07/2017	15:14 - 15:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-109	26:15		Ok
1172	Harr170701_52	01/07/2017	15:14 - 15:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-109	26:15		Ok
1173	Harr170701_53	01/07/2017	15:44 - 16:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-110	04:54		Ok
1174	Ørret170701_42	01/07/2017	15:44 - 16:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-110	05:01		Ok
1175	Ørret170629_16	01/07/2017	15:44 - 16:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-110	05:36	14:58	Ok
1176	Ørret170701_42	01/07/2017	15:44 - 16:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-110	05:57		Ok
1177	Ørret170701_43	01/07/2017	16:14 - 16:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-111	00:26		Ok
1178	Harr170701_53	01/07/2017	16:44 - 17:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-112	00:45		Ok
1179	Harr170701_54	01/07/2017	16:44 - 17:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-112	09:05		Ok
1180	Harr170701_55	01/07/2017	16:44 - 17:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-112	11:34		Ok
1181	Harr170701_56	01/07/2017	17:14 - 17:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-113	06:27		Ok
1182	Harr170701_57	01/07/2017	17:14 - 17:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-113	24:50		Ok
1183	Sik170701_4	01/07/2017	17:14 - 17:44	Sik	Kam	4	File1-[2017-06-29_07-43-38]-113	28:30		Ok
1184	Sik170701_5	01/07/2017	17:44 - 18:14	Sik	Opp	4	File1-[2017-06-29_07-43-38]-114	00:19	01:16; 04:00; 08:31; 12:29; 23:24	Ok
1185	Harr170701_58	01/07/2017	17:44 - 18:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-114	03:34		Ok
1186	Sik170701_6	01/07/2017	17:44 - 18:14	Sik	Kam	4	File1-[2017-06-29_07-43-38]-114	08:31		Ok
1187	Sik170701_5	01/07/2017	18:14 - 18:44	Sik	Kam	4	File1-[2017-06-29_07-43-38]-115	00:47	02:29	Ok
1188	Sik170701_7	01/07/2017	18:44 - 19:14	Sik	Kam	4	File1-[2017-06-29_07-43-38]-116	12:15		Ok
1189	Ørret170701_44	01/07/2017	18:44 - 19:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-116	18:20		Ok
1190	Ørret170701_45	01/07/2017	18:44 - 19:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-116	28:11		Ok
1191	Sik170701_7	01/07/2017	19:14 - 19:44	Sik	Kam	4	File1-[2017-06-29_07-43-38]-117	00:04	07:42; 15:59; 24:20	Ok
1192	Fish170701_8	01/07/2017	19:14 - 19:44	Fish	Ned	4	File1-[2017-06-29_07-43-38]-117	06:11		Ok
1193	Ørret170701_46	01/07/2017	19:14 - 19:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-117	22:06		Ok
1194	Ørret170701_47	01/07/2017	19:14 - 19:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-117	22:11		Ok
1195	Ørret170701_47	01/07/2017	19:14 - 19:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-117	22:40		Ok
1196	Ørret170701_48	01/07/2017	19:44 - 20:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-118	01:40		Ok
1197	Ørret170701_49	01/07/2017	19:44 - 20:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-118	01:51		Ok
1198	Ørret170701_49	01/07/2017	19:44 - 20:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-118	01:51		Ok
1199	Sik170701_7	01/07/2017	19:44 - 20:14	Sik	Kam	4	File1-[2017-06-29_07-43-38]-118	05:50	07:19; 16:48	Ok
1200	Ørret170701_50	01/07/2017	19:44 - 20:14	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-118	15:59		Ok
1201	Sik170701_7	01/07/2017	20:14 - 20:44	Sik	Kam	4	File1-[2017-06-29_07-43-38]-119	03:27	11:39; 16:52	Ok
1202	Harr170701_59	01/07/2017	20:14 - 20:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-119	06:24		Ok
1203	Ørret170701_51	01/07/2017	20:44 - 21:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-120	05:55		Ok
1204	Fish170701_9	01/07/2017	20:44 - 21:14	Fish	Kam	4	File1-[2017-06-29_07-43-38]-120	07:04		Ok
1205	Notat170701_1	01/07/2017	20:44 - 21:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-120	07:29		Ok
1206	Harr170701_60	01/07/2017	20:44 - 21:14	Harr	Kam	4	File1-[2017-06-29_07-43-38]-120	07:33		Ok
1207	Sik170701_8	01/07/2017	20:44 - 21:14	Sik	Kam	4	File1-[2017-06-29_07-43-38]-120	07:38		Ok
1208	Harr170701_61	01/07/2017	20:44 - 21:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-120	15:35		Ok
1209	Harr170701_62	01/07/2017	20:44 - 21:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-120	15:35		Ok
1210	Harr170701_63	01/07/2017	20:44 - 21:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-120	15:36		Ok
1211	Harr170701_64	01/07/2017	20:44 - 21:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-120	15:36		Ok
1212	Harr170701_65	01/07/2017	20:44 - 21:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-120	15:37		Ok
1213	Harr170701_66	01/07/2017	20:44 - 21:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-120	15:38		Ok
1214	Sik170701_8	01/07/2017	20:44 - 21:14	Sik	Opp	4	File1-[2017-06-29_07-43-38]-120	23:35		Ok
1215	Harr170701_67	01/07/2017	21:14 - 21:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-121	00:22		Ok
1216	Ørret170701_52	01/07/2017	21:14 - 21:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-121	06:34		Ok
1217	Ørret170701_53	01/07/2017	21:14 - 21:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-121	08:40		Ok
1218	Sik170701_7	01/07/2017	21:44 - 22:14	Sik	Kam	4	File1-[2017-06-29_07-43-38]-122	02:22	04:37	Ok
1219	Sik170701_7	01/07/2017	22:14 - 22:44	Sik	Kam	4	File1-[2017-06-29_07-43-38]-123	13:26		Ok
1220	Ørret170701_54	01/07/2017	22:14 - 22:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-123	14:41		Ok
1221	Ørret170701_55	01/07/2017	22:14 - 22:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-123	16:45		Ok
1222	Ørret170701_56	01/07/2017	22:14 - 22:44	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-123	18:23		Ok
1223	Harr170701_68	01/07/2017	22:14 - 22:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-123	21:27		Ok
1224	Harr170701_68	01/07/2017	22:14 - 22:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-123	26:01		Ok
1225	Harr170701_68	01/07/2017	22:14 - 22:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-123	26:03		Ok
1226	Harr170701_68	01/07/2017	22:14 - 22:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-123	26:05		Ok
1227	Harr170701_68	01/07/2017	22:14 - 22:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-123	26:06		Ok
1228	Ørret170702_1	02/07/2017	04:44 - 05:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-136	00:29		Ok
1229	Harr170702_1	02/07/2017	04:44 - 05:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-136	04:05		Ok
1230	Harr170702_2	02/07/2017	04:44 - 05:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-136	04:08		Ok
1231	Harr170702_3	02/07/2017	04:44 - 05:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-136	04:09		Ok
1232	Ørret170818_2	02/07/2017	04:44 - 05:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-136	10:41	16:31	Ok
1233	Ørret170818_2	02/07/2017	04:44 - 05:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-136	17:19		Ok
1234	Ørret170818_2	02/07/2017	04:44 - 05:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-136	18:38	21:51; 25:23; 26:31	Ok
1235	Harr170702_4	02/07/2017	05:14 - 05:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-137	00:59		Ok
1236	Harr170702_5	02/07/2017	05:14 - 05:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-137	01:00		Ok
1237	Harr170702_6	02/07/2017	05:14 - 05:44	Harr	Opp	4	File1-[2017-06-29_07-43-38]-137	01:01		Ok
1238	Ørret170702_1	02/07/2017	05:14 - 05:44	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-137	01:28		Ok
1239	Harr170702_7	02/07/2017	05:14 - 05:44	Harr	Kam	4	File1-[2017-06-29_07-43-38]-137	02:46	05:05; 06:29	Ok
1240	Harr170702_8	02/07/2017	05:14 - 05:44	Harr	Ned	4	File1-[2017-06-29_07-43-38]-137	03:43		Ok
1241	Harr170702_9	02/07/2017	05:14 - 05:44	Harr	Ned	4	File1-[2017-06-29_07-43-38]-137	05:55		Ok
1242	Harr170702_10	02/07/2017	05:14 - 05:44	Harr	Ned	4	File1-[2017-06-29_07-43-38]-137	06:38		Ok
1243	Sik170702_1	02/07/2017	05:44 - 06:14	Sik	Opp	4	File1-[2017-06-29_07-43-38]-138	01:39		Ok
1244	Harr170702_11	02/07/2017	05:44 - 06:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-138	11:49		Ok
1245	Harr170702_12	02/07/2017	05:44 - 06:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-138	15:58		Ok
1246	Sik170702_2	02/07/2017	05:44 - 06:14	Sik	Opp	4	File1-[2017-06-29_07-43-38]-138	16:30		Ok
1247	Ørret170702_2	02/07/2017	05:44 - 06:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-138	16:53		Ok
1248	Ørret170629_16	02/07/2017	05:44 - 06:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-138	19:39	21:06; 21:57	Ok
1249	Ørret170702_2	02/07/2017	05:44 - 06:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-138	20:16		Ok
1250	Harr170702_13	02/07/2017	05:44 - 06:14	Harr	Opp	4	File1-[2017-06-29_07-43-38]-138	20:49		Ok
1251	Ørret170702_3	02/07/2017	05:44 - 06:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-138	20:55		Ok
1252	Ørret170702_3	02/07/2017	05:44 - 06:14	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-138	21:12		Ok
1253	Ørret170702_4	02/07/2017	05:44 - 06:14	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-138	22:40		

1278	Harr170702_25	02/07/2017	06:45 - 07:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-140	27:24		Ok
1279	Harr170702_26	02/07/2017	06:45 - 07:15	Harr	Ned	4	File1-[2017-06-29_07-43-38]-140	27:36		Ok
1280	Harr170702_27	02/07/2017	06:45 - 07:15	Harr	Ned	4	File1-[2017-06-29_07-43-38]-140	27:38		Ok
1281	Harr170702_28	02/07/2017	06:45 - 07:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-140	28:37		Ok
1282	Sik170702_3	02/07/2017	07:15 - 07:45	Sik	Kam	4	File1-[2017-06-29_07-43-38]-141	00:07	06:09; 10:40; 15:03	Ok
1283	Ørret170702_13	02/07/2017	07:15 - 07:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-141	02:59		Ok
1284	Sik170702_3	02/07/2017	07:45 - 08:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-142	04:47	06:54; 15:14	Ok
1285	Ørret170629_16	02/07/2017	07:45 - 08:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-142	09:11	13:43	Ok
1286	Harr170702_29	02/07/2017	07:45 - 08:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-142	18:36		Ok
1287	Harr170702_30	02/07/2017	07:45 - 08:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-142	18:37		Ok
1288	Harr170702_31	02/07/2017	07:45 - 08:15	Harr	Ned	4	File1-[2017-06-29_07-43-38]-142	19:56		Ok
1289	Harr170702_32	02/07/2017	07:45 - 08:15	Harr	Ned	4	File1-[2017-06-29_07-43-38]-142	25:04		Ok
1290	Fish170702_2	02/07/2017	08:15 - 08:45	Fish	Opp	4	File1-[2017-06-29_07-43-38]-143	02:16		Ok
1291	Sik170702_4	02/07/2017	08:15 - 08:45	Sik	Kam	4	File1-[2017-06-29_07-43-38]-143	04:30	12:59	Ok
1292	Ørret170629_16	02/07/2017	08:15 - 08:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-143	13:42	19:17	Ok
1293	Ørret170702_12	02/07/2017	08:15 - 08:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-143	24:41		Ok
1294	Harr170702_32	02/07/2017	08:15 - 08:45	Harr	Ned	4	File1-[2017-06-29_07-43-38]-143	25:13		Ok
1295	Harr170702_33	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	25:29		Ok
1296	Harr170702_34	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	28:34		Ok
1297	Harr170702_35	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	28:35		Ok
1298	Harr170702_36	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	28:35		Ok
1299	Harr170702_37	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	28:35		Ok
1300	Harr170702_38	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	28:35		Ok
1301	Harr170702_39	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	28:35		Ok
1302	Harr170702_40	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	28:35		Ok
1303	Harr170702_41	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	28:35		Ok
1304	Harr170702_42	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	28:35		Ok
1305	Harr170702_43	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	28:35		Ok
1306	Harr170702_44	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	28:35		Ok
1307	Harr170702_45	02/07/2017	08:15 - 08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-143	28:35		Ok
1308	Ørret170629_16	02/07/2017	08:45 - 09:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-144	00:00	02:33; 18:17; 21:52; 24:24; 29:59	Ok
1309	Harr170702_46	02/07/2017	08:45 - 09:15	Harr	Ned	4	File1-[2017-06-29_07-43-38]-144	03:46		Ok
1310	Ørret170702_13	02/07/2017	08:45 - 09:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-144	06:02		Ok
1311	Ørret170702_14	02/07/2017	08:45 - 09:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-144	07:23		Ok
1312	Ørret170702_15	02/07/2017	08:45 - 09:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-144	10:26		Ok
1313	Ørret170702_16	02/07/2017	08:45 - 09:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-144	14:13		Ok
1314	Harr170702_46	02/07/2017	08:45 - 09:15	Harr	Kam	4	File1-[2017-06-29_07-43-38]-144	19:52		Ok
1315	Sik170702_5	02/07/2017	08:45 - 09:15	Sik	Opp	4	File1-[2017-06-29_07-43-38]-144	21:14		Ok
1316	Fish170702_3	02/07/2017	08:45 - 09:15	Fish	Ned	4	File1-[2017-06-29_07-43-38]-144	21:50		Ok
1317	Ørret170629_16	02/07/2017	09:15 - 09:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-145	00:01		Ok
1318	Harr170702_47	02/07/2017	09:15 - 09:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-145	00:05		Ok
1319	Harr170702_48	02/07/2017	09:15 - 09:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-145	00:05		Ok
1320	Harr170702_49	02/07/2017	09:15 - 09:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-145	00:05		Ok
1321	Harr170702_50	02/07/2017	09:15 - 09:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-145	00:05		Ok
1322	Harr170702_51	02/07/2017	09:15 - 09:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-145	00:05		Ok
1323	Harr170702_52	02/07/2017	09:15 - 09:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-145	00:05		Ok
1324	Harr170702_53	02/07/2017	09:15 - 09:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-145	14:47		Ok
1325	Harr170702_54	02/07/2017	09:15 - 09:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-145	14:47		Ok
1326	Harr170702_55	02/07/2017	09:15 - 09:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-145	17:35		Ok
1327	Harr170702_56	02/07/2017	09:15 - 09:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-145	22:53		Ok
1328	Ørret170629_16	02/07/2017	09:45 - 10:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-146	10:21	16:57; 18:20; 20:52	Ok
1329	Ørret170702_17	02/07/2017	09:45 - 10:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-146	12:16		Ok
1330	Ørret170702_18	02/07/2017	09:45 - 10:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-146	14:06		Ok
1331	Ørret170702_19	02/07/2017	09:45 - 10:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-146	14:50		Ok
1332	Ørret170702_20	02/07/2017	09:45 - 10:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-146	16:04		Ok
1333	Harr170702_56	02/07/2017	09:45 - 10:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-146	18:51	20:45	Ok
1334	Sik170702_6	02/07/2017	09:45 - 10:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-146	19:47		Ok
1335	Ørret170702_21	02/07/2017	09:45 - 10:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-146	24:38		Ok
1336	Ørret170702_21	02/07/2017	09:45 - 10:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-146	29:21		Ok
1337	Harr170702_57	02/07/2017	10:15 - 10:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-147	03:39		Ok
1338	Harr170702_58	02/07/2017	10:15 - 10:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-147	18:57		Ok
1339	Harr170702_59	02/07/2017	10:15 - 10:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-147	26:11		Ok
1340	Ørret170818_2	02/07/2017	10:45 - 11:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-148	02:55	03:42	Ok
1341	Harr170702_59	02/07/2017	10:45 - 11:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-148	10:00		Ok
1342	Harr170702_60	02/07/2017	10:45 - 11:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-148	10:00		Ok
1343	Harr170702_61	02/07/2017	10:45 - 11:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-148	10:00		Ok
1344	Harr170702_62	02/07/2017	10:45 - 11:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-148	15:39		Ok
1345	Harr170702_63	02/07/2017	11:15 - 11:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-149	03:36		Ok
1346	Harr170702_63	02/07/2017	11:15 - 11:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-149	03:36		Ok
1347	Harr170702_63	02/07/2017	11:15 - 11:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-149	03:36		Ok
1348	Harr170702_63	02/07/2017	11:15 - 11:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-149	03:36		Ok
1349	Harr170702_64	02/07/2017	11:15 - 11:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-149	07:37		Ok
1350	Ørret170818_2	02/07/2017	11:15 - 11:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-149	17:55		Ok
1351	Harr170702_65	02/07/2017	11:15 - 11:45	Harr	Kam	4	File1-[2017-06-29_07-43-38]-149	20:36		Ok
1352	Sik170702_6	02/07/2017	11:15 - 11:45	Sik	Kam	4	File1-[2017-06-29_07-43-38]-149	24:14	25:12; 28:09; 29:07	Ok
1353	Ørret170702_22	02/07/2017	11:15 - 11:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-149	28:04		Ok
1354	Ørret170702_23	02/07/2017	11:15 - 11:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-149	28:28		Ok
1355	Ørret170702_24	02/07/2017	11:15 - 11:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-149	29:30		Ok
1356	Sik170702_6	02/07/2017	11:45 - 12:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-150	00:03	05:52; 09:40; 12:35; 14:07	Ok
1357	Ørret170828_7	02/07/2017	11:45 - 12:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-150	03:53	05:52; 09:32;	Ok
1358	Ørret170702_25	02/07/2017	11:45 - 12:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-150	09:01		Ok
1359	Harr170702_66	02/07/2017	11:45 - 12:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-150	09:04		Ok
1360	Ørret170828_7	02/07/2017	11:45 - 12:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-150	09:54		Ok
1361	Harr170702_67	02/07/2017	11:45 - 12:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-150	11:29		Ok
1362	Ørret170818_2	02/07/2017	11:45 - 12:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-150	12:13		Ok
1363	Harr170702_67	02/07/2017	11:45 - 12:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-150	20:52	22:47	Ok
1364	Harr170702_68	02/07/2017	11:45 - 12:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-150	20:52	22:47	Ok
1365	Harr170702_69	02/07/2017	11:45 - 12:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-150	20:52	22:47	Ok
1366	Ørret170702_27	02/07/2017	11:45 - 12:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-150	20:52		Ok
1367	Ørret170702_28	02/07/2017	11:45 - 12:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-150	20:59		Ok
1368	Harr170702_69	02/07/2017	11:45 - 12:15	Harr	Ned	4	File1-[2017-06-29_07-43-38]-150	23:05		Ok
1369	Harr170702_68	02/07/2017	11:45 - 12:15	Harr	Ned	4	File1-[2017-06-29_07-43-38			

