SMART SAILOR

Team Members

Amelia Yolanda	Husnul Abid	June Rose Manuzon
Amelia.Yolanda@abo.fi	*Husnul.Abid@abo.fi*	*June.Manuzon@abo.fi*

SEMESTER REPORT

Data Intensive Engineering I

Table of Contents

Problem Statement

Maritime transportation has been used to carry logistics and people for commercial purposes throughout history, it is an integral part of international trade and the global economic growth. According to the *Review of Maritime Transport 2022* by the United Nations [1], more than 80% of global trade in goods by volume are transported by sea, and the percentage is even higher for many developing countries.

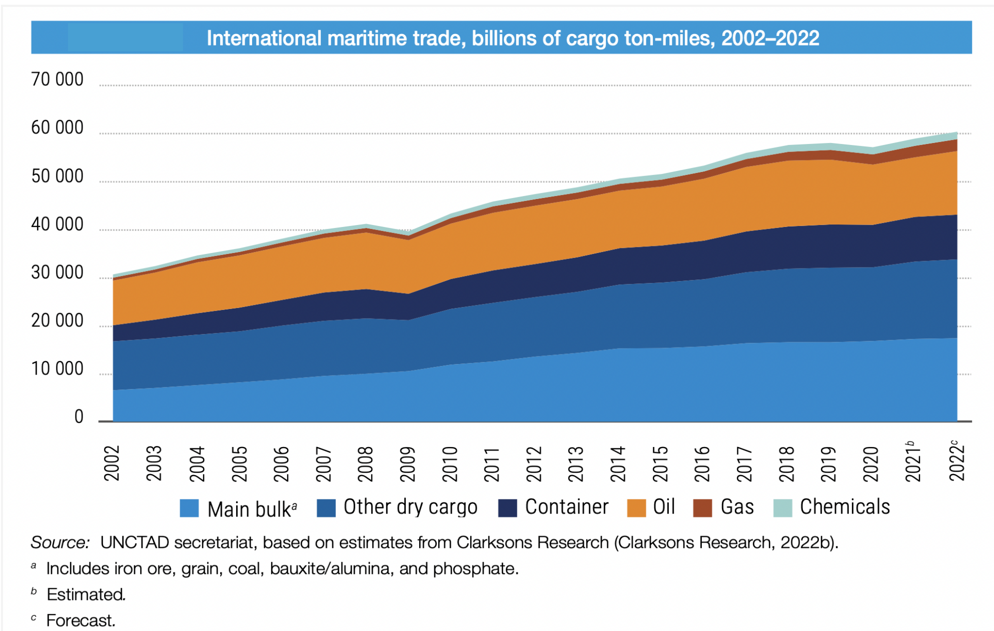


Fig 1. Yearly amount of international maritime trade in billions of cargo ton-miles

In 2021, COVID-19 and the war in the Ukraine have impacted the maritime industry and caused disruption to the supply chain including but not limited to volatile freight rates, scheduling congestion, change in shipping routes, closed ports, and new demands for shipping. In addition to that, *Safety and Shipping Review 2022* from Allianz reports that there are 3000 shipping casualties or incidents in 2021, with at least 54 ship losses (of vessels over 100 gross tonnage only) [2]. Based on the *Maritime Transport Research*, 75% out of all the marine accidents are attributable to human error [3], namely caused by physical problems (i.e. lack of sleep, excessive workload), damaging substances, communication error, distractions, navigation error, inadequate planning, or lack of training.

These reasons have initiated the public's interest in the alternative of maritime transportation in forms of autonomous vehicles. The use of maritime autonomous vehicles (MAV) allows for advance scientific endeavors, improve interstate trade, as well as promote maritime security; although it also may create challenges in regulating maritime activities [4]. MAV feasibly supports a safer and more efficient navigation of maritime vessels, and may allow improvement in terms of safety and cost-saving.

As part of the research project in Åbo Akademi University, an MAV platform (ÅBOAT) is built in order to generate a path for waypoint tracking autopilot systems through its various sensors [5]. In the continuation process of the research, a simulation software came into the equation to provide a digital twin of Åboat through the "AILiveSim" application. The digital twin allows the possibility to develop and test the MAV in an environment corresponding to the actual conditions where the boat operates along with its surroundings. This includes the flexibility of choosing the situation and scenario settings/adjustment according to the preferable testing setups.

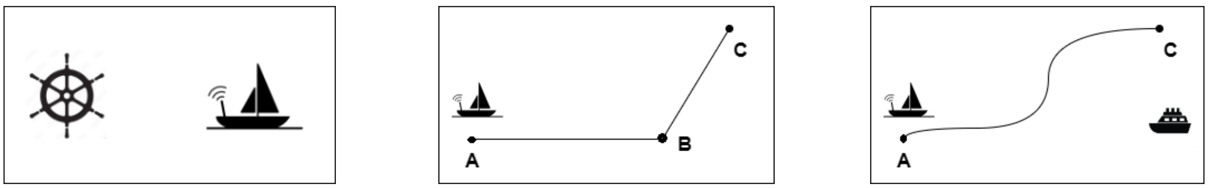
Testing the MAV through its digital twin reduces the significant time and cost that may be needed in real settings. Researchers can gather data and run the test in a safe and controlled environment without risk of getting the vehicle into harmful condition. We are designing our project, "Smart Sailor", relating to this area of interest where we use the digital twin of Åboat in the AILiveSim simulation software. We aim to test and combine various data generated from the digital environment to build situational awareness in order to benefit the usage of the autonomous vehicle.

Evolution of Åboat

Åboat had undergone numerous stages of development. A few of the stages are crucial for comprehending the project's scope and impact. One of the prior stages was to navigate the boat remotely in real time from the shore (figure 2a). Later in the development process, the researchers focused on creating a system that could steer the boat along a predetermined path (figure 2b). This system has already been implemented and using pre-stored coordinates, the system navigates the boat towards its destination.

