Jurnal
Teknologi

Full Paper

LOCATION AND ORIENTATION TRACKING IN AUTONOMOUS SURFACE VEHICLES: REAL WORLD DATA VERSUS DIGITAL TWIN PREDICTIONS

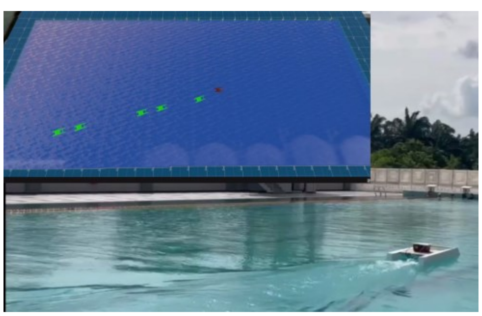
Article history
Received
6 July 2024
Received in revised form
26 August 2024
Accepted
26 August 2024
Published Online
28 April 2025

Chan Jing Yik[a], Muhammad Irfan Zainal Abidin[a], Kamarulafizam Ismail[b], Fazila Mohd Zawawi[c], Aemi Syazwani Abdul Keyon[d,e], Nur Safwati Mohd Nor[b*]

[a]Faculty of Mechanical Engineering, Universiti Teknologi Malaysia, 81310, UTM Johor Bahru, Johor, Malaysia
[b]Department of Applied Mechanics and Design, Faculty of Mechanical Engineering, Universiti Teknologi Malaysia, 81310, UTM Johor Bahru, Johor, Malaysia
[c]Department of Thermofluids, Faculty of Mechanical Engineering, Universiti Teknologi Malaysia, 81310, UTM Johor Bahru, Johor, Malaysia
[d]Department of Chemistry, Faculty of Science, Universiti Teknologi Malaysia, 81310, UTM Johor Bahru, Johor, Malaysia
[e]Centre for Sustainable Nanomaterials, Ibnu Sina Institute for Scientific and Industrial Research, Universiti Teknologi Malaysia, 81310 UTM, Johor Bahru Johor, Malaysia

*Corresponding author
nursafwati@utm.my

Graphical abstract



Abstract

The rapid advancement in autonomous technologies has driven the development and deployment of Digital Twins (DT) for various applications, including Autonomous Surface Vehicles (ASVs). This study focuses on creating and implementing a DT for an ASV to enhance environmental monitoring. The DT acts as a virtual replica of the ASV, simulating real-time data and predicting the vehicle's future state. The motivation for this project comes from the increasing need for precise and efficient environmental monitoring solutions while keeping track of the state of the ASV. The primary objective is to develop a DT for an ASV and evaluate its performance in predicting the vehicle's state in scenarios where sensor data is unavailable. The ASV and its DT were deployed in a controlled environment, with various performance metrics, including position and orientation, which are recorded and analyzed to assess the DT's predictive accuracy. When the DT received frequent updates, the mean absolute errors were relatively low: 1.411 meters for distance, 0.708 degrees for roll, 1.812 degrees for pitch, and 18.754 degrees for yaw. Extending the intervals between updates significantly affected prediction accuracy, with the MAE increased by 0.78% for roll, decreased by 21.42% for pitch, increased by 40.18% for yaw, and most significantly, the distance MAE was increased by 89.23%. The results of the study indicated that critical importance of advanced data fusion techniques and real-time processing capabilities from various sensors as well as control systems in real-time to maintain the reliability of the DT.

Keywords: Digital Twin (DT), Autonomous Surface Vehicle (ASV), Real-Time Data Integration, Predictive Modeling

Abstrak

Kemajuan pesat dalam teknologi autonomi telah mendorong pembangunan dan pengaplikasian Kembar Digital (DT) untuk pelbagai aplikasi, termasuk Kapal Permukaan Autonomi (ASV). Kajian ini mempersembahkan penciptaan dan penggunaan DT untuk ASV, yang direka untuk meningkatkan pemantauan persekitaran dan kecekapan operasi. DT berfungsi sebagai replika maya ASV, mensimulasikan data masa nyata dan meramalkan keadaan masa depan. Motivasi untuk projek ini datang dari keperluan yang semakin meningkat untuk penyelesaian pemantauan persekitaran yang efisien serta memantau keadaan ASV. Objektif utama kajian ini adalah untuk membangunkan DT yang kukuh untuk ASV dan menilai prestasinya dalam meramalkan kedudukan dan orientasi kenderaan. ASV dan DT telah digunakan dalam persekitaran terkawal dan pelbagai metrik, termasuk kedudukan dan orientasi, direkodkan dan dianalisis. Apabila DT mendapat kemas kini yang kerap, MAE adalah rendah: 1.411 meter untuk jarak, 0.708 darjah untuk sudut olengan, 1.812 darjah untuk sudut menjengkit, dan 18.754 darjah untuk sudut memutar. Walau bagaimanapun, kajian ini mendedahkan bahawa memanjangkan selang antara kemas kini akan mempengaruhi ketepatan ramalan dengan ketara. MAE meningkat sebanyak 0.78% untuk sudut olengan, menurun sebanyak 21.42% untuk sudut menjengkit, meningkat sebanyak 40.18% untuk sudut memutar, dan paling pentingnya, MAE jarak meningkat sebanyak 89.23%. Keputusan kajian menunjukkan bahawa kemas kini data yang kerap dan konsisten adalah penting untuk mengekalkan ketepatan ramalan yang tinggi. Kesimpulannya, pembangunan dan penggunaan DT untuk ASV dalam kajian ini menunjukkan potensinya untuk meningkatkan pemantauan persekitaran dan kecekapan operasi.

Kata kunci: Kembar Digital, Kapal Permukaan Autonomi, Integrasi Data masa Nyata, Pemodelan Ramalan

1.0 INTRODUCTION

Autonomous Surface Vehicle (ASV) is an unmanned, self-propelled vehicle that operates on the water's surface. ASVs rely on multiple sensors to navigate, avoid obstacles and monitor environmental conditions [1]. The deployment of ASVs has greatly streamlined activities like water sampling, underwater mapping, and natural hazard assessment, enhancing both accuracy and efficiency while reducing the need for human resources.

Various studies have propelled the evolution of ASVs, showcasing their design, capabilities, and applications across different aquatic environments. Ferri et al. [2] developed the HydroNet ASV for real-time coastal water quality monitoring. This small catamaran is equipped to measure hydrocarbon and heavy metal concentrations onboard, a feature not previously available in ASVs. Similarly, Mat Idris et al. [3] designed an ASV for monitoring inland water depths, equipped with an echo sounder and GPS for effective data collection and autonomous navigation.