1394	Harr170702_81	02/07/2017	13:15-13:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-153	16:18		Ok
1395	Harr170702_82	02/07/2017	13:15-13:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-153	16:19		Ok
1396	Harr170702_83	02/07/2017	13:15-13:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-153	16:19		Ok
1397	Harr170702_84	02/07/2017	13:15-13:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-153	16:22		Ok
1398	Ørret170828_7	02/07/2017	13:15-13:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-153	21:39	29:59	Ok
1399	Ørret170702_34	02/07/2017	13:15-13:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-153	29:10		Ok
1400	Harr170702_85	02/07/2017	13:15-13:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-153	29:29		Ok
1401	Harr170702_86	02/07/2017	13:15-13:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-153	29:29		Ok
1402	Ørret170828_7	02/07/2017	13:45-14:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-154	00:01	02:16; 12:00; 22:44; 28:50	Ok
1403	Sik170702_6	02/07/2017	13:45-14:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-154	04:32		Ok
1404	Harr170702_87	02/07/2017	13:45-14:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-154	08:32		Ok
1405	Harr170702_88	02/07/2017	13:45-14:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-154	09:26		Ok
1406	Ørret170828_7	02/07/2017	14:15-14:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-155	00:12		Ok
1407	Ørret170828_7	02/07/2017	14:15-14:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-155	06:11		Ok
1408	Ørret170702_34	02/07/2017	14:15-14:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-155	17:05		Ok
1409	Ørret170818_2	02/07/2017	14:15-14:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-155	17:19		Ok
1410	Ørret170818_2	02/07/2017	14:15-14:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-155	18:01		Ok
1411	Ørret170702_35	02/07/2017	14:15-14:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-155	18:24		Ok
1412	Harr170702_89	02/07/2017	14:15-14:45	Harr	Ned	4	File1-[2017-06-29_07-43-38]-155	26:38		Ok
1413	Ørret170629_16	02/07/2017	14:15-14:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-155	28:58		Ok
1414	Ørret170629_16	02/07/2017	14:45-15:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-156	00:14	01:11	Ok
1415	Harr170702_90	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	01:05		Ok
1416	Sik170702_6	02/07/2017	14:45-15:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-156	01:27	09:13	Ok
1417	Harr170702_91	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	01:40		Ok
1418	Ørret170702_36	02/07/2017	14:45-15:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-156	09:01		Ok
1419	Ørret170702_37	02/07/2017	14:45-15:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-156	09:04		Ok
1420	Harr170702_92	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	09:01		Ok
1421	Harr170702_93	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	09:04		Ok
1422	Harr170702_94	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	09:04		Ok
1423	Harr170702_95	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	09:06		Ok
1424	Harr170702_96	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	09:06		Ok
1425	Harr170702_97	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	09:06		Ok
1426	Harr170702_98	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	09:08		Ok
1427	Harr170702_99	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	09:08		Ok
1428	Harr170702_100	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	09:08		Ok
1429	Harr170702_101	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	09:08		Ok
1430	Harr170702_102	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	09:11		Ok
1431	Harr170702_103	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	09:11		Ok
1432	Ørret170702_38	02/07/2017	14:45-15:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-156	11:28		Ok
1433	Harr170702_104	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	11:29		Ok
1434	Harr170702_105	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	13:49		Ok
1435	Ørret170702_39	02/07/2017	14:45-15:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-156	16:50		Ok
1436	Harr170702_106	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	20:36		Ok
1437	Harr170702_107	02/07/2017	14:45-15:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-156	20:36		Ok
1438	Ørret170702_40	02/07/2017	14:45-15:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-156	29:35		Ok
1439	Lake170702_1	02/07/2017	15:15-15:45	Lake	Kam	4	File1-[2017-06-29_07-43-38]-157	05:16		Ok
1440	Ørret170629_16	02/07/2017	15:15-15:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-157	11:27		Ok
1441	Ørret170702_41	02/07/2017	15:15-15:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-157	20:13		Ok
1442	Harr170702_108	02/07/2017	15:45-16:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-158	07:19		Ok
1443	Harr170702_109	02/07/2017	15:45-16:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-158	07:19		Ok
1444	Sik170702_7	02/07/2017	15:45-16:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-158	13:12		Ok
1445	Harr170702_110	02/07/2017	15:45-16:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-158	20:58		Ok
1446	Ørret170702_42	02/07/2017	15:45-16:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-158	26:26		Ok
1447	Harr170702_111	02/07/2017	16:15-16:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-159	21:50		Ok
1448	Harr170702_112	02/07/2017	16:45-17:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-160	21:58		Ok
1449	Ørret170702_43	02/07/2017	16:45-17:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-160	23:55		Ok
1450	Sik170702_6	02/07/2017	16:45-17:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-160	25:36		Ok
1451	Sik170702_6	02/07/2017	17:15-17:45	Sik	Kam	4	File1-[2017-06-29_07-43-38]-161	02:25		Ok
1452	Ørret170629_16	02/07/2017	17:15-17:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-161	04:40		Ok
1453	Ørret170828_7	02/07/2017	17:15-17:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-161	08:11	13:29; 16:16; 25:52	Ok
1454	Ørret170702_44	02/07/2017	17:15-17:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-161	14:41		Ok
1455	Ørret170702_45	02/07/2017	17:15-17:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-161	14:51		Ok
1456	Ørret170828_7	02/07/2017	17:45-18:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-162	00:14	02:26; 04:22; 08:18	Ok
1457	Ørret170702_46	02/07/2017	17:45-18:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-162	02:24		Ok
1458	Ørret170702_47	02/07/2017	17:45-18:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-162	05:27		Ok
1459	Ørret170702_48	02/07/2017	17:45-18:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-162	07:18		Ok
1460	Ørret170702_49	02/07/2017	17:45-18:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-162	14:11		Ok
1461	Ørret170702_50	02/07/2017	17:45-18:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-162	15:40		Ok
1462	Ørret170828_7	02/07/2017	17:45-18:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-162	19:16		Ok
1463	Ørret170702_51	02/07/2017	17:45-18:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-162	21:52		Ok
1464	Sik170702_6	02/07/2017	17:45-18:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-162	22:22	25:25	Ok
1465	Ørret170702_52	02/07/2017	17:45-18:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-162	24:29		Ok
1466	Sik170702_6	02/07/2017	18:15-18:45	Sik	Kam	4	File1-[2017-06-29_07-43-38]-163	00:25		Ok
1467	Harr170702_113	02/07/2017	18:15-18:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-163	00:56		Ok
1468	Harr170702_114	02/07/2017	18:15-18:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-163	00:56		Ok
1469	Harr170702_115	02/07/2017	18:15-18:45	Harr	Ned	4	File1-[2017-06-29_07-43-38]-163	08:37		Ok
1470	Harr170702_116	02/07/2017	18:15-18:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-163	09:04		Ok
1471	Harr170702_117	02/07/2017	18:15-18:45	Harr	Ned	4	File1-[2017-06-29_07-43-38]-163	10:21		Ok
1472	Ørret170828_7	02/07/2017	18:15-18:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-163	10:30		Ok
1473	Harr170702_118	02/07/2017	18:15-18:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-163	11:59		Ok
1474	Harr170702_119	02/07/2017	18:15-18:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-163	11:59		Ok
1475	Harr170702_118	02/07/2017	18:15-18:45	Harr	Ned	4	File1-[2017-06-29_07-43-38]-163	12:35		Ok
1476	Harr170702_119	02/07/2017	18:15-18:45	Harr	Ned	4	File1-[2017-06-29_07-43-38]-163	12:35		Ok
1477	Harr170702_118	02/07/2017	18:15-18:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-163	12:57		Ok
1478	Harr170702_119	02/07/2017	18:15-18:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-163	12:57		Ok
1479	Ørret170702_53	02/07/2017	18:15-18:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-163	13:18		Ok
1480	Harr170702_120	02/07/2017	18:15-18:45	Harr	Ned	4	File1-[2017-06-29_07-43-38]-163	16:24		Ok
1481	Harr170702_121	02/07/2017	18:15-18:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-163	17:53		Ok
1482	Harr170702_122	02/07/2017	18:15-18:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-163	17:53		Ok
1483	Harr170702_123	02/07/2017	18:15-18:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-163	17:53		Ok
1484	Ørret170828_7	02/07/2017	18:45-19:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-164	18:48		Ok
1485	Ørret170828_7	02/07/2017	18:45-19:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-164	21:34		Ok
1486	Ørret170702_53	02/07/2017	18:45-19:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-164</			

1510	Ørret170703_3	03/07/2017	05:15-05:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-185	05:03	18:54; 24:45; 27:39	Ok
1511	Ørret170703_4	03/07/2017	05:15-05:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-185	05:05		Ok
1512	Fish170703_2	03/07/2017	05:15-05:45	Fish	Opp	4	File1-[2017-06-29_07-43-38]-185	11:16		Ok
1513	Harr170703_3	03/07/2017	05:15-05:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-185	25:06		Ok
1514	Harr170703_4	03/07/2017	05:15-05:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-185	25:08		Ok
1515	Ørret170703_3	03/07/2017	05:45-06:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-186	11:42	17:29	Ok
1516	Fish170703_3	03/07/2017	05:45-06:15	Fish	Ned	4	File1-[2017-06-29_07-43-38]-186	16:38		Ok
1517	Ørret170703_5	03/07/2017	05:45-06:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-186	17:44		Ok
1518	Ørret170703_6	03/07/2017	05:45-06:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-186	21:52		Ok
1519	Sik170702_6	03/07/2017	06:15-06:45	Sik	Kam	4	File1-[2017-06-29_07-43-38]-187	06:40		Ok
1520	Ørret170703_7	03/07/2017	06:15-06:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-187	09:58		Ok
1521	Ørret170703_3	03/07/2017	06:15-06:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-187	12:37	19:01	Ok
1522	Ørret170703_3	03/07/2017	06:45-07:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-188	00:00	04:31; 13:54; 22:05	Ok
1523	Harr170703_5	03/07/2017	06:45-07:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-188	14:34		Ok
1524	Ørret170703_3	03/07/2017	07:15-07:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-189	00:39		Ok
1525	Ørret170703_8	03/07/2017	07:15-07:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-189	05:55		Ok
1526	Ørret170703_9	03/07/2017	07:15-07:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-189	07:39		Ok
1527	Harr170703_6	03/07/2017	07:15-07:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-189	14:07		Ok
1528	Harr170703_7	03/07/2017	07:15-07:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-189	14:07		Ok
1529	Harr170703_8	03/07/2017	07:15-07:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-189	14:07		Ok
1530	Ørret170703_9	03/07/2017	07:15-07:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-189	24:58		Ok
1531	Lake170703_1	03/07/2017	07:15-07:45	Lake	Ned	4	File1-[2017-06-29_07-43-38]-189	28:47		Ok
1532	Harr170703_9	03/07/2017	07:45-08:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-190	05:41		Ok
1533	Ørret170703_9	03/07/2017	07:45-08:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-190	09:19		Ok
1534	Ørret170703_10	03/07/2017	07:45-08:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-190	22:30		Ok
1535	Sik170702_6	03/07/2017	07:45-08:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-190	23:12		Ok
1536	Ørret170703_11	03/07/2017	07:45-08:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-190	25:10		Ok
1537	Ørret170703_12	03/07/2017	07:45-08:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-190	29:54		Ok
1538	Ørret170703_13	03/07/2017	08:15-08:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-191	01:16		Ok
1539	Harr170703_10	03/07/2017	08:15-08:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-191	04:48		Ok
1540	Ørret170703_12	03/07/2017	08:15-08:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-191	04:59		Ok
1541	Sik170702_6	03/07/2017	08:15-08:45	Sik	Kam	4	File1-[2017-06-29_07-43-38]-191	04:59		Ok
1542	Fish170703_4	03/07/2017	08:15-08:45	Fish	Ned	4	File1-[2017-06-29_07-43-38]-191	06:27		Ok
1543	Ørret170703_12	03/07/2017	08:15-08:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-191	17:53		Ok
1544	Ørret170703_3	03/07/2017	08:15-08:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-191	17:53	29:21	Ok
1545	Ørret170703_14	03/07/2017	08:15-08:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-191	17:53		Ok
1546	Ørret170703_3	03/07/2017	08:45-09:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-192	00:06	01:13; 06:10; 12:52; 23:42	Ok
1547	Sik170702_6	03/07/2017	08:45-09:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-192	00:40		Ok
1548	Harr170703_13	03/07/2017	08:45-09:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-192	12:43		Ok
1549	Ørret170703_15	03/07/2017	08:45-09:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-192	13:29		Ok
1550	Ørret170703_16	03/07/2017	08:45-09:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-192	28:10		Ok
1551	Ørret170703_3	03/07/2017	09:15-09:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-193	01:14	07:11; 10:42; 22:22; 28:27	Ok
1552	Harr170703_14	03/07/2017	09:15-09:45	Harr	Ned	4	File1-[2017-06-29_07-43-38]-193	01:50		Ok
1553	Ørret170703_12	03/07/2017	09:15-09:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-193	07:45		Ok
1554	Ørret170703_17	03/07/2017	09:15-09:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-193	08:05		Ok
1555	Ørret170703_17	03/07/2017	09:15-09:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-193	08:09		Ok
1556	Ørret170703_18	03/07/2017	09:15-09:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-193	13:31		Ok
1557	Ørret170703_19	03/07/2017	09:15-09:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-193	14:07		Ok
1558	Ørret170703_20	03/07/2017	09:15-09:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-193	14:54		Ok
1559	Ørret170703_3	03/07/2017	09:45-10:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-194	00:36	03:28; 06:31; 11:01; 14:20; 16:58	Ok
1560	Harr170703_15	03/07/2017	09:45-10:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-194	04:20		Ok
1561	Sik170703_1	03/07/2017	09:45-10:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-194	22:54	25:33	Ok
1562	Ørret170703_20	03/07/2017	09:45-10:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-194	24:14		Ok
1563	Sik170702_6	03/07/2017	09:45-10:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-194	25:33		Ok
1564	Ørret170629_16	03/07/2017	09:45-10:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-194	27:22		Ok
1565	Ørret170629_16	03/07/2017	10:15-10:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-195	00:18		Ok
1566	Ørret170703_21	03/07/2017	10:15-10:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-195	06:28		Ok
1567	Ørret170703_22	03/07/2017	10:15-10:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-195	06:31		Ok
1568	Ørret170703_23	03/07/2017	10:15-10:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-195	24:31		Ok
1569	Ørret170703_3	03/07/2017	10:45-11:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-196	01:30	07:19; 16:38; 20:36; 22:37; 28:00	Ok
1570	Ørret170703_24	03/07/2017	10:45-11:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-196	28:00		Ok
1571	Ørret170703_3	03/07/2017	11:15-11:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-197	01:40	13:22; 20:43	Ok
1572	Harr170703_16	03/07/2017	11:15-11:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-197	01:44		Ok
1573	Harr170703_17	03/07/2017	11:15-11:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-197	01:45		Ok
1574	Ørret170703_25	03/07/2017	11:15-11:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-197	12:40		Ok
1575	Ørret170703_25	03/07/2017	11:15-11:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-197	14:21		Ok
1576	Ørret170703_25	03/07/2017	11:15-11:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-197	15:22		Ok
1577	Ørret170703_26	03/07/2017	11:15-11:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-197	18:25		Ok
1578	Ørret170703_12	03/07/2017	11:15-11:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-197	23:48		Ok
1579	Ørret170629_16	03/07/2017	11:15-11:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-197	23:53		Ok
1580	Harr170703_18	03/07/2017	11:15-11:45	Harr	Opp	4	File1-[2017-06-29_07-43-38]-197	26:00		Ok
1581	Ørret170703_27	03/07/2017	11:15-11:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-197	26:44		Ok
1582	Ørret170703_28	03/07/2017	11:15-11:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-197	26:44		Ok
1583	Fish170703_5	03/07/2017	11:45-12:15	Fish	Kam	4	File1-[2017-06-29_07-43-38]-198	10:48	18:52	Ok
1584	Ørret170818_2	03/07/2017	11:45-12:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-198	17:57		Ok
1585	Fish170703_6	03/07/2017	11:45-12:15	Fish	Ned	4	File1-[2017-06-29_07-43-38]-198	18:09		Ok
1586	Ørret170703_29	03/07/2017	12:15-12:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-199	17:09		Ok
1587	Ørret170703_29	03/07/2017	12:15-12:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-199	18:05		Ok
1588	Ørret170703_3	03/07/2017	12:45-13:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-200	06:23	10:13; 16:21	Ok
1589	Harr170703_19	03/07/2017	12:45-13:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-200	12:27		Ok
1590	Ørret170818_2	03/07/2017	12:45-13:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-200	15:37	16:28	Ok
1591	Ørret170703_12	03/07/2017	12:45-13:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-200	15:55		Ok
1592	Ørret170703_3	03/07/2017	12:45-13:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-200	16:28		Ok
1593	Ørret170818_2	03/07/2017	12:45-13:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-200	16:40		Ok
1594	Ørret170703_3	03/07/2017	12:45-13:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-200	18:22	18:50; 27:53	Ok
1595	Ørret170703_3	03/07/2017	12:45-13:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-200	29:56		Ok
1596	Ørret170703_30	03/07/2017	13:15-13:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-201	00:11		Ok
1597	Fish170703_7	03/07/2017	13:15-13:45	Fish	Ned	4	File1-[2017-06-29_07-43-38]-201	01:50		Ok
1598	Ørret170629_16	03/07/2017	13:15-13:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-201	03:08		Ok
1599	Ørret170629_16	03/07/2017	13:15-13:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-201	08:45	15:00; 26:03	Ok
1600	Ørret170703_31	03/07/2017	13:15-13:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-201	14:56		Ok
1601	Ørret170703_32									