These predetermined movements are possible with the help of various sensors that the boat is equipped with. Some of the sensors are:

- Lidar, to determine an object's distance with its surroundings using time delay with light.
- Camera, to capture the surroundings of the boat, producing image as an output.
- IMU sensors, to measure the ship's specific force or how tilted is the boat's angle.
- GPS, to capture the location of the boat.



(a) (b) (c)

Figure 2: Evolution of Åboat (left to right)

However, at this stage, the preceding system is limited in vastly autonomous behavior such as situational awareness and collision avoidance mechanism. Our goal is to merge these smart functionalities into the Åboat's core system which will enable it to detect any threat from situational awareness and path planning with collision avoidance system (figure 2c).

Partial System Architecture of Åboat

The architecture of the Åboat's entire system is extensive and complex. Therefore, we have concentrated only on the parts that would be utilized in our project. Figure 3 represents the partial system architecture which consists of many major components and data flow.

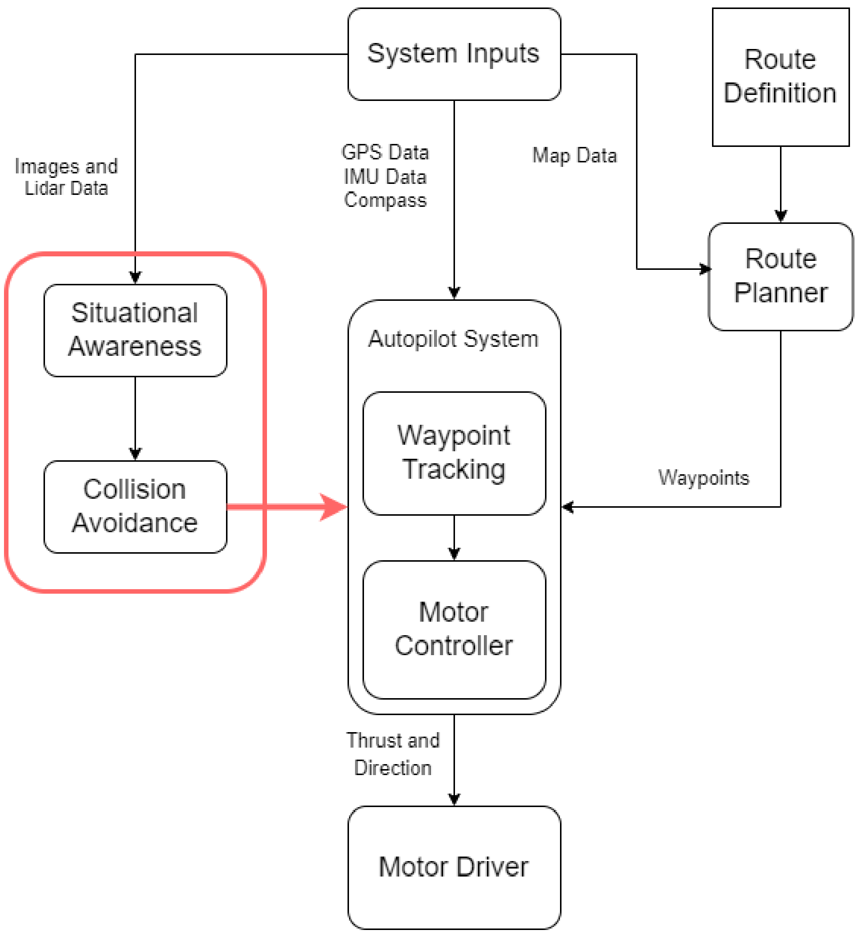


Figure 3: Partial System Architecture of Åboat

As discussed before, currently Åboat uses Camera, Lidar, GPS, IMU, Compass, etc as its system input. The autopilot system analyzes the sensor data and takes map data from the route planner which consists of waypoints. Merging these inputs the autopilot system tracks the waypoint which sends commands to the motor controller. Finally the motor controller uses a motor driver for appropriate thrust and direction.

The red section in figure 3 will be the new addition to the Åboat system which contains situational awareness and collision avoidance mechanism. Merging images with lidar data, we can detect objects with their distance which can be used as a base of situational awareness. As Åboat already has a pre-defined waypoints following algorithm, it is enough to generate new waypoints in the real-time for collision avoidance systems. This new data (waypoints) will be feed by the autopilot system that can maneuver the boat using proper thrust and direction.

