Human Swarm Interface with Predictive AI for Onsite Incident Commander in Maritime Search and Rescue Operations

Alexander Haarbye Jensen, Lasse Lundbo, Lukas Rønsholt

Software, HCI101F21, 2021-06

Master Thesis Project

STUDENT REPORT

AALBORG UNIVERSITY

Department of Computer Science
Aalborg University
aau.dk

AALBORG UNIVERSITY
STUDENT REPORT

Title: Human Swarm Interface with Predictive AI for Onsite Incident Commander in Maritime Search and Rescue Operations

Theme:
Human-computer Interaction

Project Period:
Spring Semester 2021

Project Group:
HCI101F21

Participants:
Alexander Haarbye Jensen
Lasse Lundbo
Lukas Rønsholt

Supervisor:
Timothy Robert Merritt

Copies: 4
Page Numbers: 107
Date of Completion:
June 7, 2021

Synopsis:

Maritime search and rescue (SAR) is an area which can benefit from the constant advancements in unmanned aerial vehicle (UAV) technology. Using UAVs to search an area can serve as a replacement for a helicopter which is both cost effective and can save time and thereby lives. The challenge is to implement an interface for the incident commander, who is the person in charge on the scene, to support the need functionality while maintaining a high situational awareness, performance and an appropriate workload.

Previous work on SAR operations have studied the effects of conventional-, predictive- and virtual reality interfaces in relation to situational awareness and workload. The results showed that virtual reality increased situational awareness, however the results for the effects of conventional vs predictive interfaces were inconclusive, and more research was suggested.

The research problem is to figure out what effect a predictive human swarm interface will have on situational awareness, workload and performance. A study is conducted to measure these variables and experts are brought in to develop a set of design guidelines.

The platform for the study is developed as a progressive web application which will run on an iPad Pro. The study was conducted with ten participants and through contextual interviews with three experts.

The results of the study showed significant difference in human workload in favor of the predictive interfaces. Performance measurements showed that the predictive intrusive interface had the best performance. Furthermore design implications developed in collaboration with experts are presented.

AALBORG UNIVERSITET
STUDENTERRAPPORT

Institut for Datalogi
Aalborg Universitet
http://www.aau.dk

Titel: Human Swarm Interface with Predictive AI for Onsite Incident Commander in Maritime Search and Rescue Operations

Tema:
Menneske-maskine Interaktion

Projektperiode:
Forårssemestret 2021

Projektgruppe:
HCI101F21

Deltager(e):
Alexander Haarbye Jensen
Lasse Lundbo
Lukas Rønsholt

Vejleder(e):
Timothy Robert Merritt

Oplagstal: 4
Sidetal: 107
Afleveringsdato: 7. juni 2021

Synopsis:

Vand redningsaktioner er et område som kan have gavn af den konstante udvikling af drone teknologien. Brugen af droner til at undersøge et område kan erstatte en helikopter hvilket både spare penge og tid og derfor liv. Udfordringen er at implementere en grænseoverflade for indsatslederen, som er personen der leder aktionen på stedet, og understøtte den ønskede funktionalitet mens der holdes en høj situationsfornemmelse, ydeevne og en passende arbejdsbyrde.

Tidligere undersøgelser har studeret effekten af konventionelle, forudsigende og virtuelle grænseflader i relation til situationsfornemmelse og arbejdsbyrde. Resultaterne viste at virtuel realitet forhøjede situationsfornemmelsen, men resultaterne for effekten af konventionelle mod forudsigende grænseflader var ikke afgørende, og flere undersøgelser blev forslået.

Problemstillingen er at finde ud hvilken effekt et forudsigende drone sværms grænseflade vil have på situationsfornemmelse, arbejdsbyrde og ydeevne. En undersøgelse er udført for at måle disse variabler og eksperter er bragt ind for at udvikle et set af design retningslinjer.

Platformen for undersøgelsen er udviklet som en progressiv web applikation som vil køre på en iPad Pro. Undersøgelsen var udført med ti deltagere og gennem kontekstuelle interviews med 3 eksperter.

Resultaterne af undersøgelsen viste en afgørende forskel i arbejdsbyrde til fordel for de forudsigende grænseflader. Arbejdsbyrde målinger viste at den forudsigende påtrængende grænseflade havde den bedste ydeevne. Derudover er design implikationer udviklet i sammarbejde med eksperter præsenteret.

Summary

This thesis project looks at the wicked problem of maritime search and research, which have many moving parts that affect each other. Changing one element on a search and resource operation can impact other areas, and taking on all aspects of the operation at once is impossible. For this project, we have chosen to try and improve the rescue time when a person falls into the water by replacing a considerable time-consuming factor, the rescue helicopter, with a swarm of UAVs.

To gain insight into how a SAR operation was conducted, the emergency agency process in northern Jutland was examined by interviews with key persons in the procedure. For the operation, there are two crucial people, the operations manager, and the incident commander. To further limit this project's scope, we will look at how to design the change in favor of the incident commander.

A research platform is built to examine operators' situational awareness and workload using a human swarm interface to aid in locating missing persons in a maritime environment. The main focus will be to see if having predictive elements in the interface helping the operator by using machine intelligence to try and spot people in the water.

A user study was designed to measure the chosen aspects, where ten participants would go through three variations of the interface. A conventional interface with no predictive elements, and two predictive interfaces, one with an intrusive pop-up and one with a non-intrusive pop-up for when the system found something of interest. The participants would run through a scenario based on how a SAR operation is currently conducted, with the small addition of now having the interface at their disposal.

During the project period, it was not possible to get enough domain experts to conduct a user study, so therefore the study was conducted on engineering students. In order to get some feedback from UAV experts, three expert reviews were conducted after the user study, which included a co-design process. The expert review help identify problems that might not have been discovered in the study and compare the system to the state-of-the-art systems and see if something is missing.

The results of the study showed significant difference in human workload in favor of the predictive interfaces. Performance measurements showed that the predictive intrusive interface had the best performance. The project concludes with a list of usability concerns and implications for design that we have found relevant for designing a human swarm interface, based on our research, user study, and expert reviews.

Contents

Preface

This report documents a master thesis made by three 10th semester Software students at Department of Computer Science of Aalborg University.

The master thesis implements and tests a prototype Human Swarm Interface for maritime SAR operations. The interface is developed specifically to the on-side Incident Commander.

We would like to extend our gratitude to our supervisor Timothy Robert Merrit for his continuous feedback, guidance and commitment throughout the duration of the thesis.

We would also like to express our appreciation towards Jacobo Domingo Gil at Robotto, Sean Braley, and Calvin Rubens for their in-depth expert reviews of our prototype system.

Aalborg University, June 7, 2021

Alexander Haarbye Jensen | Lasse Lundbo | Lukas Rønsholt

1 | Introduction

The use of unmanned aerial vehicles (UAVs) to assist in different tasks is becoming increasingly common as UAVs' availability increases and the price declines. Search and rescue (SAR) operations can benefit from the advancement in UAV technology if applied in a way that suits the operation. In Denmark, 16 people died in 2019 during maritime SAR operations due to the delay between the initial call and the rescue itself (Forsvarsudvalget, 2019). Today the emergency services use boats, and in special cases helicopters, to rescue individuals who have fallen into the water. The cost of SAR operations using a helicopter can exceed 60.000 Kr. which could be negated if the helicopter was replaced by UAVs (Ørsborg Johansen, 2020).

Maritime search and rescue is a wicked problem (Buchanan, 1902) with many factors determining the outcome of the operation. There are many people involved and the operation itself is a complicated process carefully designed to be as efficient as possible. This project will attempt to reduce the total rescue time by reducing the time it takes to locate a missing person and thereby save more lives by utilizing the rapid advancement in UAV and camera technology. This project will not be a full solution to the problem, but it targets the searching part of the rescue process while the rest of the process is out of scope. Nowadays in maritime SAR operation the EMA often deploys helicopters to locate the missing person. The helicopter is operated by the danish military and it takes them between 8-12 minutes just to get airborne (Nordjylland, 2015). After they are airborne they need to navigate to the desired location from the military base. These minutes are very critical in a maritime SAR operation and can easily be the difference between life and death. By applying a swarm of UAVs to a location like Aalborg Harbor these minutes will be almost eliminated.

To apply UAVs in SAR operations it is necessary to implement a system capable of controlling the UAVs during the SAR operation. Such a system needs an intuitive graphical user interface (GUI) that is tailored to the specific needs of the operators to increase the chance of the operation succeeding. SAR operations require a high situational awareness (situational awareness) from the incident commander (IC), and the system must implement the necessary tools to allow a high situational awareness. The interface designed in this project is tailored to the needs of the IC. The IC is leading the rescue team on the rescue boat and will therefore be using the interface while located on a rapidly sailing boat. The context of which the interface is being used poses clear challenges to the design and the device it is operated on, as it must be fully functional in harsh environments.

The study of this paper aims to address the research gap: *What effect will predictive features have on the situational awareness, performance and workload of the Incident Commander in maritime SAR operations?* This research gap has been a focus of some previous research, however, there have not been studies to date that have provided evidence that predictive interface elements have a positive effect on situational awareness and human workload. In this work, we build upon previous studies to examine how some predictive interfaces features might affect situational awareness and workload.

The contributions of this paper includes results from the user study in relation to situational awareness, human workload and performance. Furthermore a set of implications for design of future SAR interfaces is presented which encapsulates the findings from the study and expert reviews. These contributions can set the foundation for further studies in the area of swarm interfaces. While the results are gathered from a study focusing on SAR operations, some of the findings can be generalized to other UAV interfaces.

This paper starts by summarizing the previous work done on the 9th semester, followed by a preliminary analysis in Chapter 4. The method of the study is described in Chapter 5 which includes the continued development of the system to get it ready for the study. The remainder of the paper showcases the results in Chapter 6, discusses the study in Chapter 7 and ends with a conclusion in Chapter 9. This report is a continuation of the 9th-semester software project *"Using UAVs to Aid Decision-Making and Enhance Situational Awareness in Maritime SAR Operations"* (Jensen et al., 2020), which sets the groundworks for two separate yet similar projects to be further researched. This report makes out one of those projects and will work upon the results in (Jensen et al., 2020).

The purpose of (Jensen et al., 2020) was to investigate how a UAV swarm system interface could be designed to help the people in charge of maritime SAR operations and the first responders on-site at the scene. In particular, it is looked at how to raise situational awareness and minimize the user group's mental workload to aid them in quick and precise decision making. To narrow the focus of the maritime SAR scope, the SAR operations at the emergency services in North Jutland, *"Nordjyllands Beredskab"*, was put into focus.

Nordjyllands Beredskab's maritime SAR operations consist of several people and many tasks. The two most important persons of the operations are the Operations Manager, who first receives the call, and the Incident Commander, who is the highest-ranked first responder on-site and is in charge of the operation. These two persons were interviewed in connection with the project to determine how a SAR operation is conducted in their organization, which tasks and roles it consists of, and where it would be helpful to utilize a UAV swarm and how such a system should function. To get an overview of a typical maritime SAR operation, please consult Figure 1.1. Here the operation is outlined for the traditional approach where a helicopter is applied. However, the proposed process involving a UAV Swarm is also outlined.

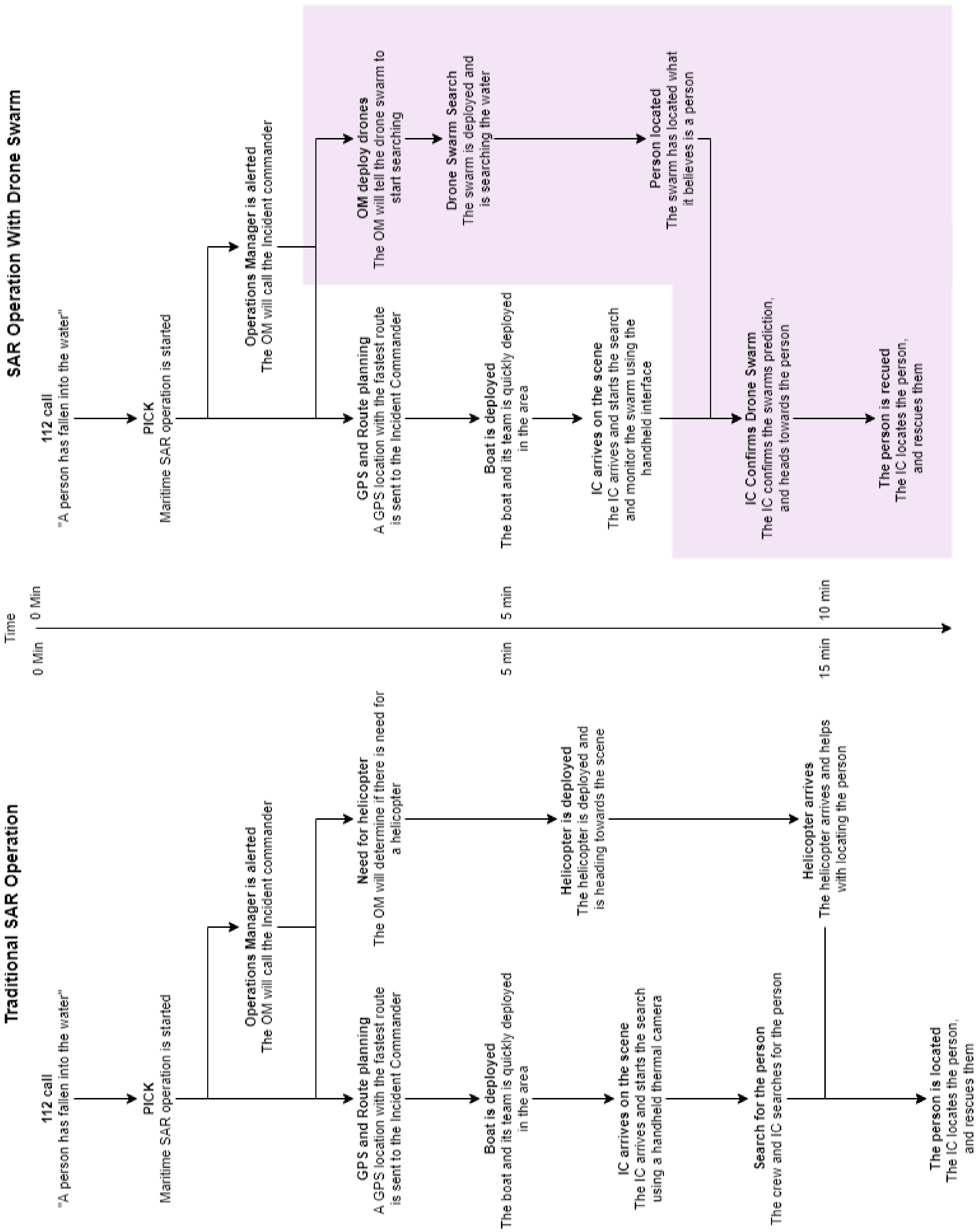

Figure 1.1: Traditional vs UAV operation

Through the related research, three main findings that guide our work include results to suggest that increased automation of swarms can reduce the mental workload of the operators, predictive interfaces can help increase situational awareness, and that a task-specific interface helps both on the situational awareness and the workload of the operator. It was noticed that the existing research only focused on utilizing a single interface to solve a task using UAV swarms, often using task specialized interfaces. Interestingly none of the studies we encountered involved the more complex team coordination and specialization observed with the EMA.

In the present day operations, the Operations Commander, Incident Commander, and helicopter pilot all have different views into the data. This specialization and segmentation enable the large rescue task to be broken into smaller sub-tasks taken up by each team's roles. For Nordjyllands Beredskab, it was concluded that having two interfaces, one for the Operations Manager and one for the Incident Commander, each designed specifically for their tasks and work environment, would benefit their SAR operations.

The base system, along with the elicited requirements, was used in two expert reviews, which purpose was to obtain feedback on the already built system and find missing requirements for the role-specific interfaces. With the input from the two experts, the requirements were revised to correct or add new requirements, making the results and contribution of (Jensen et al., 2020) a set of requirements or design suggestions, both for a general system but also role-specific for the Operations Manager and Incident Commander, along with a base system which can be adapted in several ways to create a single system.

2 | Related Work

The related work in this report will summarize some of the research done in (Jensen et al., 2020) where forward snowballing was used with (Hocraffer and Nam, 2017) as the starting point. This chapter discusses research papers that concern Human Swarm Interfaces, situational awareness, human workload, and its measurement. Therefore the relevant related work to this thesis and their state-of-the-art contributions are outlined in this chapter.

Research into Human-Swarm Interfaces is still a relatively new research area in Computer Science. However, there still exists a comprehensive range of articles studying a multitude of aspects related to human swarm interfaces. (Mi and Yang, 2013) defines four Levels of Automation (LOA) which describes how active the operator is in controlling the swarm. There are Fully Autonomous which can operate without an operator. Then there is Machine- and Human-Oriented Semi-Autonomous wherein the machine-orientated system makes most decisions but informs the operator in special needs. In human-oriented, the system continuously informs and relies on the operator. The last level is Manual Operation, in which the operator is in complete control.

(Hocraffer and Nam, 2017) finds that humans are better at the high-level and more abstract skill, where swarm systems excel at skills there are more computationally intense. This suggests that currently, the optimal LOA would be a semi-autonomous system where the operator is acting as a supervisor and prioritizing tasks, while the system takes care of tasks like navigation and detecting itself.

2.1 Situational Awareness & Human Workload

A high situational awareness is important in SAR operations, and research must be done to understand the concept fully. This section aims to summarize the work already conducted within the field regarding situational awareness and the affecting factors. (Hussein and Abbass, 2018) states that situational awareness and workload are both significant factors when trying to better decision making and enhance human performance. (Riley and Strater, 2006) found a significant correlation between a high situational awareness and strong human decision making, which directly affected the overall performance, and poor situational awareness had a negative impact. While having a high situational awareness does increase the probability of good performance, it is not a guarantee in itself according to (Endsley and Jones, 1997).

To increase situational awareness, it is crucial to have the right amount of workload as having too high or too low of a workload can negatively affect the operator. (Hocraffer and Nam, 2017) states that raising the level of automation in the system will decrease the workload and allow for a higher situational awareness. A high level of automation will reduce the need for operator intervention and give them time to do other tasks. However, this could lead to a situation where the operator is purely monitoring the system and not doing any interactions, leading to boredom. (Cummings et al., 2014) argues that boredom decreases situational awareness as operators distract themselves with their phones or other forms of entertainment. This is closely related to the findings of (Hussein and Abbass, 2018) as they found that too high a level of automation can result in "out-of-the-loop" problems where the operator has lost the broad overview of the situation and is unable to make proper decisions. This indicates that the level of automation can neither be too high or too low if a good situational awareness is to be maintained.

In (Hou et al., 2011) Intelligent Adaptive Interfaces (IAI) and Adaptive Intelligent Agents were used to increase situational awareness and decrease human workload. The paper explains that adaptive interfaces are driven by intelligent agents (automation), which aid in decision-making and thereby alters the workload level. The main challenges in regards to designing an Intelligent Adaptive Interface are discussed and listed as seen below.

- What are the reliable and cost-efficient IAI development methods?
- What are the universal usability principles that do not lead users' expectations astray?
- How and when can intelligence substantially improve the interaction?
- How do we evaluate whether the system supports users' real tasks?
- How do we design authoring tools to enable easy development and maintenance of the intelligent parts of the system (scalability)?

Each of these challenges has to be thought of when developing the intelligent interface. The paper also provides conceptual models and theories for building Intelligent Adaptive Interfaces, which can be used when designing the predictive aspects of the system. The paper does not conduct a user study to test the effectiveness of the interfaces as opposed to the intention of this project.

Work has also been done to figure out how to measure the situational awareness and workload effectively. When measuring situational awareness, the SAGAT method is a well-regarded method to use in studies. SAGAT stands for "Situation Awareness Global Assessment Technique" and was developed by (Endsley, 1988). SAGAT works by freezing the task at random during the given tasks and have the operator answer a few questions right away. The operators' responses are then checked up against the correct answers, which allows the measurement of the situational awareness. To do SAGAT, the correct answers must be known beforehand, and the task must be able to be put on pause without any undesired side effects.

To measure workload, it is possible to use the NASA-TLX method, which is a self-assessment technique (Hart and Staveland, 1988). Each participant is given a questionnaire after completing the tasks to assess their workload during the task. NASA-TLX asks the participants to rate the workload using the following six categories: Mental Demand, Physical Demand, Temporal Demand, Performance, Effort, and Frustration Level. A description accompanies each category to limit any misunderstandings of the category. The participants are asked to mark an x on a scale from low to high. The scale is a value from 0 to 100. However, this fact is not known to the participant. The result of the questionnaire is a workload score which is the product of a weighted calculation. An example is seen in Figure 2.1

DEMANDS	RATINGS FOR TASK 1:			RATING		WEIGHT		PRODUCT
MD	LOW	I x I	HIGH	30	×	3	=	90
PD	LOW	I x I	HIGH	15	×	0	=	0
TD	LOW	I x I	HIGH	60	×	5	=	150
OP	EXCL	I x I	POOR	40	×	1	=	40
FR	LOW	I x I	HIGH	30	×	3	=	90
EF	LOW	I x I	HIGH	40	×	3	=	120
				SUM			=	490
				WEIGHTS (TOTAL)			=	15
				MEAN WWL SCORE			=	32

Figure 2.1: An example of a NASA-TLX questionnaire (Hart and Staveland, 1988)

In (Agrawal et al., 2020) it is mentioned that there are nine demons of situational awareness. These demons are pitfalls that designers should consider and avoid when designing interfaces to ensure high situational awareness. The nine demons are presented in Figure 2.2.

Figure 2.2: Situational Awareness Demons - (Agrawal et al., 2020)

2.2 Predictive Interfaces

In related HCI research, it is stated that predictive interfaces are an effective tool to enhance the situational awareness of the operator. However, when it comes to research within human swarm interfaces, no clear results prove such a hypothesis. In (Roldán Gómez et al., 2017) four different human swarm interfaces are developed, and their performances in relation to situational awareness and human workload are tested and compared. The four interfaces they develop are classified as follows:

1. Conventional
2. Predictive conventional
3. Virtual reality
4. Predictive virtual reality

The predictive interface developed in (Roldán Gómez et al., 2017) is seen in Figure 2.3. More specifically, the predictive element of this interface is the warning triangle seen on the red UAV, which alerts the operator that the system finds a possible risk. In this paper, and in general, the approach when dealing with predictive interfaces is to interpret the UAVs as sources of information. The predictive element of the interface is then to filter that information. Furthermore, the paper also presents a set of design guidelines on decreasing the operator's human workload. This is mainly done by adjusting the LoA, i.e., having the system do more of the work instead of the operator. The human swarm interface design guidelines presented in this paper are as follows:

- Resistance to weather, environment, and harsh conditions.
- Reduction of the amount of information.
- Adaptation to the preferences of the operator.
- Guidance of operator attention to relevant information.
- Integration of robot position, health, status, and measurements in the same displays.
- Use of maps to show information about robots and missions.