1626	Ørret170703_43	03/07/2017	15:15 - 15:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-205	10:27		Ok
1627	Ørret170703_44	03/07/2017	15:15 - 15:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-205	25:00		Ok
1628	Ørret170703_44	03/07/2017	15:15 - 15:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-205	25:22		Ok
1629	Ørret170703_44	03/07/2017	15:15 - 15:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-205	25:27		Ok
1630	Ørret170703_45	03/07/2017	15:15 - 15:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-205	26:24		Ok
1631	Ørret170703_46	03/07/2017	15:15 - 15:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-205	29:34		Ok
1632	Ørret170703_47	03/07/2017	15:45 - 16:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-206	00:11		Ok
1633	Ørret170703_48	03/07/2017	15:45 - 16:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-206	04:17		Ok
1634	Ørret170703_49	03/07/2017	15:45 - 16:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-206	11:37		Ok
1635	Harr170703_23	03/07/2017	15:45 - 16:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-206	11:39		Ok
1636	Harr170703_24	03/07/2017	15:45 - 16:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-206	11:42		Ok
1637	Harr170703_25	03/07/2017	15:45 - 16:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-206	11:43		Ok
1638	Harr170703_26	03/07/2017	15:45 - 16:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-206	11:43		Ok
1639	Harr170703_27	03/07/2017	15:45 - 16:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-206	11:45		Ok
1640	Ørret170703_50	03/07/2017	15:45 - 16:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-206	12:53		Ok
1641	Harr170703_28	03/07/2017	15:45 - 16:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-206	13:49		Ok
1642	Fish170703_9	03/07/2017	15:45 - 16:15	Fish	Opp	4	File1-[2017-06-29_07-43-38]-206	15:37		Ok
1643	Ørret170828_7	03/07/2017	15:45 - 16:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-206	18:36	28:46	Ok
1644	Harr170703_29	03/07/2017	15:45 - 16:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-206	22:52		Ok
1645	Harr170703_30	03/07/2017	15:45 - 16:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-206	23:06		Ok
1646	Ørret170703_51	03/07/2017	15:45 - 16:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-206	23:45		Ok
1647	Ørret170828_7	03/07/2017	16:15 - 16:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-207	00:35	10:54; 28:02	Ok
1648	Ørret170703_51	03/07/2017	16:15 - 16:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-207	04:38		Ok
1649	Ørret170703_52	03/07/2017	16:15 - 16:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-207	12:01		Ok
1650	Ørret170828_7	03/07/2017	16:45 - 17:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-208	00:01	16:27	Ok
1651	Ørret170828_7	03/07/2017	16:45 - 17:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-208	17:14		Ok
1652	Ørret170828_7	03/07/2017	16:45 - 17:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-208	17:39	22:54	Ok
1653	Ørret170828_7	03/07/2017	17:15 - 17:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-209	00:23	05:23; 07:24; 12:14; 26:51	Ok
1654	Ørret170703_53	03/07/2017	17:15 - 17:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-209	04:34		Ok
1655	Ørret170703_53	03/07/2017	17:15 - 17:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-209	04:43		Ok
1656	Sik170703_2	03/07/2017	17:15 - 17:45	Sik	Opp	4	File1-[2017-06-29_07-43-38]-209	14:13		Ok
1657	Sik170702_6	03/07/2017	17:15 - 17:45	Sik	Kam	4	File1-[2017-06-29_07-43-38]-209	15:28		Ok
1658	Ørret170703_54	03/07/2017	17:15 - 17:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-209	21:09		Ok
1659	Ørret170703_55	03/07/2017	17:15 - 17:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-209	22:06		Ok
1660	Ørret170703_55	03/07/2017	17:15 - 17:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-209	22:16		Ok
1661	Ørret170828_7	03/07/2017	17:45 - 18:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-210	00:09	15:33; 20:19	Ok
1662	Harr170703_30	03/07/2017	17:45 - 18:15	Harr	Ned	4	File1-[2017-06-29_07-43-38]-210	00:11		Ok
1663	Ørret170703_3	03/07/2017	17:45 - 18:15	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-210	19:10		Ok
1664	Ørret170703_56	03/07/2017	17:45 - 18:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-210	19:40		Ok
1665	Ørret170703_57	03/07/2017	17:45 - 18:15	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-210	25:53		Ok
1666	Ørret170828_7	03/07/2017	18:15 - 18:45	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-211	00:56	06:12; 16:57	Ok
1667	Ørret170703_58	03/07/2017	18:15 - 18:45	Ørret	Ned	4	File1-[2017-06-29_07-43-38]-211	12:52		Ok
1668	Sik170703_3	03/07/2017	18:15 - 18:45	Sik	Kam	4	File1-[2017-06-29_07-43-38]-211	12:57		Ok
1669	Ørret170828_7	03/07/2017	18:15 - 18:45	Ørret	Opp	4	File1-[2017-06-29_07-43-38]-211	19:43		Ok
1670	Ørret170828_7	03/07/2017	18:45 - 19:15	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-212	00:51	06:05; 22:06; 28:51	Ok
1671	Sik170702_6	03/07/2017	18:45 - 19:15	Sik	Kam	4	File1-[2017-06-29_07-43-38]-212	17:31		Ok
1672	Harr170703_31	03/07/2017	18:45 - 19:15	Harr	Opp	4	File1-[2017-06-29_07-43-38]-212	25:54		Ok
1673	Ørret170828_7	03/07/2017	19:15 - 19:34	Ørret	Kam	4	File1-[2017-06-29_07-43-38]-213	00:07	03:24	Ok

Utførelse

Dette vedlegget vil ta for seg prosessen gruppen har gått gjennom i løpet av prosjektet. Vedlegget kan sees på som en historikk over gjennomført arbeid. Hvert hovedpunkt vil ta for seg en større periode hvor hvert underpunkt vil bestå av mindre og mer spesifikke perioder. Antall logget timer vil tilsi hvor mange timer som er ført inn i *Toggl* [63].

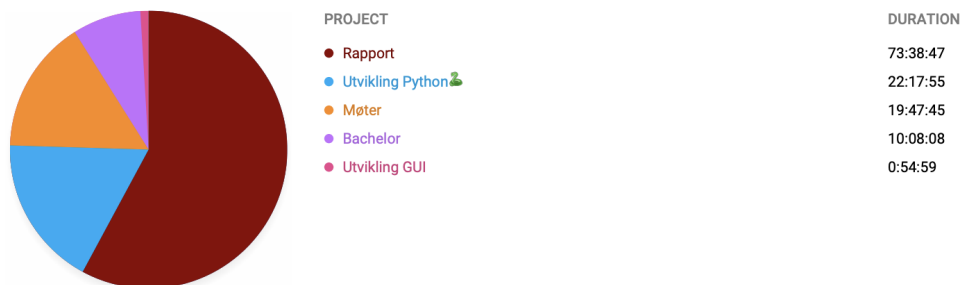
Oppstart

Forprosjekt

Periode: 7. Januar - 1. Februar

Antall logget timer: 124:58:13

Denne perioden begynte med møte med veileder, Marius Pedersen, før gruppen skrev forprosjektet og gjennomførte møte med NINA den 29.01.2020 for å snakke om oppgaven. Mye generelle diskusjoner og valg av verktøy ble gjort i denne perioden. Oppstart med bruk av Toggl og Trello, valg av Python, PyCharm og PyQt som utviklingsverktøy. Tidlig eksperimentering med ulike bildeprosessering og detekteringsmetoder.



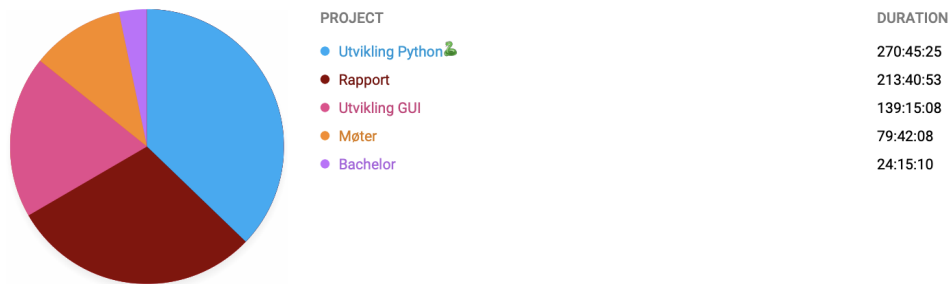
Figur E.1: Antall logget timer for *Forprosjekt* fordelt på prosjekter.

Scrum-prosessen

Periode: 21. Januar - 19. Mai

Antall logget timer:

Gjennomgang av hva som er gjort i hver sprint i korte trekk



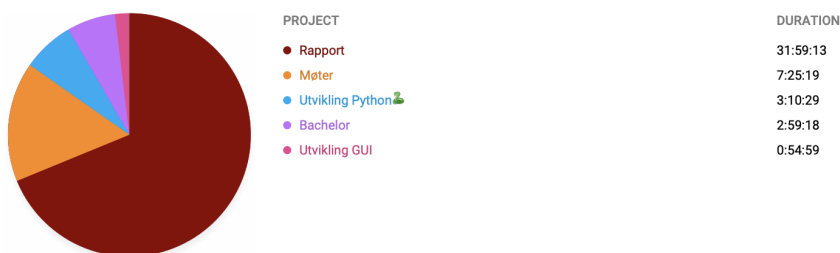
Figur E.2: Antall logget timer for *Scrum-prosessen* fordelt på prosjekter.

Sprint 1

Periode: 21. Januar - 27. Januar

Antall logget timer: 46:29:18

Hovedfokuset denne perioden var å jobbe med forprosjektrapporten og gjøre noe tidlig utforskning av prosesserings- og deteksjonsmetoder. Rapporten ble skrevet ferdig, avtalen ble undertegnet av gruppen, veileder og NINA.



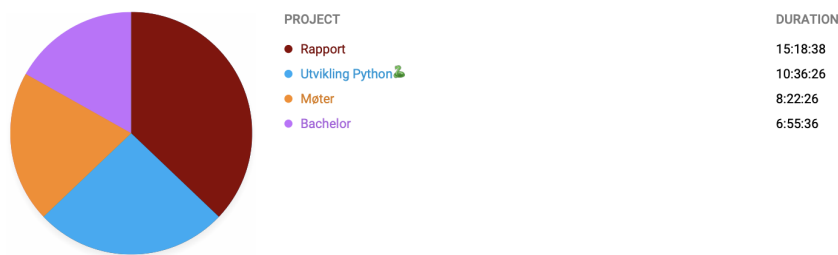
Figur E.3: Antall logget timer for *Sprint 1* fordelt på prosjekter.

Sprint 2

Periode: 27. Januar - 3. Februar

Antall logget timer: 41:23:45

En andre iterasjon av forprosjektet ble gjennomført. Korrekturlesning og ferdigstilling av rapporten rett før innlevering. Siste periode av sprinten ble brukt til noen tidlige forsøk på å sette sammen prosesseringer og deteksjoner til å jobbe sammen.



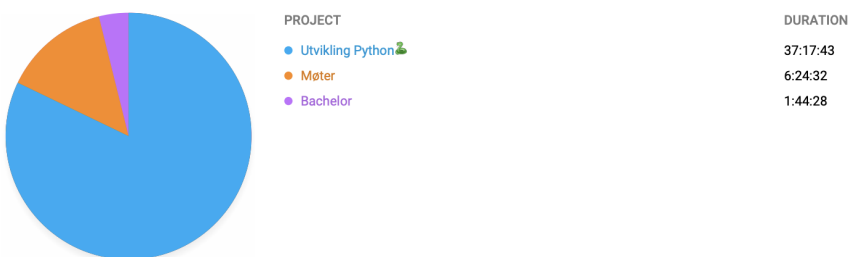
Figur E.4: Antall logget timer for *Sprint 2* fordelt på prosjekter.

Sprint 3

Periode: 3. Februar - 10. Februar

Antall logget timer: 42:41:20

Sprint 3 begynte utviklingen for fullt og gruppen ble rådet av veileder om å gå gjennom ulike preprosesseringsmetoder og eksperimentere med ferdiglage prosjekter i MATLAB. Gruppen satte av mye tid for egenlæring og oppsett av utviklingsmiljøer i denne sprinten. Store deler av utviklingsfokus gikk på utvikling av datasett og utvikling av enhetstester for den allerede implementerte funksjonaliteten.



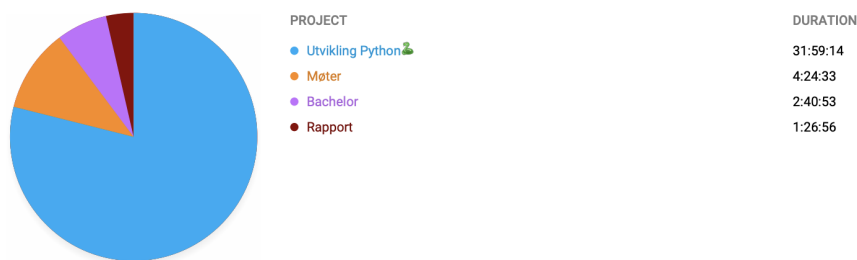
Figur E.5: Antall logget timer for *Sprint 3* fordelt på prosjekter.