Planned Process Pipeline

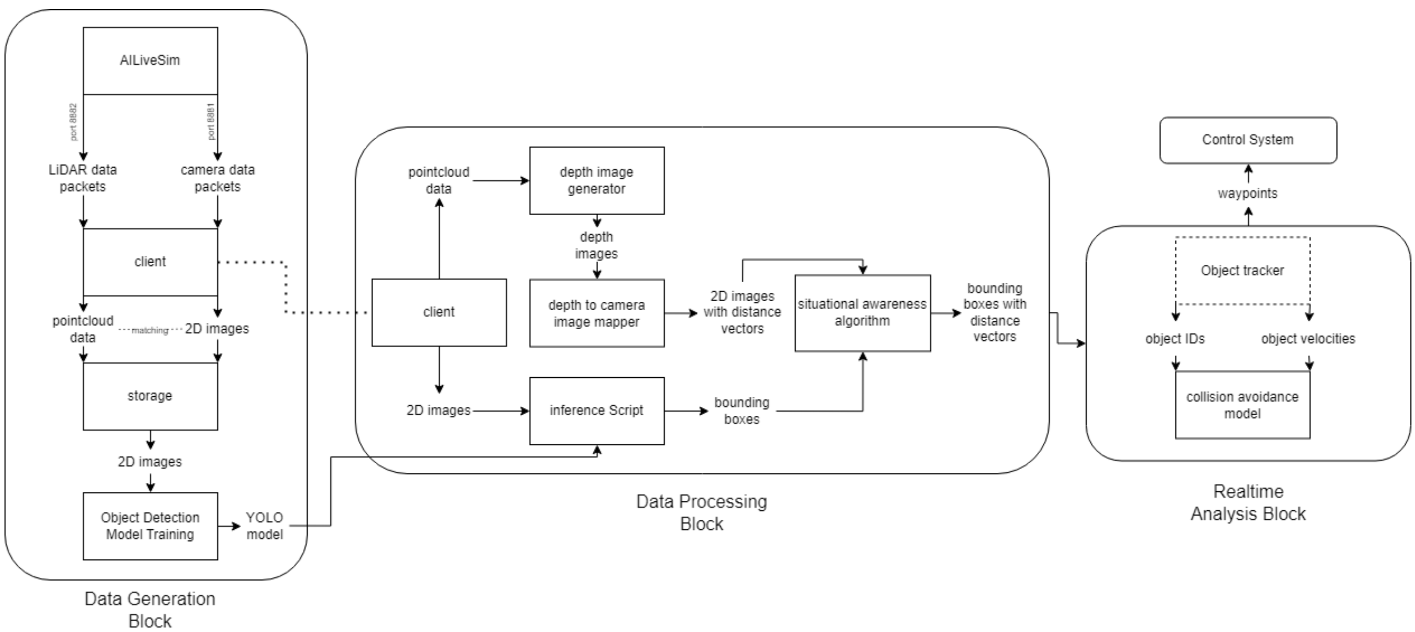


Figure 4: Simulated Input Pipeline

The planned pipeline is essentially divided into three blocks: the Data Generation Block, Data Processing Block and Real Time Analysis Block. Each block will be discussed in detail below.

Data Generation Block

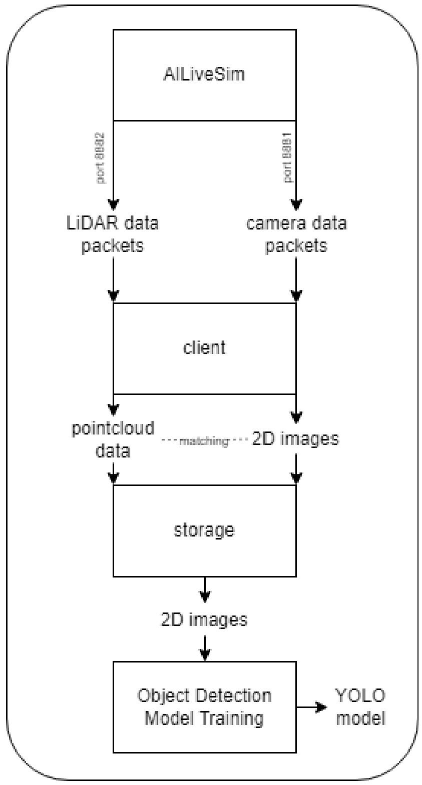


Figure 5: Data Generation Block

In the Data Generation Block, the very first component is AILiveSim. While simulations are being run on the application, data is streamed to the client through two ports. Matching data packets are transmitted and afterwards converted into two data formats, the pointcloud data format for the LiDAR data and two dimensional images for camera data. They are then put into storage for the purpose of training Object Detection and Collision Avoidance Models.

If the data will not be collected, it could also go straight from the client to the Data Processing Block.

Data Processing Block

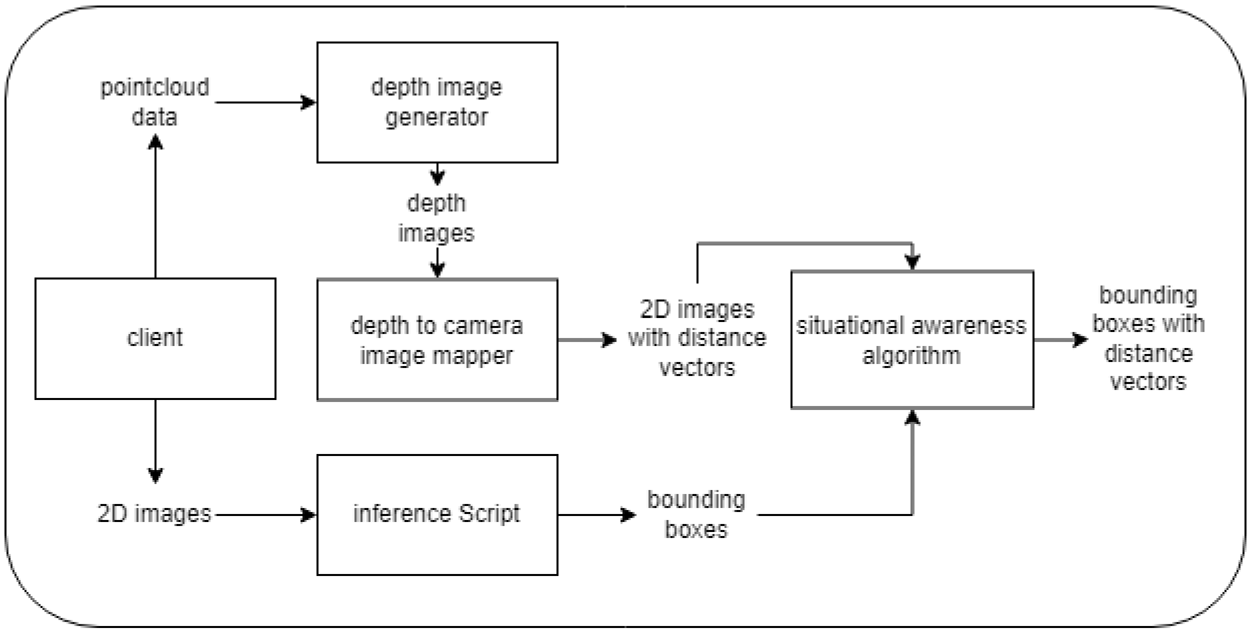


Figure 6: Data Processing Block

Continuing from the Data Generation Block, the client extracts data from the sensors and sends them to two processing stations. The pointcloud data is passed through the depth image generator to create depth images. These depth images are mapped onto the 2D images so that the individual distances of pixels from the sensors can be inferred.