The development of ASVs like Xiroi II, discussed by Martorell-Torres et al. [4], demonstrates advancements in marine task coordination, integrating an open-source Robot Operating System (ROS)-based control architecture for enhanced navigation. Wang et al. [5] introduced an ASV project focused on oil slick monitoring, utilizing specialized sampling systems to improve spatial and temporal coverage of oil spill monitoring.

Dun and Grinham [6] designed a solar-powered ASV for environmental monitoring, capable of navigating complex inland water reservoirs to measure water quality and greenhouse gas emissions. This ASV's continuous operation and integration with a floating sensor network enhance environmental monitoring efficiency. Another notable example is the Lake Environmental Data Harvester (LED) project, which utilizes ASVs for comprehensive monitoring of remote alpine lakes [7]. The LED ASV can conduct high-resolution mapping, measure various environmental parameters, and ensure real-time data collection through Internet of Things (IoT) infrastructure.

Additionally, the University of New Hampshire developed a robust, modular control system for ASVs, achieving basic autonomous navigation and preparing for advanced stages like obstacle avoidance and target tracking [8]. Carlson et al. [9] demonstrated the use of ASVs in obtaining bathymetry and ocean current measurements in hazardous environments like Greenland's rocky coastlines, icebergs, and marine-terminating glaciers, where manned vehicles face significant safety risks.

ASVs rely on various sensors for autonomous navigation and data collection. GPS sensors provide precise location data, enabling accurate navigation and obstacle avoidance [6], [10]. Inertial Measurement Units (IMUs) measure the orientation, allowing the ASV's control system to maintain the desired positioning[8]. The combination of these sensors allows ASVs to perform complex tasks autonomously, with the IMU providing the ship's heading and the GPS sensor determining the ASV's location.

In the context of autonomous systems, a Digital Twin (DT) is a highly accurate virtual model of a physical system, such as an autonomous vehicle or robotic system [11]. As autonomous systems become more complex and interconnected, scaling the DT framework to accommodate multiple systems and components, while maintaining accuracy and performance, becomes increasingly difficult. It uses real-time data from diverse array of sensors, control systems, and external environments to replicate the state and behavior of the physical unmanned system. The main problem or issue within the DT framework for autonomous systems lies in the complexity of ensuring continuous, real-time integration of this data to maintain an accurate and responsive representation. This accuracy is crucial for the DT to effectively predict the physical unmanned system's future states, optimize its performance, monitor its condition, and make informed decisions autonomously.

Wei et al. [13] developed a DT framework for real-time monitoring and analysis of ship structure deformation fields, integrating physical strain data with advanced modeling techniques for continuous monitoring and immediate feedback on structural integrity. Perabo et al. [14] demonstrated digital twin modeling of ship power and propulsion systems, enabling simulation, optimization, and predictive maintenance by identifying wear and tear early through continuous performance monitoring.

Major et al. [15] showcased the application of DTs in maritime operations, integrating various ship systems for remote monitoring and crew assistance. Raza et al. [16] developed an innovative framework for ASV development using integrated DTs within a 3D simulation environment, enhancing design and testing processes. Meanwhile, Hasan et al. [17] studied the DT framework which focusing on fault prediction and maintenance to improve reliability and efficiency of the physical system. .

Lee et al. [18] presents the development and application of a real-time digital twin for ship operations in waves. The digital twin integrates physical and virtual models, enabling real-time predictions of ocean waves and hydrodynamic performances such as seakeeping and maneuvering. The real-time digital twin system offers significant advancements in ship operation by enabling short-term, accurate predictions of ocean conditions and ship responses.

Advancements in ASV and DT technologies underscore their transformative impact on marine operations. The goal of this project is to create a DT model of an ASV, use real-time data to improve the simulation, provide realistic visualizations of the ASV's operations and surroundings, and set up efficient communication between the ASV and its DT. This study will also validate and test the DT's performance to ensure its effectiveness.

2.0 METHODOLOGY

The overall view of the Digital Twin (DT) architecture is illustrated in Figure 1. The DT is divided into three main components.

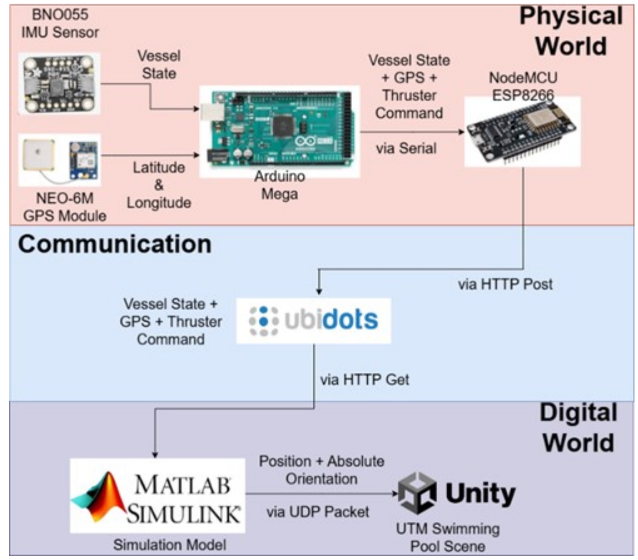


Figure 1 Overall Digital Twin Framework

The first part is the physical world, which comprises the ASV along with its array of sensors and processor units. The ASV is equipped with multiple sensors to measure and track several environmental parameters. These sensors include GPS for positioning, IMU sensor for measuring orientation and movements. The processor units on the ASV are responsible for initial data processing and ensuring that the data is accurate before it is transmitted to the next component. The second part is the communication system layer, which transfer the information between the physical world and the digital world. This bridge is essential for maintaining a continuous and seamless flow of data. The final component is the digital world, which houses the DT. This virtual representation of the ASV uses the sensor data transmitted from the physical world to simulate and predict the next state of the ASV.

2.1 Simulation Parameters

The project begins with the tuning of a physics-based model of an ASV using MATLAB Simulink. This model was modified based on an open-source Simulink model [19]. The model requires input of various physical parameters of the ASV, including thruster speed and ASV's orientation, to accurately simulate the ASV's behavior.
The simulation model functions by first calculating the total force acting on the ASV. Then, it calculates the

linear and angular acceleration of the ASV. By integrating the linear and angular accelerations, the model derives the linear and angular velocities of the ASV. Subsequent integration of the velocities yields the linear and angular positions of the ASV. Additionally, the model also provides the ASV's latitude and longitude as output. The calculated ASV states are then continuously fed back into the model, enabling continuous prediction and adjustment of the ASV's state in each iteration.