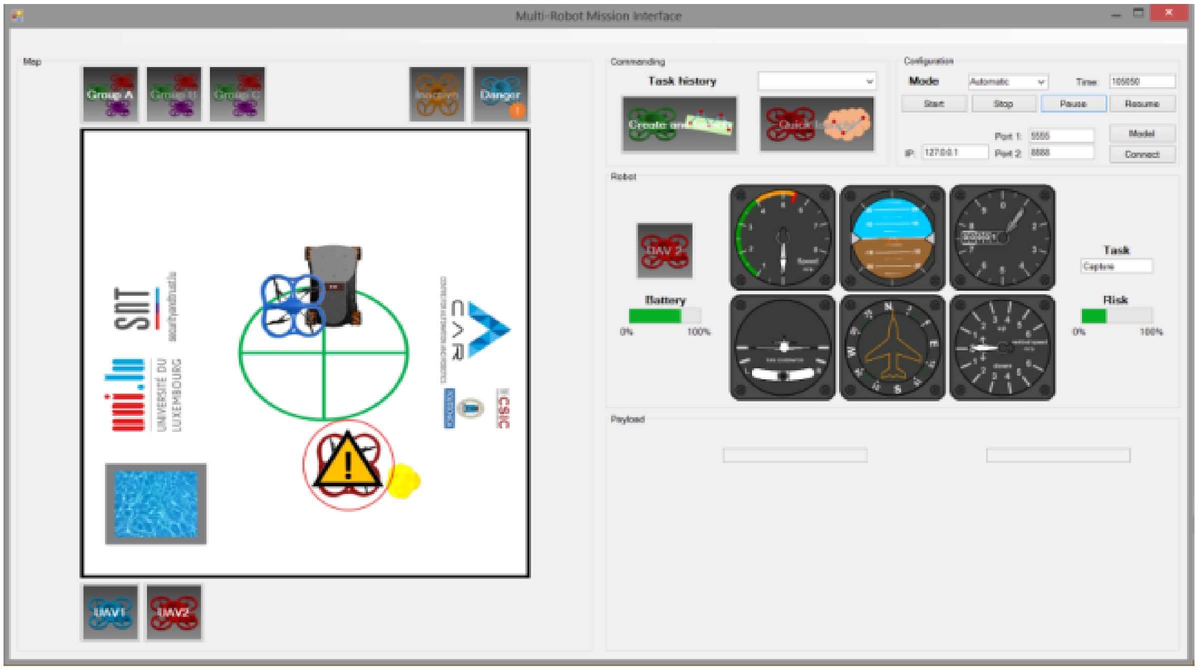

Figure 2.3: The predictive conventional interface developed in (Roldán Gómez et al., 2017)

In (Roldán Gómez et al., 2017) they conduct a series of experiments to analyze the performance of the four interfaces. For the experiments, they recruited 24 participants, and the NASA-TLX and SAGAT questionnaires are used to analyze and measure situational awareness and human workload. The analysis reveals that the four interfaces perform very similarly, i.e., there is no significant difference in the operator's situational awareness and human workload. The two interfaces with no predictive layer performed slightly better than the interfaces with predictive layers. The study also shows that the virtual reality interfaces performed better than the conventional interfaces. However, virtual reality interfaces are not considered for this project, as virtual reality interfaces are unsuitable for harsh environments like rescue boats. It would also limit the IC's possibility of communicating with his team and orientating outside the virtual reality interface. Furthermore, in (Hocraffer and Nam, 2017) it is stated that predictive interfaces are likely to reduce errors of the operator.

2.3 Task-Specific Interfaces

UAVs can be used for a wide range of tasks and operations, as they are versatile in their functionality and exist in all forms and sizes, which makes them attractive in many areas. This has lead to a lot of research into how to control UAVs and how such systems should be interacted with by humans. However, utilizing swarms of UAVs is still a relatively new area of interest, but having several UAVs working together to solve a single task leads to faster completion and allows the use of UAVs in more complex and more extensive tasks.

The article (Hocraffer and Nam, 2017) has done a comprehensive survey into the existing research of UAV swarm management and looks at challenges, limitations, and advantages of the current systems and research. One of the articles it has revived is (Peschel and Murphy, 2013), which looks at some of the roles in different missions or operations that utilize UAV swarms and which systems are used. (Peschel and Murphy, 2013) found that providing custom interfaces explicitly tailored for the operators' tasks improved the performance of the operator and reduced the error rate of the mission.

(Hocraffer and Nam, 2017) reviews three articles more about custom interfaces, one where a commercial interface is compared against a custom-built interface which resulted in a more comprehensive and intuitive interface for the operators. The last two articles looked at how UAV swarms could help firefighters find their way in smoke-filled buildings. Both found that a custom-made interface integrated into the firefighters' gear yielded the best results.

This shows that a custom build interface for the specific task at hand helps operator performance and could have an impact on the operator's situational awareness and mental workload by having tools and elements focused on assisting the operator do concrete actions and only showing the relevant information for the operation opposed to displaying all information and generic tools.

3 | Research Problem

3.1 Research Problem

This section discusses the current research and technology, as well identify and formulate the missing gap in the existing research. Finally the intended contributions of this project are outlined. Looking at the existing research within the field it it clear that having a high situational awareness can result in better decision making and increase performance. Workload has a direct effect on situational awareness, however the correct amount of workload can vary. Adjusting the level of automation can adjust the workload for the operator, but too much automation can cause issues. There is conflicting research about predictive interfaces and their effect on situational awareness, and papers have encouraged further studies to help understand the effects of predictive interfaces better. As described in (Jensen et al., 2020) the research gap being addressed is:

What effect will predictive features have on the situational awareness and workload of the Incident Commander in maritime SAR operations?

In (Roldán Gómez et al., 2017) the same research problem is addressed, however they fail to prove their hypothesis about predictive interfaces having a positive effect on situational awareness. Since the existing research fail to give evident contributions to the research problem there is still basis for conducting more research on that specific topic, which is the intention of this report.

This project is addressing the research gap found in our 9th semester project which is presented above. However this report will focus on contributing to the research about the effects of predictive interfaces using the IC's role in the system by creating a custom interface tailored specifically to his tasks.

4 | Analysis

4.1 Process Model

This section will discuss the choice of process model for this project. In the previous semester described in Jensen et al. (2020) a home grounds analysis was conducted to evaluate if the project fit an agile or a plan driven approach. The analysis revealed that the project fits well with an agile approach. However the team size has changed along with the focus of the project which warrants a reevaluation of the previous analysis. The home grounds analysis evaluates a project using 5 axes of different criteria to determine if the project is suited for an agile or plan driven approach. The axes include Personnel, criticality, size, culture and dynamism. The group did a self assessment of each criteria and filled out the polygon seen in Figure 4.1.

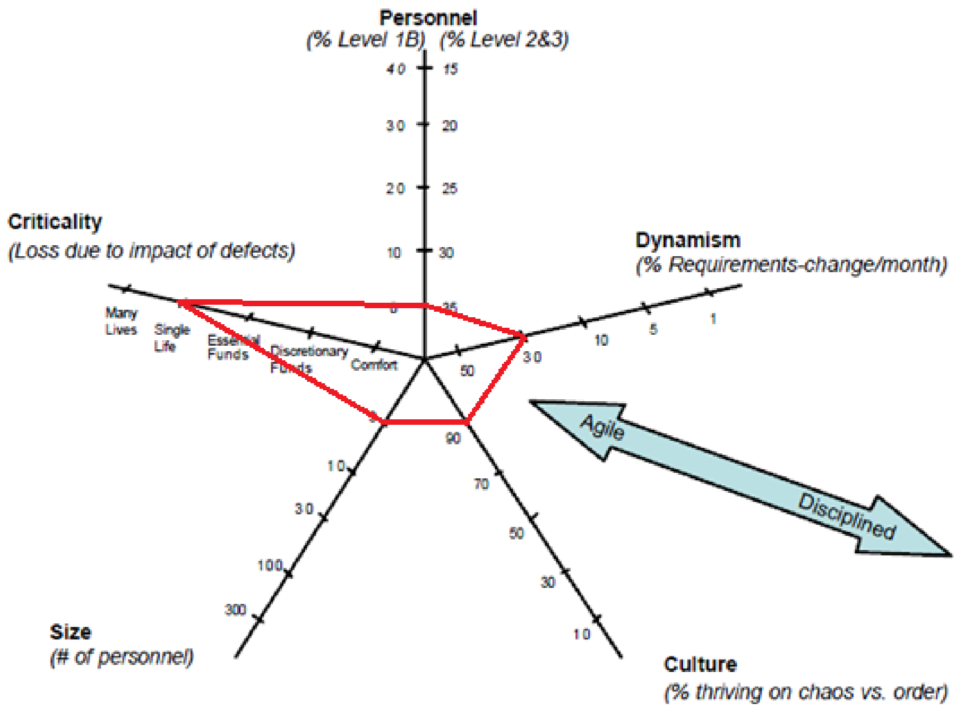

Figure 4.1: Home Grounds analysis

The results of the analysis are very similar to the one conducted last semester however the size and criticality axes have changed. The team size has been reduced from six to three members and the criticality has been increased from *Discretionary Funds* to *Single life*. The reason behind this change is the new analysis looks at the project as if it was to be used in a real world setting, whereas the old analysis looked at the project as a non deployable project.

The analysis revealed the project is still suitable for an agile approach despite the high criticality. The group decided to continue the Scrum approach with the addition of pair programming where possible. Pair programming will ensure a higher quality of programming and save time on subsequent code reviews. All roles, artifacts and meetings are in full effect as per the 3-5-3 process of scrum (Schwaber and Sutherland, 2017). The backlog is maintained on GitHub and each user story is assigned story points during the planning meeting using planning poker. The amount of story points the group is able to handle per sprint will be monitored and the planning will be adjusted accordingly.

All three members are a part of the development team while one member has been designated as the scrum master. The stakeholder Robotto functions as a product owner as described in section 4.2. Robotto will however not be present on a daily basis as the Scrum guide recommends, but meetings will be held where the direction of the product is discussed.

4.2 Stakeholders

This section presents the two stakeholders of this project. The two stakeholders are the third party company Robotto and the emergency management agency (EMA) in North Jutland called Nordjyllands Beredskab.

4.2.1 Nordjyllands Beredskab

Nordjyllands Beredskab is the emergency management agency in North Jutland and therefore they are responsible for carrying out maritime SAR operations in Aalborg. On the pre-specialization project in the fall 2020 Nordjyllands Beredskab was also a stakeholder. As Nordjyllands Beredskab is the intended user of the interface, the information gathered from them through phone meetings and emails will be used in the design process.(Beredskab, 2020).

Today there are two different approaches within the EMA when dealing with maritime SAR operations. The first approach is applied in cases where the missing person is deemed to be within close proximity of the rescue boat. Here both the search and the rescue parts are carried out by the rescue team in the rescue boat. In cases where the there is a higher uncertainty about the location of the missing person, the EMA will have the danish military launch a helicopter to assist in the search of the missing person. The helicopter search is the part that could potentially be replaced by a UAV swarm (Jensen et al., 2020).

4.2.2 Robotto

Robotto is a third party breakout company based on a bachelor project from Aalborg University. Robotto are dealing mostly with UAV wildfire detection but are interested in expanding to maritime SAR operations (Robotto, 2020). In the pre-specialization project Robotto acted like the product owner and they will do so for this master thesis project. Since Robotto is a professional company that exclusively works with utilizing UAVs they have a lot of experience and knowledge they can share which can enhance the quality of this thesis.

4.3 Hardware

This section will discuss the hardware needed for the IC interface to become a reality, as well as present the equipment used in the project and which hardware will be used in the user study.

For an actual implementation of the interface which could be used in its intended setting some special hardware is needed. Due to the harsh environment of the IC's location during a SAR operation, the hardware needs to be able to withstand being wet and possible handled roughly as everything need to happen fast, and there is no place for delicate equipment which break easily. Because of the harsh environment, the IC will more likely than not wear protection against the weather often including gloves. (Jensen et al., 2020) states the interface should be operable by touch and physical buttons. This implies that the hardware used should provide both touch and button inputs which should be functional even if the IC is wearing gloves.

4.3.1 Rugged Tablets

Figure 4.2: Getac UX10

One option for these requirements could be a rugged tablet, which is tablet size computer designed for use in commercial purposes such as warehouses or more harsh outdoor work. Rugged tablets often have an hard exterior casing with rubber covering in order to be able to withstand being handled roughly, and most of them also have a touch screen which can be used with gloves on. Some of the tablets also comes with a few physical buttons already integrated into the tablet, while the rest of them still offers the possibility to extend the functionality via I/O ports on the tablet. One option for a rugged tablet is the Getac UX10 seen in Figure 4.2, which offers physical buttons and a touch display which can be used with gloves.

Another option for rugged tablets is to take an normal tablet and install it into a special case which will protect the tablet from the environment, but normal tablets normally do not function well with gloves unless they have conductive elements in them.

4.3.2 Avionics

Figure 4.3: Garmin G1000

Another option which could be used for the display is avionics displays which are highly customizable displays with touch screens and a wide range of connected buttons and dials. An example of an avionics display can be seen in Figure 4.3, which is the Garmin G1000 display. These displays focus on high functionality and fine tuning often providing the user with a massive amount of options within the display, by nesting menus in menus which is navigated by the buttons at the bottom of the display. As the Garmin G1000 is used in critical settings like airplane cockpits it provides a solid basis that tablets like an iPad is valid to use in critical situations.

The avionics displays are designed with expert users in mind, and expects of its users to know and understand the functionality and operations that the display offers. The displays however are not handheld devices or designed to be slashed with water, and it is not certain that the touch will work with gloves on either. The fact that the display is not handheld is not a huge problem as a solution which would be integrated into the boat is also a possibility, but the fragile nature of the display might not be suitable for the environment this project is set to be used in.

4.3.3 Hardware Availability

To conduct the user study later in the project it is necessary to get a tablet which fits the requirements of the study. The group reached out to AAU's equipment office and got a list of available tablets. AAU has tablets from many different vendors including Lenovo, Samsung, Acer and Apple. However all the available tablets are running either Android OS or iOS where the group prefers a Windows OS for compatibility with 3rd party hardware accessories. Android and iOS tablets are still an option if no other alternative proves to be better.

The group has an iPad Pro available as another option, however iOS is not known for great compatibility with 3rd party hardware which might limit accessory options, however the iPad

Pro should have improved compatibility compared to its predecessors. Using a personal tablet will give the group more control and avoid spending AAU's funds to buy a completely new tablet.

4.3.4 Choice of Hardware

For the user study of this project the group has chosen to use the iPad Pro as the device to run the interface. The reason behind this choice takes its starting point in the fact that the goal of this project is to conduct a user study to investigate a hypothesis, i.e. the goal is not to release a complete system. Therefore, it is sufficient to select the available iPad Pro. One of the benefits of selecting the iPad Pro is that it takes almost no time to build and run a PWA on an iPad.

In the requirements found in Section 5.1 it is seen that the device is required to have physical buttons as a supplement to the touchscreen, as the interface is to be operated in harsh environments. This requirements can still be fulfilled using an iPad Pro since it is compatible with 3rd party hardware meaning that a device with physical buttons can easily be connected to the iPad Pro. By doing so a device with physical buttons can be simulated for the conduction of the user story.

5 | Method

5.1 Requirements

This section provides an overview of the design considerations and requirements gathered from the previous work and the additional requirements elicited through analysis and new interviews. These considerations will influence the design choices, and the requirements will be the foundation for implementing the system used for the user study. Each requirement's relevancy is evaluated before the user study, and the relevant requirements will be implemented before the study commences. The feedback from the study will lead up to a revision of the requirements.

5.1.1 Interviews

This section presents the conducted interviews concerning this project and discusses the key points gathered from the interviews that shape the future study. In the 9th semester pre-specialization project, a number of interviews were conducted (Jensen et al., 2020). At the beginning of the semester, an interview with the IC at Nordjyllands Bredskab was conducted to gather initial requirements. At the end of the semester, two interviews were conducted with Anders La Cour-Harbo and Robotto, respectively. La Cour-Harbo is a leading expert within UAV research on Aalborg University and Robotto, the product owner of the project as discussed in Section 4.1 and 4.2. These interviews act like expert reviews of the system, and they yielded a set of general requirements and a set of role-specific requirements for the IC interface. The requirements are presented below.

5.1.2 Requirements Gathering

In the (Jensen et al., 2020) report, a general set of requirements was elicited along with a group of specific requirements for the IC interface. The general requirements are seen in Appendix A.6, and the IC requirements are listed below.

1. The interface shall be predictive and notify the IC of critical situations.
2. The interface shall provide components with a focus on easy interaction.
3. The system shall be operated on a device resistant to harsh maritime environments.
4. The system shall be able to receive information from the OM.
5. The device shall be operable by physical input components.

6. The device shall be operable with gloves on, such that the operator does not need to take off protective gear to operate the device.

7. The interface shall show the location of the IC on the map, such that the IC can gain a better overview of the operation.

8. The interface shall only show the essential information for the IC. The information is easy to find, and the operator is not confused with multiple kinds of information.

These requirements were elicited from the research into related works and interviews conducted with Nordjyllands Beredskab in (Jensen et al., 2020), this report will not focus much on fulfilling the general requirements as the base system already has fulfilled a large portion of them, and continuing on these requirements does not necessarily add value for the study in this report.

5.1.3 Design Considerations & Requirements

This section presents the design considerations of the IC interface and system. To categorize the requirements, they have been split up into functional and non-functional requirements. Functional requirements describe what the system must do, while non-functional requirements define the constraints (Helen Sharp, 2019). The design requirements in focus for this project are the ones that will affect the upcoming user study. The final list of requirements making out the design rationale is seen below.

Functional Requirements

1. The interface shall support a predictive layer that notifies the IC of potential persons in the water.

2. The interface shall support a conventional interface without the predictive aspects.

3. The system shall be able to simulate an ongoing rescue situation.

   (a) The system shall simulate live video feeds by playing pre-recorded videos.
   (b) The system shall simulate UAVs moving on the map.
   (c) The system shall simulate a sailing boat on the map.

4. The system shall be operated on a device resistant to harsh maritime environments.

5. The device shall be operable by physical input components.

6. The interface shall show the location of the IC on the map, such that the IC can gain a better overview of the operation.

7. The system shall support logging of events to allow for easy time measurements.

Non Functional Requirements

1. The interface shall provide components with a focus on easy interaction.

2. The device shall be operable with gloves on, such that the operator does not need to take off protective gear to operate the device.

3. The interface shall only show the essential information for the IC. The information is easy to find, and the operator is not confused with multiple kinds of information.

4. The workload should not be higher than the IC can handle.

The requirements listed above will lay the foundation for the upcoming implementation of this master thesis project. The coming section presents prototypes for the interface design. These prototypes are based on the requirements given in this section.

5.2 Research Platform

This section aims to show the intended design of the interface through prototypes. The prototypes will be the foundation for the frontend design. The prototypes aim to account for the requirements elicited in section 5.1. The main priority is to make the interface easy to use on the tablet described in Section 4.3. The interface has two types of screens which will be shown to the IC depending on the status of the operation.

While searching in an active SAR operation the search screen is shown which includes a map with the location of the boat and UAVs, and selectable live feeds as seen in Figure 5.1. The interface has been simplified and irrelevant information has been removed, which should give a better overview compared to the design shown in the previous semester (Jensen et al., 2020). The map is draggable to allow the operator to adjust the shown area, and each UAV is labeled with a number which is also shown in the corner of the corresponding live feed. When clicking a live feed it will jump into focus in the bottom right corner, to allow the IC to have a closer look at the live feed. The live feed section has been moved to the bottom of the screen as a horizontal bar instead of the previous vertical bar to accommodate for the aspect ratio of the live feeds. The live feeds can also be selected using input from the hardware buttons instead of using the touch display in case of harsh conditions or incompatible gloves.

Figure 5.1: Initial search screen prototype

Figure 5.2 shows the interface when a body is located in the water by a UAV in the system. When compared to Figure 5.1 it is seen that the person is displayed on the map alongside a direction, distance and angle. The live video feed of the UAV that has located the body is enlarged compared to the search screen.

Figure 5.2: Initial prototype of when a body is located

As the interface should support predictive elements, there needs to be a way for the system to notify the IC when and if something is detected that he needs to address. As seen in Figure 5.3 this will be an info card which will show the livefeed containing the object, and a short text explaining which UAV triggered the notification. At the bottom three buttons are seen. One to confirm the prediction, one to ignore it, and one to get a better view of the livefeed. The confirm and ignore will both close the box, but if confirm is selected there should still be elements on the map aiding the IC in reaching the person.

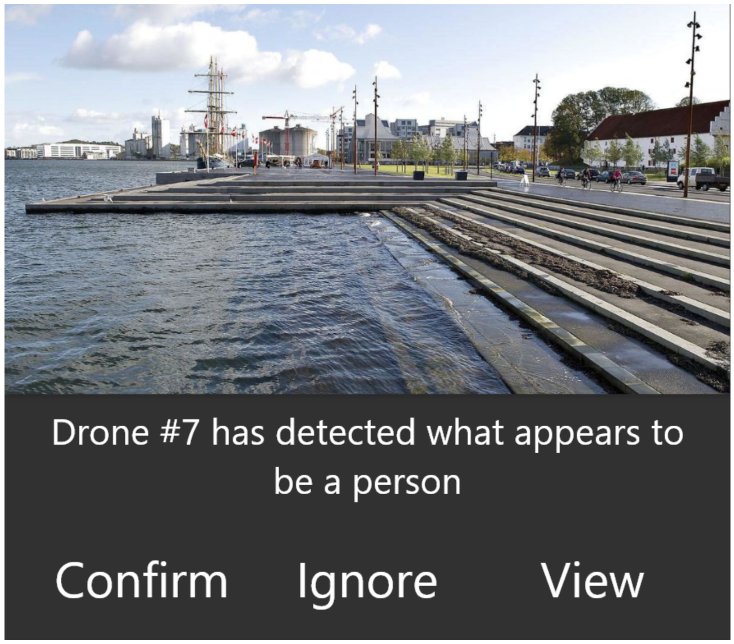

Figure 5.3: Info Card notifying about detection

5.3 Incident Commander Platform

This section describes the choices behind the system implementation in preparation for the user study. There are three main areas that have been improved to get a functional Demo up and running. The three areas consist of the study simulation on the backend, the graphical changes made to the frontend, and the added machine intelligence to enable detecting of a person on the live feeds. The following sections describe each of the three areas in detail and discuss the choices made. The part of the system being simulated can be seen in Figure 5.4.

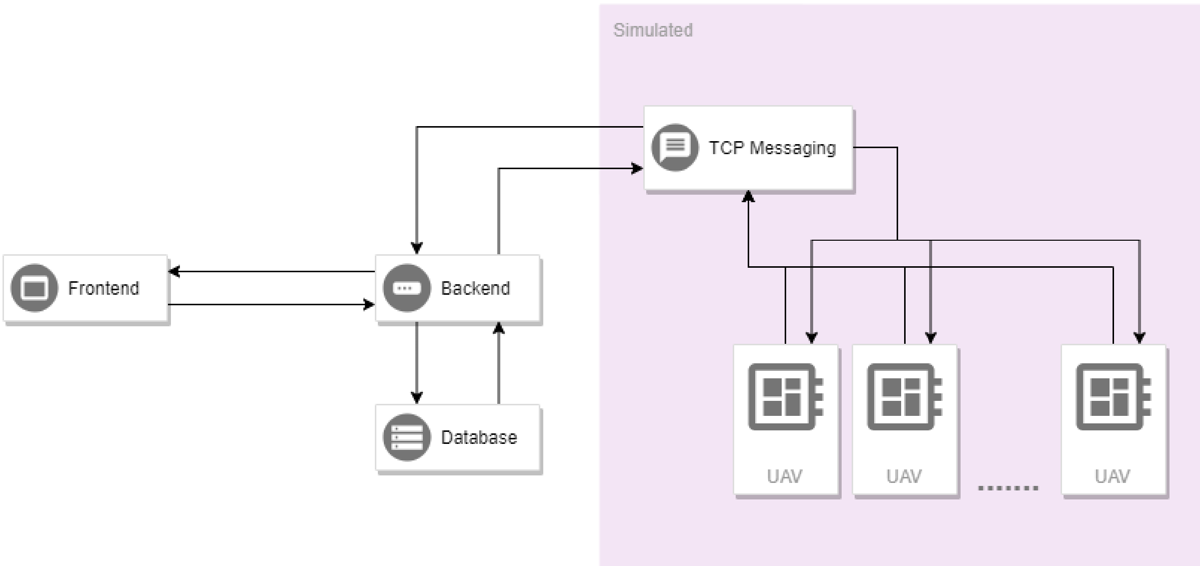

Figure 5.4: System Overview

5.3.1 Human Swarm Interface

Our 9th semester project (Jensen et al., 2020) resulted in a generic interface that could be adapted to the needs of the IC. Furthermore, a list of requirements elicited from expert interviews is also provided. Based on the findings in (Jensen et al., 2020) the prototypes in Section 5.2 was made with the focus that it should be viewed on an iPad Pro. This Section will cover how the generic interface was adapted to look like the prototypes, and utilizing the touch interface that the iPad Pro provides, as well as be able to be controlled by using a connected set of physical buttons

The final interface is seen in Figure 5.5. It is close to the design of the prototypes but deviates in few areas, having access to run it on an iPad Pro gave a better understating of how the elements in the interface should be put together.