The two dimensional images meanwhile, are inputted into the inference script. In the inference script, bounding boxes are created using the Object Detection Model that we trained previously in the Data Generation Block.

Finally, we make use of both the 2D images with distance vectors and also bounding boxes by inputting them into a Situational Awareness Algorithm. This section allows us to identify the classes of objects within a camera frame, while also understanding their distances in relation to the sensors.

Real Time Analysis Block

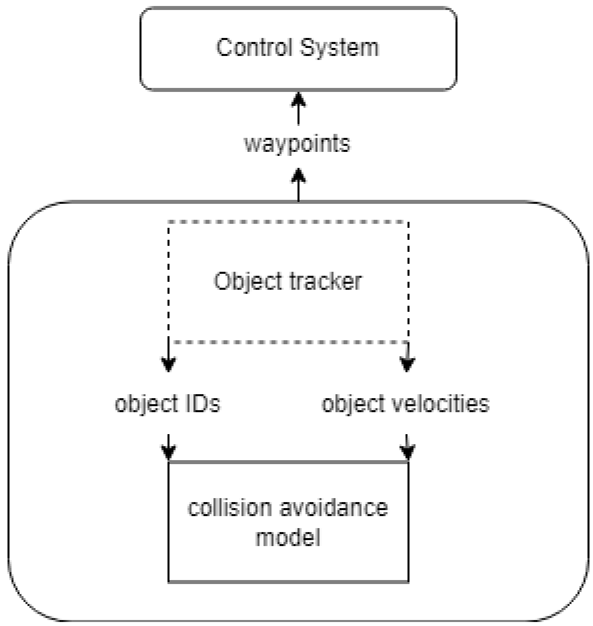


Figure 7: Real Time Analysis Block

After detecting and identifying the objects within the frame and their distances, we now use the information that we have to affect the behavior of the boat. This is done by inputting all the information mentioned into a Collision Avoidance Model. This model would output waypoints or directions that the boat must receive and execute in the control system in order to avoid collisions.

For moving collision objects, we can infer their velocities by keeping track of objects that are continuously detected in frame sequences.

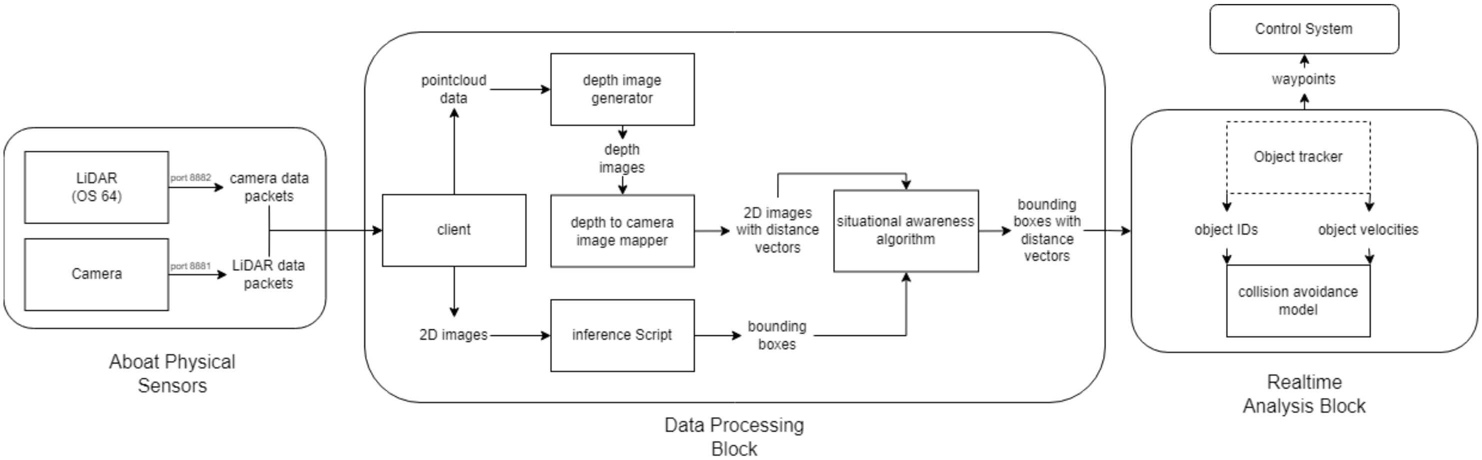


Figure 8: Pipeline with Physical Sensors as Input

In the physical setup, the Data Generation Block is removed and replaced with the physical sensors. Doing so will not impact the input or function of the Data Processing Block, the next block.

Data Collection

AILiveSim

To collect data for our research from the AILiveSim software, there are three main components that we are concerned with. These are the Scenario, Sensor, and Situation Editors. There are more components than these but these are what are primarily needed to be configured to set-up the environment for our target data.

Figure 9: Scenario Selection Menu

There are presets for different settings for the simulations in the application. For this research, the two scenarios that we use are the “TurkuWithBoats“ and “Turku_Naked”. Both feature the river Aura of Turku. Their only difference is the existence of pre-placed boats along the river.