Table 1 shows the key values for the parameters to be input into the model. This table includes various parameters that define the ASV's dimensions, mass distribution, and hydrodynamic characteristics. Most of the data were extracted from the previous study due to the utilization of the same ASV in the study [20]. The moment of the ship is calculated using the ship's dimensions and the propeller's placement relative to the center of gravity.

Furthermore, we have adjusted the model for the thrust force and moment to include our own thruster values, ensuring accurate representation of our ASV's propulsion characteristics. The input for the thruster is Pulse Width Modulation (PWM) signals, based on the specifications of the T200 Thruster, as detailed in its datasheet. The thrust force generated by the thruster is determined by inputting the voltage and PWM data into a lookup table derived from the datasheet.

2.2 Visualization with Unity

To create an accurate digital twin, the virtual world must closely resemble the physical one. By using Unity and Cesium, we can access the 3D tiles and terrains associated with the real world. Using Cesium, Google Photorealistic 3D Tiles are incorporated into the virtual environment with high precision and realism.

Table 1 Key Parameters of ASV

Parameter	Value	Description
L	$1.537\,m$	Overall length of the ASV
B	$0.615\,m$	Beam of the ASV
m	$16.17\,kg$	Mass of the ASV
rg	$[-0.29, 0, -0.11]m$	Center of gravity (CG) coordinates for the hull
$R44$	$0.24\,m$	Radius of gyration in the roll direction (44)
$R55$	$0.39\,m$	Radius of gyration in the pitch direction (55)
$R66$	$0.40\,m$	Radius of gyration in the yaw direction (66)
U_{max}	$1.5\,m/s$	Maximum forward speed of the ASV.
L_h	$0.51\,m$	Distance between the centerlines of the hulls in a multi-hull configuration.
B_{pont}	$0.10\,m$	Beam of one pontoon
y_{pont}	$0.39\,m$	Distance from the centerline to the waterline area center of one pontoon.
Cw_{pont}	0.396	Waterline area coefficient for one pontoon, representing the ratio of the waterline area to the product of length and beam.
Cb_{pont}	0.575	Block coefficient for one pontoon, representing the ratio of the volume of the submerged portion to the product of its length, beam, and draft.
l_1	$-0.12511\,m$	Lever arm distance for the left propeller
l_2	$0.12511\,m$	Lever arm distance for the right propeller
LCF	$-0.176\,m$	Longitudinal Center of Flotation, indicating the point along the length of the ASV where it is balanced in the longitudinal direction.
Aw_{pont}	$0.0609\,m^2$	Waterline area of one pontoon, calculated by multiplying the waterline area coefficient by the length and beam of the pontoon.
I_T	$0.0450\,m^4$	Total transverse second moment of area, representing the resistance to transverse bending
I_L	$0.0968\,m^4$	Total longitudinal second moment of area, representing the resistance to longitudinal bending

Furthermore, in Figure 2 showcases the initial 3D CAD model of the ASV, developed in SOLIDWORKS. It highlights the details of the ASV's design, including structural components and thruster placement. The CAD model serves as a foundational step in the digital twinning process, ensuring that every aspect of the ASV's physical structure is accurately represented for further simulation and testing purposes.

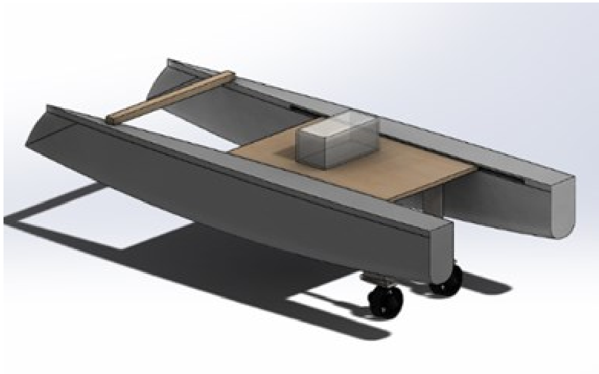


Figure 2 3D CAD Model of ASV inside SOLIDWORKS

Then the CAD model is imported into the Unity scene (shown in Figure 3). To define the ASV state in Unity and move the DT according to the data, we need the linear position of the ASV in XYZ coordinates and the orientation of the ASV. In Simulink, data strings are transmitted to Unity via UDP using a predefined IP address and port number. Upon receiving these values in Unity, the latitude and longitude data are converted to XYZ coordinates.

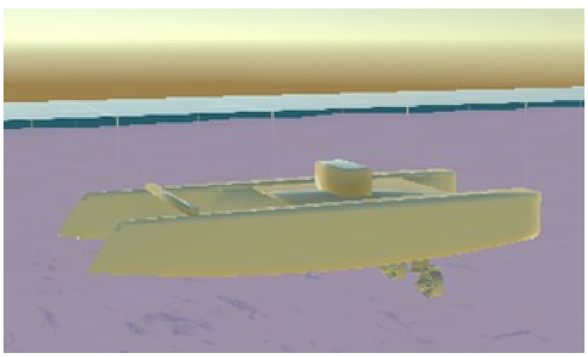


Figure 3 3D Model of ASV Imported into Unity

Additionally, rotation transformations are applied to ensure that the pitch, roll, and yaw are correctly interpreted in Unity, matching the orientation data from Simulink. Through these steps, the ASV's movement and orientation can be accurately visualized within the virtual world, reflecting its simulated behavior in real-time.

2.3 Sensor Integration

The sensor data, along with thruster commands, are processed by an Arduino Mega microcontroller. The data is then transmitted to the cloud for real-time updates via a NodeMCU ESP8266 board. The arrangement of the microcontrollers and sensors are shown in Figure 2. This method was chosen because it minimally alters the original code on the main microcontroller and offloads the computing power and memory of formatting and uploading the data to another microcontroller. This helps maintain the processing speed of the Arduino Mega and provides clarity on the tasks of each microcontroller. The cloud platform used is Ubidots, allowing for data uploads at a rate of up to one upload per second. The sensor data were first organized into JSON format before uploading it.