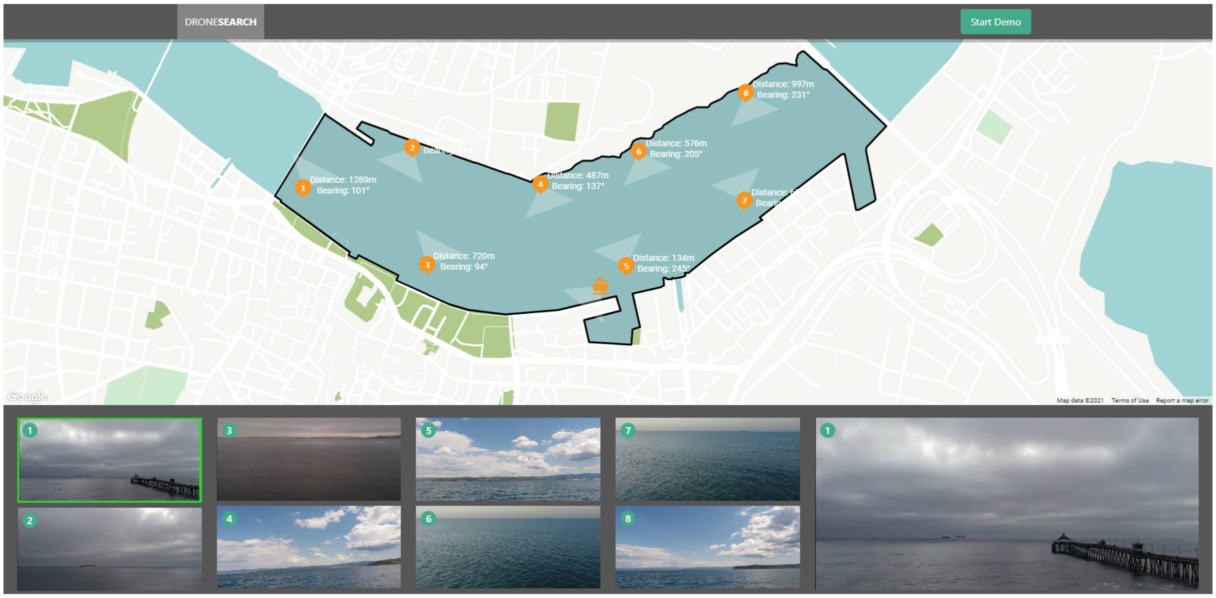

Figure 5.5: The final interface

The development of the interface was done over three development sprints, each taking about two weeks. The first sprint was focused on getting the interface to look as much as possible like the prototypes and fulfilling the requirements. The second sprint focused on the predictive elements of the interface, both the artificial intelligence, which is discussed in Section 5.3.2 and how the predictions should be expressed in the interface. The last sprint focused on fixing bugs and missing features, which was found by having a run-through of the system in the group.

The interface consists of multiple isolated UI elements and is dependent on services to provide it with cross-functionality and data. The primary elements of the interface are described in the following sections.

The Livefeed element:

The live feed element seen in Figure 5.6 will stream a single live feed from a provided URL. The UAV that the feed belongs to can easily be identified by the circle in the top left corner. The number in here is the number of the UAV the feed belongs to.

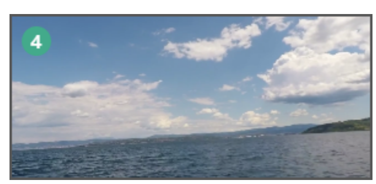

Figure 5.6: Livefeed Element

For the purpose of the user study, this component will also activate the predictive elements in the interface. Suppose the player is given a URL that has been appended with a prediction. In that case, it will then, based on static timestamp, activate the predictive elements when the video reaches this timestamp.

The Livefeeds Bar:

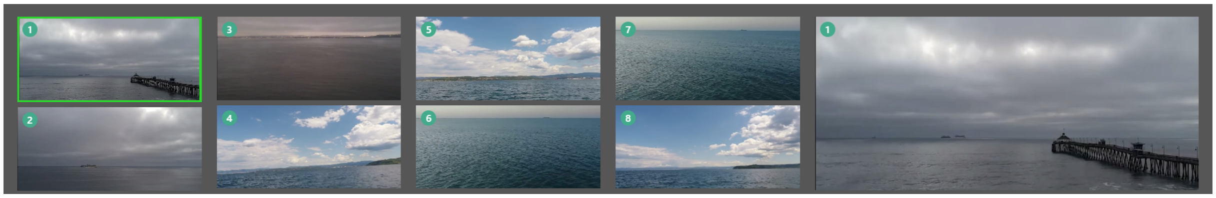

Figure 5.7: The Livefeeds bar

The live feeds bar seen above in Figure 5.7 consists of up to eight different live feeds and then a larger view of one of the feeds. The number of live feeds, their size, and the larger view's size is a delicate balance as the bar can not take up to much of the screen space, but it still needs to contain enough feeds without the feeds being so small that you can't see details in them. The correct balance and sizes were found by trial and error on the iPad Pro to have the right sizes for the screen.

As mentioned, there is a large view in the bar. Any of the eight feeds can be selected for this by either touching / clicking a feed or navigating to a feed using the numpad keys (4,5,6,8) and pressing Enter on the feed. The feed currently being displayed in the large view will be marked with a green border to indicate that it is the selected feed.

The Fullscreen view:

A third view of a live feed in the interface is the fullscreen view. This view can be activated by double-clicking/touching or pressing Enter again on an already selected feed in the live feed bar.

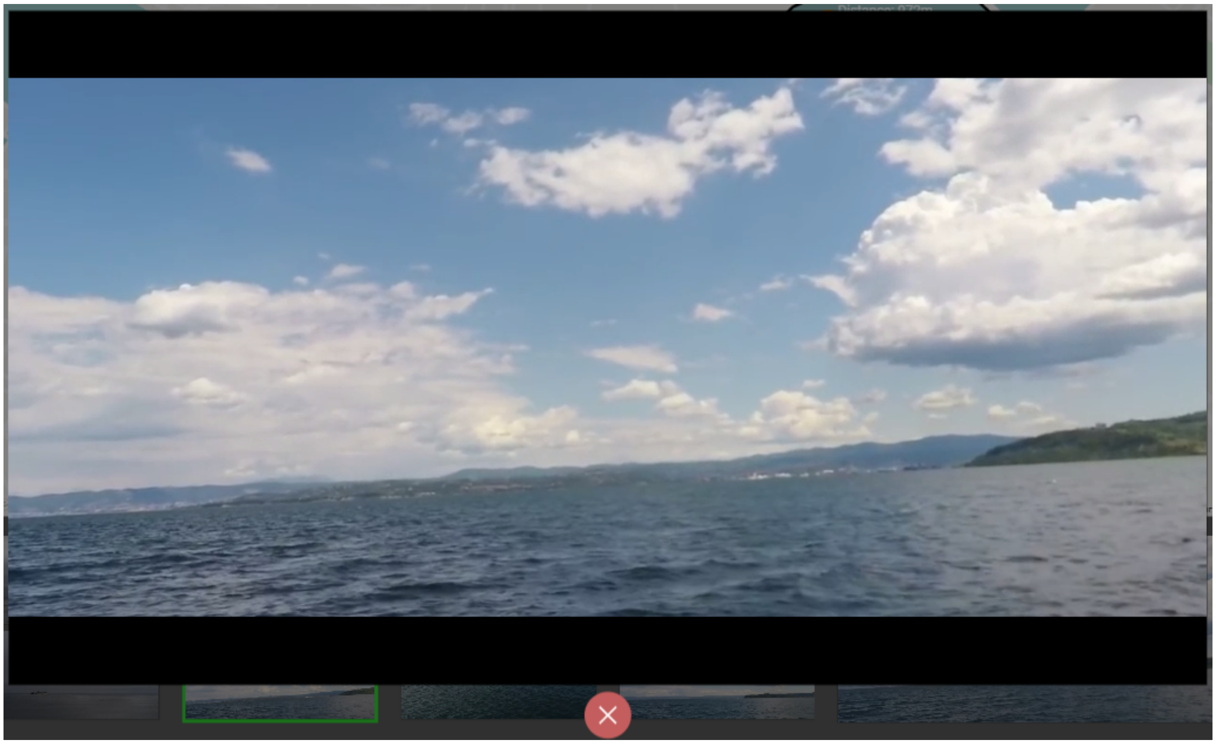

Figure 5.8: The fullscreen view

When opening the fullscreen view, the feed will open in a popup, taking up most of the screen space as seen in Figure 5.8. This option offers a much larger view of a feed if the user is in doubt or needs a better look at something on a feed and can be closed again by touching/clicking on the red circle floating under the view or pressing Enter again.

The Map and its markers:

Figure 5.9: The Map

The map seen in Figure 5.9 is the main element in the interface taking up the most space. The map is responsible for showing the position of all the moving parts in the system, such as the UAVs, the boat, and the located person. The map itself is an embedded version of google maps, with a custom styling on it. The custom styling sets the colors of the map but also removes all the company markers and text from the map, as this is unnecessary information for the IC.

The UAV Marker

The UAVs in the system are represented by the marker seen in Figure 5.10. The marker contains the UAVs identification number in the center, and then from emerging from the marker is a 90-degree cone representing the orientation that the UAV is pointing. Next to the marker is two labels, the first one shows the distance to the boat, and the second one is the bearing from the boat to the UAV. These labels help the IC get an overview of the UAVs' positioning and quickly navigate to one of them if needed.

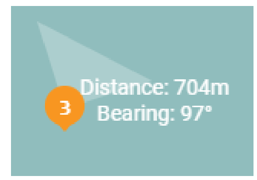

Figure 5.10: UAV marker

The UAV markers can also have different colors, orange representing an idle state and green representing an active state such as searching.

The Person Marker

The person marker is placed on the map after a prediction has been confirmed. It indicates the person's current position based on the UAV that detected the person in the water. The pin is placed in front of the UAV and is red to draw extra attention to it. The marker does not currently have any distance and bearing label as it would often overlap with the label of the UAV. The difference between the person and the drone is negligible, and creating new labels is deemed unnecessary.

Figure 5.11: Person Marker

The Boat Marker

The boat marker seen in Figure 5.12 is represented by a boat icon and has an orientation cone the same way that the UAVs have. The boat marker is there to help determine where the IC is located in relation to the UAVs and the found person and the orientation cone, which allows him to get a sense of which direction the UAVs are without looking at a compass.

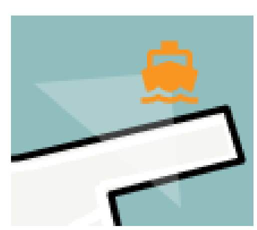

Figure 5.12: Boat marker

The Info card

The info card is a popup that appears on the screen's right-hand side when the predictive interface is active. It is triggered by object detection on the live feeds. The card includes the feed that triggered the popup to prevent the IC from finding the feed in the live feed bar himself. The info card asks the IC to either confirm or ignore the detection of the object or view the feed in full screen. If the IC confirms the detection, a new pin will appear on the map to show the detected object's location as described above. If the detection is ignored, nothing will happen, and the info card will disappear. The info card is seen on Figure 5.13

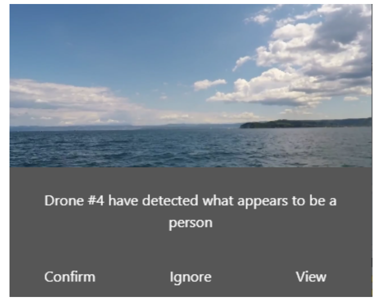

Figure 5.13: Info Card

For more details on how some of these elements are implemented consult the sprint logs in Appendix A.1 and A.2. The elements in the interface rely on a few services to get the required data and work in sync with each other. The most important services are:

- **The UAV service:**
  Will request information about the UAVs in the backend system to provide the interface with the UAVs positions and statuses. This information is updated once every second.
- **The Predictive service:**
  Offers the functionality to have elements only be visible when a prediction has been made, along with a way to tell if a prediction has been made and which UAV the prediction belongs to.
- **The Map service:**
  Contains functionality regarding the map and mathematics dealing with GPS navigation.

The interface contains more services providing the elements with functionality or data, the services created or changes in this project can be read more about in Appendix A.1 and A.2.

5.3.2 Convolutional Neural Network

This section describes the use and reasoning behind the implementation of a convolutional neural network. As defined by Section 5.1, the system must be predictive to inform the IC about critical situations. As soon as that requirement was elicited, it became clear that the system needs artificial intelligence to assist in detecting human beings on the live feeds. For the implementation, the group had two options going forward. Make a new Deep Neural Network from scratch, do the training on personal hardware, or find a suitable framework with the functionality needed for this project.

The first option of creating a neural network from scratch is interesting. However, very time consuming. While the group has previously made a convolutional network from scratch, including the training in (Jensen et al., 2018), it would still require a lot of research to acquire the latest knowledge on the technology and ways of deep learning. Optimization would also be a challenge as the created prediction model will have to run on several live feeds simultaneously in real-time. Gathering a data set for the training could also prove difficult as several thousands of images are required to train an effective neural network.

The second option of finding a suitable framework is more realistic with the given time frame. By using an existing neural network, the group can skip the process of finding a data set and training the new model. Finding the proper framework is essential as it needs to have the right training to look for humans in the water, and it needs to be fast. After weighing each option's pros and cons, the group decided to go with option two and find a framework instead of spending time creating one from scratch, which is a whole project in itself. The search for a suitable framework came to an end with YOLOv3.

YOLOv3 - Object Detection and Classification

YOLOv3 is the third iteration of a convolutional neural network with object detection and multi-label classification (Redmon and Farhadi, 2018). YOLO goes through several steps during the processing of an image, the first being bounding box prediction. The coordinates for each corner of the bounding box are calculated to encapsulate the object. Several bounding boxes may contain the same object. However, an objectness score is calculated for each box using logistic regression, and the box with the best fit is kept going forward. Logistic regression is a method used to determine a true or false value for a given prediction (Swaminathan, 2018). In this case, the logistic regression will return 1 if the bounding box is the best fit for the object.

Once the bounding boxes have been calculated, the next step is class prediction. YOLO uses 53 convolutional layers for feature extraction, which can be seen in Figure 5.14 and has opted not to use a softmax as that would imply that each bounding box only has one correct classification. Softmax returns a probability distribution meaning all the values sum up to 1, with the most likely class having the highest value. This is created for one-or-the-other classifications but is counterintuitive when an object can have multiple labels simultaneously, such as animal and dog.

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Figure 5.14: Darknet53

All of these calculations are done for each image given to the framework, and the framework is capable of doing these calculations several times per second given the proper hardware. This allows YOLO to process video feeds at a reasonably high frame rate. This is one of the main reasons for choosing YOLO for this project. Using an RTX 2080 Super, it was possible to maintain a stable frame rate at 60, double the amount of frames recorded by a typical web camera. In a real-world scenario running YOLO on multiple camera feeds would require linearly scaling hardware resources with the number of live feeds.

It is important to mention that YOLO is not a perfect fit for this project, and the source code was modified and recompiled to suit the needs of the project. YOLO is a multi-label classification framework, but it was only necessary to find humans and nothing else in this project. During the initial testing of YOLO on some of the live feeds, the framework would return items such as surfboard and boat, which would trigger the predictive interface. To avoid this, the changes made to the source code limit the classification only to report the person classification and discard the rest. The same model is used, and no retraining was necessary.

For the demo, the YOLO framework is run on pre-recorded live feeds, and the output containing the bounding boxes is saved and uploaded to the server and used as predictive live feeds for the frontend. The model was run on a personal PC as the server does not have a graphics card to speed up the computation. This also limits the ability to have the server do the real-time processing of the live feeds, which is another reason the pre-recorded live feeds have been pre-processed. If the system was to be operational, the server would need multiple graphics cards to have enough computational power to do real-time prediction on all of the live feeds.

5.3.3 User Study Simulation

This section discusses the simulation changes implemented on both the frontend and the back-end of the system. Simulation changes covers the changes made to the system in order for it to support the upcoming user study. The study design of the upcoming user study is presented in Section 5.4. However, the user study is not going to take place in the natural setting of the IC in a maritime SAR operation i.e. on a rescue boat.

As previously mentioned the user study is going to utilize an iPad Pro with a connected numpad keyboard while the participant is sitting down with the iPad and numpad keyboard laying on a table in front of them. In order for this setup to work in a user study on an interface that is designed to be operated by an IC in a rescue boat, various simulations are needed. The simulations implemented in the system are:

- Simulate UAVs searching an area
- Simulate live video feeds
- Simulate sailing rescue boat

Simulate UAVs searching an area

The first activity that needs to be simulated in order for the system to be applicable for the upcoming user study is to simulate searching UAVs on the interface's map. On Figure 5.15 it is seen that an area of Aalborg Harbor is outlined in black, this area highlights the intended search area. Inside this area eight UAVs are seen marked with the numbers 1-8. Furthermore, the position of the rescue boat is also visualized inside the search area. Obviously, it is not possible to visualize movement on a picture however, Figure 5.15 is a snapshot of an ongoing search where UAVs 1-8 search the intended search area.

Figure 5.15: The interface: UAVs Searching

The search is launched by clicking the start/pause/continue button in the top right corner of the interface. When the start demo button is clicked an endpoint is called this endpoint triggers a method on the backend that handles the simulation of the UAVs. The algorithms implemented to enable the simulation of searching UAVs are described in great details in the sprint log Section A.2.3.

Simulate live video feeds

The next element of the system that needs to be simulated is the live video feeds. In a real world scenario where the UAVs were not simulated they would provide live video feeds. However, for the sake of the upcoming user study these need to be simulated as well. This simulation is done by having recorded videos stored on the server, these videos are then played when the demo is started. The video feeds are outlined on the bottom of Figure 5.16.

Figure 5.16: The interface: Video feeds

The videos are made from collected maritime UAV footage found on the internet. The last thing missing in relation to the simulated live video feeds is a person in the water. The research conducted in the master thesis is revolving around maritime SAR operations. Therefore it is needed to simulate a person on one of the live feeds in order for the user study to be relevant to the research question of this thesis. To accommodate for this a swimming person has been edited onto the recorded videos. It is then this very person the operator is supposed to detect and then "rescue" during the user study.

Simulate sailing rescue boat

The final simulation of the system is the simulation of the rescue boat. As mentioned, the user study is not going to take place in its natural setting i.e. in a rescue boat. Therefore, it is necessary to simulate the rescue boat on the map in order to conduct a meaningful study.

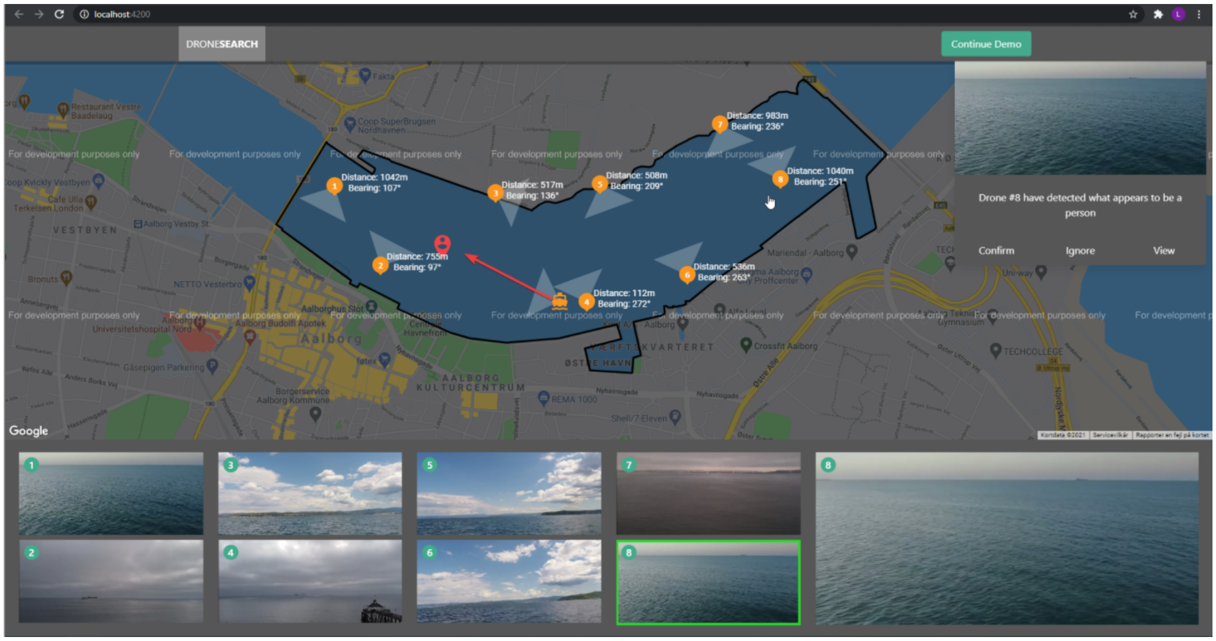

Figure 5.17: The interface: Rescue boat sailing

The intention behind the rescue boat in the user study is, when the operator detects a person somewhere in the water, he should navigate the boat to the location of the missing person by saying the bearing and distance out loud. This has been implemented so that a study moderator can click anywhere on the map and the boat will then sail towards the desired location. It has been visualized on Figure 5.17. Here the moderator has clicked on the person icon on the map and the rescue boat is therefore on its way there.

This third simulation rounds op the simulations implemented in the system to mimic real world activities. It is deemed that these simulations have a minimal effect on the intended measurements of the upcoming user study. The goal of the user study is to measure the operators situational awareness and human workload during a maritime SAR operation. These measurements are conducted on two version of the interface with predictive elements and on a version without predictive elements. The performance of the interfaces are then compared in terms of situational awareness and human workload.

5.4 Study Design

To better understand how big of the impact of the predictive elements in our interface have, we present a two part user study on the interface. The first part of the study will be conducted on naive users, which will receive training in the interface, as they are not SAR experts. This is done as to measure if the predictive elements have an positive impact on situational awareness and human workload. The second part of the study will be conducted with expert SAR or UAV operatives, but there will be fewer participants. The experts will be invited to a contextual interview to promote a co-design session of how the interface could be improved as to better aid their needs in the field.