Sensors Editor

Figure 10: Sensors Editor Menu

In the menu showed in Figure 10, different sensors can be placed on different maritime vehicles. There are a variety of sensor types that can be taken advantage of based on the configuration file inputted in this section. In this research, we use the basic camera setting with no distortion, and the OS1 64 format for the LiDAR sensor data.

Situation Editor

Figure 11: Situation Editor Menu

The objects or interacting elements in the application can be found in the Situation Editor Menu as show in Figure 11. Objects can be placed and their arrangement can be saved for later reuse. This image shows the Aboat placed in a sample situation with boats and a buoy around it.

Configuration Files and Python Scripts

AILiveSim also comes with sets of configuration files and python scripts that are used in tandem with the application. With the configuration files, the flexibility of AILiveSim could further be expanded. Sensor parameters such as samples per second, range, or resolution are available for modification depending on the type of sensor used.

The Python scripts on the other hand contain sample code that are used to manipulate or receive data from the application. One such script is used for connecting to the application via a socket server and receiving packets of data sent from digital sensors.

Data Analysis

Analysing Images

It is possible to read and store the images from AILiveSim cameras in widely used and accessible file types like PNG and JPEG. Every picture is in RGB format and the dimension of each image is 480*720*3. We have used Python pillow library to import and analyse these images.

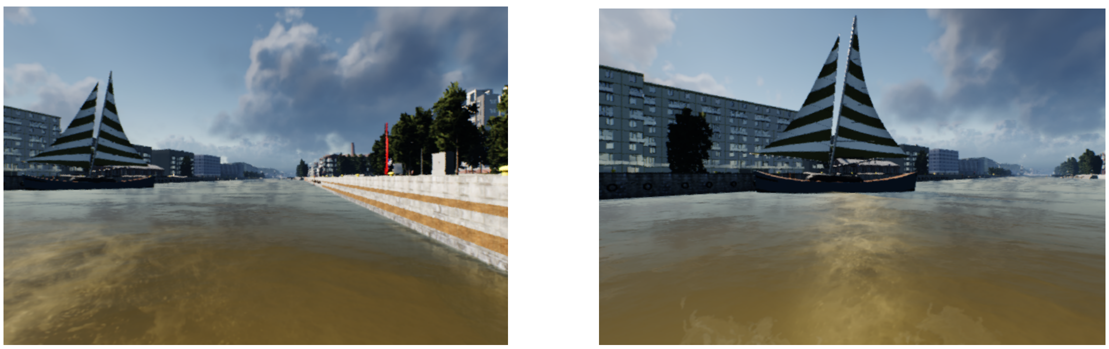


Figure 12: 480*720*3 RGB images from AILiveSim

Analysing Lidar Data

The lidar sensor from AILiveSim provides 3D point cloud as *pcd* files. These files consists of around 131,000 rows which represents 3D points with *x, y, z* and depth *r.* We were able to plot these files using Python open3d library (figure 13).

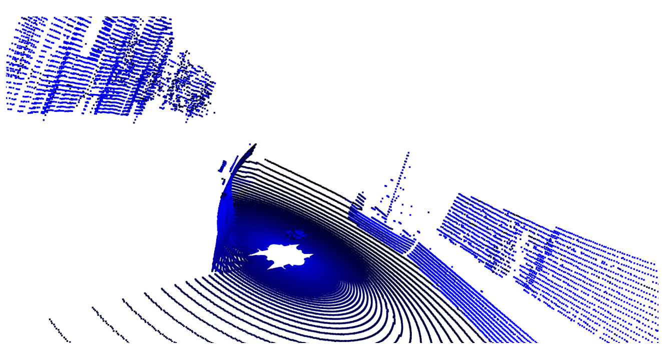


Figure 13: Plotting Lidar point clouds

With the open3d library, can also take the partial points from the *pcd* file which helps us to limit the horizontal or vertical view to analyse specific region.

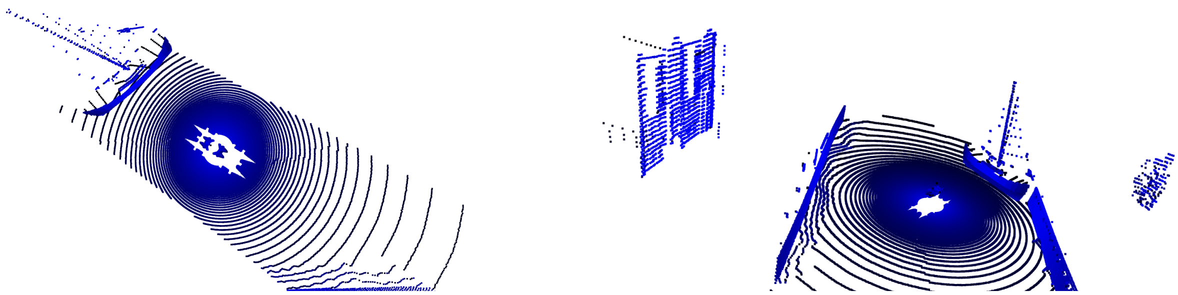


Figure 14: Partial data points from figure 13

Converting Lidar 3D data points to 2D data points

We take into account the lidar's point cloud data from AILiveSim which provides us 3D data points. However, It is challenging to perform object detection in 3D data points, and this activity takes more processing time and computational resources. Also, fusing 3D data points with 2D images is not feasible. In this scenario, We have decided to convert these 3D cloud data points into 2D data points.

In order to flatten the view of a lidar sensor to a 2D image we have to project the points in 3D space into a cylindrical surface that can be unwrapped, to a flat surface. These 3D points can be substantially projected and discretized into a 2D point map while preserving their distance characteristic using the following projection functions [6].

$$\theta = atan2(y,\ x)$$

$$\varphi = arcsin\{\ z\ /\ \sqrt{}(x^2 + y^2 + z^2)\}$$

$$r = \theta/\Delta\theta$$

$$c = \varphi/\Delta\varphi$$

where $p = (x, y, z)$ denotes a 3D point and (r, c) denotes the 2D map position of its projection. We fill the element at (r, c) in the 2D point map with 2-channel data (d, z) where $d = \sqrt{(x^2 + y^2)}$ Note that x and y are coupled as d for rotation invariance around z. An example of the d channel of the 2D point map is shown in figure 15. Elements in 2D positions where no 3D points are projected into are filled with (d, z) = (0, 0).

