# CHAPTER 6

# Visual pattern recognition in navigation simulator interfaces: a method for automatic reconstruction of vessel motion parameters

Yevgeniy Kalinichenko
Oleksandr Koliesnik
Oleg Safyan

## Abstract

In this chapter, it is presented a method for automatically determining the navigation and control parameters of a vessel, which is based on reading the simulator screen without access to internal telemetry or system data. The presented approach considers the display of the simulator not only as an interaction tool for the operator, but also as a visual representation of the state of the internal model, because the navigation parameters are coded using graphic indicators that create a graphical user interface, which methodologically can be understood as a structured visual environment where stable areas correspond to certain parameters.

By automatically identifying these areas and interpreting their contents, the method determines synchronized time series that describe the vessel's movement. Once set up, the system can read numerical and graphical indicators and convert them into data sequences up to 10 Hz, allowing quantitative information on vessel movements to be obtained directly from the user interface image without software integration or access to internal modules.

The experimental validation was performed on a fully functional bridge simulator which user interface displays all navigation and engineering parameters, and its results showed that the most important navigation parameters can be reconstructed very reliably using only the interface image, while the reconstructed time series supports the correct sequence of events. The average absolute error is a maximum of 1° for the course and remains below 2% for propeller revolutions, which is sufficient for maneuver analysis and trajectory studies.

A key advantage of this approach is its independence from the internal architecture of specific simulation systems. This allows the method to be applied on various platforms, even where access to internal data is restricted or unavailable.

DOI: 10.21303/978-9908-845-05-0.ch6

The method is currently protected by two patent applications filed with the National Intellectual Property and Innovation Agency of Ukraine.

Overall, the results demonstrate that the visual interface of a navigation simulator can serve as an independent and universal data source for investigating ship dynamics and supporting experimental studies.

**Keywords**

Image recognition, navigation simulator, interface analysis, ship motion parameters, automated data collection, ship dynamics, bridge simulator.

## 6.1 Introduction

Over the past decades, digital simulation technologies have significantly expanded the role of navigational simulators in maritime research and training. Modern full-mission bridge simulators reproduce not only the visual layout of a ship's wheelhouse, but also the operational logic of navigation systems and ship control. As a result, they provide a controlled environment in which ship dynamics, environmental influences, and operator decisions can be studied in conditions that are as close as possible to real navigational practice. At the same time, a practical difficulty appears when such studies require access to reliable navigation data, especially when internal simulator outputs are limited or unavailable.

In this chapter, this problem is addressed by proposing a method for extracting navigation and control parameters directly from the visual interface of the simulator. The goal is to develop an approach that makes it possible to reconstruct vessel motion without relying on internal telemetry, using pattern recognition methods applied to elements of the graphical interface. Despite these advantages, obtaining experimental data from simulators remains a technically challenging task.

In many cases, researchers have to rely on internal telemetry channels, proprietary service logs, or specialized application programming interfaces to access parameters describing the ship's motion. Such solutions often depend on the specific architecture of the simulator software. As a result, experimental procedures developed for one system cannot always be reproduced on another platform. When the internal structure of the simulator is closed or poorly documented, the parameters necessary for analysis may remain unavailable to the researcher.

The general theme of this monograph, pattern recognition in surveillance and diagnostic systems, suggests a different approach to solving this problem. What matters here is not the recognition itself. It is how the simulator is treated as a data source. The graphical interface of a navigation simulator continuously displays a large amount of operational information about the state of the vessel. From a methodological point of view, it can be considered not only as a tool for interaction with a person, but also as a visual representation of the internal state of the model.

In other words, the simulator screen can be considered as a structured visual environment in which navigation parameters are encoded using graphic indicators.

Analysis of these indicators using pattern recognition methods allows obtaining quantitative information about the movement of the vessel without direct access to the internal software modules of the simulator.

The method described in this chapter is based on this idea.

Instead of using internal telemetry streams, the proposed approach analyzes the graphical interface of the simulator itself. The display is considered as an external projection of the internal state of the simulator, and the graphic elements that appear on the screen are interpreted as a representation of navigation and control parameters.

By automatically identifying these elements, the method reconstructs synchronized time series describing the movement of the simulated vessel.

The experimental verification of the approach was carried out using a full-mission bridge simulator.

The technical solution presented here is currently protected by two pending patent applications filed with the Ukrainian National Office for Intellectual Property and Innovation (UANIPIO): application for invention No. a202600852 and application for utility model No. u202600851.