5.4.1 Setup

The interface will be presented on an iPad Pro with a connected numpad keyboard for the participant to use Though the numpad it is possible for the participant to stop a UAV, change between the live feeds on the screen, or view a live feed in fullscreen mode. This setup can be seen in Figure 5.18

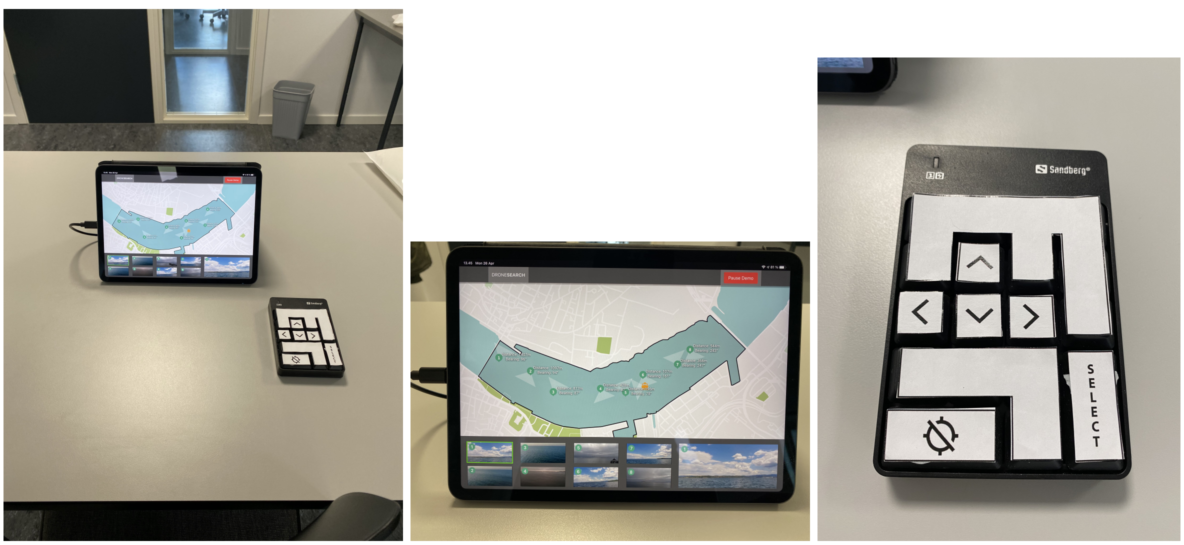

Figure 5.18: First: Participant setup, Second: The Tablet, Third: The keypad

During the study, only one conductor will converse with the participant. The rest of the group will be sitting a small distance away, but still close enough to observe clearly what is going on and listen in on the conversations. The conductor will have a computer available from where he can also see the interface, but also have an admin menu available from which he can pause the demo, turn on or off the predictive elements or change which predictive mode the interface should be in. The changes are synced with the participant's interface so changes will happen immediately. The conductor's setup can be seen next to the participant's in Figure 5.19

Figure 5.19: Conductor setup next to the participant

The observation team, consisting of the rest of the group have their own setup. One members will act as a "technician" and will monitor the admin page, the same page the conductor has open, and makes sure that the interface is set up right for the tasks at hand. The technician will also create custom log entries between participants, so it can be clearly separated later. The other member will act more as an observer and note down questions and feedback during the study, as well as write down the participants answers to the post survey interview. This setup can be seen in Figure 5.20.

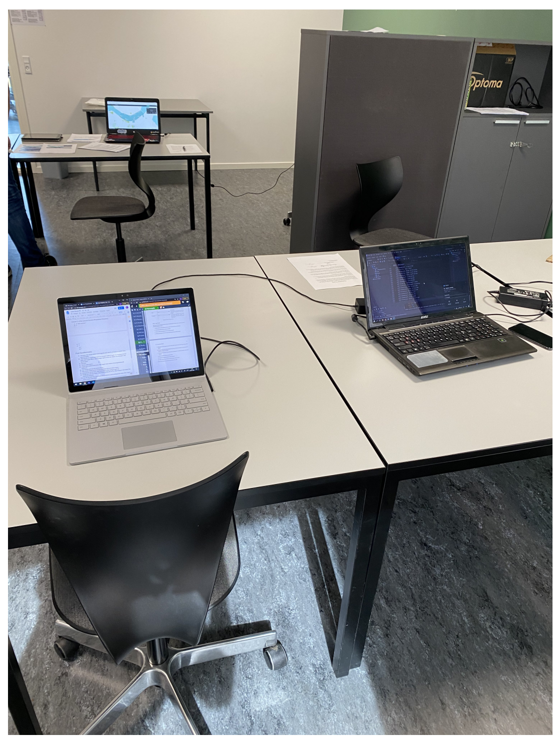

Figure 5.20: The observation team's setup

The study will be conducted using the Wizard of Oz technique, as there are multiply parts of the system which are simulated. Instead of having UAVs flying around, their movement is simulated in the interface. The video feeds in the interface are also pre-recorded and some of them pre-processed by a neural network.

The interface will create a log over certain actions, such as when the demo starts and stops or chooses one of the option in the predictive information. The logs will contain timestamps and a description of the action, which will help get accurate time measurements later on when analyzing the data.

5.4.2 Participants

The participants for the study consist of ten students from the Computer Science department at Aalborg University, all students were male and in the age group of 22 to 26. As they are all studying computer science they should be able to pick up on the interface relatively easy, as they are accustomed to using all kinds of technology. However none of the students are domain experts, so some deviations are to be expected in the results.

5.4.3 Training

In order to make the participants familiar with the interface, and avoid confusion in the actual study about how to operate the interface, all participants will go though a quick training exercise. The training will teach the participants to navigate the interface, and understand the basic elements that they will use in the study. A quick list of tasks is set up for the training which the participants will go though. The training tasks are seen below.

Training Tasks

1. When you feel ready please start the demo.
2. Select feed 3 to put it into focus.
3. Navigate to feed 6 using the numpad keys 4,5,6,8 to put it into focus using enter.
4. Make feed 6 go into fullscreen mode by either double clicking on it or pressing enter again.
5. Exit fullscreen mode.
6. Please put another drone of your choice into focus.
7. Now stop the drone you have selected by pressing the numpad key 0.
8. Tell the helmsman to sail to the stopped drone, by saying the bearing and distance of the drone out loud.
9. Can you see anything different on live feed X?

5.4.4 Tasks

This section presents the actual tasks of the user study. These tasks are completed after the completion of the training tasks.

1. When you feel ready please start the demo.
2. Figure out which drone discovers a person, and stop the UAV.
3. Make the helmsman sail to where you determine the person is.
4. When the boat reaches the destination, please pause the demo.

5.4.5 Process

The study has three conditions which will be tested, the first condition "No predictive elements" will act as the base line for the study to see if the predictive elements have an effect. The second condition will be "Intrusive elements" which will have the prediction layer enabled and present the information in a intrusive way which require immediate action. The third and last condition is "Non-intrusive elements" which also have the predictive layer active, but will present the information in a more subtle way not demanding immediate attention but still makes it self known.

When the participant arrives they are introduced to the project in broad terms and told how long they can expect the usability test to last. The hardware will be explained and the participant will undergo the training described in Section 5.4.3. Once the training is done the participant is left in the room with one study conductor who will assist if needed. The conductor will be minimally invasive and simply observe during the study. At certain points the conductor will pause the study to allow the use of SAGAT as described in section 2. The SAGAT question will be asked at the same time for each participant and will be asked before the person appears and after the participant has acknowledged that a person is present in a live feed.

Each of the tasks is gone through chronologically and when the last task is done, the study ends. The participant will then be asked to fill out a NASA-TLX questionnaire to asses workload.

NASA-TLX Questionnaire

As described in Section 2 the NASA-TLX questionnaire is a self assessment tool to measure workload. The questionnaire is defined below in Table 5.1 Each category is meant to allow the participant to reflect on their workload in each respective categories during the demo. The rating is a scale of low to high or poor to good. The participant will put a mark on the line in the ratings field which will correspond to an approximate value between 1 and 100. After the questionnaire is handed to the participant they will be asked to rank the categories according to which category contributed most to least to their overall workload. This ranking will be used as the weight in the workload calculations after the study.

Title	**Rating**	**Description**
Mental Demand	Low ______________________ High	How much mental and perceptual activity was required (e.g., thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?
Physical Demand	Low ______________________ High	How much physical activity was required(e.g.. pushing, pulling, turning, controlling, activating, etc.)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?
Temporal Demand	Low ______________________ High	How much time pressure did you feel due to the rate or pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic?
Performance	Poor ______________________ Good	How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals?
Effort	Low ______________________ High	How hard did you have to work (mentally and physically) to accomplish your level of performance?
Frustration	Low ______________________ High	How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

Table 5.1: NASA-TLX Questionnaire

SAGAT Questionnaire

In Chapter 1 it is advocated based on last semesters research that SAGAT is state of the art when it comes to measuring situational awareness during a user study. Therefore, SAGAT is applied to measure situational awareness in this master thesis. The idea behind SAGAT is to ask the participant questions during the user study. The intention in this user study is to pause the demo at two fixed timestamps and then have the participant answer the following questions:

1. Before pausing the demo, how many UAVs were moving and how many were stopped?
2. What is the distance from the rescue boat (you) to UAV #7?
3. Which UAV is closest to the rescue boat (you)?
4. Have you located a person? (If yes, ask below)
   (a) Which UAV found the person?
   (b) Where is the person?

The first timestamp will be sometime *before* a person is present on one of the video feeds and the second timestamp will be sometime *after* a person is present on one of the video feeds. The answers given by each participant is then analyzed and the overall situational awareness is compared between the predictive and the conventional interface. The results yielded by the SAGAT Questionnaire are presented in Chapter 6.

Post Study Questionnaire

After the survey is completed and the participant have evaluated them self using the NASA-TLX questionnaire, a short semi-structured interview is conducted using the questions below.

1. Which version of the interface did you prefer? (Conventional vs Predictive)
2. What did you like and what did you not like?
3. Did you prefer the peripheral or the centric info card?
4. What do you think of the size of the video feeds are they visible enough?
5. What did you think about the colors of the UAVs and the boat on the map, do they make sense?
6. Did you change your behavior/approach when you knew the interface would alert you to possible people in the water?
7. Did you like the way the video feeds were represented, or could you imagine a better way to do it?
8. Do you find the fullscreen mode useful?

9. Does the numpad navigation seem natural?
10. Was it clear for you which live feed belongs to which drone?

The interview is held to obtain some more personal preferences and thought about the design of the interface, which will aid determinate which elements or approach would give the best user experience while still being effective. The interview is also to get more information about the participants behavior and choices in the study, and if they did something which was not predicted before hand could have been an option, the interview serves as a good place to ask why and how they though of it.

6 | Results

This Chapter will present the result from the conducted user study explained in Section 5.4. The first part presents the findings in relation to human workload, the second part is about situational awareness, and the third and last part presents performance data. This Chapter will only present the results objectively while the discussion about the results will take play in Chapter 7.

Using the results from the study, it will be attempted to disprove the following two main null hypotheses to obtain information relevant for the research question proposed in Section 3.1.

- There is no significant difference in workload between the conventional interface and the predictive interfaces.
- There is no significant difference in operator performance between the conventional interface and the predictive interfaces.

6.1 Human Workload

In this section, the results yielded from the user study concerning human workload are presented. In Section 2 it is argued that the NASA-TLX method is the state of the art when it comes to measuring human workload. Therefore, it is applied to analyze the results of the user study in this project.

NASA-TLX is a questionnaire that each participant answers after completing the tasks. The participants completed the same tasks in this user study using three different interfaces, one conventional interface and two predictive interfaces. Each participant answered the NASA-TLX questionnaire twice, one time after using the conventional interface and then again after using the two predictive interfaces. Each participant either started with the conventional interface followed up by the two predictive interfaces or vice versa in a randomized order. This means that each participant answered the questionnaire directly after using the interface(s) in question.

The NASA-TLX questionnaire is composed of fixed generic questions, and they are seen in section 5.4. These are questions where the participants are asked to provide a rating between 1 - 10 on the mental, physical and temporal demand of the interface in question and his own performance, effort, and frustration. After they have provided each of the six categories with a rating, they are asked to rank the six categories between 0 and 5 accordingly, to which they deem the most important.

The questionnaire data is then processed by multiplying rating with rank. This is done participant-wise. Meaning, the rating is given by a participant to each of the six questions is multiplied by the question's rank provided by the participant.

In Figure 6.1 the results of the NASA-TLX questionnaire are seen. The results are an average of the ratings multiplied by the associated ranking. It is seen that there is an evident differentiation between the conventional interface and the predictive interfaces, especially when it comes to mental demand, performance, and effort. The conventional interface scores are more than twice as high as the predictive interfaces regarding mental demand and effort. These are clear indications that the predictive interfaces decrease the workload of the operator. Furthermore, performance scores about 50% higher in the predictive interfaces. This is very positive as they evaluate their performance is better when using the predictive interfaces.

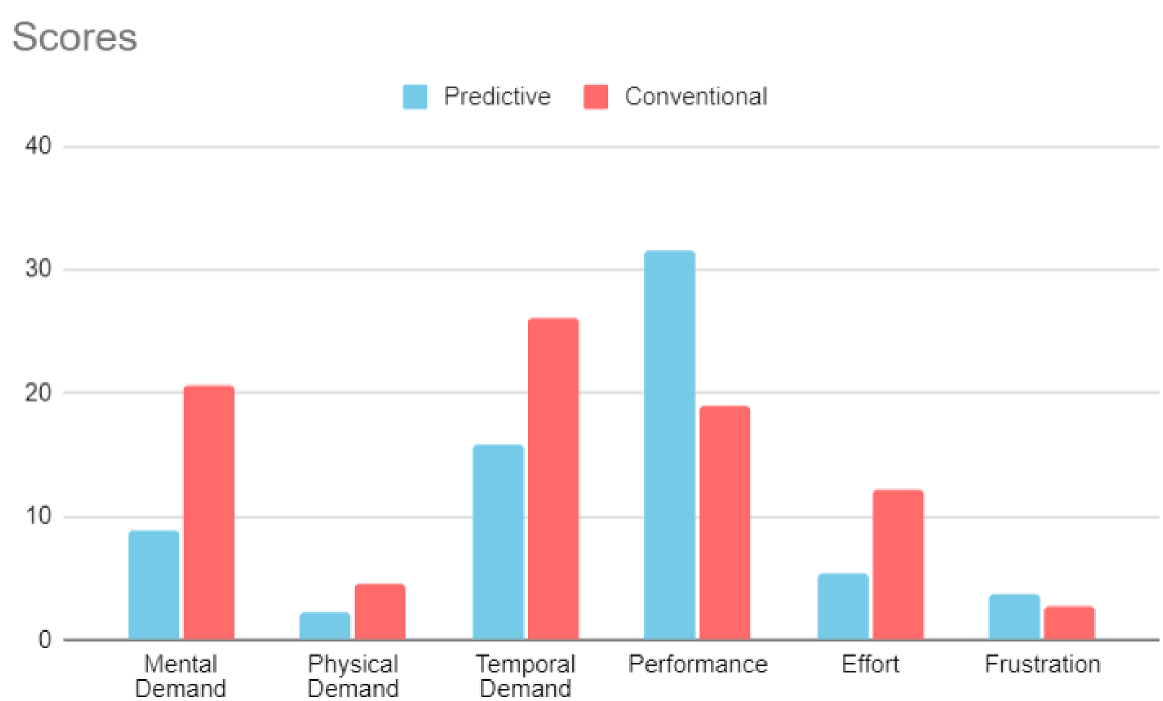

Figure 6.1: NASA-TLX: Results in averages

To sum it up, the NASA-TLX questionnaire results clearly indicate that the operator's human workload is decreased when using the predictive interfaces. And at the same time, the performance of the operators is improved significantly.

6.1.1 NASA-TLX t-tests

To further analyze the results of the NASA-TLX questionnaire, a two-sample t-test has been conducted on the questionnaire data of the conventional interface versus the questionnaire data of the predictive interfaces. The t-tests are run on each of the six questions: mental demand, physical demand, temporal demand, performance, effort, and frustration. The null hypothesis that is investigated in these t-tests is *There is no significant difference in workload between the conventional interface and the predictive interface*.

The results of the six two-sample t-tests are seen in Figure 6.2. Here it is seen that a p-value below 0.05 is obtained in three of the six areas. The three areas that did not obtain a p-value below 0.05, meaning that there is no significant difference between the interfaces, are physical demand, temporal demand, and frustration. These p-values are expected to be above 0.05 as there should be no significant change in the physical and temporal demand of the operator regardless of the interface in question.

Physical demand is not relevant as the participants sit down and interact with the system regardless of the interface type, i.e., the interfaces should have little to no physical demand. Also, as the user study is concerned with rescuing drowning people, the temporal demand is expected to be more or less the same regardless of the interface in question. Finally, it is positive that both interfaces scores low in frustration.

It is also seen that the p-values obtained for mental demand, performance, and effort are below 0.05. Therefore, there are significant differences between the conventional and predictive interfaces when looking at these three areas. This is very positive as these are the three areas considered key areas when looking at the interfaces developed for this thesis. As seen on Figure 6.2 the p-values are very low:

- Physical demand p-value: 0.0062
- Performance p-value: 0.036
- Effort p-value: 0.018

These p-values clearly state that the null-hypothesis *There is no significant difference in workload between the conventional interface and the predictive interfaces* can be rejected as these three areas are the relevant NASA-TLX criteria in this project.

Mental Demand

	Predictive	Conventional
Mean	8,8	20,6
Variance	30,84444444	156,0444444
Observations	10	10
Pearson Correla	0,5496574743	
Hypothesized M	0	
df	9	
t Stat	-3,547813795	
P(T<=t) one-tail	0,003118612971	
t Critical one-tail	1,833112923	
P(T<=t) two-tail	0,006237225943	
t Critical two-tail	2,262157158	

Physical Demand

	Predictive	Conventional
Mean	2,2	4,6
Variance	4,622222222	32,93333333
Observations	10	10
Pearson Correla	0,7636776618	
Hypothesized M	0	
df	9	
t Stat	-1,754532644	
P(T<=t) one-tail	0,05661651625	
t Critical one-tail	1,833112923	
P(T<=t) two-tail	0,1132330325	
t Critical two-tail	2,262157158	

Performance

	Predictive	Conventional
Mean	31,6	18,9
Variance	95,6	170,9888889
Observations	10	10
Pearson Correla	-0,01077621592	
Hypothesized M	0	
df	9	
t Stat	2,447088691	
P(T<=t) one-tail	0,01846633952	
t Critical one-tail	1,833112923	
P(T<=t) two-tail	0,03693267905	
t Critical two-tail	2,262157158	

Effort

	Predictive	Conventional
Mean	5,4	12,1
Variance	24,26666667	81,65555556
Observations	10	10
Pearson Correla	0,5731013652	
Hypothesized M	0	
df	9	
t Stat	-2,859490319	
P(T<=t) one-tail	0,009398003731	
t Critical one-tail	1,833112923	
P(T<=t) two-tail	0,01879600746	
t Critical two-tail	2,262157158	

Temporal Demand

	Predictive	Conventional
Mean	15,8	26,1
Variance	143,0666667	166,1
Observations	10	10
Pearson Correla	0,1724108609	
Hypothesized M	0	
df	9	
t Stat	-2,035672067	
P(T<=t) one-tail	0,03613447949	
t Critical one-tail	1,833112923	
P(T<=t) two-tail	0,07226895897	
t Critical two-tail	2,262157158	

Frustration

	Predictive	Conventional
Mean	3,7	2,8
Variance	36,23333333	29,73333333
Observations	10	10
Pearson Correla	0,6479231233	
Hypothesized M	0	
df	9	
t Stat	0,587929801	
P(T<=t) one-tail	0,2855175003	
t Critical one-tail	1,833112923	
P(T<=t) two-tail	0,5710350006	
t Critical two-tail	2,262157158	

Figure 6.2: NASA-TLX: t-test results

6.2 Situational Awareness

A part of the research question presented in Section 3.1 is concerned with situational awareness in SAR operations. As mentioned in Section 5.4 a SAGAT questionnaire was conducted twice during the study on each interface. The first one after approximately 25 seconds, and the last one after they had located the person and given correct instructions to the helmsman. This means that each participant answered the SAGAT six times in total during the study.

In the first round of SAGAT questions of each interface, all the participants answered all the questions correctly. However, the second round after the person has been located the participants had difficulty answering one of the questions correctly in all of the interfaces.

The answers to the SAGAT have been expressed as True/False tables, with six tables in total, one for each round of SAGAT questions. The tables for the first round of each interface will not be presented as they are the same with True for all questions. The tables for the second round of question show difficulties concerning questions 1, which is seen on Figure 6.3
As seen in Figure 6.3 question one had some mixed responses. The questions is *"How many drones were moving and how many were stopped, before the demo was paused"*. Some par-

Conventional	Question 1	Question 2	Question 3	Question 4	Question 4a	Question 4b
Participant 1	T	T	T	T	T	T
Participant 2	T	T	T	T	T	T
Participant 3	T	T	T	T	T	T
Participant 4	F	T	T	T	T	T
Participant 5	T	T	T	T	T	T
Participant 6	T	T	T	T	T	T
Participant 7	F	T	T	T	T	T
Participant 8	T	T	T	T	T	T
Participant 9	F	T	T	T	T	T
Participant 10	F	T	T	T	T	T

Peripheral	Question 1	Question 2	Question 3	Question 4	Question 4a	Question 4b
Participant 1	F	T	T	T	T	T
Participant 2	F	T	T	T	T	T
Participant 3	F	T	T	T	T	T
Participant 4	F	T	T	T	T	T
Participant 5	T	T	T	T	T	T
Participant 6	T	T	T	T	T	T
Participant 7	F	T	T	T	T	T
Participant 8	T	T	T	T	T	T
Participant 9	F	T	T	T	T	T
Participant 10	F	T	T	T	T	T

Intrusive	Question 1	Question 2	Question 3	Question 4	Question 4a	Question 4b
Participant 1	F	T	T	T	T	T
Participant 2	T	T	T	T	T	T
Participant 3	F	T	T	T	T	T
Participant 4	F	T	T	T	T	T
Participant 5	T	T	T	T	T	T
Participant 6	T	T	T	T	T	T
Participant 7	F	T	T	T	T	T
Participant 8	T	T	T	T	T	T
Participant 9	F	T	T	T	T	T
Participant 10	F	T	T	T	T	T

Figure 6.3: Second round of SAGAT, Top left: Conventional, Top right: Peripheral, Bottom: Intrusive

ticipants answered the question wrong in all interfaces, like participants seven, nine, and ten. Some participants answered it correctly in the convectional interface but then fail in one or both of the predictive interfaces. Lastly, there are the few who answer the question correctly in every interface. This shows that something has either been unclear about the question or confusing in the interface.

6.3 Operator Performance

This section presents the results extracted from the log data of the application. More specifically, the extracted data is the response time of the participants from the moment the person appeared on one of the live feeds to the moment of their acknowledgment of the person. This is done through exact timestamps logged by the application. In both the conventional and predictive interfaces, the starting point of the time measure is when the person appears on a live feed. However, the endpoint of the time measure differs between the interfaces. In the conventional interface, the acknowledgment is recorded as when the user paused the UAV which had found the person. In the predictive interfaces, the acknowledgment is set to be when the participant clicked the confirmation button in the popup window. Each of these events was logged and extracted from the SQL database.

To disprove the second null hypothesis: *There is no significant difference in workload between the conventional interface and the predictive interface.* a One-way repeated measures ANOVA test was conducted on the data, the result of the ANOVA test can be seen in Figure 6.4

SUMMARY	Count	Sum	Average	Variance
Participant 1	3	32,189	10,72966667	25,29761433
Participant 2	3	13,649	4,549666667	1,114156333
Participant 3	3	31,949	10,64966667	11,35528233
Participant 4	3	16,89	5,63	1,912323
Participant 5	3	10,462	3,487333333	0,6682763333
Participant 6	3	20,287	6,762333333	1,171654333
Participant 7	3	14,994	4,998	0,701461
Participant 8	3	15,864	5,288	6,625429
Participant 9	3	17,698	5,899333333	10,16069733
Participant 10	3	12,961	4,320333333	0,7541923333
Conventional	10	75,345	7,5345	15,27077139
Predictive Intrusive	10	47,386	4,7386	2,529373156
Predictive Peripheral	10	64,212	6,4212	10,05423773

ANOVA

Source of Variation	SS	df	MS	F	P-value	F crit
Rows	170,7927227	9	18,97696919	4,275337646	0,00424238666	2,456281149
Columns	39,62545487	2	19,81272743	4,463626337	0,0266516521	3,554557146
Error	79,8967178	18	4,438706544			
Total	290,3148954	29				

Figure 6.4: One-way repeated measures ANOVA

The p-value is less than 0.05, which disproves the null hypothesis and indicates a difference in performance (response time) between the interfaces. To figure out where the difference lies, a paired samples t-test was performed on each of the individual interfaces against one another. The third null hypothesis states: *There is no significant difference in operator performance between the conventional interface and the predictive intrusive interface.* The t-test result can be seen on Figure 6.5

	Conventional	Predictive Intrusive
Mean	7,5345	4,7386
Variance	15,27077139	2,529373156
Observations	10	10
Pearson Correlation	0,3082690527	
Hypothesized Mean Differe	0	
df	9	
t Stat	2,365638317	
P(T<=t) one-tail	0,02110479457	
t Critical one-tail	1,833112923	
P(T<=t) two-tail	0,04220958915	
t Critical two-tail	2,262157158	

Figure 6.5: Paired samples t-test, conventional vs intrusive

As the p-value is less than 0.05, the null hypothesis has been denied, and it is proven that there is a significant statistical difference between the conventional and predictive intrusive interface.