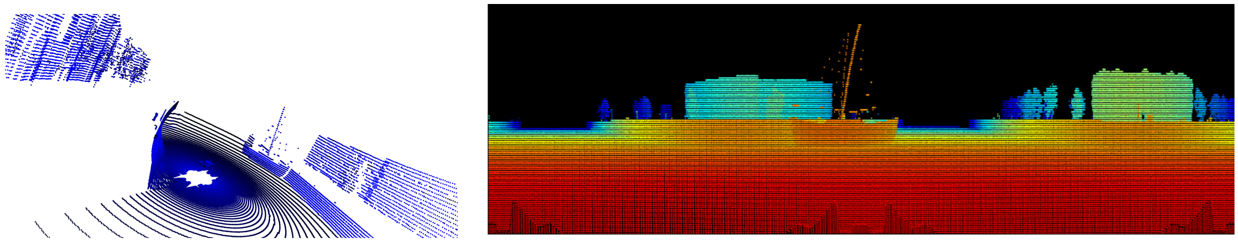


Figure 15: Conversion of Lidar 3D data points to 2D data points

The code has been implemented here. The ***h_res*** and ***v_res*** variables are very much dependent on the Lidar sensor used. According to the spec sheet for the LIDAR OS-1-64, it has the following important characteristics:

- A vertical field of view of 45.0 degrees from +22.5 degree to -22.5 degree
- A horizontal field of view of 360 degrees.
- Rotation rate can be selected to be between 10-20Hz.

The hyper-parameters are set on these values. Finally, we have the 3D points projected to 2D coordinate points, with a minimum value of (0,0), we can plot those points data into a 2D image. We can produce images based on depth, height and reflectance.

Lidar-Camera Fusion

Lidars and cameras are two types of complementary sensors. While cameras provide high-resolution shape and texture information, Lidars provide low-resolution shape and depth data. Monocular detection techniques directly anticipate 3D boxes from 2D images rather than depending on the lidar point cloud [7]. These methods have a number of difficulties since 2D images lack depth information. As a result, most monocular detectors must implicitly or explicitly forecast depth for each 2D image pixel, which is frequently a highly challenging operation.Many researchers combined these two sensor data and created a fusion to detect objects. In [8], bounding boxes are drawn into the 3D cloud points and combined fusion network which provides improved 3D detection over prior methods.

In our project, we focused on to merge 2D images with converted 2D lidar points which preserves the depth information as well. We projected the converted 2D lidar points on the corresponding images with respect to the image height and width (figure 16).