Sprint 4

Periode: 10. Februar - 17. Februar

Antall logget timer: 34:45:30

Sprint 4 begynte med videre utvikling av datasett hvor de 2 nye diskene som ble utlevert av NINA på møtet den 29. Januar ble gjennomgått. Dette datasettet var ikke tidligere gjennomgått av NINA så gruppen ble her nødt til å bruke mer tid på å gå gjennom videomaterialet for å kunne laget et godt datasett. Gruppen begynte utviklingen av en automatisert testsuite for å gjøre det mulig å kjøre mange tester med ulike parametre og hente ut statistikk sammenhengende.



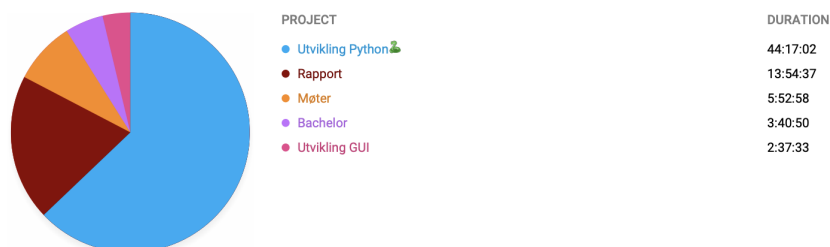
Figur E.6: Antall logget timer for *Sprint 4* fordelt på prosjekter.

Sprint 5

Periode: 17. Februar - 2. Mars

Antall logget timer: 70:23:00

Sprint 5 har hovedsakelig bestått av utvikling av testsett. Ferdigstilling av utviklingssettet og hva som skulle inngå i testsettet har stått sentralt. I tillegg har en god del tid vært satt av til å skrive på rapporten for å komme i gang med denne. Grunnet flytting av veiledningsmøte og karrieredag så ble sprinten forlenget. Dette var for å at vi skulle få fullført en del av oppgaven som var planlagt, men også fordi en ny sprint på veldig få dager ville vært tilnærmet lik meningsløst.



Figur E.7: Antall logget timer for *Sprint 5* fordelt på prosjekter.

Sprint 6

Periode: 2. Mars - 9. Mars

Antall logget timer: 37:50:37

Sprint 6 har hovedsakelig bestått av videre arbeid med CVAT, testsuite og innsanking av kunnskap om GUI. CVAT har tatt lengre tid enn forventet grunnet diverse problematikk og for lav tidsbruk. For å kunne arbeide med GUI, så har gruppen satt av tid til å sette seg inn i hvordan PyQt5 fungerer og benyttet seg av ulike kilder på nettet for å få en forståelse av hvordan utviklingen bør foregå. I tillegg så har gruppen laget en skisse og en prototype for å visualisere ønsket brukergrensesnitt.



Figur E.8: Antall logget timer for *Sprint 6* fordelt på prosjekter.

Sprint 7

Periode: 9. Mars - 16. Mars

Antall logget timer: 43:30:23

Sprint 7 har hovedsakelig bestått av videre arbeid med CVAT og utvikling av GUI. Gruppen har i Sprint 7 fokusert på å utvikle GUI-en som deteksjonen skal gjøres gjennom. Store deler av arbeidet har gått til innsanking av kunnskap for å få en bedre forståelse for utvikling i PyQt. Prototypen har også blitt revidert etter innspill fra veileder og diskusjon i gruppen.



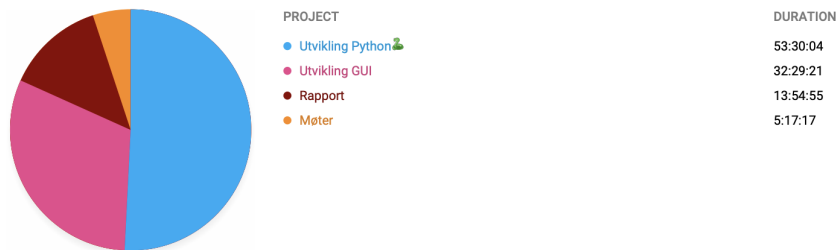
Figur E.9: Antall logget timer for *Sprint 7* fordelt på prosjekter.

Sprint 8

Periode: 16. Mars - 1. April

Antall logget timer: 105:11:37

Sprint 8 så ble arbeidet med å utvikle det grafiske brukergrensesnittet videre fokuset for Thomas og GUI-en er nå tilnærmet lik ferdigstilt. For Vebjørn så har Sprint 8 hovedsakelig bestått av mye arbeid med CVAT. Etter 16-17 av de 20 testsettfilene var annoterte, oppstod et problem med CVAT slik at alt arbeid utover de 12 første filene gikk tapt. Utvikling av analysescript av resultatfiler har blitt utført, men har vist seg å ha betydelige problemer med ukjent opphav.



Figur E.10: Antall logget timer for *Sprint 8* fordelt på prosjekter.

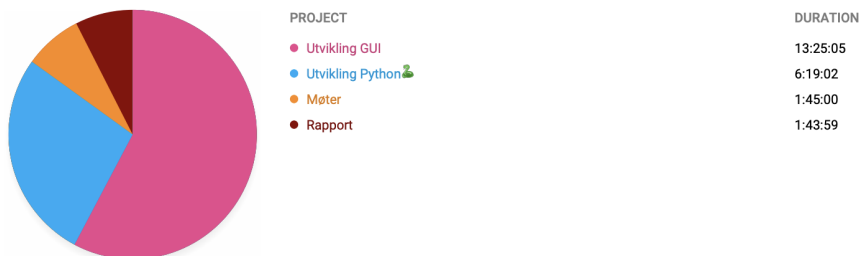
Sprint 9

Periode: 1. April - 14. April

Antall logget timer: 23:13:06

Sprint 9 så ble hovedfokuset å jobbe videre med ferdigstilling av CVAT og GUI-en. Brukerguide er påbegynt for GUI-en og kun mindre problemer rundt den kjørbarefilen for Windows gjenstår før GUI-en er fullført for versjon 1.0. Forsøk på recovery av resterende CVAT-filer ble ikke vellykket. Annotering av utviklingssett er ferdig, resterende annotering av testsettet er outsourcet.

Grunnet påskeferie ble sprinten strukket ut over et lengre tidsrom med en lavere forventning til antall loggete timer.



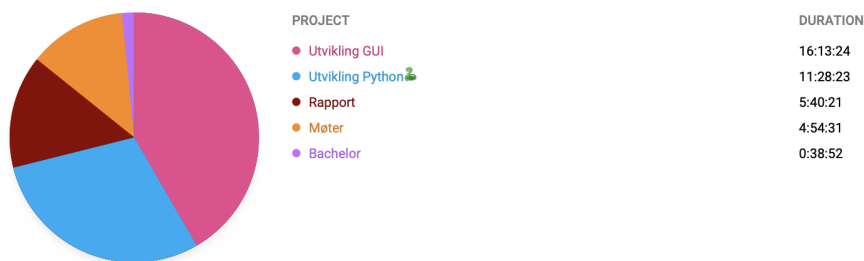
Figur E.11: Antall logget timer for *Sprint 9* fordelt på prosjekter.

Sprint 10

Periode: 14. April - 21. April

Antall logget timer: 38:55:31

Sprint 10 så ble hovedfokuset å jobbe videre med ferdigstilling av resultatanalyse og GUI-en. Brukerguide er nesten ferdig og mindre problemer knytte til GUI-en og den kjørbare filen er fikset. GUI-en er for øyeblikket i versjon 1.2 etter at disse endringene har blitt gjort. Omskriving av resultatanalyse er tilnærmet ferdig, trenger kun input fra veileder på hvordan den endelige scoren skal regnes ut/se ut. Rapporten har fått en midlertidig struktur som skal arbeides på av gruppen sammen med veileder. Gruppen har også fokusert på å skrive videre på rapporten og fylle inn fra arbeid fra foregående sprintene, som er blitt gjort, men ikke dokumentert.



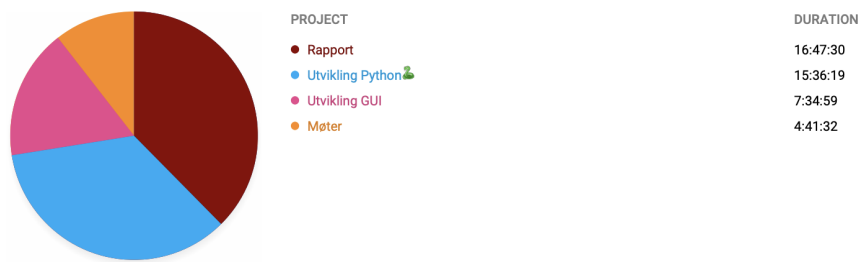
Figur E.12: Antall logget timer for *Sprint 10* fordelt på prosjekter.

Sprint 11

Periode: 21. April - 28. April

Antall logget timer: 44:40:20

Sprint 11 så ble hovedfokuset å jobbe videre med ferdigstilling av resultatanalyse og GUI-en. GUI-en har fått mindre justeringer og forbedringer, og er nå i versjon 1.3 etter at disse endringene har blitt gjort. Rapporten har fått mer og mer fokus utover sprinten og begynner nå å ta form. Rapportens utvikling vil diskuteres med veileder for videre input.



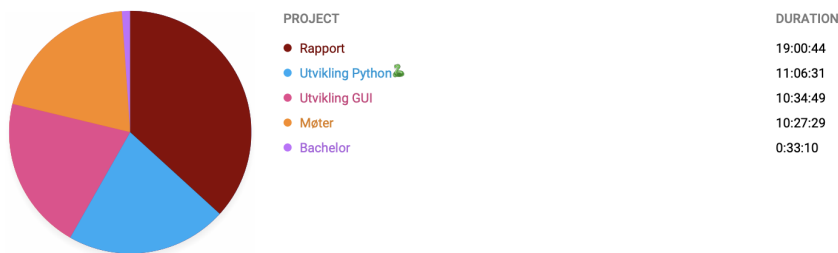
Figur E.13: Antall logget timer for *Sprint 11* fordelt på prosjekter.

Sprint 12

Periode: 28. April - 05. Mai

Antall logget timer: 51:42:43

Sprinten har gått med til ferdigstilling av førsteutkast om bildebehandling og datasett. Resultatanalyse outputter nå et bilde med boksplokk av jaccard index for alle deteksjoner gjort i korrekt frame, separat for hver videofil. Scriptet er laget men ikke testet for testsettet.



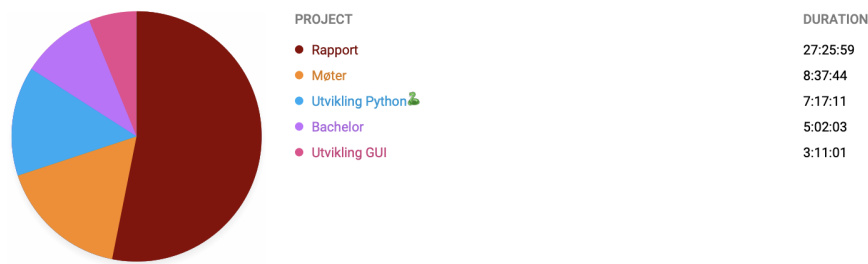
Figur E.14: Antall logget timer for *Sprint 12* fordelt på prosjekter.

Sprint 13

Periode: 05. Mai - 12. Mai

Antall logget timer: 51:33:58

Sprinten har gått med til ferdigstilling og revidering av en del kapitler i rapporten. Brukertesting av NINA og andre brukere har hatt høy prioritet og vært meget verdifull for gruppen. Tilbakemeldingene fått fra brukertesting har blitt vurdert av gruppen, og flere endringer i applikasjonen har derfor blitt gjort. Litt tid har gått til utvikling på output av confusion matrix og forenkling av testoppsettet for rammesammenligninger. Rapport om datasett er tilnærmet skrevet ferdig, seksjon om testing er to av tre seksjoner ferdig.



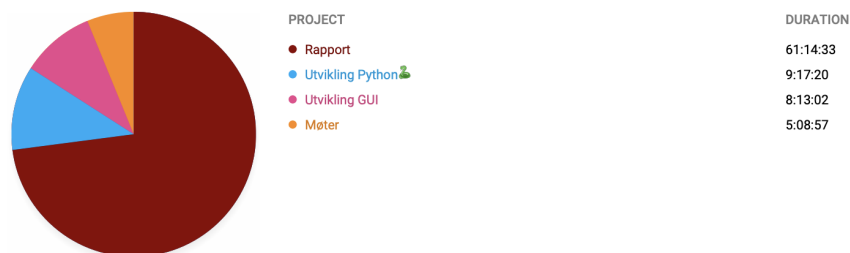
Figur E.15: Antall logget timer for *Sprint 13* fordelt på prosjekter.

Sprint 14

Periode: 12. Mai - 20. Mai

Antall logget timer: 61:14:33

Sprinten har gått med til ferdigstilling av rapporten og oppgaven i sin helhet. Gruppen har vært i dialog med NINA for å sikre at terminologi er i henhold til deres ønske. GUI og Brukerguiden har fått en siste revidering og blitt ferdigstilt for innlevering.



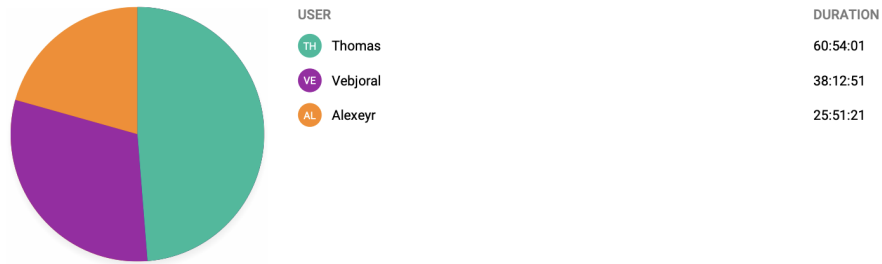
Figur E.16: Antall logget timer for *Sprint 14* fordelt på prosjekter.

Tidsbruk

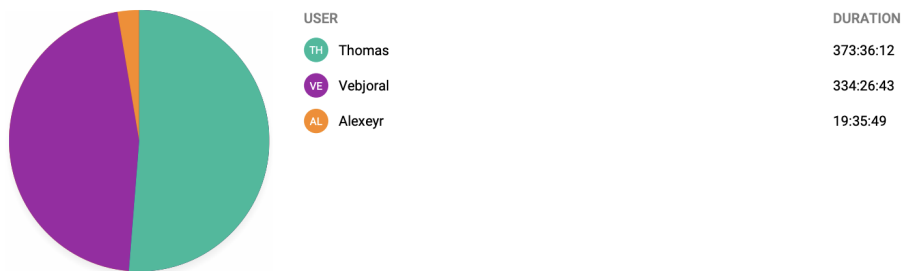
For prosjektet så valgte gruppen å benytte seg av *Toggl* [63] for å holde oversikt over tidsbruken under hele prosjektet. Programmet ble brukt for å sørge for at alle medlemmer la ned ønsket mengde med timer hver uke, men også for å kontrollere at gruppen ikke brukte for mye tid på deler som ikke var like essensielle.

Gruppen ble enige om å kun føre effektive timer. En reell arbeidstid på 75% av total tid ble ansett som fornuftig og mulig å overholde. Dette førte til at kravet på 30 timer per person i uken ville tilsvare 22,5 time logget på *Toggl*. I begynnelsen var gruppen ikke like gode på å føre timer da programmet ikke var fullstendig satt opp internt og det ennå var litt usikkerhet rundt hva som skulle logges og hvordan.

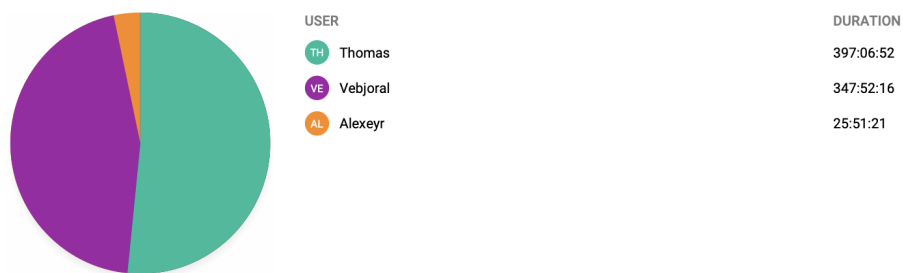
Gruppen bestod av 3 personer frem til Forprosjektet var ferdig og antall timer som ble logget per uke sank derfor etter reduksjonen i antall medlemmer. Totalt endte gruppen på 770 timer med effektivt arbeid for hele bachelorprosjektet.