The fourth null hypothesis is as follows: *There is no significant difference in operator performance between the conventional interface and the predictive peripheral interface.* The results of the t-test is seen in Figure 6.6

	Conventional	Predictive Peripheral
Mean	7,5345	6,4212
Variance	15,27077139	10,05423773
Observations	10	10
Pearson Correlation	0,8235831446	
Hypothesized Mean Differ	0	
df	9	
t Stat	1,587994574	
P(T<=t) one-tail	0,0733750300	
t Critical one-tail	1,833112923	
P(T<=t) two-tail	0,14675006	
t Critical two-tail	2,262157158	

Figure 6.6: Paired samples t-test, conventional vs peripheral

This time the p-value is above 0.05, which means it is not possible to reject the null hypothesis. Therefore there is no significant difference between the conventional interface and the predictive peripheral interface.

Lastly, the two predictive interfaces were put up against each other in a t-test. The results are seen in figure 6.7

	Predictive Intrusiv	Predictive Peripheral
Mean	4,7386	6,4212
Variance	2,529373156	10,05423773
Observations	10	10
Pearson Correlation	0,4793664178	
Hypothesized Mean Dif	0	
df	9	
t Stat	-1,91145197	
P(T<=t) one-tail	0,04412813236	
t Critical one-tail	1,833112923	
P(T<=t) two-tail	0,08825626472	
t Critical two-tail	2,262157158	

Figure 6.7: Paired samples t-test, intrusive vs peripheral

Once again, the p-value is above 0.05, and there is, therefore, no significant difference between the predictive intrusive and the predictive peripheral interface.

All factors potentially influencing the results and the results themselves will be discussed in Chapter 7.

6.4 Post Study Interview

This section presents the findings from the post study interview questions. Each participant encountered a fixed set of post study questions in a semi structured interview. Below the general consensus for each of the questions is presented.

Which version of the interface did you prefer? (Conventional vs Predictive) It was clear which of the interfaces was the favorite among the participants. 9 out of 10 choose the predictive intrusive interface as their preferred interface. The general reasoning behind their decision was that it grabbed their attention more and the feed was bigger. One participant explained that the peripheral version made him unsure how long the pop up notification had been there.

What did you like and what did you not like? The comment that was given the most was that the participants wished the live feeds at the bottom of the screen were bigger. One participant also noted that it was weird that the live feed of a stopped UAV continued playing. Another comment was to highlight the UAV marker on the map of the UAV which live feed is currently selected.

Did you prefer the peripheral or the centric info card? As revealed by question one, almost all the participants preferred the centric info card. It quite simply grabbed their attention more and made them act quicker. One participant did however say that the experience of the centric popup depends on the amount of false positives from the predictive algorithm.

What do you think of the size of the video feeds are they visible enough? Once again the consensus was that the participants wanted bigger live feeds. One participant commented that his entire focus was on the live feeds, and therefore he wished the sizes of the map and live feed bar were swapped.

What did you think about the colors of the UAVs and the boat on the map, do they make sense? Most of the participants said the colors made sense, however almost half of the participants said they did not actually notice the color of the UAV markers changing. One participants made the point that red seems like a failure color maybe a yellow color should be used instead for a stopped UAV.

Did you change your behavior/approach when you knew the interface would alert you to possible people in the water? Many participants said they took a more relaxed approach when they knew the system would notify them if it found a person. During the conventional interface, some of the participants said they switched live feeds more often, or looked more intensely at the live feeds in general.

Did you like the way the video feeds were represented, or could you imagine a better way to do it? Nine out of ten participants answered that they would have preferred if the video feeds were bigger. The reason they gave was that the devoted almost all their attention to on the video feeds and almost no attention on the map which takes up 2/3 of the screen.

Do you find the fullscreen mode useful? In general the participants did not use the fullscreen mode. They used the enlarged view which they deemed big enough. Only two out of eight participants answered that they found the fullscreen mode useful.

Does the numpad navigation seem natural? Eight out of ten participants answered that they preferred the numpad interface over the touch interface. They majority answered that the numpad interface felt intuitive to use.

Was it clear for you which live feed belongs to which drone? In general it made sense for the participants which UAV that belonged to which video feed as both the video feeds and the UAV are numbered. Only one out of the ten participants answered that is was not clear to him.

Based on the interview findings listed above, two main tendencies can be extracted. First, the participants preferred the centric pop-up as it gave them a better understanding on what is on the video feed. Lastly, the participants in general thought that the video feeds should be bigger as it is a bit hard to see details on the small video feeds.

The main findings in terms of results show that the two main null-hypothesis presented in the beginning of the section both can be disproved. This is shown by the different t-tests and ANOVA carried out both on the NASA-TLX questionnaire data, as well as the operator performance data. Furthermore, it was found that it is the predictive intrusive yielding the results which disprove the hypothesis. This is seen when comparing it with the conventional interface.

6.5 Expert Reviews

The user study was conducted on naive users, and therefore not the intended users of the interface. Even with training and understanding of the situation there might still be information or feedback that they would not be able to provide. Therefore reviews with three experts who develop and research UAVs were conducted to get their view on our interface, and promote a co-design process in order to leverage their knowledge about current design, features and limitations.

Expert reviews is a quick and cost efficient way of getting feedback with a limited amount of expert participants (Korhonen et al., 2009). During an expert review it is common to have the experts consider usability heuristics as presented in (Nielsen, 1994, Chapter 5). The heuristics discussed during the expert review included simple and natural dialogue, minimize memory load, feedback etc.

The expert reviews were conducted through virtual meetings and lasted about 45 minutes. The meetings started out with the group doing a ten minute presentation to set the scene, showed a demo and presented the results from the user study. After the presentation initial question from the experts were answered and the co designing began. The experts already had many good points themselves, and after their initial thoughts and suggestions they were asked about specific things such as component size, colors, information relevancy and usability.

The first expert review was conducted with Jacobo Domingo Gil, who is the COO of Robotto, and has been included in the project since the pre-specialization (Jensen et al., 2020). Here he was also included in expert reviews. After Jacobo's review, the next reviews were with Calvin Rubens and Sean Braley who are doing research on UAVs and UAV swarms and authors of papers such as (Braley et al., 2018). Below follows a list of takeaways from each of the expert reviews.

6.5.1 Notes from Jacobo Domingo Gil

- **Draw a line between the IC and the person to aid in navigating to the person.** Currently there is no graphical element showing the path from the boat to the person once the person has been located. Jacobo suggests a line should be drawn to clearly indicate the direction the boat must take in addition to the existing information.

- **Make the UAV that have located a person more noticeable by being different from the others.**
  The current implementation changes the color of a UAV to red when it has found a potential person. However, the difference could be made larger, either by increasing the size of the drone icon, or even by making it flash to grab extra attention.

- **Make the connected information larger when a UAV is clicked on.**
  Jacobo pointed out that it might be a good idea to increase the size of the text next to the UAVs if a UAV icon was clicked. This will help with text size and overlapping issues.

- **Get information from the OM interface about where the person fell into the water, or request predictions made in the OM interface.**
  Jacobo is aware that there is another interface being developed with focus on the operations manager, and the suggests there should be a way of accessing the information in the operations manager's system.

- **The interface could be used by another person in the boat, if needed.**
  A comment was made that interface itself could be handed to another person in the boat if the IC was busy doing something else. Even without training it is still possible to observe the live feeds

- **Bigger pop-ups are better**
  Jacobo stated that the bigger the popups the better. He wants the user to notice the notification straight away to avoid any delays.

- **Pop-ups do not make it appear that the interface know more then the IC, but is just aiding the IC.**
  It is important to keep the IC in control. Jacobo mentions that its good the interface prompts the user to take action instead of doing it automatically. This creates the sense that the user knows more than the system and is still in control.

- **Search area could be divided into quadrants, the search team could handle one quadrant and the UAVs search the rest.**
  This ties into making the UAV paths visible. If the UAVs are searching certain quadrants, the boat could search a completely different quadrant to optimize the search and potentially find the person faster.

- **Field of view cones on the UAVs should by dynamic based on what the UAV can see and the visibility in the area.**

A feature suggested was to make the size of the vision cones dynamic. They should change size depending on the weather visibility, altitude and obstructions etc.

6.5.2 Notes from Calvin Rubens

- **On the fly verification of the flight path.**
  Calvin suggested the interface should make it clear where the UAVs are going to go. If the IC has no idea of the UAVs' flight path then he might feel out of control. Knowing the path of the UAVs can be important to plan the boats future movement.

- **Highlighting of hot spots which would affect path planning.**
  Highlighting of hotspots such as where the person was last seen, or potential persons in the water could be shown on the interface. These hotspots could have influence on the planned path for the UAVs.

- **Dynamic influence of the flight path.**
  Calvin suggested that the IC could have an influence on the flight path of the UAVs. This does make sense, however the group imagines it is a job for the OM to direct the UAVs to avoid increasing the IC's mental workload.

- **Show the area already covered by the UAVs.**
  This suggestion could prove beneficial for the IC when planning where to look next. Heatmaps displaying the area already searched allows for a better understanding of which parts have been left unsearched.

- **Dynamic sizing of the live feed bar.**
  Resizing the live feed bar to suit the need of the user could improve the user experience. The IC might want a bigger map at times, and bigger live feeds at other times.

- **Intrusive pop is great for grabbing attention.**
  Calvin says the intrusive popup is great as it grabs the attention of the IC straight away. He did however stress that the popup could get annoying with too many false positives.

6.5.3 Notes from Sean Braley

- **Make intended actions clear**
  Sean agreed with Calvin that the path of the UAVs should be visible, and all other intended actions the UAVs might take should be visible to the IC. This is to prevent any feeling of uncertainty the IC might have regarding the UAVs and their actions.

- **Component sizing: humans can only look at one component at a time.**
  Sean made an interesting note regarding peripheral vision. Even if the live feeds were made bigger, the IC will still only be able to focus on one feed at a time. making the feeds bigger would also spread them further apart, increasing the chance of missing details on the feeds furthest away from the feed the IC is focusing on.

- **Toggleable information: UAV heatmaps and paths**
  Adding too much information to the interface can overload the IC, and therefore it is suggested that the information should be toggleable to avoid clutter.

- **Map orientation relative to the boat.**
  When looking at a map, many people are used to the map being oriented in the user's current heading. The map is north aligned in the current interface, and there is no option to change this. Adding this feature would add the increased personalization.

7 | Discussion

This chapter discusses the conducted user study and the findings thereof. Furthermore, it discusses the different contributions of this thesis, and finally, the possible limitations are discussed. Some parts of the discussion will uncover more work that could be done to the project, which will be presented in chapter 8.

7.1 User Study

The user study aims to investigate if there are any differences in human workload, situational awareness, and performance depending on the type of interface. This section discusses the factors that could have an influence on the results of each of these three areas.

The NASA-TLX self-assessment technique was used to measure the workload of the participants. The results showed significant differences in three of the categories, which is seen in Section 6.1. During the study, some participants had to ask about a few of the questions to make sure they understood them correctly. The answers to the questions had to be defined as a scale from one to ten, but some participants were unsure if 1 or 10 meant good or bad. While the raw data shows no outliers, which would indicate a misunderstanding of the question, it is still possible that some participants could have misunderstood a question. The questions themselves were given as defined by (Hart and Staveland, 1988) and were not altered from the original description. To counteract this in future studies, the moderator could go through the questions with the participants. However, this might have adverse effects as the participant might feel inclined to provide higher scores if he feels watched by the moderator. Overall the NASA-TLX technique was a success in the study and provided valuable data for the project.

To investigate the performance, the timestamps from the logs were used to measure the participants' reaction time. Using a three-way ANOVA, the results showed differences between the interfaces, and there were no significant problems collecting this data. Then a Bonferroni style post hoc test was conducted to measure the individual differences between each of the interfaces. The group is aware that this might introduce a type I error, but it still provides valuable insights.

To measure situational awareness, the SAGAT questionnaire was used. The results gathered from the questionnaire did not show any difference between the conventional and the predictive interfaces. Many of the participants answered wrong on the first question regarding the amount of moving UAVs or simply did not know. This could be because of the participants not being told which questions they were going to receive. To combat this and possibly get more meaningful data, the participants should be informed beforehand what to look out for and what questions are coming up.

As mentioned, the study did not reveal any significant differences in situational awareness between the interfaces. However, it is possible to discuss the interfaces in relation to the nine demons from (Agrawal et al., 2020) as presented in Chapter 2. The nine demons are design pitfalls that interface designers should consider in order to provide optimal conditions for situational awareness.

One of the demons, "attention tunneling" describes how a user can become overly focused on one source of information and ignore the rest of the information available. During the study, this was seen when the participants were unable to answer correctly to a question regarding the UAV statuses because they had been focusing on the live feeds only. The participants might also have had a "faulty mental model" which is also one of the nine demons, as they were in a completely new environment and might have had a different understanding of which parts of the system were the most important ones. The users also reported a lower mental demand when using a predictive interface in comparison to the conventional interface. This could potentially lead to the out-of-the-loop demon if the operator becomes too accustomed to the automation of the system and therefore neglects to look at the live feeds.

7.2 Contributions

The contributions of this report is based upon the conducted user study and expert reviews. The intention behind the user study is to see if having predictive elements in an interface for the incident commanders in maritime SAR operations have a measurable effect on the operator. For this both the human workload and situational awareness of the operator as well as his performance while using the interface are measured.

The contributions of this project is classified using the four types of HCI contributions explained in (Jacob Wobbrock, 2016). The first contribution of the project is the developed research platform, which is classified as an *Artifact* contribution. The results and gathered data of the user study is the *Empirical* contribution, and lastly there is the implications for design, which is seen in Section 7.2.2, and is the *Theoretical* contribution of this project.

The results in Section 6.1 clearly indicate that having the predictive element available had a noticeable difference on the operator's mental demand. When looking at the NASA-TLX results in Figure 6.1 mental demand shows a decrease by more then 50% and temporal demand is close to a 50% reduction as well. These reductions are also shown in the post study interviews where the participants felt like they had to switch camera more often and look more intensively at the feeds in order to spot the person in the conventional interface, but in the predictive interface they felt they had more time to analyze each feed before going on to the next one. This suggests that the operator took some sort of reassurance in the predictive elements. Feeling that if a person was located on another feed than the one the operator was currently looking at, they would be notified by the interface, giving the feeling of having less workload and more time for each task.

This decrease in mental and temporal demand also comes with a 70% increase in performance as seen on Figure 6.1. This indicates that the participants deemed they had better and was more reassured that the task was performed to the best of their ability. It can be understood that not only does the predictive elements give the operator more time and less pressure to perform his tasks, but it also makes him more confident in his choices and actions.

One last significant change is the reduction in the operators effort using the interfaces. Here there are a 40% decrease of effort in the predictive interface, which shows that the while the operators felt they had more time, higher performance, they also felt they had to put in less effort in to obtain these results.
The results from the NASA-TLX can also be seen in response times that was measured in order to see if there was an actually measurable effect in the performance of the participants. Section 6.3 goes over the logged time measurements of the operators actions in the interfaces, and in Figure 6.4 it is seen that there is a significant change in the interfaces, and based on the mean value of the interfaces it is seen that the predictive interfaces have a lower mean value then the conventional. To see which interface was the one with the highest impact, the Bonferroni approach was taken.

In Figures 6.5 and 6.6 the two predictive interfaces have been set up against the conventional interface. From these it can be seen that the significant difference lies with the Intrusive interface, this results also matches up with the feedback given in the post study interview where nine out of ten would chose the intrusive interface as their preferred one.

7.2.1 Usability Concerns

This section presents some general usability concerns which were discovered during the study and the expert reviews. Some of these concerns are well known, and many of them can have an influence on human workload. Human workload is a key aspect when designing mission critical interfaces like human swarm interfaces. When lives are at stake it is very important that the human workload of the operator is kept to a minimum, as too high workload can result in mistakes as attention has to be devoted to the task at hand. Below is a list of the gathered usability concerns.

- *Customizable notifications:* Notifications can both be a help, but also an annoyance. Therefore it is recommended to allow the user to specify how the notifications are displayed. Being able to change the intrusiveness of the popup will allow the user to decide how important and time critical the notifications of the system are.

- *Togglable Information:* is a key aspect of personalization. When designing mission critical interfaces it is important that the user is not overloaded with information. However, at times some information might be required. E.g. in maritime SAR human swarm interfaces it could improve the operators situational awareness if he was able to see where the UAVs have already searched and the path they are taking. However, to display this information all at once might clutter the interface and overload the operator with information i.e. decrease situational awareness. Therefore togglable information is a powerful tool, as information like UAV paths can be visible only when the operator deems it necessary.

- *No scroll:* Having to scroll in an interface might lead the user to over look information, resulting in a loss of situational awareness. All information should be viewable in the screen and not hidden off-screen.

- *Touchscreen:* The touchscreen is the modern way of interacting with a tablet interface, and support for touch navigation is important as it would be expected by the user to have that option when holding a tablet. Touch navigation may also be quicker in certain situations compared to other input methods. As it says in Section 4.3, touchscreens are widely used in mission critical operations which the Garmin G1000 is an example of.

- *Larger info on selected UAV:* When showing information next to each UAV on the map, having the ability to focus or enlarge one drone's information is useful as it will separate it from the rest of the UAVs, making it clear that this is the information that should be acted on at this time.

- *Coordinates not easy to grasp:* When showing the location of the UAVs it is not very meaningful to display the coordinates as the main source of their location. People do not have the mental capacity to grasp a number with 8 or more decimals and therefore the position of the UAVs should be visualized on a map instead. Coordinates are important for internal algorithms however.

- *Component sizing:* The size of the components has to be considered when creating the interface. A person has limited peripheral vision and no matter how big the components are, the user can still only look at one thing at once. Bigger components allow for displaying greater detail while smaller components might allow the user to use peripheral vision to notice changes in other components.

- *Only relevant information:* The interface should only show the information that is relevant for the user, in a way that makes sense. Overloading the user with information will only increase their workload as they will have to analyze more information in order to find the information that he needs. It is also necessary to consider how the information is represented to make it as easy for the user as possible. One example could be showing a UAV's battery as the amount of minutes it can fly instead of the battery percentage. The percentage does not give the user any useful information, but knowing how long a UAV can fly is useful.

- *Make the intended actions as clear as possible:* The interface should be aware of the actions that the user might take based on the information in the interface, and should support these actions by providing the user with a clear projections of the information or guidance they need to perform the action. In the case of guiding the crew to the located person, the distance and direction to the person should be clearly visibly and other information should be less noticeable. Here it would also be useful to plot a path from the crew to the person so the user can see how they are going to reach the person.

7.2.2 Implications for Design

This section presents the implications for future design of Human Swarm Interfaces in the maritime SAR context. The implications are presented in a list of concepts with underlying prescriptions in accordance with the structure for design implications presented in (Sas et al., 2014). This means to explain the implications based on the findings from this thesis and back them up by claims from related HCI research.

- **Personalization**
  The concept of personalization is an important aspect of interface designing as it can help

increase the user's situational awareness as described by (Hocraffer and Nam, 2017). No person is the same, and having the ability to make adjustments to the interface on the fly will allow the user to take control of the components of the interface.

- *Resizable live feeds:* The size of the live feeds turned out to be a subjective opinion and can change depending on the task at hand. Making the live feeds of the interface resizable will allow the user to adjust them to the specific needs of the current situation. In the study, the majority of the participants mentioned that they would have liked different sizing of the live feeds. This goes hand in hand with the personalizing aspect that is advocated for in (Hocraffer and Nam, 2017).
- *Map orientation:* It is important to consider how the map should be orientated, should it be true north at the top always or should it re-orientate related to the user, so that the top is straight ahead. Allowing the option to switch between orientations might help the user get a better overview of the situation. This implication was yielded from the expert reviews where a UAV expert suggested that it might be the best thing to do. It provides the user with a clearer idea of which UAVs in the area corresponds to the UAVs in the interface. This also plays well into the interface personalizing concept in (Hocraffer and Nam, 2017) if this mode is toggleable.

- **Interface Navigation**

  The concept of Interface Navigation in this list concerns a swarm interface on a tablet in a harsh environment. The subcategories describe practices that enhance the navigation experience in this specific setting, but can not be used as broad guidelines for desktop interfaces.

  - *Physical inputs:* When designing an interface, it is essential to consider the environment the interface is supposed to be used in. If you are designing for harsh environments like onside maritime SAR operations, you should consider physical input buttons/dials. As there might be things limiting the touch interface, such as water on the screen. Therefore, it is advised to support both a touch interface and a physical input interface. In the study, many participants preferred the physical input buttons to the touchscreen for navigation. Physical input buttons are also mentioned as one of twelve usability heuristics in (Inostroza et al., 2013).

- **Predictions**

  Having the human swarm interface might help the user with his tasks, but the mental workload of managing and understanding all of the information is still high. Having predictive or intelligent elements in the interface can help decrease human workload. The elements will analyze and interpret the data for them and present it in a more user-friendly way that does not require a high mental workload for the user to understand. The different types of predictions presented here are not the only way to go. However, they are state-of-the-art artificial intelligence technologies.

- *Object detection:* When searching for objects of interest on live feeds, it is helpful to alert the user of any detected objects that fit the given criteria. Object detection can pose a challenge that is false positives. These can have a negative effect on the experience, but with a good detection model, it is a crucial functionality of the system. The predictive interfaces in the study yielded the best results here object detection is the main predictive element. This aligns with the fact that object detection is the state of the art when it comes to SAR human swarm interfaces, as mentioned in (Bejiga et al., 2016).
- *Predicted path of the missing person:* Predicting the direction of a moving object could be a help to the user if the algorithm is reliable. During the expert interviews, this fact was discussed that seeing how the person might move could help the IC better understand how to reach the person or see if they are heading towards any dangerous areas. This prescription can prove challenging to implement if all the parameters of all influencing factors are not known to the system. Predicting the path of a moving target is an area that is still being heavily researched, but it is understood that it can help in SAR operations as described in (Macwan et al., 2011)
- *Predicted location of missing person:* When a UAV locates a person, it is advised to predict the location of the person and display it to the user. This gives the user a quick understanding of where the person is and how far away they are from the person. During the SAGAT in the user study, the participants were asked to point and explain where the located person is. In the conventional interface there was no marker, and the participants were vaguer in their explanation compared to the predictive interfaces that had a marker where the person was. When displaying the predicted location, the human workload is decreased, as the operator does not have to keep in mind which UAV on the map has located the person. It is explained in the previously mentioned paper (Macwan et al., 2011) that predicting a location of a person can be of assistance in SAR operations.

- **Situational Awareness**
  Situational awareness is vital in critical systems where lives are on the line. Better awareness will increase performance, and when creating an interface for swarms, it is important to design for this concept. Below are some prescriptions that can be applied to interfaces

dealing with any UAV monitoring; however, some prescriptions are specific to search and rescue.