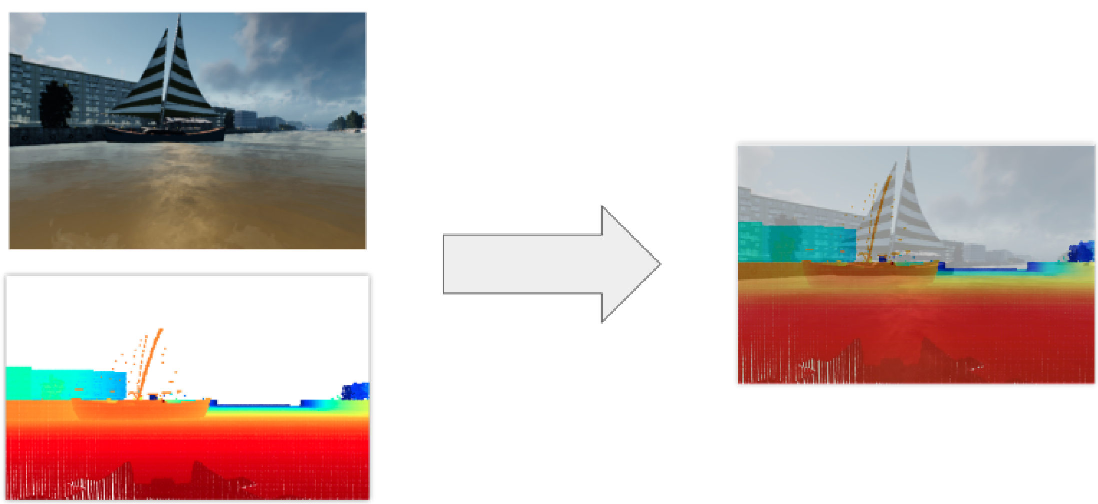


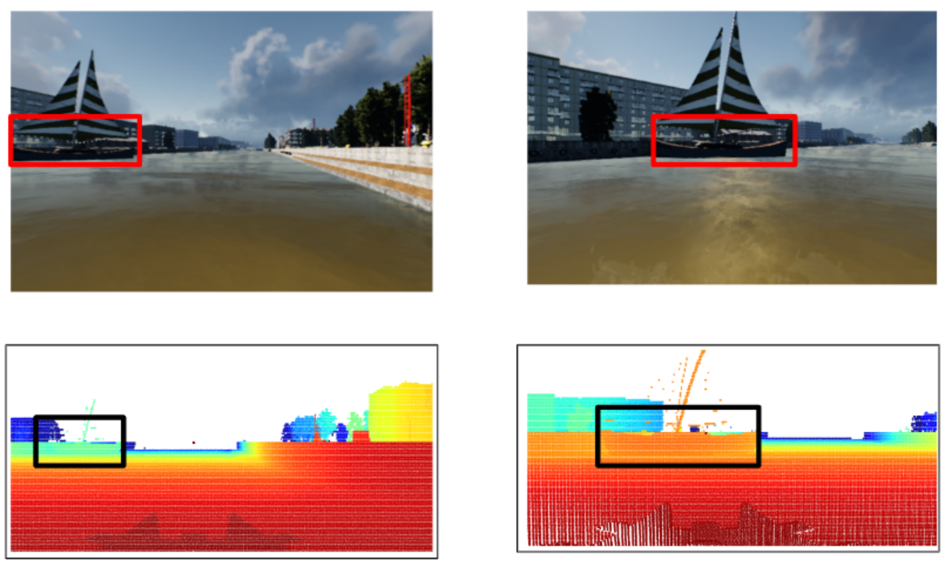
Figure 16: Lidar-Camera Fusion

This projection has been done with geometric functions. This process gives us the depth information of each pixel in the images. Moreover, object detection techniques can more easily be integrated with this technology. Fusing cloud data points with images provides the distance on a particular image area more accurately than before which can be plugged in the situational awareness system.

Situational Awareness

The main advantages of multi-sensor situational awareness systems are redundancy, which allows an observation to be cross-validated from various sources, and increased availability and integrity through complementary sensing that allows it to detect targets that cannot be detected by one sensor [9]. Although these multi-sensor situational awareness are well known in other autonomous fields like cars and airborne vehicles, the maritime field has received little attention because of the difficult environment at sea. Using lidar and camera fusion, we can significantly increase the available data that can be used as inputs of the situational awareness system.

From the lidar cloud points and image fusion, we have the RGB images with depth properties which denote the distance of a pixel from the ship. Implementing object detection in these images, we can identify objects and the pixel location of the bounding boxes. These pixel locations can be analyzed further with the lidar data to extract the distance property (figure 17).



(a) (b)

Figure 17: Two sets of situational awareness

For object detection, we are using the YOLO algorithm which stands for 'You Only Look Once'. YOLO reframes object detection as a regression problem and presents the class probabilities of the detected objects in the output photos. The main advantage of YOLO is it just requires a single pass over the image to identify objects and detect bounding boxes. YOLO has a reputation for processing real-time scenarios quickly and accurately. After getting the bounding boxes, we take the median of the distance of it from the lidar cloud points. Mean can affect the distance calculation as many pixels of the bounding boxes might not be the objects. We can export these data in json format for further processing.

Detecting objects with distance twice per second in real-time gives us the surrounding objects' location, their movement direction and speed. Analyzing these data in real-time, we can have the whole situational awareness system by knowing neighboring objects and their distance.

SWOT Analysis

The following are analysis of proposed process and methods for the smart sailor project:

Strengths

- **Self generated dataset through digital twin**: We have access to the AILiveSim simulation software that allows us to generate dataset from various sensors for our solution. The sensor projects the real conditions of the boat and is available to be set up accordingly (in terms of features, position, or amount of sensors). It is also possible to segregate sensors based on the boat's feasibility.
- **Combining data from more than one source**: We are aiming to achieve higher accuracy of object detection by analyzing Lidar and camera data to be used in comparison to receive better possibilities, the frame per rate movement from both data may be useful to know the velocity of the moving object surrounding our vessel.
- **Adaptability in different conditions**: Maritime transportation controls are affected to some point based on weather or other environment conditions. Using AILiveSim allows us to test the solution in various weather/situations, e.g. the radius of the Lidar sensor may be decreased when the weather is foggy, hence using the digital twin might help us to ensure the solution works in any kind of situation.
- **Existing reference for the digital twin**: We have access to AILiveSim training documentations, and also the guidance from the previous year's students to run the simulation software. One of the tips that we received early on is to minimize the amount of camera sensors as it may affect the rendering process of AILiveSim.

Weaknesses

- **Limited resources on computational power**: As the simulation software requires GPU power to run, so far we can only access it through the computer in Agora, and one of our computers.
- **Difficulty in accessing AILiveSim**: The simulation software has some deficiencies that require us to reload the whole application whenever the screen suddenly freezes. Some parts of the user experience are also not recognizable and can cause confusions as there are many components that need to be modified, for example, to load and operate the boat.
- **Trial and error needed to acquire proper use of the dataset**: Since we are trying to combine data that are very different in nature (from the Lidar and camera sensors), we need to find ways to transform both data to have a result that can be used consecutively.
- **Adjust method with maritime regulation**: Since the usage of autonomous vehicles is not commercial yet, we may need to adjust solutions based on the rules and regulations accordingly.

Opportunities

- **Possibility to provide dataset for ML relating to maritime**: We are trying to combine data from Lidar and camera sensors for greater accuracy. The combined dataset later might be useful for other research purposes, or it may be used as an enrichment for the current available dataset.
- **Contribute to find safer and more efficient way to operate a ship**: Through this project, we are aiming to create situational awareness that can help to improve the path planning process of an MAV. This process may relatively contribute to reducing the amount of distance and therefore save time and cost of operating a vessel.
- **Deploy on real-life Åboat**: Since this project is hosted by Åbo Akademi, we have a chance to deploy and test our model in the real-life situation.
- **Publication opportunity**: We are collecting and using data from the 3D simulation software, while a lot of publications in maritime transportation covers only the mathematical model and/or 2D simulations.

Threats

- **Issues in AILiveSim updates**: The new updates from the simulation software often cause issues such as unable to open the application before doing an update in our drive. This may cause some bottleneck when we need to access the software accordingly.

- **Replacement in maritime transportation**: With eco-friendly technology being developed rapidly, there might be possibilities that the maritime industry finds other transportation options that may result in less emission in the future. We might need to adapt the technology used for our solution to be relevant by then.