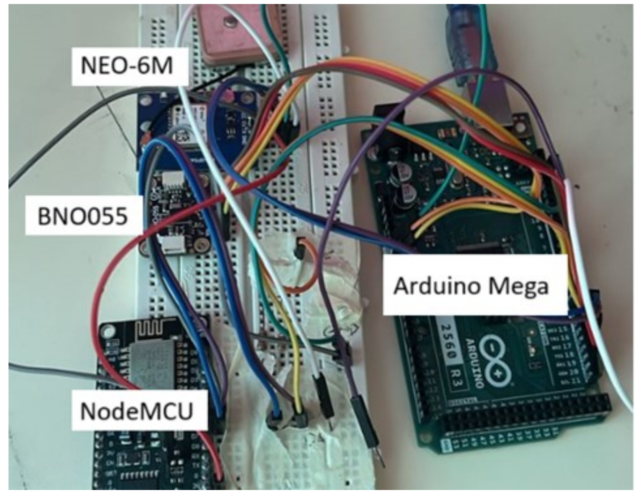


Figure 2 Connection between microcontroller and sensors

Real-time sensor data is used to update the ASV's state, using the BNO055 IMU sensor and NEO-6M GPS module to provide orientation, velocity, acceleration, and position data. The BNO055 sensor is selected to obtain the orientation data of the ASV. This sensor provides a range of data outputs which includes absolute orientation (Euler Angle and Quaternions), angular velocity, linear acceleration and magnetic field strength.

To calculate angular acceleration, we can derive it from the angular velocity by taking the difference between consecutive angular velocity measurements and dividing it by the time interval between them. However, gyroscope data tends to be noisy and fluctuates easily. To address this, we utilize the quaternion values method in [21], which result from sensor fusion and onboard algorithms, making them more stable. By converting the quaternions to Euler angles, we can reduce the angular velocity data's susceptibility to noise and provide more stable data. We can derive angular velocity and angular acceleration using the formula

$$\omega_n = (\theta_n - \theta_{n-1}) / \Delta t \tag{1}$$
$$\alpha_n = (\omega_n - \omega_{n-1}) / \Delta t \tag{2}$$

Where ω is angular velocity, θ is angular is position, α is angular acceleration and Δt is time interval.

2.4 Real Time Model Update

For the digital twin to accurately represent the real-world ASV, it requires constant updates with real-time data. This ensures that the model aligns closely with the actual behavior of the ASV. The model is updated with new data obtained from the Ubidots cloud platform. Angular and linear acceleration, velocity and position integrators are reset with this new data as initial points to ensure accurate tracking of the ASV's movement. This allows the model to adjust and accurately reflect the dynamic conditions affecting the ASV's motion.

The flowchart in Figure 3 represents a dynamic system updating process that involves updating simulation model and predicting states based on available data. The process begins with an initial step of acquiring the initial data set which includes acceleration (a_n), velocity (v_n), position (x_n). Following this, the next state of the system is predicted $(a'_{n+1}, v'_{n+1}, x'_{n+1})$. Then, if real data $(a_{n+1}, v_{n+1}, x_{n+1})$ is available, it will be used to update the model. If real data is not available, the system uses simulation data as the input state to continue the simulation process. This loop repeats, ensuring that the model continually adapts based on the most recent data.

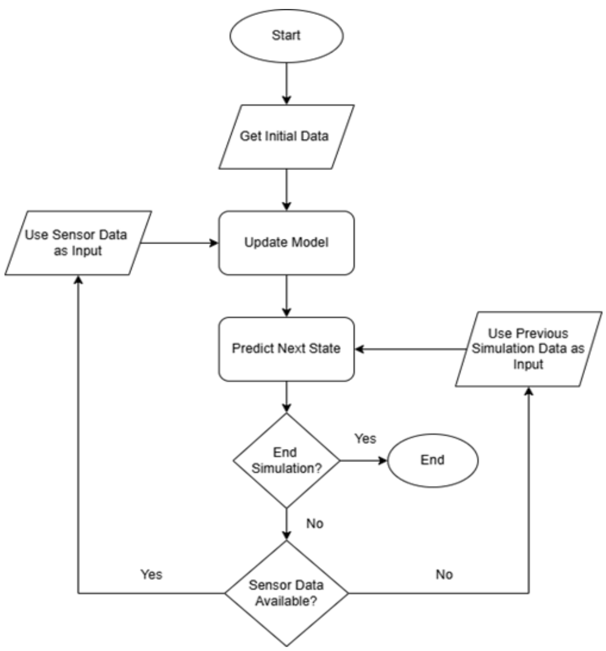


Figure 3 Flowchart of the Model Update System

2.5 Testing and Data Collection

The validation and testing of the DT for the ASV involves experimentation and comparison with real-world data to ensure accuracy and reliability in simulating the ASV's behavior. This section outlines the methodology and findings from testing the DT against the physical ASV.

To begin, the ASV was deployed on a mission with predefined start and end points set using GPS coordinates inside UTM Swimming Pool, which is approximately shown in Figure 4. Point 1, marked with a red dot, is located at the northern end of the pool, while Point 2, indicated by a yellow dot, is positioned towards the southern end. The ASV starts from the right bottom side of the pool and need to move itself to Point 1, wait a few seconds and then move to Point 2. These waypoints are selected to test the ASV's navigation and manoeuvrability within the controlled environment. During this mission, the ASV navigated autonomously towards the endpoint, continuously publishing sensor data to the cloud via mobile data whenever a connection was available.

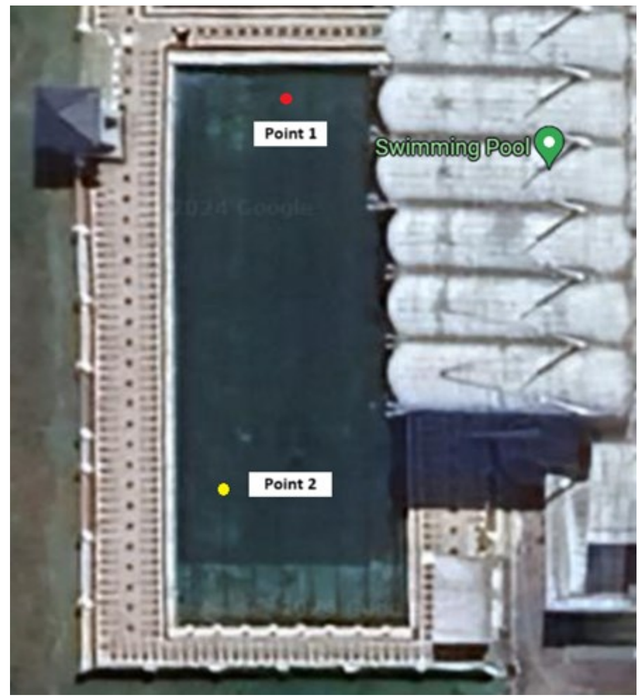


Figure 4 Preset Waypoints (from Google Maps)

The cloud-stored data enabled the DT to operate in two modes. The first mode, which is the Real-time Simulation, involved running the DT concurrently with the ASV using live data feeds. This real-time operation tested the DT's ability to accurately reflect the ASV's current state and behaviours. The second mode, Time-delayed Simulation, utilized previously uploaded data to simulate real-time conditions with a time delay. This approach allowed for thorough validation of the DT's capabilities without the need for constant outdoor testing, ensuring that the DT could replicate real-time operations accurately.