- *Initial position of person:* having the initial location of where the missing person is believed to have fallen into the water, as discussed in the expert interviews, this can help the user in deciding what should be the first course of action or in which direction that they should focus there and the team's attention towards and let the swarm handle the rest. The initial location of a person is crucial for the prediction of future movement. This would be the *Target Information* mentioned in (Macwan et al., 2011).
- *Heatmaps showing already searched areas:* Knowing where the UAVs have searched already will give the user knowledge, which can help determine where to look next. If the object of interest can change position, it would be beneficial to let the heatmaps fade over time to show that the object might have moved to an already searched area. Having a heatmap or a way to see where the UAVs already have searched was something that both a few participants in the user study mentioned would have been nice to have, and it was also brought up in the expert reviews, where there was a general consent that it would help to know where the UAVs had search and an indication of how long or how well the area was searched. A similar approach is presented in (Morse et al., 2010) where the term *see-ability map* is introduced. It uses a heatmap to show the area which has been seen by a UAV and has different gradient depending on how thoroughly the area was searched.
- *See planned UAV path:* It was discussed in the expert interviews that UAVs are still a new technology, and it is not always clear what they are doing by just looking at them. This can lead to a sense of uncertainty or loss of control which is never good in life or death situations. Showing the location of the UAVs and their planned path and actions can give the user a higher certainty and a sense of control over the UAVs even when they do not. This goes hand in hand with the concepts mentioned in (Agrawal et al., 2020). More specifically, the demon "enigmatic autonomy" which states that it must be apparent to the user which actions the robots can take.

These implications for design wraps up the research contributions of this project. The main contribution of this project is the results of the conducted user study. Here it was discovered that a predictive interface has a significant effect on operator performance and operator workload. Furthermore, the implications for design contribute as suggestions on how to design a good human swarm interface.

7.3 Possible Limitations

This section discusses the limitations of this project. These are limitations both in terms of the interface and the user study. In this kind of project, which aims to address a wicked problem by developing a prototype system, limitations are inevitable. In terms of the human swarm interface that has been developed for this master thesis project, it has some clear limitations as opposed to a real-world system:

- **Simulated UAVs**
  The study raises a concerns for ecological validity as all UAVs in the system are simulated, meaning that their positioning and movement on the map are fictive as well. The study could have been conducted utilizing real UAVs however, it would require much more development and time which does not fit the schedule of a master thesis project. The results however, are still valid as it is concerning human workload, situational awareness and operator performance. These factors are not directly influenced by the fact that the UAVs are simulated. As they are mainly a result of the operator's monitoring of the video feeds. Future studies could involve real UAVs.

- **Simulated video feeds**
  The study raises a concern for ecological validity as the video feeds are pre-recorded videos that are simulating live UAV footage. The missing person the operator is looking for in the user study is edited into the videos. If the system supported real UAVs the study could have been carried out using live video feeds instead. However, it is reasonable to assume that the results of the user study are still valid as the pre-recorded videos used are maritime UAV footage, just like they would be in a real world scenario, the only difference being that they would be live. For future studies the system could support live UAV footage.

- **User study not conducted in real setting**
  The study also raises a concern for ecological validity as it is not carried out in its natural setting. This could have been improved by arranging with Nordjyllands Beredskab to conduct the study on open water in a rescue boat. However, due to COVID-19 among other things this was not possible. It is reasonable to presume that this would have had an effect on the results as the real environment compose many more distraction then a lab study. However, despite the fact it is not ideal to perform lab studies on this system, the findings are still useful as it shows that the predictive interfaces yield significantly better results on human workload and operator performance. For future user studies it should be carried out in a more natural setting.

- **Participants not real Incident Commanders**
  The study raises a concern for ecological validity as the participants are not Incident Commanders like the intended users of a final system. Obviously it would have been ideal to conduct the user study with real incident commanders as participants which was not possible due to COVID-19. However, the results are still deemed valid as they show a significant difference on human workload and operator performance. For future user studies educated incident commanders should be involved as participants.

- **User study conducted on 10 participants**
  The study results raises a concern for validity as only 10 participants are involved. In the ideal world more participants would have been involved in order to provide more backing for the results. The participants were fellow students at the Department of Computer Science at Aalborg University. Here COVID-19 also played a role as it was only in late April when the university opened, but only on 30% capacity, that the group was allowed to recruit other students as participants. This also resulted in the group only being able to requite ten participants for the user study. However, when looking at similar research papers, there are also cases where around ten participants are involved. Therefore it is deemed not to have a significant impact on the results of the user study. For future user studies more participants should be involved in order to be more confident on the findings.

8 | Future Works

This Chapter presents future work and invites others to take up certain topics for future studies.

8.1 Situational Awareness

This project set out to study which effect a conventional interface versus a predictive interface has on situational awareness, workload, and performance. As discussed in Chapter 7, the study was not able to prove and significant difference in one of the three categories, namely situational awareness. To get more knowledge of situational awareness in the two different interfaces, it will be necessary to conduct further studies. The lesson taken away from experience gained in the conducted study is that the SAGAT questionnaire needs more questions that require the participant to focus on specific aspects of the interface. The participant also needs to be told what questions are coming up so they can prepare and keep a close eye on the relevant UI elements.

Some of the challenges during the study were that the participants were not domain experts but only had general knowledge about UAVs and the quick introduction they were given about the topic. The research shows a learning effect on the system, but this was expected as it was conducted with naive users. It will not be possible to avoid the learning effect unless the features are tested by domain experts using systems that are already used daily. For future work in this, a more prolonged study could be conducted on interfaces already deployed and in use, and perhaps even be tested in more life-like scenarios.

Doing SAGAT many times during a run-through of an interface over several interfaces also takes a lot of time, and paired with the NASA-TLX questions, and the performance measurements, the overall length of the study increased. To go in-depth with situational awareness, it would be beneficial to conduct a survey with situational awareness as the sole dependent variable. To combat fatigue in a lengthier study, it would be plausible to use a between-group design. However, this would require more participants. During this study, the participant was allowed to look at the interface while answering the SAGAT questionnaire. However, it might give interesting results to hide the interface and force the participant to rely on his memory.

8.2 Features

During the research, design of the user study, and the feedback from the participants, a collection of features have come up which might improve the interface and give a better overview of the situation for the operator.

8.2.1 Heatmap

To give the operator a better understanding of what has been searched and how recently, a heatmap could be drawn over the map marking where the UAV's have flown and what would have been inside their field of vision. One of the participants also brought this up in the post-study interview, that they thought it would be nice to see which areas had been searched.

Drawing a heatmap does require some thought, as the system is intended for maritime environments. This means that factors such as wind and stream can actually change an already searched area, and the person in the water is also under constant movement. So an area already explored can suddenly be changed entirely, and the missing person can appear in an already searched area. To accommodate this, a heatmap that moved with the current could be implemented, or the heatmap could slowly disappear based on how long ago the UAV searched the area.

8.2.2 Personalization

In (Hocraffer and Nam, 2017) it is explained how adapting the interface to the user can increase performance and reduce errors. This can be done by a completely custom interface for the specific user or let the user choose what they want to see and how the interface should act during certain situations. In this study, nine out of ten participants choose the intrusive interface as the preferred one, as it made them feel secure that they had not missed the person. As one participant expressed with the peripheral interface, he might have reacted right away on it. Still, he was uncertain when the pop-up appeared and afraid that he might have overlooked it for a time, leaving the person in the water. So for him, this gave him doubt in his actions. However, the study was not conducted on expert users, and it is a possibility that a user who is used to the stressful situation might find the peripheral better.

So, therefore, it could be worth considering not constricting the interface to one or another approach but letting the users of the interface personalize it to fit them best.

8.3 Future User Studies

As a future work of this project, it would be highly relevant to conduct additional user studies. The user study that has been conducted tries to measure performance, situational awareness, and human workload all in one study. A suggestion for future studies could be to split it up into three studies, one measuring each of the three areas.

In terms of measuring performance and human workload, the user study that has been conducted and documented in Section 5.4 and 6 yielded good results even though it was conducted on näive users. However, only measuring operator performance or human workload involving more than the ten participants who have a better knowledge of the topic could produce different results.

However, when talking about situational awareness, the user study was not able to show any effect. One of the reasons here is the wrong execution of the SAGAT method. A study that only focuses on situational awareness and has a well-designed SAGAT questionnaire has a good chance of yielding better situational awareness results. This is a clear-cut future work as this study failed to show an effect on situational awareness. In the ideal world with no COVID-19 restrictions, further user studies should involve more participants than ten as it is easier to get better results if more participants are involved. When more participants are involved, the effect of outliers is minimized, which is crucial.

It could also be possible to refine and develop a more functional prototype, which could be used for field testing in collaboration with search and rescue experts. This would help verify the processes and identify any issues that might arise using the new capabilities offered by the system.

9 | Conclusion

This project studies the differences in situational awareness, human workload, and performance between a conventional human swarm interface and two predictive human swarm interfaces for maritime SAR operations. This was done through a prototype system capable of monitoring a swarm of UAVs using an interface specifically built for the incident commander. While the system made in (Jensen et al., 2020) which this system is built upon, can link up to real UAVs using embedded software created with the system, this study was conducted using simulated live video feeds from UAVs and scripted UAV moment. But in the future, a study using actual UAVs could be done.

Section 5.1 explains how functional and non-functional requirements were gathered using the work from (Jensen et al., 2020), which were then used to develop the artifact contribution namely the research platform suitable for the user study. Section 5.4 describes the setup of the conducted user study and procedure that followed. The study results are presented in Chapter 6, and the outcome and any factors influencing the results are discussed in Chapter 7.

The study focuses on situational awareness, human workload, and performance. The study revealed a significant difference in performance between the interfaces using a three-way ANOVA. And a Bonferroni post hoc test showed the predictive intrusive interface to be the best performing interface. The study also uncovered a significant difference in three of six categories of workload measurements: Mental demand, Performance, and Effort. Once again, the predictive interfaces showed the best results. The study was not able to reveal any significant difference in terms of situational awareness.

There are more studies to be conducted in these areas, especially within the subject of situational awareness. Talks about future studies can be read in Chapter 8 where the group attempts to convey the essential factors which could help set up better studies for these subjects in the future.

This report also has a theoretical contribution a list of design implications as seen in Chapter 7. These implications are collected from the study and the three expert reviews and serve as a good starting point for future Human Swarm Interfaces.

This master thesis concludes in an empirical contribution saying that the predictive interfaces reduce mental demand, give a better mental performance, lower the effort required i.e. decreases human workload. It also decreases reaction time compared to the conventional interface. No findings regarding situational awareness can be made from the study.

A | Appendix

A.1 Sprint One

This section covers the first implementation sprint, which focus were on the user interface and getting it to fulfill the requirements for the IC, as well as incorporate some of the feedback received on the interface in (Jensen et al., 2020). The sprint took place from February 18th to March 3rd.

A.1.1 Planning

According to the project plan the purpose of the first sprint was to implement the graphical changes which were elicited in the previous semester and from the interviews conducted in section 5.1.1. These changes were summarized to a requirement list in section 5.1. The group used planning poker in accordance with scrum to evaluate the time necessary for each user story. For the first sprint, knowing the the amount of story points the group is able to complete during a sprint can be difficult as there is no previous sprint data to to rely on. The goal was set to 20 story points to accommodate for the decrease in group members compared to the 30 points goal from last semester.

The user stories chosen were all related to the frontend to change the interface into one that is specific for the IC. the stories are seen in Table A.1

#	User story	Story points	System
1	As IC, I want the live feeds bar to be at the bottom, to utilize the screen size better.	5	Frontend
2	As IC, I want to be able to navigate the interface using physical inputs, so I can use it in my work environment.	3	Frontend
3	As a developer, I want a more smooth updating of the drone positions, so the page doesn't appear to be reloading.	2	Frontend
4	As IC, I want the orientation of UAVs and boat showed, so I know which way they are searching.	2	Frontend
5	As IC, I want relevant information about the UAVs next to the UAV, so I can get an orientation about the swarm.	2	Frontend
6	As IC, I want to be able to enlarge a live feed by double clicking it, to allow an even greater view.	2	Frontend
7	As IC, I want to be able to see the entire search area on the map, so I know where the UAVs will search.	1	Frontend
8	As IC, I want the boat to be visible on the map, so I know the rescue team's location.	1	Frontend
9	As IC, I want the live feeds bar to put a live feed into focus when I select a feed, to allow a better view.	1	Frontend
10	As a developer, I want the application as a PWA, so that the user can load it onto their devices.	1	Frontend
11	As IC, I want the drone status bar removed, as it provides irrelevant information.	½	Frontend

Table A.1: Sprint 1: User Stories

The total story points ended up being 20.5 which is very close to the points goal for this sprint. the implementation of these stories will be covered in the coming sections and the stories are reviewed in the section A.1.5.

A.1.2 Live Feed Bar Changes

This section covers the implementation of user story #1, #6, #9 from Table A.1, as they are all concerning changes to the live feed bar.

The live feed bar itself was originally placed at the right hand side of the screen as a vertical expendable menu as seen in Figure A.1. This functioned well on a PC monitor as it allowed for more space for the map and could be hidden away when not needed. The design however is not desirable on tablets as it requires scrolling to see all the feeds and it does not utilize the aspect ratio of a tablet in landscape mode.

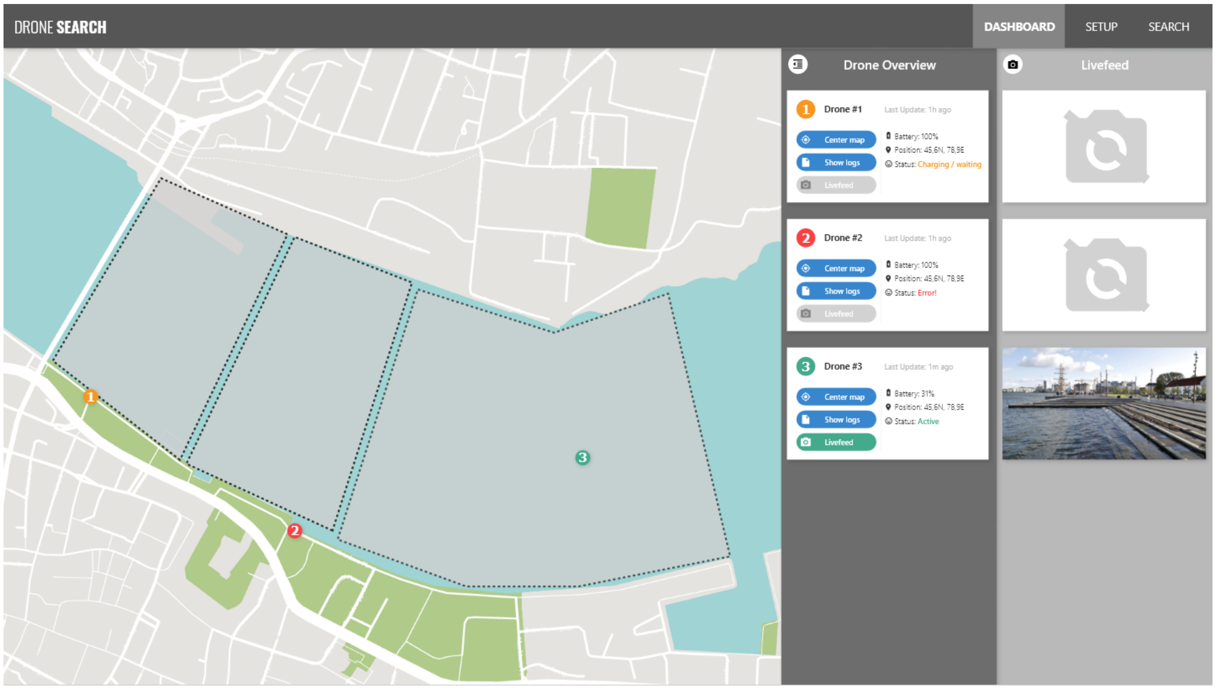

Figure A.1: Old Overview Page

User story #1 will address this issue by implementing a horizontal live feed bar containing up to eight live feeds without the need for scrolling. Having more than eight live feeds will however require the use of a scroll feature. The new live feed bar will also contain a larger video frame to allow a video to be put into focus by the IC which will be further discussed in the implementation of user story #9. The new live feed bar consists of five columns made in HTML, and the first four columns have two feeds stacked vertically while the last column is a wider column with only one video to make use of the available space.

As a continuation of user story #1 where the live feed bar was moved to the bottom of the screen user story #9 was implemented. User story #9 states the the IC should be able to select which specific which live feed should be shown as the selected feed on the right hand side of the bottom bar. The prototype for the intended bottom bar is shown in Figure A.1.

Figure A.2: The New Live Feed Bar

In Figure A.2 the new live feed bar is shown. The eight different live feeds are presented on the left hand side of the bar and the focused live feed is presented in the enlarged feed on the right hand side. Furthermore, it is seen that the fifth live feed has a green outline which indicated that it has been put into focus by the operator. The operator can at any time change the focused live feed to whichever feed he desires.

User story #6 states that a by double clicking a live feed it should be displayed in full screen mode. Full screen mode is a simple popup with a video taking up about 90 percent of the screen. The edges surrounding the video are darkened, but transparent to allow the user to understand that the full screen video is a popup and that the previous is screen is still readily available if the popup is closed. Full screen mode can be activated by double clicking any of the feeds from the 4 columns with the vertically stacked videos. Double clicking the enlarged video will, as of now, not bring up full screen mode, however this might change depending on user feedback. The full screen mode can be seen in Figure A.3

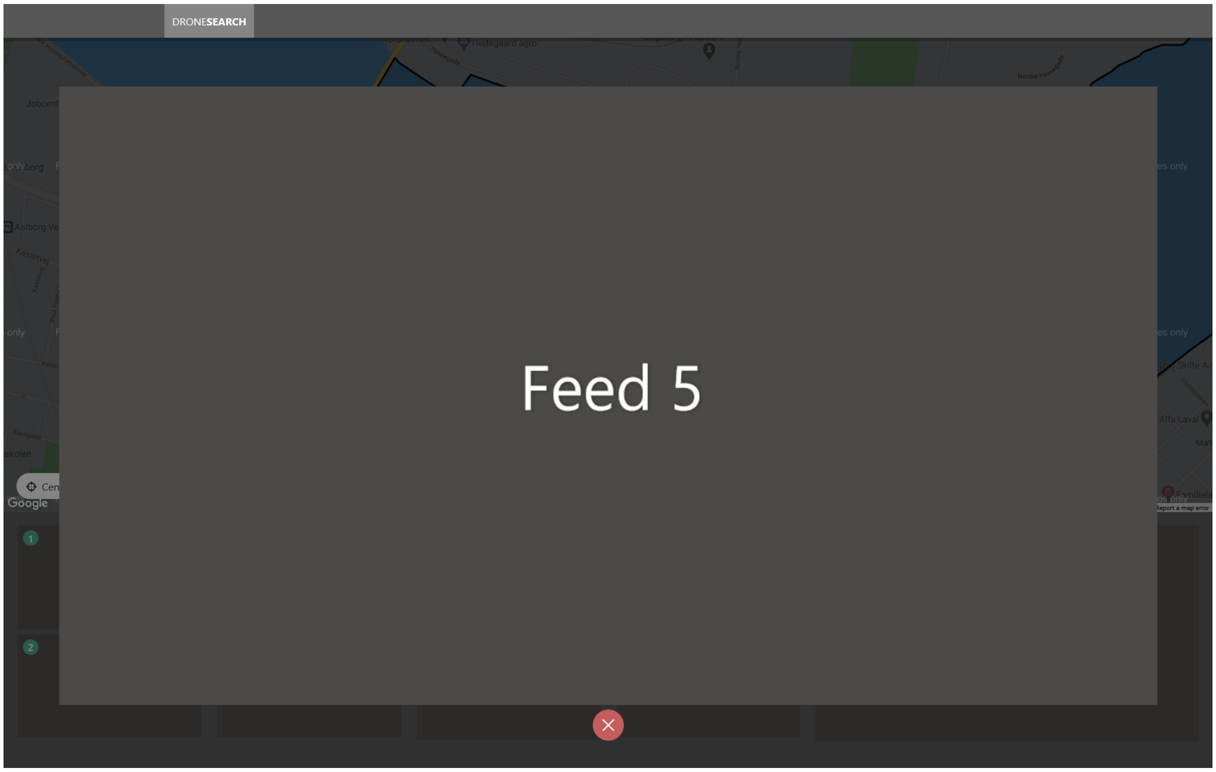

Figure A.3: The Full Screen Mode

A.1.3 Map Changes

From the interface feedback in (Jensen et al., 2020), it was decided to redo some of the elements about the drones and the map in the interface. For the IC the UAV status bar took up a lot of space showing some not to relevant information for his tasks, so this lead to User Story 11 in table A.1. However the IC still needs information about the UAVs, and based on last semesters feedback it was decided to visualize the UAVs view, show relevant information next to the UAV on the map but also show the boat's position on the map, which are stories four, five, seven, and eight. The last user story for the map is story three, to get a more smooth interface with moving markers on the map instead of blinking markers.

The first story completed was story three, which was a simple change in the way that the interface updates the UAV information. The initial implementation had a list over the UAVs with their relevant data. Each time the information was updated, this list would be overwritten with a new list containing the updated information. This implementation lead to the UAV markers on the map blinking each time there was an update. The fix to get a more smooth updating without blinking markers, was to go into the existing list and update the information on the UAVs instead of overwriting. Just updating the relevant information on the UAVs stopped the markers for blinking as they were no longer being overwritten but just updated, which made the updating much more smooth.

In order to get the boats position on the map, a service was created to retrieve the position of the boat, but since we currently do not have a boat or anything in the backend to provide the boats position, the service simply provides a hard coded position. However the service is made such that it can easily use an API to get the position, and then the rest of the interface would use that information instead automatically with having to make more changes. This is possible by the service exposing an Observable containing the position, and when a new position is available this observable will get the updated position. The map will create a new marker for the boat, the same way as with the UAVs, anchored to the position in the observable. This way when the observable changes, the marker for the boat moves with it.

With the boat on the map, the orientation of the UAV's and the boat could be visualied. Here orientation is the bearing of the UAVs and boat and a field of view. In order to achieve this a second marker for each UAV had to be place on the map with the same position as the UAV, the marker itself would be a dynamical created SVG triangle. The triangle would be created by using the cosine and sine functions to get the missing two points of the triangle, which would then be used in the SVG to form the marker.

To remove the status bar with the UAV information, the HTML tag which placed it on the page, was simply removed. The code for the actual list and cards were kept as it could be useful later as reference or implemented in a new way. But there should still be information about the UAVs on the map, this was solved by placing the relevant information next to the drone on the map. For the information displayed it was chosen that the bearing of the UAV should be displayed along with the distance from the boat to the drone. The bearing between two points can be calculated using the formula for the forward azimuth, which can be seen in Listing A.1.

```
public calulateBearing(lat1, lon1, lat2, lon2): number {
    const y = Math.sin(lon2 - lon1) * Math.cos(lat2);
    const x =
      Math.cos(lat1) * Math.sin(lat2) -
      Math.sin(lat1) * Math.cos(lat2) * Math.cos(lon2 - lon1);
    const rad = Math.atan2(y, x);
    return Math.round(((rad * 180) / Math.PI + 360) % 360); // in degrees
}
```

Listing A.1: Forward Azimuth

The distance between two positions can be calculated using the haversine formula which gives the shortest distance over the earth's surface. The implemented haversine can be seen in Listing

```
  public calulateDistance(lat1, lon1, lat2, lon2): number {
    const R = 6371e3; // earth mean radius
    const Phi1 = (lat1 * Math.PI) / 180;
    const Phi2 = (lat2 * Math.PI) / 180;
    const DeltaPhi = ((lat2 - lat1) * Math.PI) / 180;
    const DeltaLamda = ((lon2 - lon1) * Math.PI) / 180;

    const a =
      Math.sin(DeltaPhi / 2) * Math.sin(DeltaPhi / 2) +
      Math.cos(Phi1) * Math.cos(Phi2) * Math.sin(DeltaLamda / 2) *
          Math.sin(DeltaLamda / 2);
    const c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1 - a));

    return Math.round(R * c); // in metres
}
```

Listing A.2: Haversine Formula

In order to display the information on the map, a slightly modified map popup was used. Normally when using a map you can click on a location and a small box will open with information about it, we used that same box but made it always open, removed all styling from it so it was transparent, and lastly removed the option to close it. This way we ended up just having the text next to the markers, and it would move together with the marker.

The final map can be seen in Figure A.4.