Upcoming Milestones

Current Progress

Before heading into the discussion of future Milestones, here is a quick summary of the accomplishments so far to understand the current status of the research:

- Underwent literature review relating to collision avoidance, object detection datasets, algorithms, and various sensors that are used to analyze and predict collisions,
- Discussed with previous year's students in charge of this project about the continuation plan, and consulted with Kai Jämsä relating to Åboat's hardware architecture (sensors, connection for self-driving technology),
- Installed and performed workshops on AILiveSim to become familiar with the setup of the environment/situation, scenarios and vehicles, and various sensors and ways to control the vehicles.
- Collected trial and error samples for various sensors in AILiveSim (including setting up the sensors to mirror those of the physical boat) to become familiar with the dataset collection process; and
- Mapped LiDAR sensor data in forms of point-cloud data and combined image data to Lidar data to improve accuracy in object detection.

Future Plans

These are the goals for the upcoming periods 3 and 4:

- Compile and finalize dataset from camera-lidar sensor fusion.
- Use the generated dataset (including the mapping process between data from both sensors) as a base for training for object detection.
- Merge object detection to be used in collision avoidance.
- Possibility to deploy the model into the physical Åboat.
- Compile results into report paper.

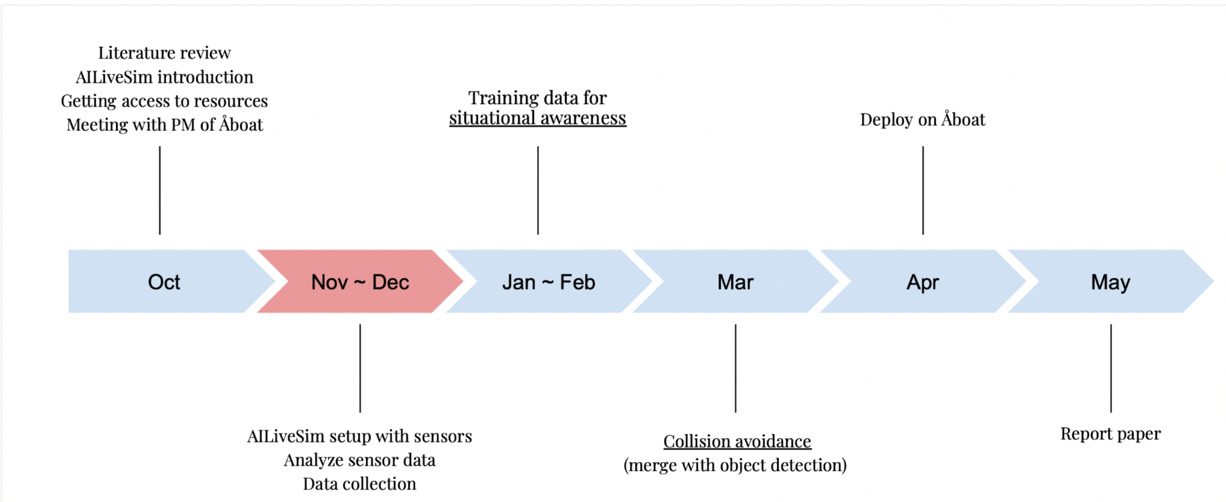


Figure 18: Upcoming milestones

Conclusion

In summary, situational awareness and collision avoidance in maritime has a huge potential and can take leverage of the recent improvement on artificial intelligence and machine learning. As testing the solution in a real life boat is expensive and also risky, the digital twin of Åboat in AILiveSim is playing a vital role in this scenario. We have collected two different sensor data from this software. Converting 3D lidar data points to 2D data points and lidar-camera fusion are primary tasks for the situational awareness system. In this semester, we have worked with these two important components which are the groundwork of situational awareness and collision avoidance. Also we have designed the pipeline of the overall system. Additionally, our solution's strengths, weaknesses, opportunities and threats have all been carefully considered. In next semester, we are planning to finish the project by implementing object detection, situational awareness and collision avoidance.

Reference

1. *Review of Maritime Transport 2022. UNCTAD*. [Online] Available at: https://unctad.org/rmt2022 (Accessed: December 14, 2022).
2. *Safety and Shipping Review 2022. AGCS Global*. [Online] Available at: https://www.agcs.allianz.com/news-and-insights/reports/shipping-safety.html (Accessed: December 14, 2022).
3. Sánchez-Beaskoetxea, J. *et al.* (2021) *Human error in marine accidents: Is the crew normally to blame?*, *Maritime Transport Research*. Elsevier. Available at: https://www.sciencedirect.com/science/article/pii/S2666822X21000083#bib0032 (Accessed: December 14, 2022).
4. Klein, N., Guilfoyle, D., Karim, M. S. and McLaughlin, R. (2020) “MARITIME AUTONOMOUS VEHICLES: NEW FRONTIERS IN THE LAW OF THE SEA,” International and Comparative Law Quarterly, Cambridge University Press, 69(3), pp. 719–734. doi:10.1017/S0020589320000226
5. Storbacka, M. (1970) *Development of autonomous navigation systems for maritime applications*, *Doria*. Available at: https://www.doria.fi/handle/10024/181815 (Accessed: December 14, 2022).
6. Bo Li, Tianlei, Tian - “Vehicle Detection from 3D Lidar Using Fully Convolutional Network”, *2016*.
7. Xiaozhi Chen, Kaustav Kundu, et al -. “Monocular 3d object detection for autonomous driving”, *Conference on Computer Vision and Pattern Recognition 2016*.
8. Darshan Bhanushali, Robert Relyea, et al. - “LiDAR-Camera Fusion for 3D Object Detection”, *International Symposium on Electronic Imaging 2020*.
9. Sarang Thombre, Zheng Zhao, et al - “Sensors and AI Techniques for Situational Awareness in Autonomous Ships: A Review”, *IEEE Transactions On Intelligent Transportation Systems 2022.*