Figur E.17: Antall logget timer for *Forprosjektet*.



Figur E.18: Antall logget timer for *Scrum-prosessen*.



Figur E.19: Antall logget timer for *Bachelorprosjektet*.

Brukertesting Referat

Testperson 1

Bruker 1 testet ut versjon 1.5 av programvaren. Brukeren oppfattet ikonet, utformet som en mappe, som intuitiv nok til å forstå at ikonet må trykkes på for å velge filer som skal benyttes. Etter at filene var lastet inn så reagerte brukeren på at ved valg av modus for deteksjonen, så var det mulig å velge begge modusene samtidig, og mente at avkryssingsboksene kanskje heller skulle være radioknapper for at kun én modus skulle være mulig. Brukeren satte spørsmål til hvorfor modusen «Sakte» var valgt som standard ved oppstart av programmet. Brukeren opplevde også at boksene til knappene muligens var litt for små og ønsket derfor større bokser. I tillegg til at de ble opplevd som små så ytret brukeren et ønske om farge på knappen som skal være primærhandling, men også et ønske om å rokere om noen av knappene. Utenom ønskene ytret ovenfor så syntes brukeren at applikasjonen så fin og strukturert ut. Da brukeren ble bedt om å gjøre et oppslag i brukerguiden som er tilgjengelig i applikasjonen, så opplevde brukeren at det var vanskelig å oppdage brukerguiden som er illustrert som et spørsmålstegn og brukte derfor en del tid på å finne frem til brukerguiden. Brukeren kunne se for seg å ta i bruk applikasjonen til daglig og syntes applikasjonen var enkel å ta i bruk etter 15 minutter med bruk.

Testperson 2

Bruker 2 opplevde at enkelte tekstelementer i applikasjonen ble krympet grunnet skjermforholdene som maskinen til brukeren kjørte med. Brukeren kjørte versjon 1.5 av applikasjonen, samme som bruker 1. Brukeren brukte kort tid på å komme frem til at mappen skulle benyttes for å kunne laste inn filer til applikasjonen, men overser muligheten til å velge modus før «Kjør deteksjon» ble valgt. Brukeren opplevde derfor å få en melding om beskjed om at «én modus må velges først», noe brukeren gjorde. Brukeren kommenterte at teksten i meldingsboksen var litt vel liten. Da brukeren skulle forsøke å starte brukerguiden fra applikasjonen så var brukeren veldig fokusert på nedre halvdel av applikasjonen og brukte derfor en god del tid på å finne frem til brukerguiden som er plassert i det øvre venstre hjørnet. Brukeren ga uttrykk for et ønske om en meny for de ulike valgene som kan gjøres for deteksjonen, men også at de kunne gjøres mer intuitive ved bruk av symboler. Brukeren forvekslet «Kjør testing» og «Kjør deteksjon» knappen ved forespørsel om å kjøre deteksjon på noen valgte filer, opplevde at «Kjør deteksjon» ikke var godt nok fremhevet som en primærhandling. Ved gjentatt bruk så opplevde brukeren at applikasjonen var enkel å ta i bruk og oppfattet applikasjonen som intuitiv når større deler av applikasjonen hadde blitt utforsket.

Testperson 3

Bruker 3 var første personen til å teste ut versjon 1.6 av applikasjonen. Versjonen introduserte en del forbedringer som de tidligere brukerne hadde ytret et ønske om. Brukeren oppfattet nesten umiddelbart at mappen måtte velges for å laste inn filer til applikasjonen. Brukeren navigerte hurtig til de ulike valgene som kan tas før deteksjonen kjøres. Brukeren stilte spørsmål hva som foregikk da programmet kjøre modus «Hurtig» uten at utviklingsmodus var aktivert og skulle gjerne hatt en respons på hva applikasjonen gjorde underveis. Knappen for å kjøre deteksjonen ble hurtig valgt av brukeren og ble ved senere anledning fremhevet av brukeren som veldig enkel å ta i bruk med den nye fargen som skiller den fra de andre knappene. Bruker 3 brukte 15 sekunder på å finne frem til brukerguiden, noe som var betydelig hurtigere enn bruker 1 og 2. Brukeren oppfattet designet i applikasjonen som veldig behagelig og syntes det var et veldig fint brukergrensesnitt. Eneste ønsket brukeren hadde til applikasjonen var å legge på tekst på mappen som skal symbolisere valg av fil, og at en forklarende tekst som «Hjelp» eller «Søk» ville fungert fint som en erstatning for spørsmålstegn-symbolet som brukerguiden er tilknyttet.

Testperson 4

Bruker 4 opplevde applikasjonen som meget intuitiv og var den brukeren som benyttet minst tid til å utføre de ulike oppgavene som brukeren ble tildelt. Bruker 4 testet også ut versjon 1.6 av applikasjonen. Brukeren ytret et ønske om «Drag & Drop»-funksjonalitet da brukeren skulle laste inn filer, men syntes at applikasjonen fungerte bra slik den var. Ved kjøring av deteksjonen uten utviklingsmodus aktivert, så reagerte brukeren på at ingen informasjon om progresjon i applikasjonen var synlig. Etter at brukeren ble forklart at terminalen / konsollen som programmet er knyttet til, viser informasjon om progresjon i programmet, så hadde brukeren et ønske om at dette skulle være lettere tilgjengelig. Brukeren ville gjerne hatt den type informasjon i samme vindu som selve applikasjonen, slik at status til programmet ville vært mer tydelig til enhver tid. Utenom ønsket om status så var brukeren fornøyd med applikasjonen og opplevde ikke applikasjonen som uforståelig ved gjennomføring av oppgavene. Brukeren fant også frem til brukerguiden, som tidligere hadde vært den oppgaven som tok lengst tid, på veldig kort tid.

Brukerguide

DAF

-

Deteksjon av Fisk i Video

Brukerguide GUI

Utgave 1.0

Skrevet av

Thomas Faack Bjørnstad og Vebjørn Sundli Alseth

Innholdsfortegnelse

Om Brukerguiden	4
Vanlige Spørsmål (FAQ)	5
Systemkrav	6
Hvordan installere programvaren	7
Hvordan bruke programmet	8
Vindu 1: Innlasting	9
Kort beskrivelse av “Vindu 1: Innlasting”:	9
Hvordan laste inn filer:	9
Hva slags filer du kan laste inn:	9
Støttede filformater:	9
Informasjon om filvelgeren:	9
Skjermbilder	10
Figur 1.1: Oppstart av programmet	10
Figur 1.3: Filvelgeren kommer frem	11
Figur 1.5: Velg åpne for å bekrefte valg av filer	12
Vindu 2: Valgte filer	13
Kort beskrivelse av “Vindu 2: Valgte filer”:	13
Hva tabellen forteller:	13
Hvordan endre modus for deteksjonen:	13
Modus 1: Sakte	13
Modus 2: Hurtig	14
Hvordan aktivere Utviklingsmodus:	14
Hvordan kjøre testing på datasettene:	14
Hvordan legge til nye filer:	15
Hvordan gå tilbake til Start igjen:	15
Hvordan kjøre deteksjonen:	15
Skjermbilder	16
Figur 2.1: Programmet etter innlasting av filer	16
Figur 2.2: De ulike filene og “filveien” som er valgt av bruk	16
Figur 2.3: Filenes størrelse i MB (megabyte) og lengde i tid.	17
Figur 2.4: Modusen som deteksjonen skal kjøres i. En modus må velges for å fortsette.	17
Figur 2.5: Aktivering av utviklingsmodus	18
Figur 2.6: Valg av datasett	18
Figur 2.7: Om brukeren vil kjøre testing på ett av datasettene	19

Vindu 3: Ferdige filer	21
Kort beskrivelse av "Vindu 3: "Ferdige filer":	21
Hvordan laste inn nye filer:	21
Eksportering av data:	21
Om .CSV fil:	21
Skjermbilder	22
Figur 3.1: Programmet etter innlasting av filer	22
Figur 3.2: Output fra program (.CSV-fil)	22
Utviklingsmodus	23
Ordforklaringer	24

Om Brukerguiden

[Link til GitHub og Rapport](#)

Brukerguiden er utviklet av Thomas Faack Bjørnstad og Vebjørn Sundli Alseth. Brukerguiden og programvaren er et resultat av et bachelorprosjekt som er gjennomført av gruppens medlemmer på vegne av Norges Institutt for Naturforskning (NINA).

Oppgaven har blitt utført med hjelp fra Marius Pedersen, som veileder for gruppen, og kontaktperson fra NINA, Knut Marius Myrvold. Målet med bachelorprosjektet har vært å utvikle en programvare som skal gjøre det enklere å føre statistikk på fisk i en fisketrapp.

Programvaren er hovedsakelig utviklet for å gjennomføre deteksjon på fisk i video. Og kan med mindre modifikasjoner gjennomføre deteksjon på andre objekter som er i bevegelse.

Vanlige Spørsmål (FAQ)

Hvorfor detekteres ikke fisken helt i begynnelsen av videoen?

Deteksjonen benytter seg av bakgrunnssubtraksjon. Denne funksjonen har behov for en historikk med bilder før den fungerer optimalt. Derfor kan brukere ved enkelte tilfeller oppleve dårlige deteksjoner av fisk i begynnelsen av en video.

Hvorfor krever programmet administratortilgang ved installasjon?

Programvaren har behov for administratortilgang ved installasjon. Dette er ikke noe utviklerne har valgt bevisst, men er et resultat av at installasjonsfilen er opprette ved bruk av verktøyet NSIS.

Hvorfor kjøre filer og programmet lokalt og ikke over ekstern disk?

Om programvaren må hente videomaterialet fra en ekstern disk så vil dette forårsake at en del tid går på å laste videofilene fra den eksterne disken til datamaskinen. Om brukeren opplever at deteksjonen går sakte og laster videofiler fra en ekstern disk, så er det anbefalt å lagre videofilene lokalt på maskinen.

Om du ikke finner spørsmålet ditt her i brukerguiden, så kan du forsøke å kontakte en av utviklerne.

Systemkrav

Applikasjonen er blitt utviklet med tanke på at deteksjon skal være mulig å gjennomføre i sanntid. Applikasjonen er ikke blitt testet på andre datamaskiner enn gruppens egne, ettersom at gruppen ikke har fått muligheten til å teste på andre datamaskiner. Applikasjonen vil trolig kjøre på vesentlig lavere systemkrav, men ved betraktelige reduseringer så vil målet om deteksjon i sanntid, trolig ikke overholdes. Systemkravene gjelder når filene og applikasjonen ligger lokalt på maskinen og ikke på en ekstern disk.

Operativsystemer:

- MacOS: MacOS Catalina (Versjon 10.15.0) og nyere
- Windows: Windows 7 og nyere

Maskinvare: (tilsvarende en MacBook Pro (15-inch, 2018))

- Prosessor: 2,6 GHz 6-Core Intel Core i7
- Minne (RAM): 16 GB 2400 MHz DDR4
- Lagring: Minimum 100 MB med ledig lagringsplass
- Grafikk: Intel UHD Graphics 630 1536 MB

Hvordan installere programvaren

Programvaren er tilgjengelig på [GitHub DAF](#). Herfra laster du ned installasjonsfilen som er merket med navnet “**DAF**”.

Når filen er lastet ned på maskinen så kjører du filen ved å dobbeltrykke på filen, eller høyreklikker, og velger **Åpne/Kjør**.

Når du starter installasjonsfilen blir du møtt med et vindu som vil be om at brukeren gir programmet administratortilgang for å kunne installere programmet. Brukeren må da godta dette for å installere programmet.

Etter at programmet har startet, får brukeren muligheten til å bestemme hvor programmet skal installeres. Brukeren velger ønsket område på PCen og velger “Installer”.

Etter at installasjonen er fullført, så vil det komme en melding med at installasjon nå er fullført. Om du skulle oppleve problemer her burde du sjekke at du oppfyller systemkravene som er listet opp under “**Systemkrav**”, før du eventuelt leser “**Ofte stilte spørsmål**” eller kontakter utviklerne.

Når programvaren er installert så vil du finne en mappe med programvaren der du valgte å installere programmet. Mappen vil inneholde en kjørbare fil ved navn “DAF” og en mappe med navn “Vedlegg”, her ligger to tilhørende bilder som programmet trenger og brukerguiden du leser nå.

Dersom du finner hovedmappen, med mappen og filen beskrevet ovenfor, der du valgte å installere programvaren, var installasjonen vellykket. Brukeren kan nå kjøre programmet ved å starte filen med navnet “DAF”. For informasjon om programmet, se “Vindu 1: Innlasting” og videre.

Hvordan bruke programmet

Kort om bruk av programmet

Programvaren er konstruert slik at det skal være enkelt for brukeren å laste inn filer som skal kjøres deteksjon på. Programmet gjør det mulig å laste inn flere filer samtidig og eksporterer automatisk data fra deteksjonen.

Programmet består av 3 ulike vinduer. Ved oppstart så vil vindu 1 - Innlasting, være det vinduet som møter brukeren ved oppstart. Etter at brukeren har lastet inn filene sine, så blir brukeren møtt med vindu 2 - Valgte filer. Der vil brukeren kunne gjøre ulike valg for deteksjonen før deteksjonen kjøres, etter at deteksjonen er fullført så vil vindu 3 - Ferdige filer vises.

Brukeren har etter innlasting muligheten til å legge til flere filer. Om dette er ønskelig må brukeren trykke på knappen “Legg til fil”. Da vil filvelgeren komme frem igjen og gjør det mulig å velge nye filer som brukeren vil ha med.

Brukeren har etter innlasting mulighet til å velge mellom ulike moduser. Hver modus har ulike parametere som dermed vil gi ulikt utslag på deteksjonen. De ulike modusene sine parametere forklares mer detaljert på “**Vindu 2: Valgte Filer**” under undertittelen “**Valg av modus**”. Brukeren har også mulighet til å velge om “Utviklingsmodus” skal være aktiv eller ikke, men også om brukeren heller vil kjøre testing av ett av datasettene.

Brukeren har etter valg av modus muligheten til å kjøre selve deteksjonen. Da vil brukergrensesnittet stå stille til deteksjonen på de ulike filene er gjennomført. Etter at deteksjonen er fullført så vil programmets siste vindu vises.

I det siste vinduet, hvor alle filene igjen er listet opp i en tabell, har brukeren muligheten til å starte en ny innlasting eller bare krysse ut av programmet. Om “Ny Innlasting” velges så vil hele programmet starte på nytt og brukeren vil bli sendt tilbake til det første vinduet igjen. Alle filer som programmet har kjørt deteksjon på vil ha en tilhørende resultatfil som automatisk eksponeres til stedet hvor filene som ble lastet inn er lagret.

Vindu 1: Innlasting

Kort beskrivelse av “Vindu 1: Innlasting”:

- Bruker vil bli møtt med denne skjermen når brukeren starter programmet. Vinduet består av en mappe og en tekst. Brukeren kan her trykke på mappen slik teksten oppfordrer, og navigere seg fram til filene som brukeren vil kjøre deteksjon på. Filvelgeren er hovedsakelig beregnet for videofiler av formatene: **.mp4**, **.m4v**, **.mov**. Det kan hende at enkelte andre videoformater kan fungere.

Hvordan laste inn filer:

Ved oppstart av programmet vil det se ut som **Figur 1.1**. Det blå mappe-ikonet er selve knappen for innlasting av filer. Ved å trykke på denne mappen så vil filvelgeren starte.