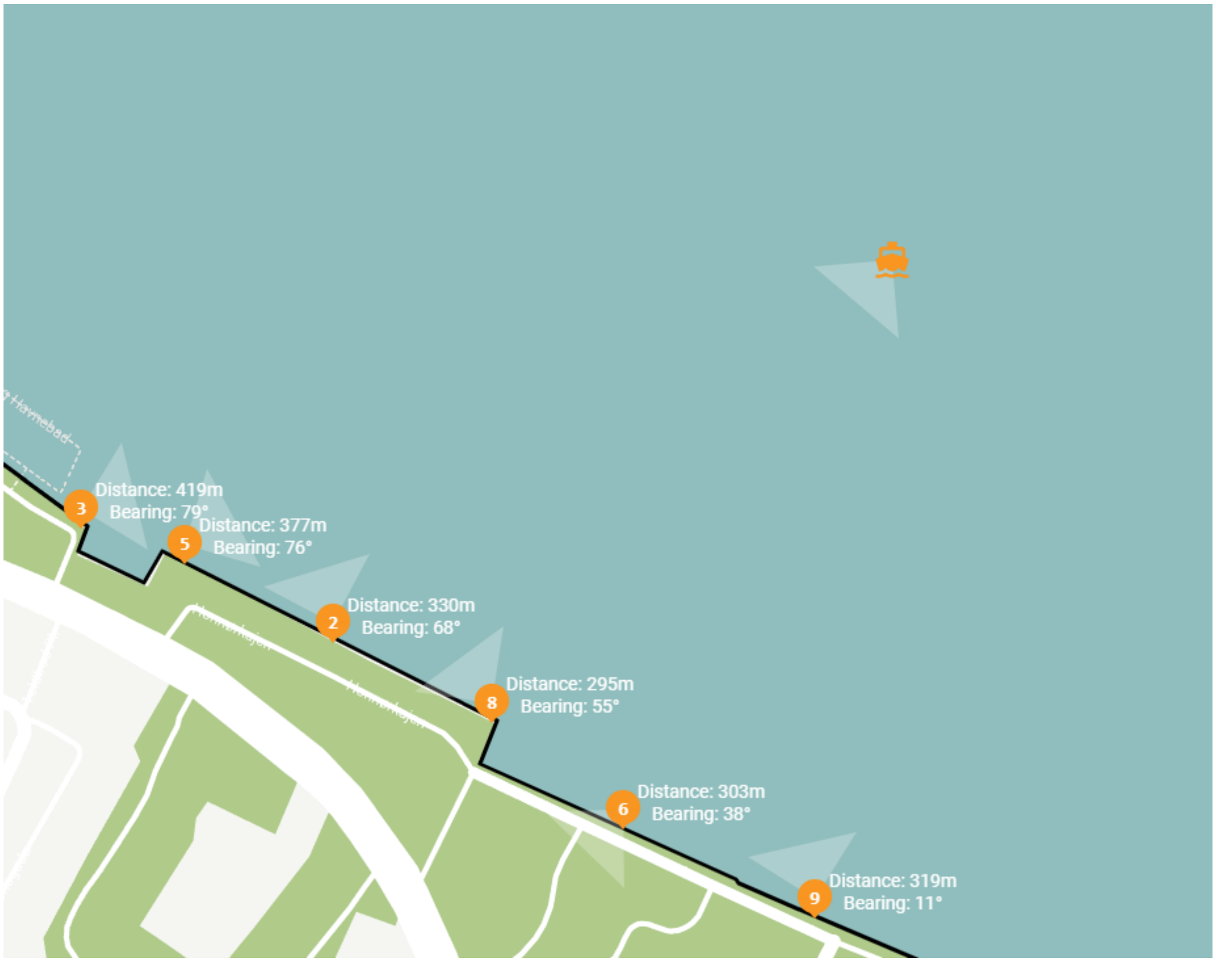

Figure A.4: The new map

A.1.4 Creating a PWA

The last two user stories 2 and 10, about having the interface as a PWA and able to use physical input to control the interface. A PWA or Progressive Web App is a web page which can be saved as an app on devices. It includes functionality as caching, offline ability, and access to native functions using web workers. Luckily setting up a Angular project as a PWA is a easy process, and most of the work is automated. Angular have a CLI which can be used to add PWA support to a project, the CLI will create the necessary files and modify configurations to include the new files and enable the PWA. For this project we just went with the default configuration of the PWA that Angular configures, but otherwise it would be possible to change how the caching works, and what the behavior should be when offline.

As the interface is indented to be used on a device with physical buttons on it, and in our study an iPad Pro with a connected numpad, we needed the interface to be able to listen to for button presses. Luckily there is no difference in the way this is handled on a computer and a tablet, so in our angular project a hostlistner was created to listen for keyboard events. It was then made so that when the buttons 4, 5, 6, or 8 was pressed a white border could be moved left, down, right, or up on the livefeeds. Enter could then be used to select a livefeed to be showed in the larger view to the right, and if enter was pressed again the livefeed would open up on the fullscreen mode, where enter again would close it.

One last thing before being able to fully run the PWA on the iPad was to get the backend hosted on a server, as the local development servers on our computers was not reachable by the iPad. A team member had a cloud server which was hosted by a 3rd party. The database and API was setup and installed on the cloud server, as well as added as a system service so that the API would always be running. With the Backend hosted, the iPad would be able to reach it when using the interface.

A.1.5 Review

At the end of the sprint a review meeting was held to go over the implemented user stories and see if there was something missing. The development team was done with the assigned user stories halfway into the sprint, which made room to for other tasks, such as setting up the backend on a hosted server so that interface could be compiled for the iPad.

As mentioned in Section A.1.1 the development team aimed for 20.5 points this sprint, which was completed halfway though the sprint. The reason for this is a combination of having over-estimated some of the smaller user stories, and setting the estimated points too low for a two week sprint with full days as the development team do not have any lectures any more to account for. For sprint two the amount of story points chosen for the sprint can be increased to accommodate for the experience gained in sprint 1.

A.1.6 Retrospective

After the review meeting, a retrospective meeting was held to look at the process. The development team have worked a lot together before in the past, during that time have developed a well functioning process for development. However this sprint was different, as in the start of the semester the original team split into two teams working on two projects. This meant that the development team this time would only consist of three members instead of the usual six members.

This lead to the belief that maybe less story points should be assigned to the sprints, as there were less developers in the same two week sprints. A little though was given to the fact that more time was also available for the team as there was no lecture to attend this semester, but it's impact was not fully factored in. Another problem the team had to overcome was that pair programming had always been used, but being three members made it difficult to have pair programming on all the user stories. Instead it was decided that on the bigger and more complex stories, the team would assign two members to do pair programming, and the third member would then work on the smaller stories which could easily be done by one person. The development team also remained in the same voice call, even though working on separate stories, so that one member would not have to sit alone, but also so that all members to follow the stories and get feedback from all members.

A.2 Sprint Two

As the stories in sprint one was completed ahead of time, it allowed sprint two to start before time. The sprint therefore took place from March 2nd to March 15th. The focus of the sprint was to create the predictive elements in the system and prepare it for a user study.

A.2.1 Planning

#	User story	Story points	System
1	As a developer, I want a predictive layer on the live feeds, so the IC can be notified about critical situations	5	Backend
2	As IC, I want to be notified about critical situations, so I can quickly respond to them	3	Frontend
3	As a developer, I want the Test person to click on the map to move the boat, and log the time and GPS of the click	2	Frontend
4	As a developer, I want the frontend to use the bearing from the backend, so the drones show the correct bearings	2	Frontend
5	As a developer, I want a video feed to be able to trigger an action when the video reaches a certain timestamp, so the interface can appear to be predictive.	2	Frontend
6	As a developer, I want a script that changes the drones locations, so that it appears they are moving.	2	Backend
7	As a developer, I want the drone that found the person to stop moving when the predictive layer is activated, so the person pin can be set.	1	Frontend
8	As a developer, I want the person to be placed in front of the drone which detected them.	1	Frontend
9	As a developer, I want the drones to start and stop when clicking the button, so the demo can play.	1	Frontend
10	As a developer, I want the enlarged and fullscreen views to download the stream at the specified time, instead of downloading the entire video.	1	Frontend
11	As a developer, I want a hotkey to toggle predictive elements in the interface, so I can easily switch in the user survey.	1	Frontend
12	As IC, I want the found person to be displayed on the map, so I can rescue them.	1	Frontend
13	As a developer, I want the drone that found the person to stop moving.	1	Backend
14	As a developer, I want the drone bearings to be in the database, so they can be used in the frontend	1	Backend
15	As a developer, I want an endpoint to return 8 video URLs, so the frontend can get the right videos.	1	Backend
16	As a developer, I want the survey videos hosted on the server, so the frontend can use them.	1	Backend
17	As a developer, I want it to be possible to start and stop the demo, so I can ask the test person questions.	½	Frontend

Table A.2: Sprint 2: User Stories

For the second sprint, the amount of story points aimed for was increased from 20 to 25 in compliance with the Sprint 1 review in Section A.1.5. The stories focused on getting the system ready for the user study in terms of demo functionality and system architecture. Changes were made to the backend to simulate an active mission, and machine intelligence was applied to the live feeds. The frontend got tweaked to make the simulated mission seem as authentic as possible. The most significant changes will be discussed in the following sections. The total amount of planned story points ended up being 26.5, which is slightly above the aim, but considering the early completion of Sprint 1, it should not be an issue.

A.2.2 Livefeed Intelligence

To enable the detection of people in the water the system must implement a machine intelligence model which can be applied to the live feeds. In a real setting the feeds must go through this model in real time to do object detection, however for the demo it is sufficient to run pre-recorded videos through the model and simply use the saved output in the demo itself.

For this a pre-trained model, using the YOLOv3 model which is a FCNN was used. The trained model is used for general object detection, but for this it will only be used for detecting people. In order to run the model on the videos, a program called DarkNet was used. DarkNet would run the model on the video frame for frame and draw bounding boxes on the found objects, but it would do it for all the detected objects. Instead of changing the model to only detect people, the source code of DarkNet was changed such that it would only draw the box if the label was "person".

A.2.3 Backend Changes

This section will describe the implementation process of the most important user stories from Section A.2.1 related to the backend. The backend has undergone a few changes during Sprint 2 in preparation for the user study, including taking care of live feed URLs' delivery to the front end and a demo service.

Live feed URLs

Up until Sprint 2, the URLs of the live feeds shown in the frontend have been statically defined in the code. However, it was necessary to make a more dynamic URL delivery to enable a switch between the predictive videos and the non-predictive videos. It was also essential to make sure the list of live feeds only included one video with a person to avoid confusing the study participant. This is done by grabbing a random video from the videos, including a person from either the predictive or non-predictive list. The rest of the list is filled up with videos without a person, resulting in the frontend receiving seven clean videos and one video with a person.

Demo Service

Since the demo will occur without real drones flying in the air, it is necessary to simulate virtual drones realistically. To accomplish this, the backend will run a demo script when enabled by the frontend. The demo script will run asynchronously on the ASP.NET server and is controlled through the REST requests. The demo service has a substantial amount of functions to make it work, but this section will provide an overview of the overall functionality. When a *StartDemo* request is received from the frontend, the script will start going through a loop which can be seen in Listing A.3.

```
1  private void DemoLoop()
2  {
3      //Continue looping untill a cancel is requested
4      while(!_cancellationToken.IsCancellationRequested)
5      {
6          int i = 0;
7          foreach (DemoDrone drone in _demoDrones)
8          {
9              //Skip if paused
10             if (drone.Paused)
11             {
12                 i++;
13                 continue;
14             }
15
16             // Advance the drone by one step
17             AdvanceDrone(drone, i++);
18         }
19
20         Thread.Sleep(_stepDelay);
21     }
22 }
23 }
```

Listing A.3: The Main Demo Loop

The loop goes through the list of drones and calls the *AdvanceDrone* method as seen on line 17. If the drone is currently paused, it will be skipped and remain in its current position. Line 20 calls the *Sleep* method with a set interval of 2000 ms, which indicates the update rate. The

AdvanceDrone method sets the drone coordinates to the next set of coordinates on its path. If the drone's current path is complete, then a new path will be calculated given a bearing and a distance. The new position is then updated in the database. The *AdvanceDrone* method can be seen in Listing A.4

```
1  private void AdvanceDrone(DemoDrone drone, int droneIDX)
2  {
3      if (drone.Linecoords.Count == 0)
4      {
5          double nextBearing = drone.Reverse ? _droneBearings[droneIDX,
               drone.CurrentStep, 0] + 180 : _droneBearings[droneIDX,
               drone.CurrentStep, 0];
6
7          //Calculate the next line of points
8          drone.Linecoords =
9              GetNextPathLine(drone.DroneObj.CurrentPosition.Latitude,
                   drone.DroneObj.CurrentPosition.Longitude,
10             nextBearing, _droneBearings[droneIDX, drone.CurrentStep, 1]);
11
12         [...]
13     }
14     var nextCoords = drone.Linecoords.Dequeue();
15
16     [...]
17
18     //Set Coords to the next coords on the path
19     drone.DroneObj.CurrentPosition.Latitude = nextCoords.Item1;
20     drone.DroneObj.CurrentPosition.Longitude = nextCoords.Item2;
21     drone.DroneObj.CurrentPosition.Bearing = bearing;
22
23     //Update the drone in the DB
24     _droneService.UpdateDrone(drone.DroneObj.UUID, drone.DroneObj);
25 }
```

Listing A.4: The AdvanceDrone Method

Line 3 checks if the current path is empty, and line 5 calculates the next path by using the formula presented in Listing A.5. The new coordinates are then taken from the front of the list on Line 14, and the drone is updated in the database on line 24.

```
private Tuple<double, double> GetNextCoordinate(double l1, double lg1, double b,
    double d)
    {
        b = ToRad(b);
        l1 = ToRad(l1);
        lg1 = ToRad(lg1);
```

```

        double l2 = Math.Asin(Math.Sin(l1) * Math.Cos(d / _R) + Math.Cos(l1) *
            Math.Sin(d / _R) * Math.Cos(b));
        double lg2 = lg1 + Math.Atan2(Math.Sin(b) * Math.Sin(d / _R) *
            Math.Cos(l1), Math.Cos(d / _R) - Math.Sin(l1) * Math.Sin(l2));

        return new Tuple<double, double>(ToDeg(l2), ToDeg(lg2));
    }
```

Listing A.5: The GetNextCoordinate Method

The formula is an implementation of (Veness, 2020) which calculates the next point given a starting latitude and longitude, distance, and a bearing. This formula is called several times to calculate several coordinates along a long straight path until the next turn. When a turn occurs, the formula is given a different bearing, and the process repeats itself as a result of the *DemoLoop* method. The script updates the drones live in the database, which is then retrieved by the frontend. The drones will continue to fly in the simulated path until the demo is stopped and reset.

A.2.4 Frontend Changes

Coming from Sprint 1, the frontend was in a good state with only a few elements missing to be able to perform in the intended way. One change that was made, was to have the drones use the bearing from the database instead of getting a randomly assigned bearing. This was made possible as the backend had been updated to contain bearings, which is described in Section A.2.3.

One essential element that was missing was the ability to show a located person on the map, therefore a service was made which would be used to obtain the position of a located person, which would be a static position as the backend did not support this functionality. A new marker could then be placed on the map listening to the location of the service.

Predictive System

A new part of the system implemented in this sprint, is the predictive service and its connecting elements in the interface. The predictive service has two data streams or observables which the rest of the system can listen to. The first observable exposes a boolean value representing if the predictive elements should be active. The second observable exposes the data about the prediction, which is the index of the drone which has activated the prediction service.

The predictive service also has an exposed timestamp, which correlates to the timestamp of a person appearing in the drone feeds. Each video feed will then use this timestamp to know when to activate the predictive service.

```
1  onPlayerReady(api: VgApiService) {
2    this.playerApi.emit(api);
3
4    this.source.pipe(take(1)).subscribe((url: string) => {
5      if (url?.includes('out')) {
6        interval(100).subscribe(() => {
7          if (
8            api.currentTime < this.pred.TIMESTAMP + 0.5 &&
9            api.currentTime >= this.pred.TIMESTAMP
10         )
11           this.pred.enablePredictive(url);
12       });
13     }
14   });
15 }
```

Listing A.6: LiveFeed Component

The function in Listing A.6, which is executed when the video player is ready to play, will look at the name of the video it has been assigned on line 5. If the video name contains the word *'out'*, it is an indication that this video have been though the CNN explained in Section A.2.2, and that when this video reaches the timestamp the predictive service should be activated. On line 6 a subscription which will run every 100ms is created, which will check if the time of the video have reached the timestamp. The check have been made with a small interval of 0.5sec, to ensure that check activates. If the check is inside the interval the predictive service will be activated with the video which then in the service is matched with the drones to get the index.

After the prediction service has been activated a few things are connected to it, which will then be enabled in the interface. The first thing is the person marker, this marker will only be shown if the service is active. The next thing that will activate is an info card which will inform the user that a drone has found something, which can be seen in Figure A.5. The card has three options, the first is to confirm the prediction and close the card, while the person marker remains. The second option is to

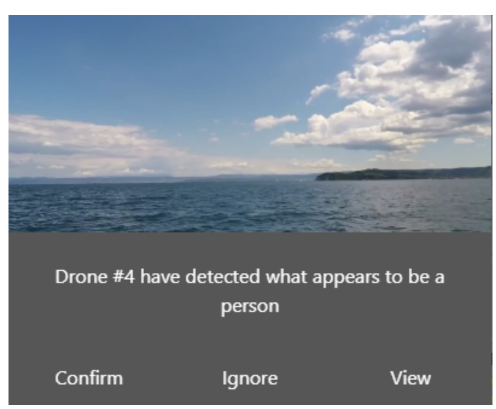

Figure A.5: Info Card

ignore the prediction, which will remove the card and person marker by disabling the predictive service. The last option, view, will open up the fullscreen view with the video.

User Study Preparations

This section discusses the changes implemented in the frontend to accommodate for the upcoming user study of the system. The upcoming user study is not intended to take place in the system's natural setting i.e. in the rescue boat. Therefore, the system needs to simulated a number of activities in order to accommodate for that fact. The first change made to the frontend to simulate a real world SAR operation is that the operator can now navigate the boat by clicking on the map. On Figure A.6 a picture of an ongoing demo is shown, here the operator has selected to move the boat. User story 8 was also implemented which enables visualization of a persons position when the system detects a person in the water which is also seen on the figure. The drone that detects the person in the water will stop moving and monitor the person until the search is recalled.

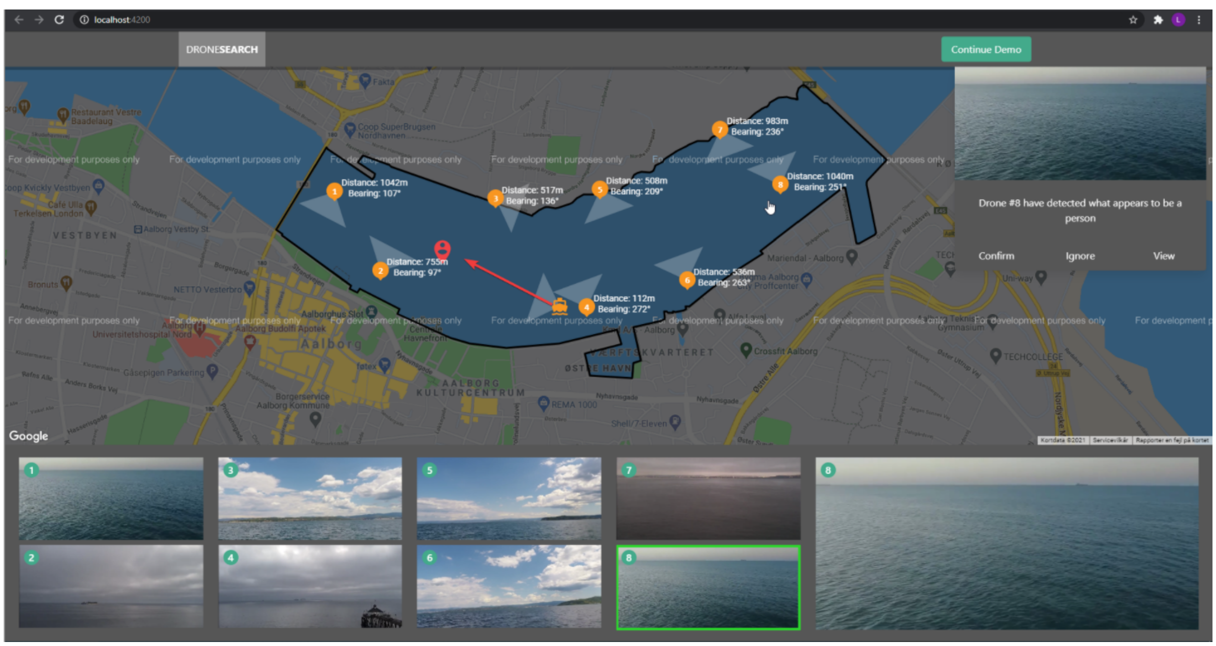

Figure A.6: Dashboard: Boat moving

For the purpose of conducting user studies and measuring situational awareness the functionality to start and pause a demo has also been implemented. The functionality of pausing a demo is essential to be able to evaluate situational awareness since it's part of the SAGAT method to do user surveys during the study. The button to do so is seen in Figure A.6. Furthermore it is also possible to toggle the system between being in predictive and non predictive mode. This enables easy switching between the two modes as they are to be compared performance wise in the user study.

A.2.5 Review

At the end of this sprint a review meeting was held. Here the development team went over the user stories selected in the planning meeting, these stories are seen in Section A.2.1. The development team implemented all the selected user stories within the allocated time slot for this sprint, actually it was completed one day in advance.

In Sprint 1 20.5 story points was planning and implemented a head of time. Therefore, more story points were planned for the second sprint namely 26.5 story points. All 26.5 story points were also completed in this sprint. The development team estimates that it is a fitting number of story points for future sprints.

A.2.6 Retrospective

After the conclusion of the second sprint, a new retrospective meeting was held. A few things had been done a bit differently compared to the first sprint. This sprint contained a lot of smaller assignments, so instead of utilizing pair programming all the time it was only used when a member had a problem. The rest of the time, each development member had their own story. The team continued being in a voice call all day, in order to discuss or ask question about the tasks. Since pair programming was not used it was important to be more detail oriented when doing code reviews before code was merged with the code base.

After the sprint the development team agreed that this sprint could be the last development sprint unless new discoveries are made, as the system is now in a state that could be used for a study. If there was to be more development it would just be minor corrections or additions to make it fit the tasks of the study, but at this time the exact tasks are unknown but the system should be able to accommodate most tasks with minor changes as it is now.

A.3 Sprint Three

Sprint 3 took place from March 22nd to March 29th. The Sprint is a follow up to the discoveries made while making a rundown of the planned demo as a few changes had to be made. The sprint was only half the length to accommodate for the smaller amount of story points.

A.3.1 Planning

The planning for Sprint 3 was straight forward as the user stories were already known before the planning started. The stories were elicited from the demo rundown which revealed a few flaws which had to be corrected before putting the demo to use. The user stories are relatively small with 1 story point assigned to each story. This became a total of 10 story points which is why only one week has been set aside for Sprint 3 instead of the usual two weeks. The most significant changes include a new logging system for accurate measurements, a server side boat implementation and the ability to stop drones. All stories are seen in Table A.3

#	User story	**Story points**	**System**
1	As a developer, I want the frontend to be able to load the position of the boat from the backend and also update its position when clicking the map with a hidden key+mouse combination	1	Frontend
2	As a developer, I want the boat to be in the database with endpoints to load and change its position	1	Backend
3	As an IC, I want to be able to enter fullscreen mode by double clicking the enlarged video feed	1	Frontend
4	As IC, I want an option to stop a drone, so I can keep looking at potential person.	1	Frontend
5	Reset not resetting everything	1	Frontend
6	Info card appears in non-predictive	1	Frontend
7	Fix card not playing	1	Frontend
8	As a developer, I want to try out an intrusive option of the info card, to see if it works better in the study.	1	Frontend
9	Fullscreen view is black on iPad	1	Frontend
10	As a developer, I want to save logs from the frontend in the database, so i can check them later.	1	Backend

Table A.3: Sprint 3: User Stories

A.3.2 Story Implementations

This section covers the implementation of the user stories presented in Sprint 3's planning section A.3.1. Since these are minor user stories i.e. they have all been assigned one story point they are grouped into meaningful categories when discussing their implementation in the remainder of this section.