With the digital twin running, the accuracy of the model was determined by calculating the error between the predicted position and orientation of the ASV and the actual values. The error was computed from the value updated and the predicted value at the same timeframe. The position error between the predicted latitude and longitude can be determined using the following equation which is based on the Haversine Formula:

$$d = 2r \arcsin(\sqrt{\sin^2\frac{\Delta\phi}{2} + \cos\phi_1 \cos\phi_2 \sin^2\frac{\Delta\lambda}{2}}) \tag{3}$$

Where d is the distance between the two points (along the surface of the sphere), r is the radius of the Earth (mean radius = 6,371 km), ϕ_1 and ϕ_2 are the latitudes of point 1 and point 2 in radians, $\Delta\phi$ is the difference between the latitudes of the two points (in radians), $\Delta\lambda$ is the difference between the longitudes of the two points (in radians).

Then, the total error throughout the simulation can be determined for position (latitude and longitude) and orientation (pitch, roll and yaw) by finding the Mean Absolute Error (MAE) via the equation below:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |Predicted\ Value_i - Sensor\ Value_i| \quad (4)$$

Where n is the total number of data points.

For comparison purposes, the data file was modified to remove some of the data points to simulate a scenario where the time interval for each update was more than 2 seconds. The error was then calculated using the same method as above to analyze the impact of reduced update frequency on the digital twin's accuracy. By conducting these tests, we could determine the discrepancies between the DT and the real physical system.

3.0 RESULTS AND DISCUSSION

Figure 7 shows the Digital Twin (DT) alongside the Autonomous Surface Vehicle (ASV). In this figure, the green ASV represents the historical path and updates of the positioning, while the red ASV indicates the predicted estimated position when sensor data is not available. The digital twin model is continuously updated with real-time data. As new data is uploaded to the cloud, the ASV's position is refreshed to align with the actual sensor data collected from the physical ASV, ensuring the digital twin accurately mirrors real-world conditions.

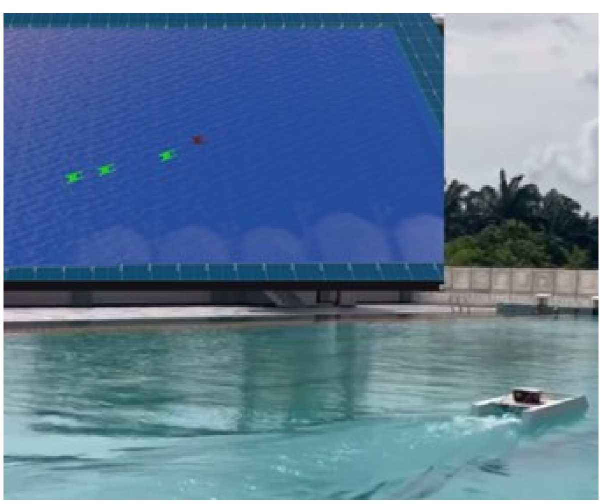


Figure 5 Digital Twin and Real Time Ship View

Figure 8 provides a detailed comparison of the predicted and actual locations of the ASV during deployment, visualized through a series of latitude and longitude coordinates with timestamps indicating the sequence of data points. The close alignment between the predicted (red) and actual (green) paths in the plot indicates that the digital twin model generally performs well in tracking the ASV's movements. However, any noticeable deviations between the two paths highlight instances where the model's predictions diverge from the actual data, which can be attributed to sensor inaccuracies, complex environmental conditions, and limitations in the simulation model's capabilities.

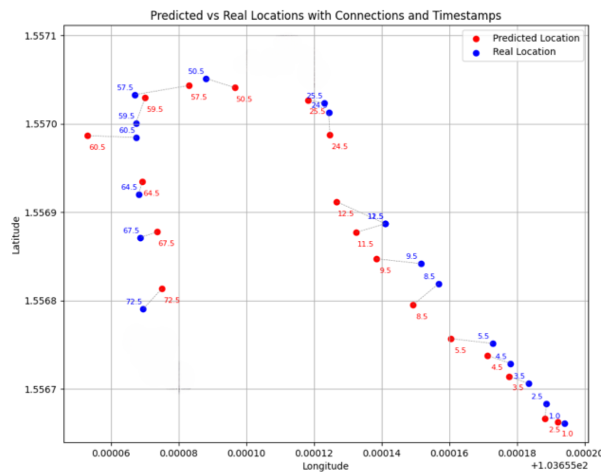


Figure 6 Predicted vs Real Locations with Timestamps

3.1 Digital Twin Error Analysis

Figure 7 illustrates the roll, pitch and yaw angle variation of the ASV throughout the deployment which indicates the tilt of the ASV around its longitudinal axis, the front-to-back tilt of the ASV and its rotation around the vertical axis, respectively. The consistent intervals of red vertical lines across the figures show the sensor data update to the simulation model. The roll angle generally remains stable around 0 degrees, suggesting that the ASV maintained a mostly level orientation inside simulation. Similarly, the pitch angle data indicates that the ASV maintained a relatively stable vertical orientation for most of the deployment. However, the yaw angle data shows significant changes across time. This happen as the ASV have done some directional adjustments during the deployment to move to its designated locations. This demonstrates the ASV's responsiveness and adaptability in real-time, which are crucial for autonomous navigation and maintaining accurate positioning in dynamic environments. These adjustments are critical for autonomous systems to ensure safe and efficient operation in unpredictable maritime environments. This approach is validated by studies such as that of Zhang et al. (2021), which highlights the importance of real-time sensor data updates for maintaining the accuracy of ASV navigation control system. The DT's ability to mirror real-time conditions and predict future states based on sensor data is essential for effective autonomous navigation and collision avoidance in ASVs.