Hva slags filer du kan laste inn:

I Versjon 1.7 så kan brukeren velge filter for **.mp4**, **.m4v** og **.mov**. Om ingen av disse filtypene er ønskelig kan brukeren også velge at alle filer skal vises gjennom filteret, “All Files”. Programmet er kun lagd for videofiler og videoformater som er støttet er;

Støttede filformater:

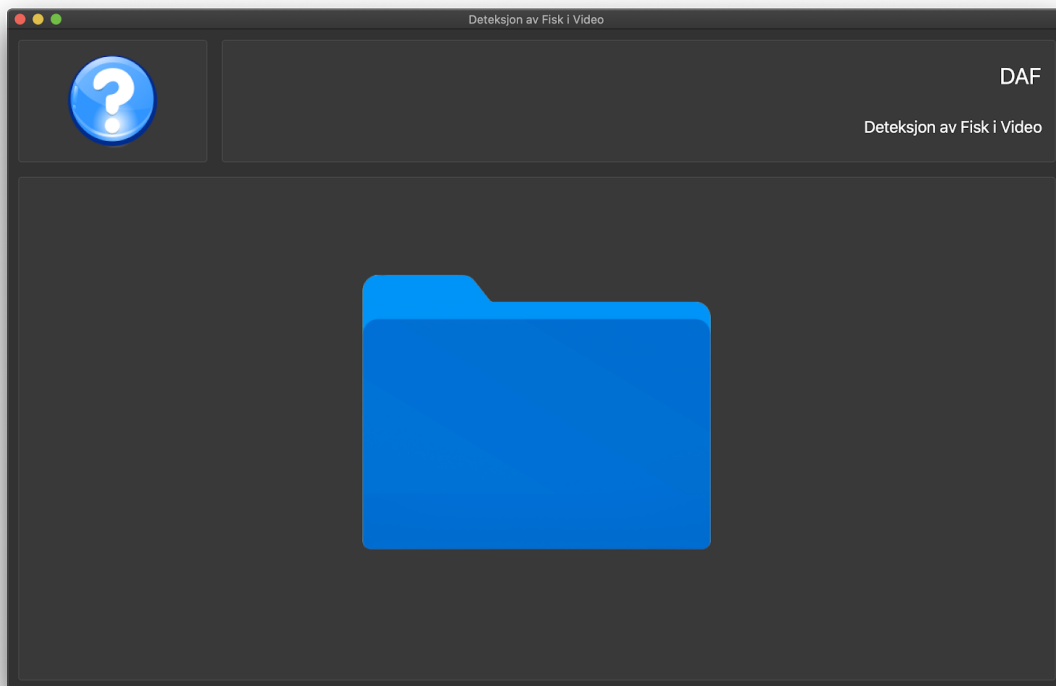
- MP4 (.mp4)
- M4V (.m4v)
- MOV (.mov)

Informasjon om filvelgeren:

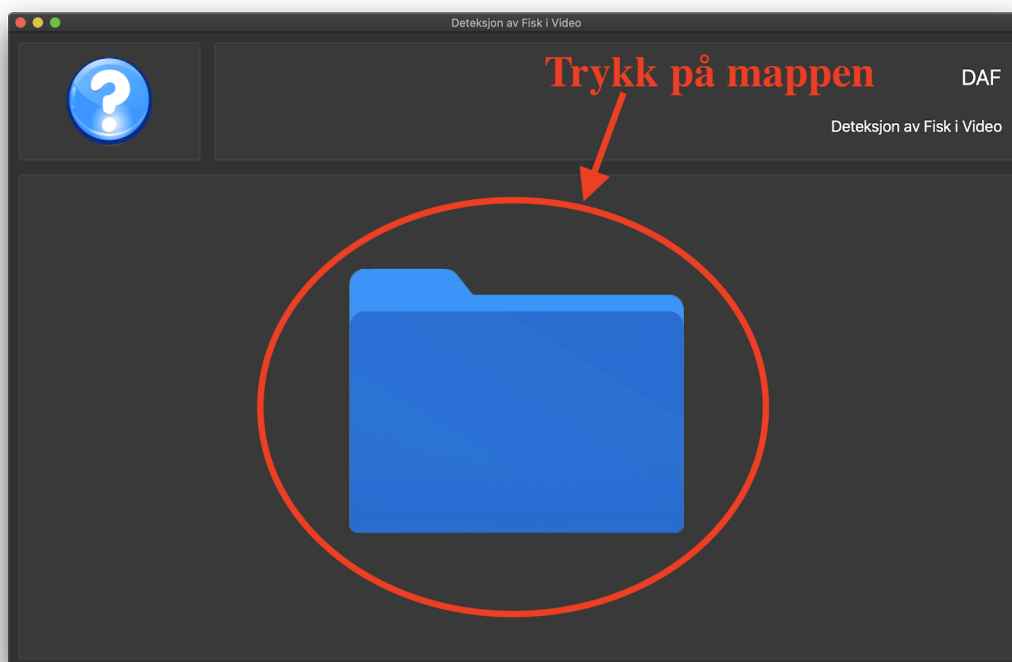
Filvelgeren starter opp der programmet er installert. Brukeren må derfor navigere seg frem til mappen hvor de ønskede filene er lagret. Det er mulig å velge flere filer samtidig (ved å holde inne Windows: **CTRL** eller MacOS: **Command** og markere flere filer) og hurtigtaster som markere alle videoer (Windows: **CTRL + A**, MacOS: **Command + A**) vil fungere for å velge alle filene samtidig. Etter at alle ønskede filer er markert er det bare å trykke på “Åpne” for å laste inn de valgte videoene.

Skjermbilder

Figur 1.1: Oppstart av programmet



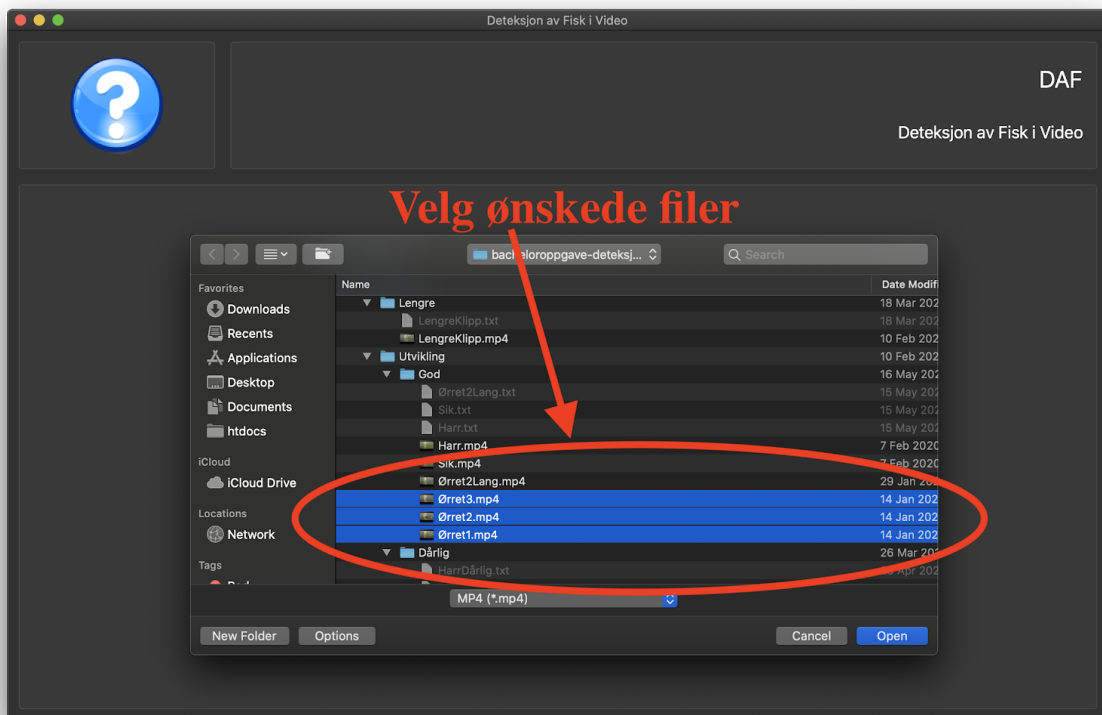
Figur 1.2: Trykke på mappen



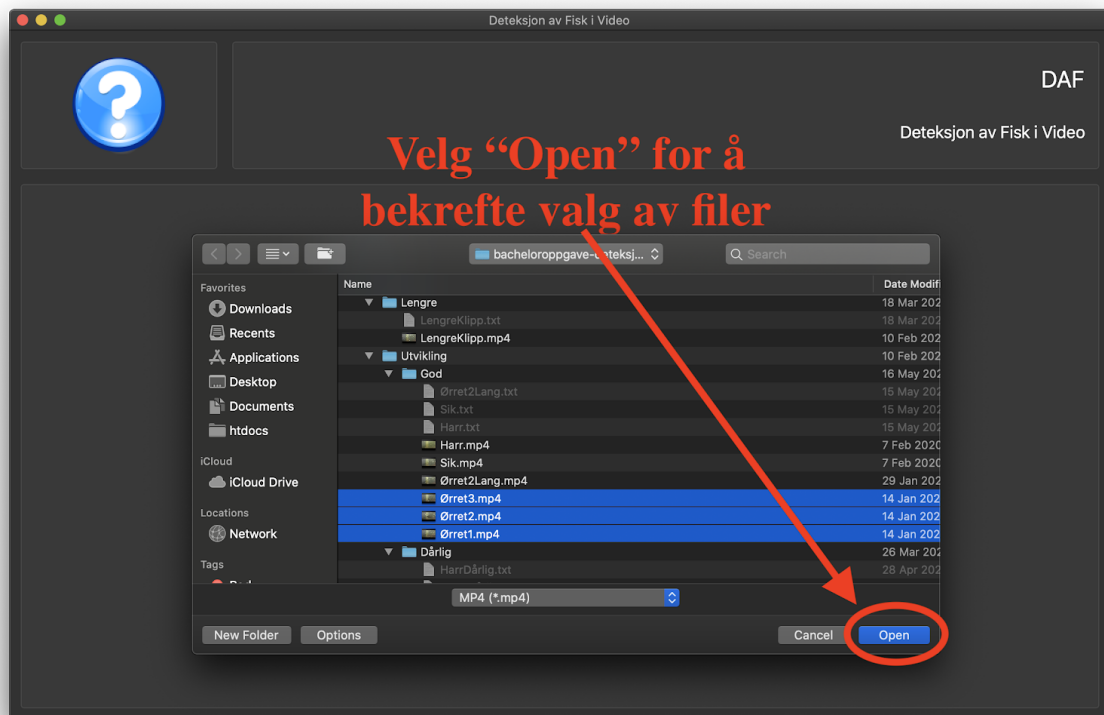
Figur 1.3: Filvelgeren kommer frem



Figur 1.4: Valg av ønskede filer



Figur 1.5: Velg "Open" (åpne) for å bekrefte valg av filer



Vindu 2: Valgte filer

Kort beskrivelse av “Vindu 2: Valgte filer”:

- Bruker vil bli møtt av denne skjermen (**Figur 2.1**) når brukeren har valgt filer som skal lastes inn. Vinduet består av en tabell med de ulike filene, ulike moduser som kan velges, og knapper for å endre filer, endre hvor resultatene skal eksporteres til, gå tilbake, og for å kjøre deteksjonen. Brukeren er nødt til å velge en modus før de kan fortsette med å kjøre deteksjonen.

Hva tabellen forteller:

Tabellen (**Figur 2.2**) viser brukeren hvilke filer som er valgt, filens størrelse i MB (megabyte) og videolengde.

Hvordan endre modus for deteksjonen:

Programmet har i Versjon 1.7 kun 2 ulike moduser som deteksjonen kjører i. Brukeren kan fritt velge mellom én av disse modusene, men må velge én av dem for å kunne kjøre deteksjonen da programmet er avhengig av å vite hvilke parametere som skal benyttes. Modusen velges ved å endre hvilken av radioknappene som er valgt.

Modus 1: Sakte

Modusen “Sakte” benytter parametere som vil ta lengre tid for programmet å kjøre gjennom de valgte filene. Denne modusen bør benyttes om videomaterialet består av uklart vann og/eller mye turbulens. At programmet tar lenger tid vil også gi programmet muligheten til å øke treffsikkerheten og dermed mer presist fortelle hvor det eventuelt befinner seg fisk i videomaterialet. At programmet kjører saktere skyldes valgene av parametere, per Versjon 1.7 så vil de ulike parameterne være:

Modus 2: Hurtig

Modusen “Hurtig” benytter parametere som vil bruke kortere tid på å gå gjennom videomaterialet, men på potensiell bekostning av presisjon i deteksjonen. Denne modusen anbefales å benyttes når det klart vann med lite turbulens. Programmet vil gå hurtigere gjennom deteksjonen og dermed bli hurtigere ferdig med prosessen. At programmet kjører saktere skyldes valgene av parametere, per Versjon 1.7 så vil de ulike parameterne være:

Hvordan aktivere Utviklingsmodus:

Om brukeren ønsker å se hvordan deteksjonsalgoritmen utfører arbeidet med filene, så er det mulig å aktivere “Utviklingsmodus”. Da vil programmet vise flere mindre vinduer med videoene som er valgt. Hvert enkelt av vinduene vil vise effekten av de ulike prosesseringsteknikkene, men også den originale videoen. Denne modusen bør kun aktiveres om brukeren ønsker å se hva som faktisk utføres av deteksjonsalgoritmen. Da denne modusen krever mer prosessering og vil øke tiden det tar å gjennomføre deteksjonen på filene.

Hvordan kjøre testing på datasettene:

Om brukeren ønsker så kan det kjøres deteksjon på datasettene som er vedlagt rapporten. For å kunne kjøre testing på ett av datasettene må ønsket datasett krysses av og knappen “**Kjør Testing**” må velges. Om brukeren velger begge datasettene eller ingen, men allikevel trykker på knappen “**Kjør Testing**” så vil en feilmelding produseres. Programmet har en hardkodet sti til området hvor datasettene som skal testes på skal ligge. Kravene for å kunne kjøre testing på ett av de valgte datasettene er at datasettene lastes ned som **.ZIP** fil og “pakkes ut” i området der programvaren originalt ble installert. Det vil si at mappen med “Utviklingssett” og “Testsett” skal ligge på samme nivå som mappen med Vedlegg og selve programmet. Det gjør det dermed mulig for andre å endre filene som ligger i denne mappen, men vær OBS på at dette krever tilhørende fasitfiler.

Hvordan legge til nye filer:

Om brukeren ønsker å legge til flere filer som skal benyttes, så kan brukeren velge knappen, “Legg til fil” (**Figur 2.8**). Da vil filvelgeren fra **Figur 1.3** komme frem igjen og nye filer kan velges. De valgte filene vil så lastes inn og brukeren kan fortsette.

Hvordan eksportere resultater til bestemt mappe:

Om brukeren ønsker å eksportere resultatene til en bestemt mappe eller område på datamaskinen, så er det mulig å endre dette gjennom denne knappen. Brukeren vil få opp en filvelger som gjør det mulig for brukeren å navigere til den mappen som er ønskelig at resultatene skal lagres ved. Om brukeren ikke velger hvor mappen skal ligge, så vil filene automatisk bli lagret der filene ble lastet inn fra.

Hvordan gå tilbake til Start igjen:

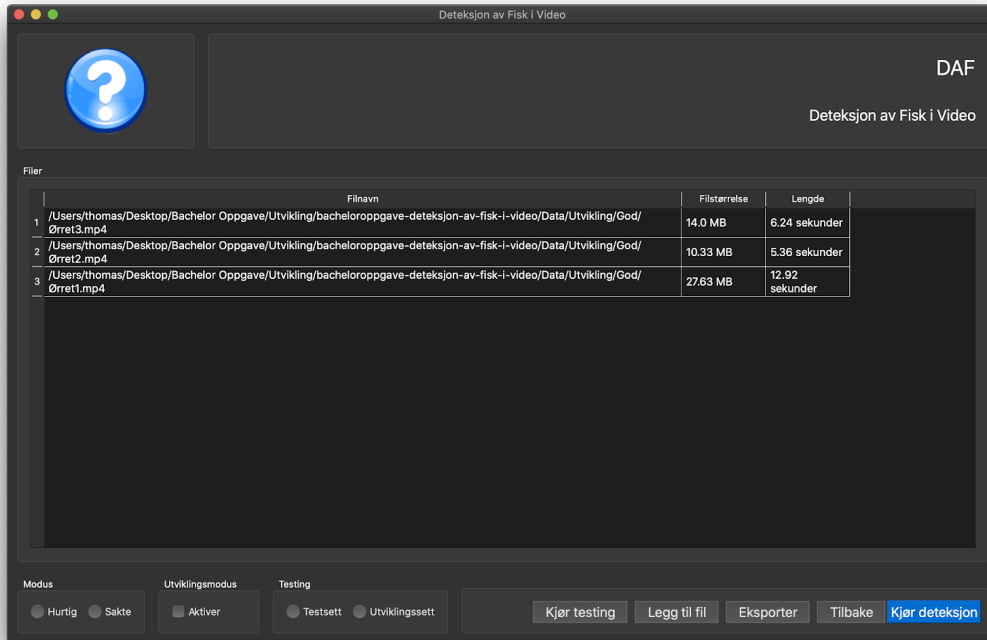
I **Figur 2.10** kan vi se hvordan brukeren kan gå tilbake til start igjen. Ved å gå tilbake til start så vil marking av valgte filer fjernes og programmet være i samme status som om det hadde blitt lukket og kjørt på nytt igjen.