Intrusive Interface

As it was decided that there was to be two versions of the info card, the non intrusive and the intrusive card, there needed to be a way for the study conductor to switch between the two interfaces. The card itself was already made, and had a CSS class which would change its size and position to be more intrusive. A listener for the key 'i' was added to the list of key listeners already active, this would then emit true on a event which the info card listens to. When the info card receives true on this event, the CSS class would be appended the card, and it would change to the intrusive info card.

Server Side Boat

After a discussion about how the IC was supposed to make the boat move it was decided that instead of clicking on the map the IC would say the bearing and distance out loud. At this point a study moderator would have to move the boat on a separate device to simulate the helm man sailing the boat to the given location. To support this the boat had to be moved from existing only on the frontend client to be a part of the database. Having the boat in the database would allow for synchronization of the boat between multiple frontends. In compliance with user story one and two, the backend had new endpoints made to get, register, update and delete boats from the database. There is no simulation of the boat on the backend which means the calculations of the boat position is still handled by the frontend.

These user stories created a new table in the database called *boats* which include a unique ID and a foreign key to a position in the already existing *position* table. The change of the database scheme was handled by the EntityFramework as described in (Jensen et al., 2020, c. 5.2.1).

Stopping a UAV

In the predictive version of the interface the system will automatically stop the UAV in question when the system detects a person in the water. However, in the the conventional version of the interface no such intelligence is applied. Therefore, the functionality to manually stop a drone has been added to the system. The intention is then, when the operator detects a person in the water he/she should stop the UAV that has found the person in order to keep getting video footage of the person during the remainder of the rescue operation.

Logging System

To get accurate time measurements a logging system was implemented which allowed for easy logging of actions in the frontend. The logging system adds and entry to the new *logs* table in the backend. The *logs* table consists of a unique ID and a string which is received from the frontend through the REST framework. At the time of implementation, actions such as start/stop demo, click events of dialog boxes, and predictive triggers are logged with timestamps.

Bug Fixes

It was found that, even though the videos would start and play fine, on the desktop browser when the fullscreen window was opened, the videos on the iPad did not. It was found that this was due to the API event listen to, while firing on the desktop browsers each time the video was ready to play, iPad OS was slightly different and did not fire this event. The fix was to change to a different event which achieved the same, but the iPad would also fire, so the code was changed from using the **canPlay** event to the **loadedMetadata** event.

A similar bug was that the video in the info card was not playing either when it appeared, this was was a more simple fix as it simply just needed to listen to the API event, and then start playing when it was fired. This was for some reason not implemented in the start.

Furthermore, it was also found that when double clicking the enlarged video feed it would not be displayed in fullscreen mode as when double clicking one of the regular video feeds. This was obviously a very minor fix, however it enhances the homogeneity of the system, as the same thing now happens when each video feed is double clicked.

Another bug in the system was that the info card would still appear in the non predictive mode, when it was not supposed to. To fix this a simple check was added to the code that activates the predictive elements, to see if the predictive mode was set to true, and if not stop executing.

Lastly it was discovered that the functionality to reset a demo was not working properly as the UAVs on the backend can only reset their position when they are not moving i.e. when the demo is paused. Therefore, it is now ensured in the fronted that the operator pauses the demo before resetting it.

A.3.3 Review

After sprint three was concluded the user stories was reviewed again to make sure that everyone had been made such that all problems found in the initial rundown of the demo was now solved. Of the 10 issues which were found, all were fixed though the sprint, and the system is ready to the next rundown before the actual study begin.

A.3.4 Retrospective

The Sprint 3 retrospective meeting was held shortly after the review. While changing the length of a sprint is unconventional according to the Scrum method it was necessary due to the amount of story points left in the backlog. Besides the change of sprint length, the rest of the sprint went on as normal and no process model complications arose. This is the last sprint of this project and no further changes to the process model will be necessary.

A.4 User Study Protocol

In this section a detailed manuscript for the user studies are presented. It contains all content all the way from the beginning to the end of the user study. It starts out with a detailed description of the setting of the project, and the functionality of the system as well as the participants role as IC.

A.4.1 Participant Brief

Hello and welcome to this study. The study is split into two parts, a training session and a study session. The training session will teach you how to use the system by giving you instructions on how to solve a set of tasks. In the study section you will then use the system to complete a mission. You will be going through three different versions of the interface and you will be asked to answer a few questions, both during the study and after the study. In this study you are taking the role as an Incident Commander of an emergency service for maritime search and rescue operations. This user study takes approximately 30 minutes.

The normal procedure for how such an operation is conducted can be seen on the left side of Figure A.7, please read this procedure now and do ask question after if there is something which is unclear.

The aim of this study is to improve this process and eliminate the need of usage of the helicopter. For this we will use a drone swarm which will help search the area, and you (the IC) will be able to monitor this swarm using the provided interface. The proposed method utilizing the drone swarm can be seen on the right side of Figure A.7, and it is this procedure we will go though with you today. More specifically the parts in the purple area, please read this procedure now and again ask questions if there is something unclear for you.

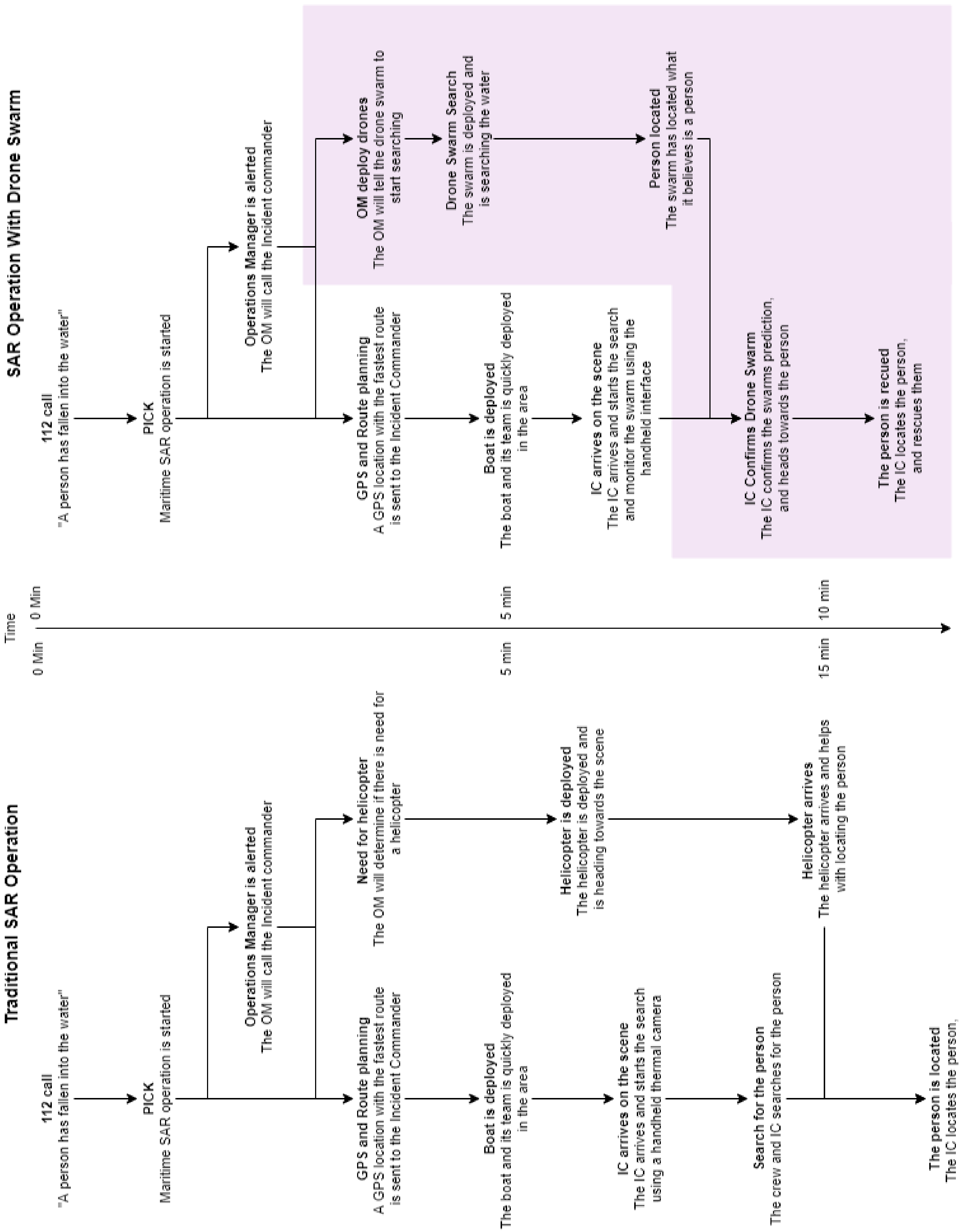

Figure A.7: Traditional vs drone operation

The study starts with a person has called and reported that a person has fallen into the water in Aalborg harbor. The operations manager at the central has called you, and after a drive you are now on site of the incident on board the rescue boat. To locate the person in the water you will be using the new drone rescue system which provides you with live video feeds from different drones and a map of the area. The drones have already been launched by the operations manager and are ready for you to use. This is where the rescue mission starts. Your job as IC is to lead the rescue operation from the boat meaning that you will be monitoring the video feeds in search of the missing person. If you identify a person in the water you are asked to give the helmsman directions on how the reach the location of the missing person.

As you may have guessed the natural setting of a maritime SAR operation is obviously in a boat on open water. Since this is a user study of a prototype system for research purposes this study will not take place in it's natural setting. The rescue operation will instead be simulated as the video footage you are about to see. You will monitor and operate the interface at the desk you are sitting at now using an iPad Pro and a connected numpad keyboard.

Do you have any questions so far about the study or the fictional situation?

A.4.2 Participant Training

The first part of the training session is a presentation of the interface that you are about to monitor and operate. There are two main components of the interface the first begin the map that displays the position of the eight drones as well as the position of the rescue boat that you and your team will be located in. As you can see each drone has a distance and a bearing associated with it. This is data that you are expected to say out loud in order to give the helmsman directions. More specifically, when you locate a person in the water you should provide the helmsman with the bearing of the drone that has found the missing person.

The last main component of the interface is the bar in the bottom of the screen, here the eight video feeds are displayed. On the right side of the bottom bar you see a larger video feed this is the selected video feed which means that it is a duplication of the feed which has a green outline on the left hand side of the bar. Lastly you can make a video feed go into fullscreen mode by double clicking on it or pressing enter on the numpad.

If you have any question please let me know, otherwise we will proceed to the training tasks.

Training Tasks

1. When you feel ready please start the demo.
2. Select feed 3 to put it into focus.
3. Navigate to feed 6 using the numpad keys 4,5,6,8 to put it into focus using enter.
4. Make feed 6 go into fullscreen mode by either double clicking on it or pressing enter again.
5. Exit fullscreen mode.
6. Please put another drone of your choice into focus.
7. Now stop the drone you have selected by pressing the numpad key 0.
8. Tell the helmsman to sail to the stopped drone, by saying the bearing and distance of the drone out loud.
9. Can you see anything different on live feed X?

You have now completed all the training tasks. Do you feel ready to precede or do you have any questions?

A.4.3 Study

Now it is time for you to use the system in a real world scenario. Your objective as Incident Commander is to find the person in the water as quickly as possible. The scenario ends when the boat is at the location of the missing person. The tasks are on the paper in front of you please read them. You will be asked to pause the demo twice during each run. Please let me know when you are ready to start the scenario.

Conducting the Study

This section is repeated for each of the different interfaces

1. When you feel ready please start the demo.
2. Figure out which drone discovers a person, and stop the UAV.
3. Make the helmsman sail to where you determine the person is.
4. When the boat reaches the destination, please pause the demo.

After 25 seconds: Please pause the demo, we have a few questions for you to answer. these questions will just give us an idea of your understanding of the system. [Sagat]

After the person has been located: Please pause the demo once again, we will run through the questions from before. [Sagat]

1. Before pausing the demo, how many UAVs were moving and how many were stopped?
2. What is the distance from the rescue boat (you) to UAV #7?
3. Which UAV is closest to the rescue boat (you)?
4. Have you located a person? (If yes, ask below)
    (a) Which UAV found the person?
    (b) Where is the person?

After the boat reaches the person: Alright, you can pause the demo now, you successfully arrived at the person and was able to help them aboard the ship. Now I have a different set of questions for you to answer, There are no wrong or right answers, so just fill them out as you feel.

Title	Rating	Description
Mental Demand	Low ____________________ High	How much mental and perceptual activity was required (e.g., thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?
Physical Demand	Low ____________________ High	How much physical activity was required(e.g.. pushing, pulling, turning, controlling, activating, etc.)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?
Temporal Demand	Low ____________________ High	How much time pressure did you feel due to the rate or pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic?
Performance	Poor ____________________ Good	How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals?
Effort	Low ____________________ High	How hard did you have to work (mentally and physically) to accomplish your level of performance?
Frustration	Low ____________________ High	How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

Table A.4: NASA-TLX Questionnaire

A.4.4 Post Study

You have now been through the three different representations of the system, and we just have a few last questions for you.

1. Which version of the interface did you prefer? (Conventional vs Predictive)
2. What did you like and what did you not like?
3. Did you prefer the peripheral or the centric info card?
4. What do you think of the size of the video feeds are they visible enough?

5. What did you think about the colors of the UAVs and the boat on the map, do they make sense?
6. Did you change your behavior/approach when you knew the interface would alert you to possible people in the water?
7. Did you like the way the video feeds were represented, or could you imagine a better way to do it?
8. Do you find the fullscreen mode useful?
9. Does the numpad navigation seem natural?
10. Was it clear for you which live feed belongs to which drone?

Thank you so much for participating and have a nice day.

A.5 User Study Consent Form

Consent Form

The purpose of this study is to study UAV swarm interfaces utilizing predictive elements, and how to present the findings of the predictive elements in a meaningful way. The study is split in two parts, one training session and the actual study which will cover three categories of interfaces. During the study, the participant will be asked few questions, after each completed category the participant will be asked to measure their own performance on a provided form. At the end of the study, there will be an interview to discuss the participant's experience and offer a chance for more feedback.

The participant can and is encouraged to ask questions about the study or the process at any time during the study, or the facilitator after on the email provided at the bottom of this document. The participant can choose not to answer one or more questions without any consequences. The participant partaking is completely voluntary and can at any time decide to stop the study without reason or consequences.

The participant is informed that all data recorded during the study, will only be used for the project and will be anonymized such that there is no personal information linking the participant to the data. The participant is furthermore informed that their anonymity will be ensured and any information provided in the study will be classified.

The participant is informed that the study will be recorded, and actions taken in the interface collected and that all data will be deleted when the project is over. The participant is furthermore informed that the recordings will be stored securely and will not at any time be published.

By signing this document the participant agrees to the above statements, and that their demographic information such as age and gender can be used in the project, as well as selected screenshots or pictures from the study can be used in the project or possibly as part of a published paper about the project.

Participant	Date
Facilitator	Date

Contact Information:
HCI101F21@cs.aau.dk

A.6 General Requirements From Last Semester

1. The system shall help the operator maintain a low mental workload.
2. The system shall help the operator maintain a high situational awareness.
3. The system shall have high reliability as it is important that it does not freeze or crash during a critical operation.
4. The system shall have high performance as time is critical.
5. The system shall provide a GUI capable of defining and adjusting search zones.
6. The system shall provide a GUI showing the position of each UAV on a map.
7. The system shall provide a GUI showing the status of each UAV in the system.
8. The system shall provide pop-ups during status changes to increase situational awareness.
9. The system shall provide a REST API to allow the frontend to interact with the system.
10. The system shall provide the functionality to launch and recall the UAVs.
11. The system shall provide communication between the UAVs and the backend to provide video stream, GPS coordinates, and current status.

12. The interface shall not rely on scrolling, as it can make it difficult for the operator to find the relevant information.
13. The interface shall only present information in numbers if absolutely necessary.
14. The interface shall avoid showing too much information, as it decreases situational awareness.
15. The interface shall show a precise alternative for GPS coordinates, as they are hard to communicate.
16. The interface shall show points of interest on the map to increase the situational awareness.
17. The interface shall display UAV information next to the UAV's icon, such that operators do not need to search for the information.
18. The interface shall use instant pop-ups instead of notifications.

Bibliography

A. Agrawal, S. J. Abraham, B. Burger, C. Christine, L. Fraser, J. M. Hoeksema, S. Hwang, E. Travnik, S. Kumar, W. Scheirer, J. Cleland-Huang, M. Vierhauser, R. Bauer, and S. Cox. The next generation of human-drone partnerships: Co-designing an emergency response system. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–13, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450367080. doi: 10.1145/3313831.3376825. URL https://doi.org/10.1145/3313831.3376825.

M. B. Bejiga, A. Zeggada, and F. Melgani. Convolutional neural networks for near real-time object detection from uav imagery in avalanche search and rescue operations. In *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 693–696, 2016. doi: 10.1109/IGARSS.2016.7729174.

N. Beredskab. Nordjyllands beredksab, 2020. URL https://www.nordjyllandsberedskab.dk/.

S. Braley, C. Rubens, T. R. Merritt, and R. Vertegaal. Griddrones: A self-levitating physical voxel lattice for 3d surface deformations. In *Conference on Human Factors in Computing Systems - Proceedings*, volume 2018-April, 2018.

R. Buchanan. Wicked problems in design thinking. 01 1902.

M. Cummings, C. Mastracchio, K. Thornburg, and A. Mkrtchyan. Boredom and distraction in multiple unmanned vehicle supervisory control. *Interacting with Computers*, 25, 05 2014. doi: 10.1093/iwc/iws011.

M. Endsley and W. Jones. Situation awareness information dominance & information warfare, 02 1997.

M. R. Endsley. Situation awareness global assessment technique (sagat). In *Proceedings of the IEEE 1988 National Aerospace and Electronics Conference*, pages 789–795 vol.3, 1988. doi: 10.1109/NAECON.1988.195097.

Forsvarsudvalget. Skibsfartens og luftfartens redningsrÅds årligeredegørelse for sø- og flyveredningstjenesten i danmark 2019, 2019. URL https://www.ft.dk/samling/20191/almdel/FOU/bilag/112/2234765/index.htm.

S. G. Hart and L. E. Staveland. *Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research*, volume 52 of *Advances in Psychology*. 1988.

Y. R. Helen Sharp, Jennifer Preece. *Interaction Design : Beyond Human-Computer Interaction*. 2019.

A. Hocraffer and C. S. Nam. A meta-analysis of human-system interfaces in unmanned aerial vehicle (uav) swarm management. *Applied Ergonomics*, 58:66 – 80, 2017. ISSN 0003-6870. doi: https://doi.org/10.1016/j.apergo.2016.05.011. URL http://www.sciencedirect.com/science/article/pii/S0003687016300989.

M. Hou, H. Zhu, M. Zhou, and G. Arrabito. Optimizing operator–agent interaction in intelligent adaptive interface design: A conceptual framework. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 41:161 – 178, 04 2011. doi: 10.1109/TSMCC.2010.2052041.

A. Hussein and H. Abbass. Mixed initiative systems for human-swarm interaction: Opportunities and challenges. In *2018 2nd Annual Systems Modelling Conference (SMC)*, 2018. doi: 10.1109/SYSMC.2018.8509744.

R. Inostroza, C. Rusu, S. Roncagliolo, and V. Rusu. Usability heuristics for touchscreen-based mobile devices: Update. In *Proceedings of the 2013 Chilean Conference on Human - Computer Interaction*, ChileCHI '13, page 24–29, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450322003. doi: 10.1145/2535597.2535602. URL https://doi-org.zorac.aub.aau.dk/10.1145/2535597.2535602.

J. K. Jacob Wobbrock. *Research contributions in human-computer interaction*. 2016.

A. H. Jensen, J. Alsen, L. Lundbo, L. Rønsholt, M. T. Boisen, and R. E. Nielsen. The intelligent food dispenser, 2018. URL https://projekter.aau.dk/projekter/da/studentthesis/the-intelligent-food-dispenser(6caf6c52-3a6a-4e78-b7de-3fec383d467a).html.

A. H. Jensen, J. Alsen, L. Lundbo, L. Rønsholt, M. T. Boisen, and R. E. Nielsen. Using uavs to aid decision-making and enhance situational awareness in maritime sar operations, 2020. URL https://projekter.aau.dk/projekter/da/studentthesis/using-uavs-to-aid-decisionmaking-and-enhance-situational-awareness-in-maritime-sar-operations(0 .html.

H. Korhonen, J. Paavilainen, and H. Saarenpää. Expert review method in game evaluations: Comparison of two playability heuristic sets. In *Proceedings of the 13th International MindTrek Conference: Everyday Life in the Ubiquitous Era*, MindTrek '09, page 74–81, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605586335. doi: 10.1145/1621841.1621856. URL https://doi-org.zorac.aub.aau.dk/10.1145/1621841.1621856.

A. Macwan, G. Nejat, and B. Benhabib. Target-motion prediction for robotic search and rescue in wilderness environments. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(5):1287–1298, 2011. doi: 10.1109/TSMCB.2011.2132716.

Z. Mi and Y. Yang. Human-robot interaction in uvs swarming: A survey. *IJCSI Internation Journal of Computer Science Issues*, 10(2):273–280, 2013.

B. Morse, C. Engh, and M. Goodrich. Uav video coverage quality maps and prioritized indexing for wilderness search and rescue. pages 227–234, 01 2010. doi: 10.1145/1734454.1734548.

J. Nielsen. *Usability Engineering*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1994. ISBN 9780080520292.

R. Nordjylland. Læge i luften, 2015. URL https://rhnordjylland.rn.dk/service/nyhedsbase-tidligere-sygehus-vendsyssel/2015/10/laege-i-luften.

J. Peschel and R. Murphy. On the human-machine interaction of unmanned aerial system mission specialists. 43(1):53–62, 2013. doi: 10.1109/TSMCC.2012.2220133. URL https://www.scopus.com/inward/record.uri?eid=2-s2.0-84879217365&doi=10.1109%2fTSMCC.2012.2220133&partnerID=40&md5=e83920387c9bf1da56004fe1ebed1ac0.

J. Redmon and A. Farhadi. Yolov3: An incremental improvement. 04 2018.

J. M. Riley and L. D. Strater. Effects of robot control mode on situation awareness and performance in a navigation task. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 50(3):540–544, 2006. doi: 10.1177/154193120605000369. URL https://doi.org/10.1177/154193120605000369.

Robotto. Robotto: Saving lives – saving nature, 2020. URL https://robotto.dk/vision/.

J. Roldán Gómez, E. Peña-Tapia, A. Martín-Barrio, M. Olivares-Mendez, J. Cerro, and A. Barrientos. Multi-robot interfaces and operator situational awareness: Study of the impact of immersion and prediction. *Sensors*, 17:1720, 2017. doi: 10.3390/s17081720.

C. Sas, S. Whittaker, S. Dow, J. Forlizzi, and J. Zimmerman. Generating implications for design through design research. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, page 1971–1980, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450324731. doi: 10.1145/2556288.2557357. URL https://doi-org.zorac.aub.aau.dk/10.1145/2556288.2557357.

K. Schwaber and J. Sutherland. The scrum guide, 2017. URL https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf.

S. Swaminathan. Logistic regression — detailed overview, 2018. URL https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc.

C. Veness. Movable type scripts, 2020. URL http://www.movable-type.co.uk/scripts/latlong.html#destPoint.

M. Ørsborg Johansen. Privat helikopter-eftersøgning kostede Flyvevåbnet 60.000 kr., 2020.