The rest of the chapter is organized as follows. Section 6.2 reviews previous research on marine simulators and visual pattern recognition methods. Section 6.3 examines the navigation simulator interface as a structured visual environment and outlines the conceptual framework of the proposed approach. Section 6.4 describes the architecture of the method and its principles of operation. Section 6.5 presents the results of the experimental verification. The final section summarizes the main results of the study.

## 6.2 Related works

Over the past twenty years, research on navigation simulators has been significantly intensified. This is due to the fact that simulators have begun to play an important role not only in training but also in scientific research into ship control.

Z. Munim et al. show in their systematic review that simulator-based research is becoming more methodologically structured, especially in terms of scenario evelopment, data measurement and analysis [1]. But their work focuses mainly on how such research is conducted, rather than how navigational data can be obtained independently of the specific architecture of the simulator.

C. Sellberg considers simulators primarily as a teaching and evaluation tool and clearly demonstrates their value for reproducing complex navigational situations [2]. However, the simulator is considered primarily as a pedagogical environment, while the movement of the vessel itself is not distinguished as an independent object of quantitative analysis.

H. Tasher et al. investigate marine simulators from technical, training and organizational perspectives and point to the increasing complexity of their systems [3]. This paper is useful for understanding a broader evaluation system, but it also proceeds from the assumption of internal simulator data availability, and does not address alternative ways of obtaining navigation parameters without direct access to the system.

Modern simulators reproduce the operation of the ship's bridge. They combine the movement of the vessel, the influence of the environment and the actions of the boatmaster. Thanks to this, simulators gradually became not just educational tools, but full-fledged experimental sites where navigation processes can be studied in controlled conditions.

Usually, such research is focused on three directions: creation and evaluation of the simulators themselves, analysis of the behavior of ship drivers and modeling of ship movement.

There are many works devoted to the use of simulators in education and science. Reviews show that simulators are widely used to study decision-making, maneuvering, and situation assessment in difficult environments.

These reviews also show a common limitation, namely that the issue of obtaining data usually remains within the simulator environment itself and is rarely considered as a separate methodological problem.

The advantage of simulators is that they allow to safely reproduce rare or dangerous situations. This makes it possible to obtain data that is difficult or impossible to collect in a real voyage.

At the same time, such studies have their own difficulties. It is necessary to correctly create scenarios, determine what exactly to measure, and correctly interpret the received data.

Another very important difficulty in comparative or repeated studies is the dependence on the internal outputs of the simulator, its service logs or its own interfaces, which prevents even well-planned experiments from being transferred from one platform to another.

In most cases, the simulator is seen as an environment where the behavior of the vessel can be observed in different conditions. In parallel, the behavior of the boatmaster is also studied: how it perceives information, assesses the situation and makes decisions during its change. In recent years, special attention has been paid to the so-called non-technical skills, the level of cognitive load and the peculiarities of the distribution of visual attention.

C. Hetherington et al. emphasize the importance of the human element in maritime safety and show that decision-making and cognitive factors often determine the outcome of navigational situations [4]. At the same time, their work does not consider the problem of obtaining quantitative data on vessel motion.

F. Saeed et al. propose a method for assessing non-technical skills in a bridge simulator using evidential reasoning [5]. Their approach allows evaluating operator performance, but the vessel itself is treated mainly as a background for such assessment.

O. Atik and O. Arslan use eye-tracking methods to analyze how operators interact with navigation information [6]. This makes it possible to study visual attention in detail, although the parameters of vessel motion are not analyzed independently.

V. Ronca et al. apply neurophysiological monitoring to assess operator state during emergency training in a full-mission simulator [7]. Their results give additional insight into cognitive load, but again focus on the operator rather than on vessel dynamics.

To study these aspects, various methods of observing the operator's activities are used. The most common are eye tracking systems, behavioral coding methods, as well as neurophysiological monitoring, which allows assessing the operator's state while performing navigation tasks in a simulation environment. Such approaches provide valuable information about the role of the human element in the process of controlling a vessel. However, in most such studies, the dynamics of the vessel itself are secondary. In most cases, vessel motion is considered mainly as a consequence of operator decisions, while the possibility of analyzing movement parameters as an independent object of quantitative study is addressed much less often.

Another notable line of research is related to the mathematical and physical description of ship maneuvering. The main attention in such works is paid to the development and refinement of hydrodynamic models that allow describing the motion of a ship under the influence of external factors – wind, waves and currents. As a rule, it is about constructing systems of differential equations that reflect the interaction between the engine, steering control and external forces acting on the hull.

After formalizing the model, its parameters are refined taking into account experimental measurements or data obtained during modeling. Such models are widely used to analyze the maneuvering characteristics of a ship, predict its trajectory and develop algorithms for automatic motion control.