Hvordan kjøre deteksjonen:

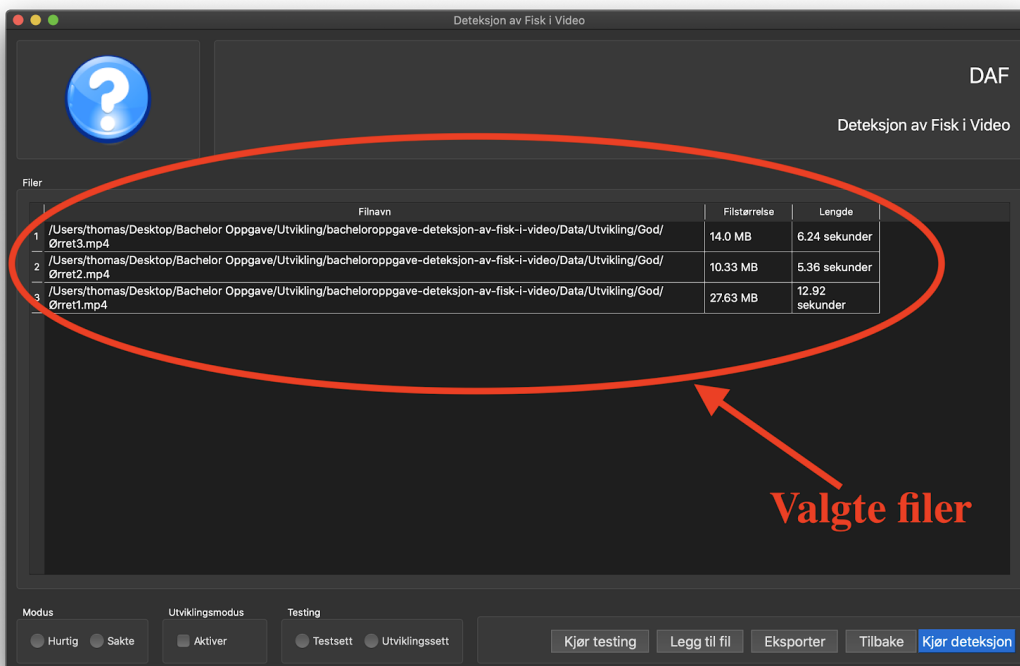
For at programmet tillater at brukeren fortsetter så må én, og bare én modus være valgt, og minimum én fil må være lastet inn. Om disse kravene er oppfylt vil programmet la brukeren fortsette. Om kravene ikke er oppfylt vil brukeren få en meldingsboks hvor et problem vil være beskrevet.

Skjermbilder

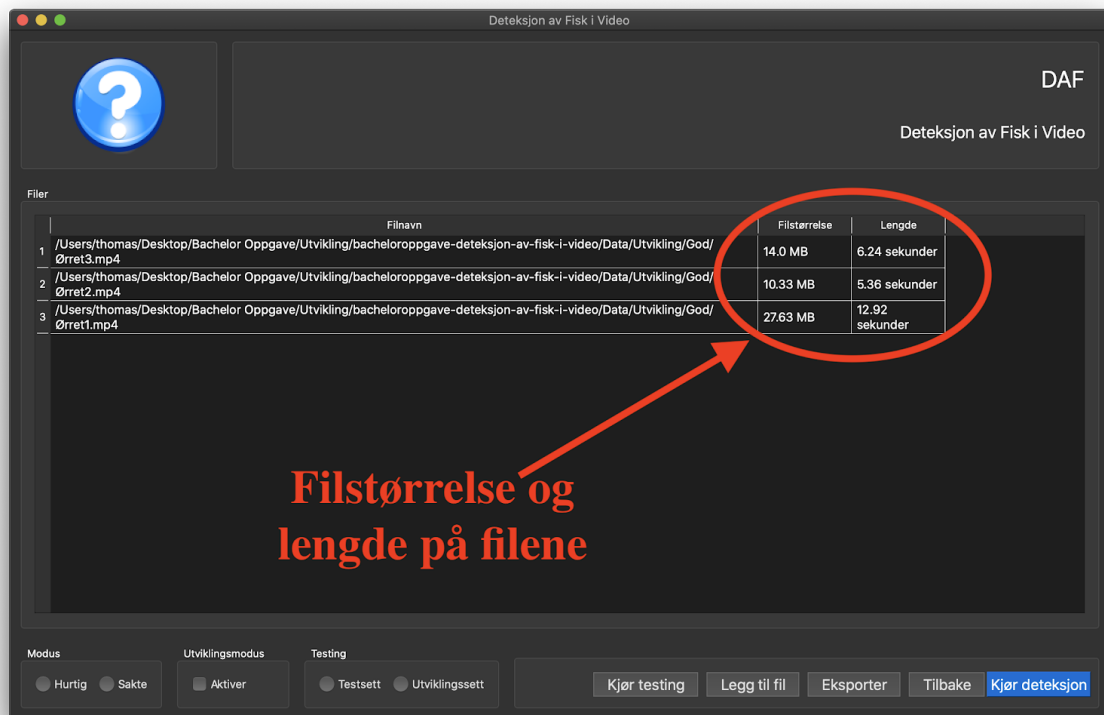
Figur 2.1: Programmet etter innlasting av filer



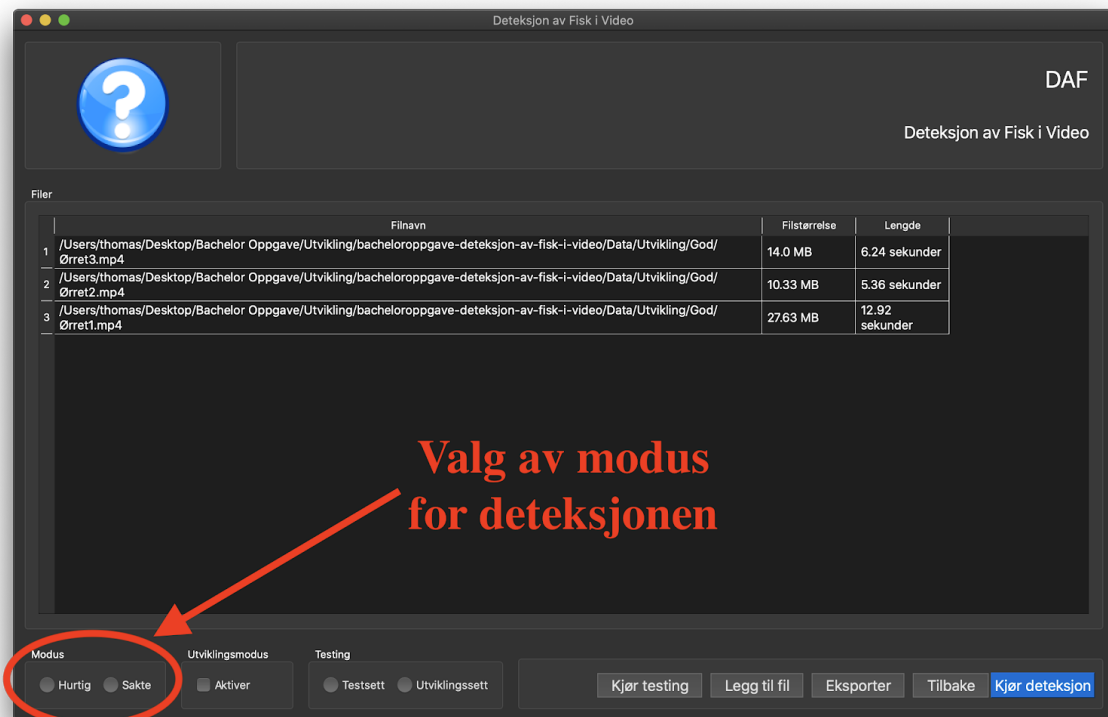
Figur 2.2: De ulike filene og "filveien" som er valgt av bruk



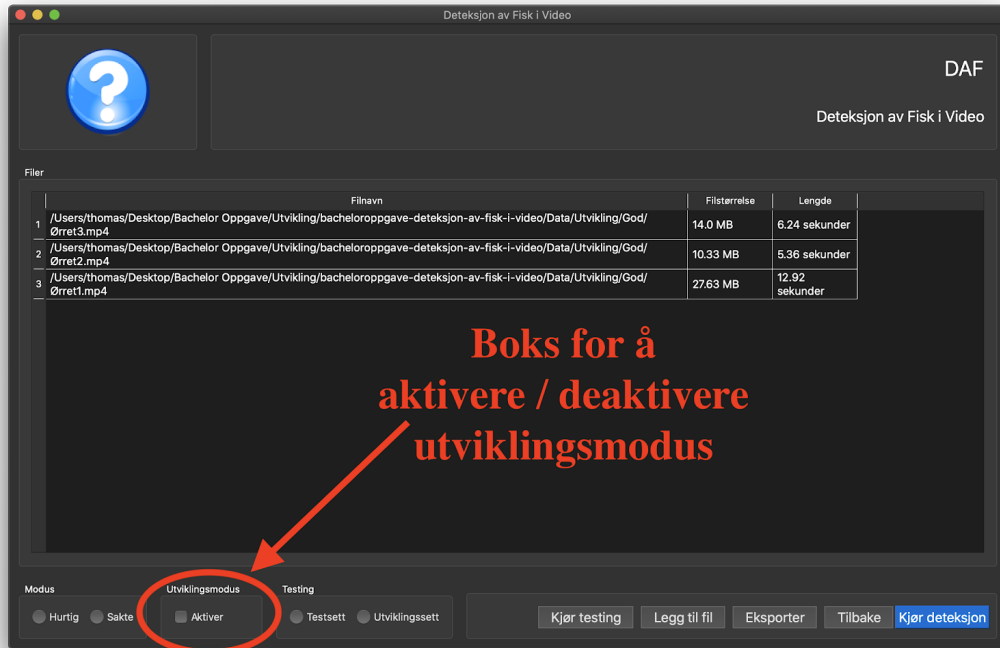
Figur 2.3: Filenes størrelse i MB (megabyte) og lengde i tid.



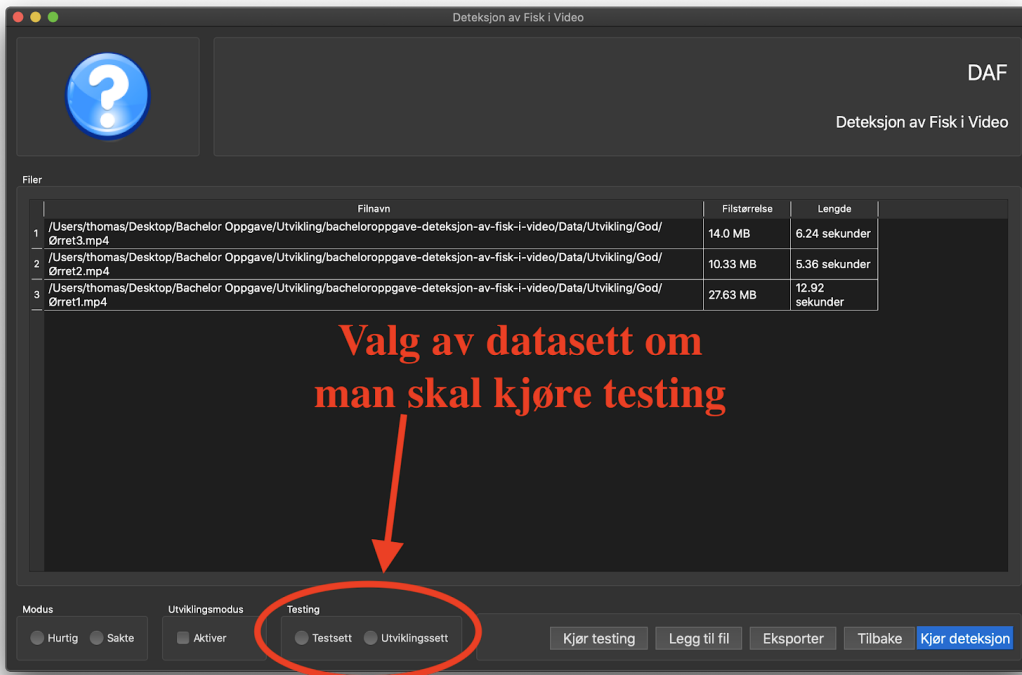
Figur 2.4: Modusen som deteksjonen skal kjøres i. En modus må velges for å fortsette.



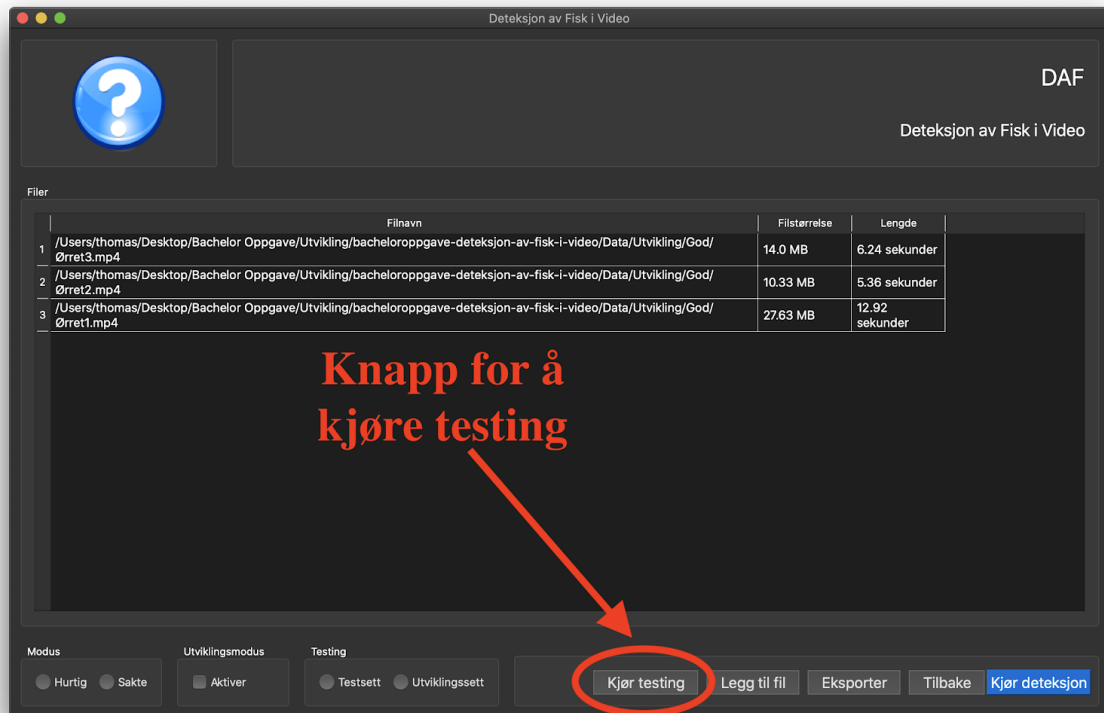
Figur 2.5: Aktivering av utviklingsmodus