The spike of data that occurs after each update can be attributed to the error between predicted state and sensor data. Also, it can be due to the sensor providing inaccurate values. The latter reason could be the sensors report faulty acceleration or velocity data. For instance, in the 60th second, the roll angle reaches an unreasonable value of -60 degrees, which would suggest that the ASV is capsizing which is unlikely and indicates a sensor error. This highlights the need for error-checking algorithms to filter out erroneous data points and improve the reliability of the sensor readings.

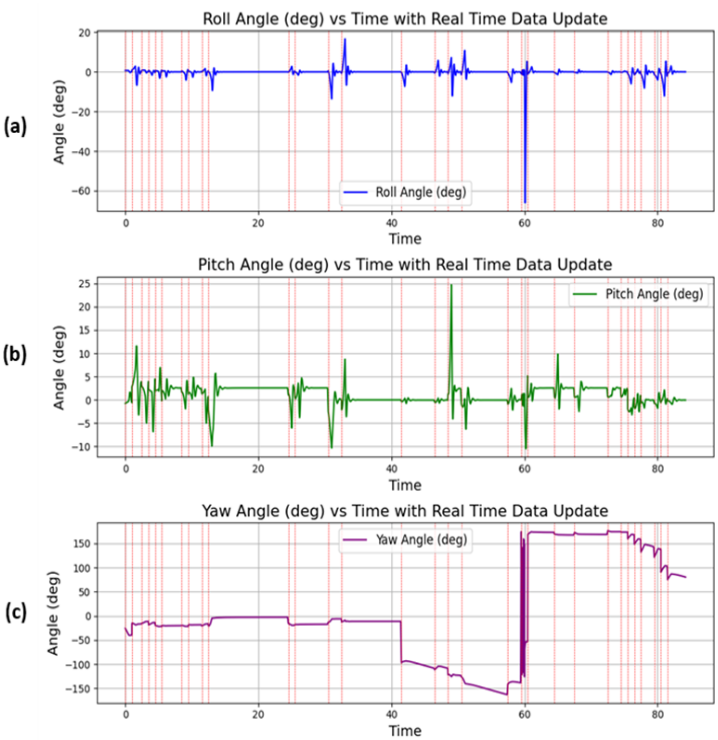


Figure 7 a) Roll, (b) Pitch, and (c) Yaw of ASV with Real-Time Data Updates

Table 2 shows the compiled error metrics used to measure the accuracy of the DT in predicting the ASV future state. The minimum absolute error values for roll, pitch, yaw, and distance demonstrate that the model can achieve high accuracy under certain conditions, with errors as low as 0.100 degrees for roll and 0.083 meters for distance. However, the maximum absolute error values, particularly for yaw (137.956 degrees) and distance (3.208 meters). The Mean Absolute Error (MAE) shows an average error of 0.708 degrees for roll, 1.812 degrees for pitch, 18.754 degrees for yaw, and 1.411 meters for distance. These values indicate that while the model performs reasonably well, there is room for improvement, especially in predicting the yaw angle and positional accuracy. Overall, this analysis underscores the need for enhanced sensor integration and more advanced modeling techniques to better capture the complexities of the ASV's real-world behaviour.

Table 2 Error Metrics for Roll, Pitch, Yaw and Distance

	Roll Angle (deg)	Pitch Angle (deg)	Yaw Angle (deg)	Distance (m)
Minimum Absolute Error	0.100	0.080	0.819	0.083
Maximum Absolute Error	3.150	5.046	137.956	3.208
Mean Absolute Error (MAE)	0.708	1.812	18.754	1.411

3.2 Impact of Update Interval

In this section, we investigate the effect of increasing the update interval on the error metrics by modifying the update intervals to be more than or equal to 2 seconds. This analysis aims to determine how less frequent updates impact the accuracy of the digital twin's predictions.

Figure 8 illustrates the comparison between the predicted and real locations of the ASV with an update interval of more than or equal to 2 seconds.

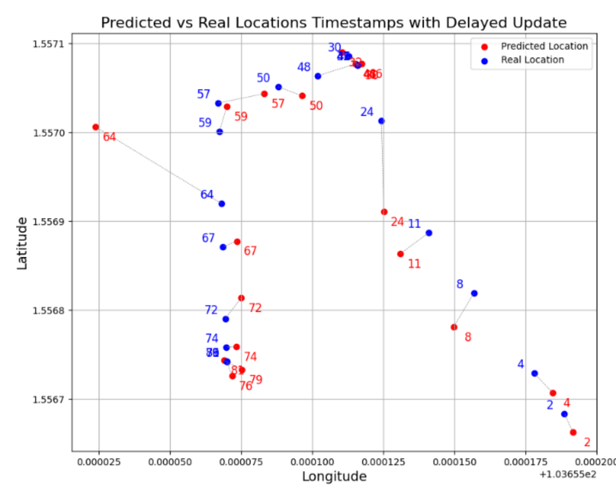


Figure 8 Predicted vs Real Locations with Delayed Update

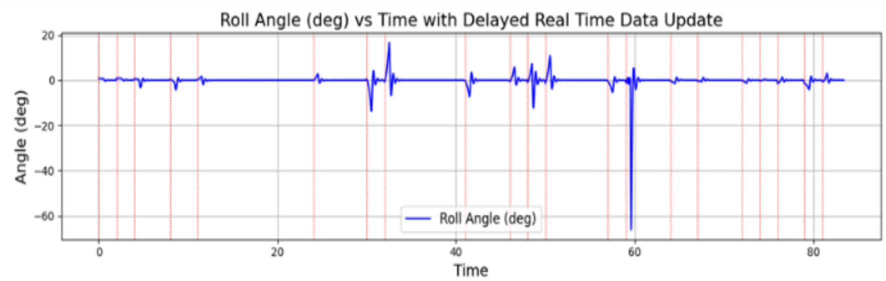
This figure is used to view the overall impact of less frequent data updates on the accuracy of the digital twin's predictions. This figure reveals that less frequent updates lead to increased discrepancies between the predicted and actual ASV locations. This can be seen by the longer distance between the blue points (actual location) and the red points (predicted location) as compared to Figure 8.

Figure 9 shows the plot of roll, pitch and yaw angle throughout the deployment with a longer sensor update interval. Compared to Figure 7, which feature more frequent updates, the angles in these figures exhibit smoother graphs. This smoothing indicates a reduction in the frequency of abrupt changes due to the extended update intervals. However, this apparent smoothness comes at the cost of increased error in accurately tracking the ASV's orientation when the data is not updated frequently.