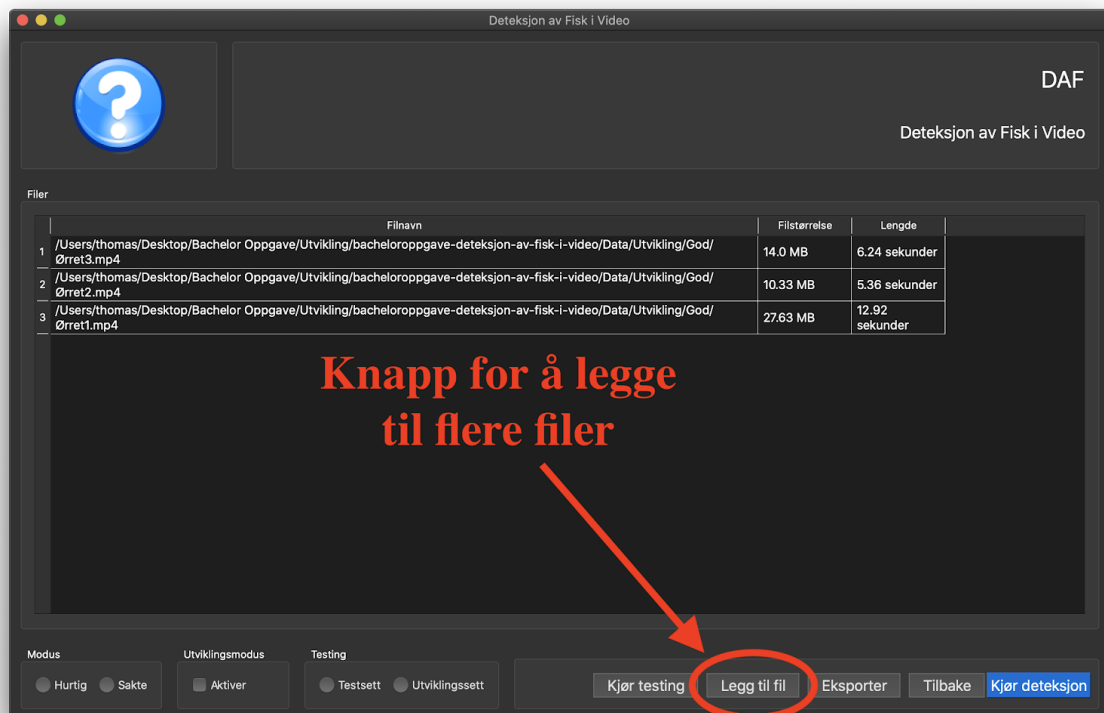
Figur 2.6: Valg av datasett



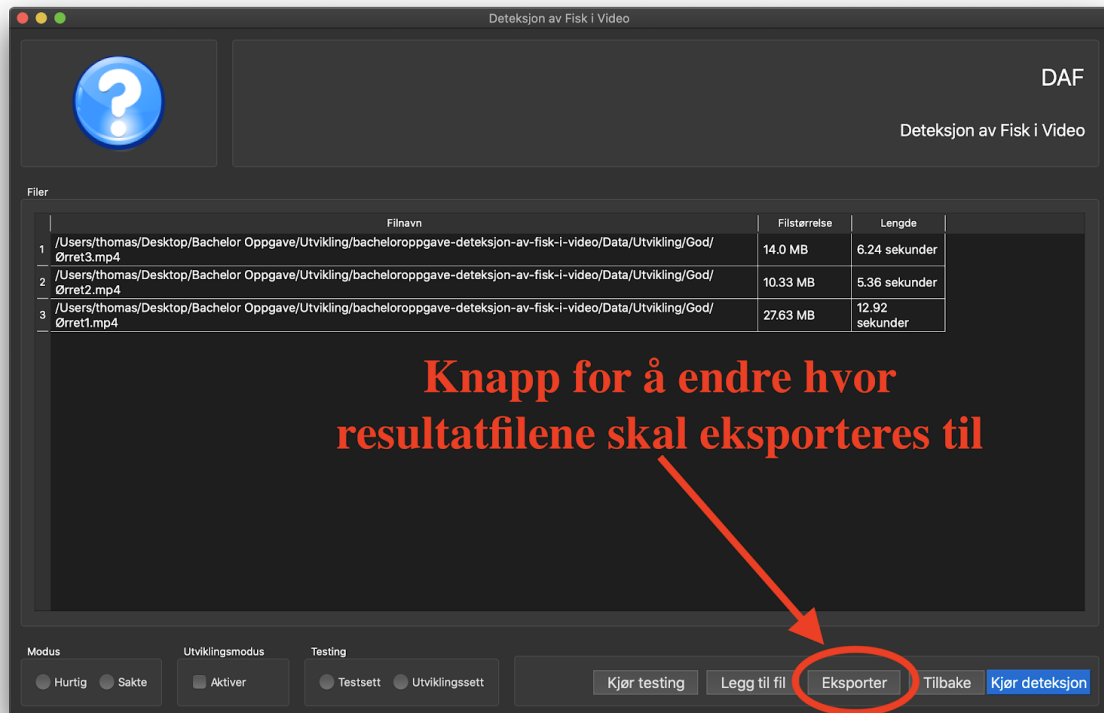
Figur 2.7: Om brukeren vil kjøre testing på ett av datasettene



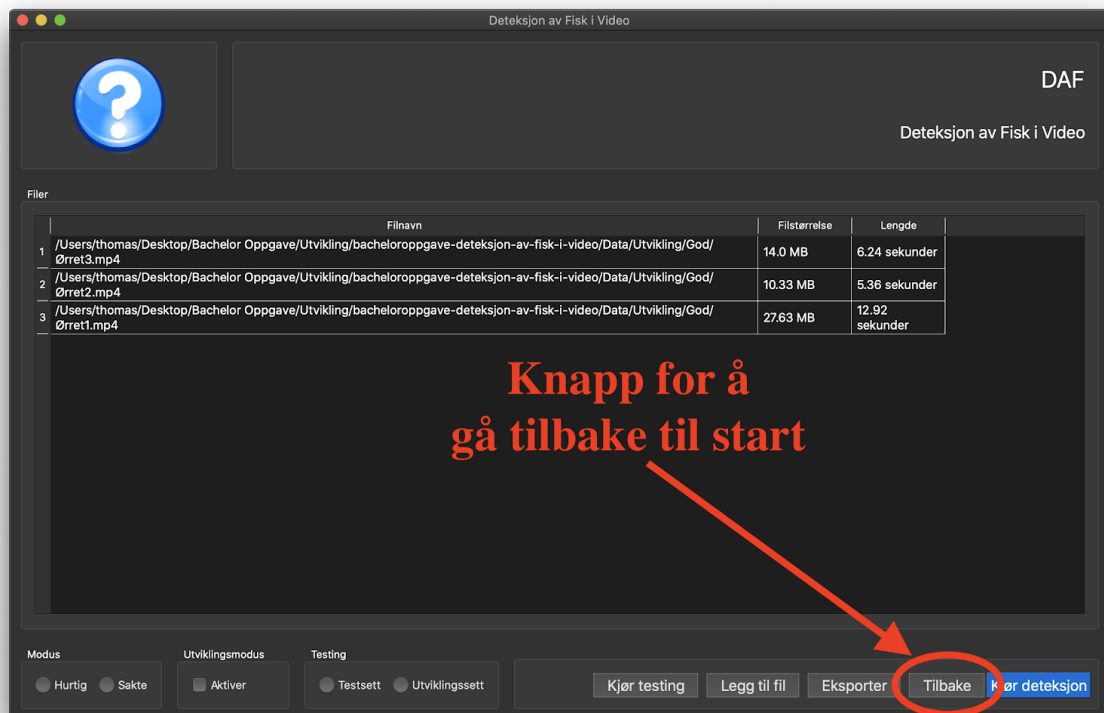
Figur 2.8: For å endre ønskede filer så velges knappen "Endre". Da vil filvelgeren fra Figur 1.3 kommer frem igjen..



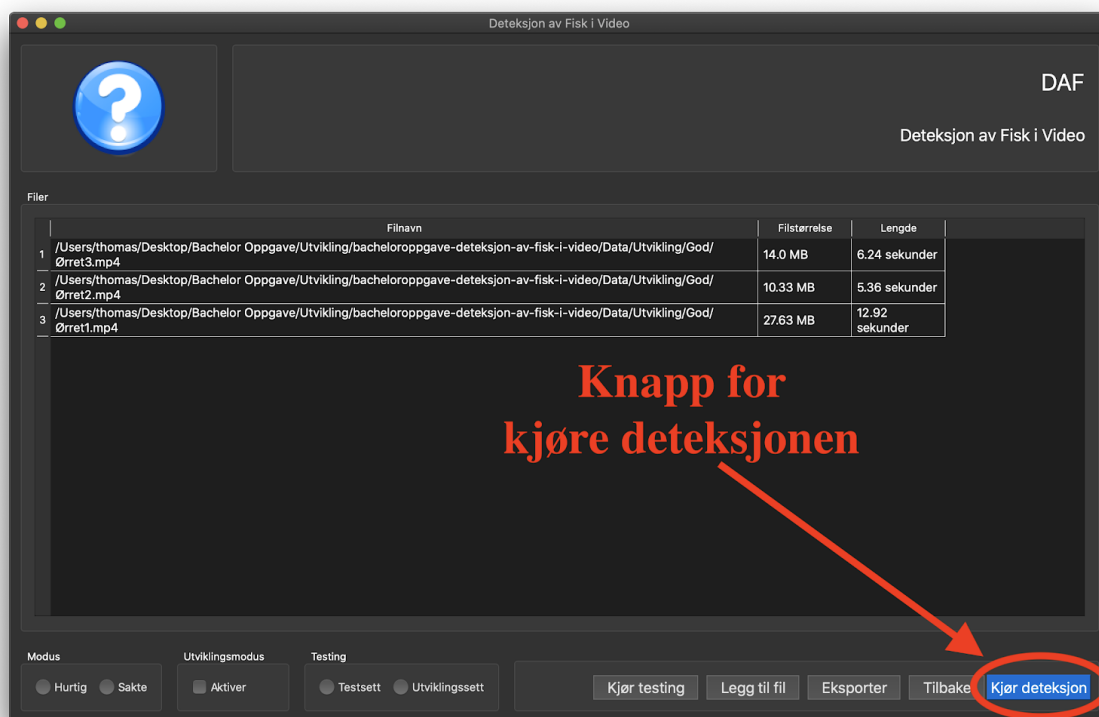
Figur 2.9: Om brukeren ønsker å bestemme hvor filene skal eksporteres



Figur 2.10: Om brukeren ønsker å gå tilbake til start.



Figur 2.11: Etter at én modus er valgt så kan brukeren fortsette.



Vindu 3: Ferdige filer

Kort beskrivelse av “Vindu 3: “Ferdige filer”:

- Bruker vil bli møtt av denne skjermen (**Figur 3.1**) når deteksjon har kjørt ferdig. Vinduet består av en tabell med de ulike filene og knapp for å starte en ny innlesing. Data fra deteksjonen vil automatisk bli eksportert som en **.CSV** fil slik at brukeren trygt kan avslutte programmet eller starte en ny innlesing om ønskelig.

Hvordan laste inn nye filer:

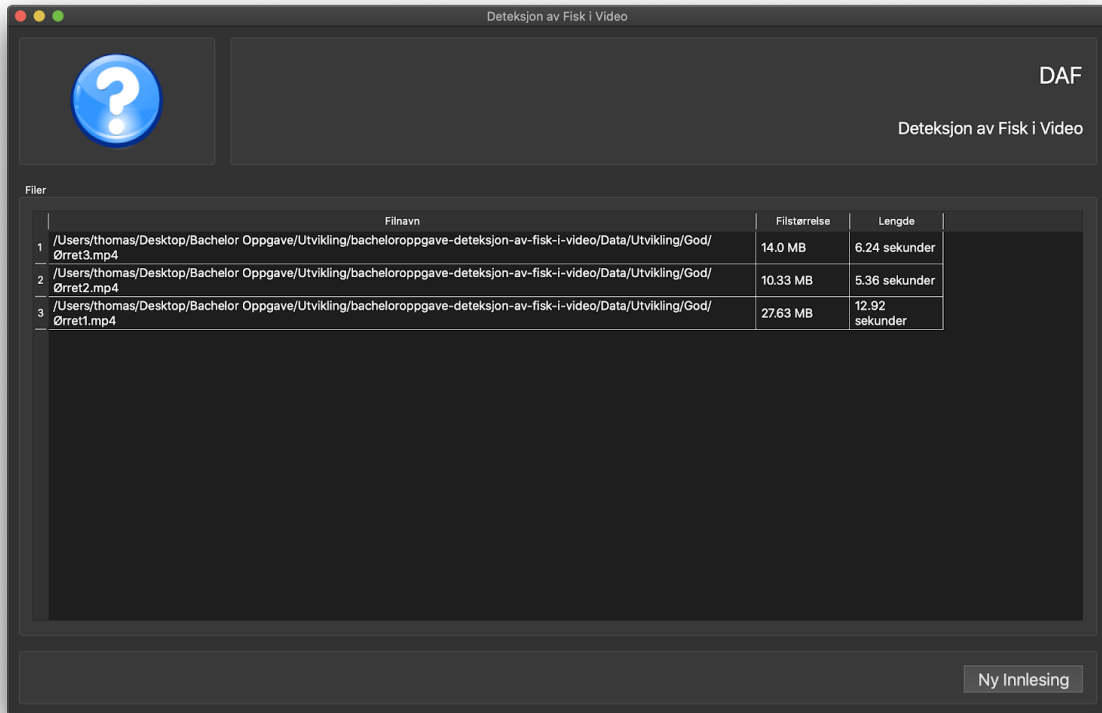
Om brukeren ønsker å gjennomføre flere innlesninger så er det ikke behov for å avslutte programvaren og starte den på nytt. Brukeren kan enkelt velge knappen “Ny Innlasting” for å starte en ny innlasting. Da vil programmet ta brukeren til den første vinduet igjen og brukeren gjennomfører samme prosedyrer som ved forrige gjennomføring.

Eksportering av data:

Programmet eksporterer automatisk alle filene som brukeren har valgt å kjøre deteksjonen på. Filene vil ha et **.CSV**-format og dermed kunne enkelt åpnes av de aller fleste regneark-programmer som blant annet Microsoft Excel. Filene vil eksporterer til der de innleste filene ligger slik at både videofil og resultat fil ligger på samme sted.

Skjermbilder

Figur 3.1: Programmet etter innlasting av filer



Utviklingsmodus

Utviklingsmodus i programvaren er en modus som kan benyttes av alle brukere, men vil være unødvendig å aktivere med mindre brukeren ønsker å se arbeidet som utføres av deteksjonsalgoritmen.

Deteksjonsalgoritmen består av flere funksjoner som utfører ulike prosesseringsteknikker på de valgte filene. Forskjellen mellom Utviklingsmodus **Aktivert** og **Deaktivert** (ikke aktivert) er at programmet kjører funksjonskallet `cv.imshow(frame)`. For å forklare funksjonen kan vi først se på `cv` som er en referanse til OpenCV biblioteket. `imshow` er en funksjon i OpenCV biblioteket som kjører, og `frame` er det bildet vi ønsker å vise. Her vil `frame` være resultatet fra de ulike prosesseringsteknikkene.

Ordforklaringer

Systemkrav - En oversikt over krav som en maskin på tilfredsstillende

Operativsystem - Den grunnleggende programvaren til en datamaskin

Maskinvare - De fysiske komponentene til en datamaskin

GB - Gigabyte, enhet for lagring (Eks. 1GB = 1000MB)

MB - Megabyte, enhet for lagring (Eks. 1MB = 1000KB)

GUI - Grafisk Brukergrensesnitt, fra engelsk Graphical User Interface

Parameter - En størrelse som kan ha ulike verdier

Prosesseringsteknikker - Teknikker som benyttes for å gjøre endringer på bildet

Datasett - Et sett med videofiler

OpenCV - (Open Source Computer Vision Library) er bibliotek med funksjoner hovedsakelig rettet mot datasyntese. Biblioteket er gratis for bruk under åpen kildekode BSD-lisens.

NSIS - (Nullsoft Scriptable Install System) er et skriptdrevet installasjonsforfatterverktøy for Microsoft Windows. Verktøyet benyttes for å skape installasjonsfil av en \textit{zip} mappe og er kompatibel med GPL-lisensen.

Møtereferat

Møte med NINA

Dato: 29.01.2020

Sted: NTNU i Gjøvik

Prosjektgruppen møtte NINA på Campus i Gjøvik og fikk i tillegg til en introduksjon og besvarelser på en rekke spørsmål, også overlevert enda mer videomateriale enn det gruppen allerede hadde blitt tildelt ved oppstart av prosjektet. Prosjektgruppen stilte spørsmål til:

- Konkrete krav NINA har til sluttproduktet
- Differanse mellom det utleverte videomaterialet og andre fisketrapper
- Hvor mye tid som legges ned i gjennomgang av videomaterialet
- Hva den innhentede informasjonen benyttes til
- Utformingen av det grafiske brukergrensesnittet
- Hvilket format resultatfilene skal eksporteres i
- Viktigheten av en brukerguide

Møte med NINA

Dato: 11.05.2020

Sted: Microsoft Teams

Prosjektgruppen møtte NINA på Microsoft Teams da coronaviruset (COVID-19) hindret fysiske møter. I møtet så ble status på prosjektet presentert for NINA og prosjektgruppen gjennomførte brukertesting av to av NINAs forskere. I tillegg til en gjennomgang av det grafiske brukergrensesnittet så ble et utkast av rapporten overlevert til NINA, slik at prosjektgruppen kunne få innspill fra NINAs forskere.