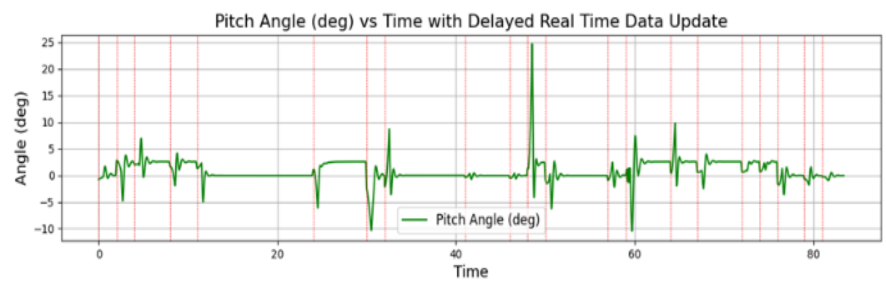
The comparison between figures with different update intervals highlights several important points. Extended update intervals result in smoother graphs for roll, pitch, and yaw angles, indicating fewer apparent fluctuations. However, this smoothing effect compromises accuracy and the ability to detect rapid changes. Although the overall shape and trends of the graphs remain consistent with those seen in the previous scenario. The general behavior of the ASV is captured, but the finer details and rapid adjustments are lost. This might lead to potential inaccuracies in the digital twin's predictions. Additionally, when the model is not updated regularly, the angle and distance errors accumulate over time. Without accurate data to correct these errors promptly, the digital twin cannot accurately represent the actual state or predict a state within certain tolerances.

Table 3 shows the same metrics to determine the accuracy of DT in Table 2 and presents the highlights the percentage changes for roll, pitch, yaw angles, and distance when the sensor update interval is extended. For the roll angle, the maximum absolute error decreases by 33.34% to 2.100 degrees, while the MAE slightly increases by 0.78% to 0.714 degrees. For the pitch angle, the maximum absolute error reduces by 17.77% to 4.149 degrees, and the MAE decreases by 21.42% to 1.424 degrees. For the yaw angle, the minimum absolute error drops by 5.69% to 0.773 degrees, but the MAE significantly increases by 40.18% to 26.289 degrees. Regarding distance, the maximum absolute error dramatically rises by 254.34% to 11.369 meters, and the MAE nearly doubles with an 89.23% increase to 2.670 meters.

(a)



(b)



(c)

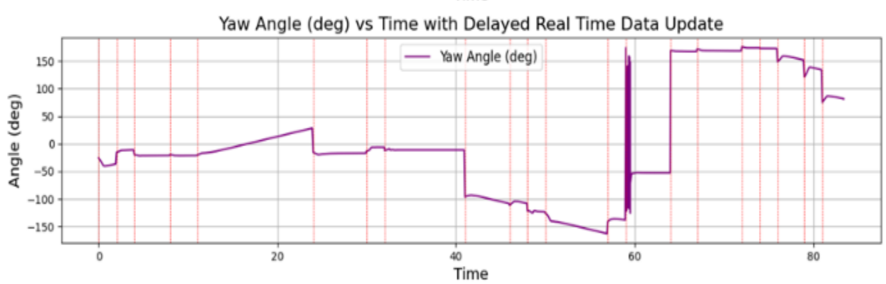


Figure 9 a) Roll, (b) Pitch, and (c) Yaw of ASV with Delayed Real-Time Data Updates

Table 3 Combined Error Metrics and Percentage Changes for Delayed Sensor Update

	Roll Angle (deg)	Pitch Angle (deg)	Yaw Angle (deg)	Distance (m)
Minimum Absolute Error	0.100 (0.00%)	0.080 (0.01%)	0.773 (-5.69%)	0.101 (20.89%)
Maximum Absolute Error	2.100 (-33.34%)	4.149 (-17.77%)	138.098 (0.10%)	11.369 (254.34%)
Mean Absolute Error (MAE)	0.714 (0.78%)	1.424 (-21.42%)	26.289 (40.18%)	2.670 (89.23%)

These metrics highlight the critical importance of frequent, accurate data updates for maintaining the fidelity of the digital twin. While the model shows the roll and pitch angles with minimal changes in error metrics, the yaw angle and distance metrics are significantly affected by delayed sensor updates. The substantial increase in maximum and mean absolute errors for yaw and distance indicates that the model struggles to accurately predict these parameters when data updates are less frequent. This suggests a need for optimizing update intervals to ensure the digital twin can reliably predict ASV's state in dynamic environments.

4.0 CONCLUSION

The novel contribution of this study lies in the development and implementation of a Digital Twin (DT) specifically tailored for an Autonomous Surface Vehicle (ASV), with a focus on real-time data integration and the impact of update intervals on predictive accuracy. The study provides new insights into how the frequency of data updates affects the DT's ability to accurately replicate and predict the ASV's behavior in dynamic environments. By integrating real-time data from the ASV's sensors, the DT provided accurate simulations and predictions of the ASV's state, validating its effectiveness in a controlled environment. The results indicated that the positional errors averaged 1.411 meters, demonstrating a degree of accuracy in the DT's ability to mirror the real-world positioning of the ASV. The performance of the Digital Twin (DT) for the Autonomous Surface Vehicle (ASV) was evaluated against key metrics such as positional accuracy and orientation stability. The DT achieved a mean absolute error (MAE) of 1.411 meters for positional accuracy and within 0.708 degrees for roll, 1.812 degrees for pitch, and 18.754 degrees for yaw. These results are comparable from similar studies, such as Zhang et al. (2021), which also emphasize the importance of frequent data updates for maintaining high accuracy in DT systems. Future work should aim to refine these metrics further and explore additional studies from the latest advancements in DT applications for autonomous systems.

Acknowledgement

N.S.M.N thanked Kurita Asia Research Grant (grant number: 23Pmy133, cost centre number: R.J130000.7324.1U023, provided by Kurita Water and Environment Foundation (KWEF). K.I thanked UTM Fundamental Research Grant with cost centre number: Q.J130000.3824.22H50. A.S.A.K thanked Kurita Asia Research Grant (grant number: 23Pmy005, cost centre number: R.J130000.7354.1U025, provided by Kurita Water and Environment Foundation (KWEF).

Conflicts of Interest

The author(s) declare(s) that there is no conflict of interest regarding the publication of this paper.

References

[1] E. Zereik, M. Bibuli, N. Mišković, P. Ridao, and A. Pascoal. 2018. Challenges and Future Trends in Marine Robotics. *Annual Reviews in Control*. 46: 350–368. Doi: 10.1016/j.arcontrol.2018.10.002.

[2] G. Ferri, A. Manzi, F. Fornai, F. Ciuchi, and C. Laschi. 2015. The HydroNet ASV, a Small-sized Autonomous Catamaran for Real-time Monitoring of Water Quality: From Design to Missions at Sea. *IEEE Journal of Oceanic Engineering*. 40(3): 710–726. Doi: 10.1109/joe.2014.2359361.

[3] M. H. Bin Mat Idris, M. Al Azhari Bin Che Kamarudin, M. I. Sahalan, Z. Bin Zainal Abidin, and M. M. Rashid. 2016. Design and Development of an Autonomous Surface Vessel for Inland Water Depth Monitoring. *2016 International Conference on Computer and Communication Engineering (ICCCE)*. Accessed: Jun. 29, 2024. [Online]. http://dx.doi.org/10.1109/iccce.2016.47

[4] A. Martorell-Torres, E. Guerrero-Font, J. Guerrero-Sastre, and G. Oliver-Codina. 2022. Xiroi II, an Evolved ASV Platform for Marine Multirobot Operations. *Sensors*. 23(1): 109. Doi: 10.3390/s23010109.

[5] D. T. Wang, W. P. Meurer, T. N. Nguyen, G. W. Shipman, and D. Koenig. 2021. Preservation of Oil Slick Samples on Adsorbent Teflon Fabric: Potential for Deployment Aboard Autonomous Surface Vessels. *Marine Pollution Bulletin*. 169: 112460. Doi: 10.1016/j.marpolbul.2021.112460.

[6] M. Dun and A. Grinham. 2010. Experimental Evaluation of an Autonomous Surface Vehicle for water quality and Greenhouse Gas Emission Monitoring. *2010 IEEE International Conference on Robotics and Automation*. Accessed: Jun. 29, 2024. [Online]. http://dx.doi.org/10.1109/robot.2010.5509187.

[7] A. Odetti *et al.* 2024. Lake Environmental Data Harvester (LED) for Alpine Lake Monitoring with Autonomous Surface Vehicles (ASVs). *Remote Sensing*. 16(11): 1998. Doi: 10.3390/rs16111998.

[8] C. Gagnon, A. Roemer, R. Mitchell, S. Hurley, J. Hensel, and X. Mcnulty. 2017. Autonomous Surface Vehicle (Asv). https://repository.library.noaa.gov/view/noaa/46264.

[9] D. F. Carlson *et al.* 2019. An Affordable and Portable Autonomous Surface Vehicle with Obstacle Avoidance for Coastal Ocean Monitoring. *HardwareX*. 5: e00059. Doi: 10.1016/j.ohx.2019.e00059.

[10] T. Ren, D. Wang, and Y. Zou. 2023. Autonomous Catamaran Unmanned Surface Vessel Based on GPS And Radar. Accessed: Jun. 29, 2024. [Online]. http://dx.doi.org/10.20944/preprints202305.0998.v1

[11] A. Rasheed, O. San, and T. Kvamsdal. 2020. Digital Twin: Values, Challenges and Enablers from a Modeling Perspective. *IEEE Access*. 8: 21980–22012. Doi: 10.1109/access.2020.2970143.

[12] L. Wright and S. Davidson. 2020. How to Tell the Difference between a Model and a Digital Twin. *Advanced Modeling and Simulation in Engineering Sciences*. 7(1). Doi: 10.1186/s40323-020-00147-4.

[13] P. Wei, C. Li, Z. Jiang, and D. Wang. 2024. Real-Time Digital Twin of Ship Structure Deformation Field Based on the Inverse Finite Element Method. *Journal of Marine Science and Engineering*. 12(2): 257. Doi: 10.3390/jmse12020257.

[14] F. Perabo, D. Park, M. K. Zadeh, O. Smogeli, and L. Jamt. 2020. Digital Twin Modelling of Ship Power and Propulsion Systems: Application of the Open Simulation Platform (OSP). *2020 IEEE 29th International Symposium on Industrial Electronics (ISIE)*. Accessed: Jun. 30, 2024. [Online]. http://dx.doi.org/10.1109/isie45063.2020.9152218.

[15] P. Major, G. Li, H. Zhang, and H. P. Hildre. 2021. Real-Time Digital Twin of Research Vessel for Remote Monitoring. *ECMS 2021 Proceedings edited by Khalid Al-Begain, Mauro Iacono, Lelio Campanile, Andrzej Bargiela*. Accessed: Jun. 29, 2024. [Online]. http://dx.doi.org/10.7148/2021-0159.

[16] M. Raza, H. Prokopova, S. Huseynzade, S. Azimi, and S. Lafond. 2022. Towards Integrated Digital-Twins: An Application Framework for Autonomous Maritime Surface Vessel Development. *Journal of Marine Science and Engineering*. 10(10): 1469. Doi: 10.3390/jmse10101469.

[17] A. Hasan, A. Widyotriatmo, E. Fagerhaug, and O. Osen. 2023. Predictive Digital Twins for Autonomous Surface Vessels. *Ocean Engineering*. 288: 116046. Doi: 10.1016/j.oceaneng.2023.116046.

[18] J.-H. Lee, Y.-S. Nam, Y. Kim, Y. Liu, J. Lee, and H. Yang. 2022. Real-time Digital Twin for Ship Operation in Waves. *Ocean Engineering*. 266: 112867. Doi: 10.1016/j.oceaneng.2022.112867.

[19] Junxun Luo. 2024. Simulate Navigation Algorithms of An ASV (https://github.com/MartinJXLuo/Simulate-Navigation-Algorithms-of-An-Autonomous-Surface-Vessel-ASV-/releases/tag/v1.0.0), GitHub. Retrieved June 29, 2024.

[20] Menchon, M. E. A. 2023. Design and Fabrication of an Unmanned Surface Vehicle (USV) for Malaysian Sea. Undergraduate Final Year Project Report. Faculty of Mechanical Engineering, Universiti Teknologi Malaysia.

[21] L. Zhang, S. Li, X. Wu, and Y. Chen. 2021.Gyro Denoising and Fusion Quaternion Attitude Estimation Based on Advanced Filtering Techniques. *IEEE Sensors Journal*. 20(5): 2571–2579. Doi: 10.1109/JSEN.2020.2994177.