For example, V. Frett et al. analyze maneuvering data from simulators and show that it can be used to study ship dynamics, but they rely completely on data recorded inside the simulator [8].

H. Li et al. look at uncertainty in maneuver simulations and highlight the need for accurate parameters, although their approach still assumes access to model or simulator data [9].

P. Pires da Silva studies how sensitive maneuvering models are to changes in parameters, showing how much results can vary. This helps to understand the models better, but does not explain how these parameters can be obtained in practice [10].

H. He et al. describe a simulator used for testing automatic control algorithms, where the system is treated as closed and data can only be accessed through internal tools [11].

H. Yasukawa and Y. Yoshimura present the MMG method, which is widely used to predict ship maneuvering. It provides a clear modeling framework, but assumes that all required parameters are already known or measured [12].

J. Lee et al. propose an improved model that includes wave effects using a two-time-scale approach. This gives a more realistic description, but still depends on predefined input data [13].

D. Kim et al. develop a CFD-based model for maneuvering in currents. It provides detailed results, but requires high computational effort and carefully prepared input data [14].

At the same time, the source of data for such studies is most often either the mathematical models themselves or the internal telemetry of navigation simulators. As a result, the task of reconstructing navigation parameters from external observations without referring to the internal data of the system, as a rule, remains beyond the scope of these works.

In practice, navigation data is usually taken straight from the simulator software. This is done through logs, telemetry, or built-in export tools that record parameters like course, speed, and control actions. The data is accurate, but it creates a dependency.

Once a study uses internal simulator data, it becomes tied to that specific system. In practice, this is not always convenient. Access can be limited, poorly documented, or different from one simulator to another. Because of this, even a good experiment can be hard to repeat on another platform. Often, extra integration or custom tools are needed.

In other fields, things are handled differently. Computer vision is already used to read data from screens in industrial monitoring, diagnostics, and control systems.

For example, R. Smith describes the Tesseract OCR engine, which can read text directly from images and is widely used to extract data from visual interfaces [15]. This shows that even simple visual elements can be turned into structured data without internal access.

R. Szeliski gives a general overview of computer vision methods, including how to detect objects, recognize them, and track them in images [16]. These ideas can be applied to reading interface elements on a screen.

T. Wuest et al. show that in many industrial systems, data is taken from external observations rather than internal sources, and machine learning is used to process visual and sensor data [17].

M. Rehman et al. review computer vision methods in construction, where system states are reconstructed from images and video, showing that visual data alone can be enough to track complex processes over time [18].

The screen is treated as a source of information: indicators are located, their values are read, and the system state is reconstructed without using internal data.

This approach is common in industry, but in maritime simulators it is still rare. This creates a clear gap.

On one side, maritime research produces a lot of data about ship motion and human performance – but almost always through internal simulator logs. On the other side, computer vision already has tools that can extract the same kind of data from images – but these methods haven't really been adapted to ship simulators.

So, in most studies, one simple idea is still overlooked: what if the simulator interface itself could be used as an independent and universal source of navigation data?

This observation is the starting point of this work. If to change the simulator screen view, it is possible to see what it does as an operator interface. Indeed, the screen displays the current state of the ship. All instruments and indicators on the screen – steering, engine thrust, rotational speed, ambient parameters – reflect the values of the variables calculated in the simulation. The graphical elements of the interface can therefore be considered as a visible visualization of the internal parameters of the system.

In this interpretation, the interface acts as a structured visual space. In it, different areas of the screen correspond to specific navigation indicators, and parameters are displayed through stable graphic forms. If to analyze these interface elements, it is possible to reconstruct the value of the ship's movement parameters and the time change. This approach allows to receive data directly from the interface image without connecting to the internal telemetry modules of the simulator. Thanks to this, it becomes possible to conduct research even in cases where access to internal software is limited. The method proposed in the work arose at the intersection of two areas of research. The first is related to the use of marine simulators in scientific experiments. The second is with the development of computer vision and pattern recognition methods. By applying image recognition methods to the interface of the bridge simulator, it is automatically possible to read the readings of the devices and form synchronized time series of navigation parameters directly from the screen image. The analysis of existing studies shows that these two directions developed separately from each other for a long time. Works devoted to navigation simulators mainly focus on training of boatmasters, analysis of operators' behavior and improvement of vessel propulsion models. At the same time, methods of obtaining quantitative information from graphic interfaces of various technical systems are actively developing in the field of computer vision. Despite the apparent closeness of these ideas, their joint use in marine modelling has been virtually unaddressed.

Therefore, the interface of the navigation simulator, although it constantly reflects a large number of parameters of the ship's operation, was not perceived as an independent source of data for analysis for a long time. However, if one considers the screen as a visual representation of the state of the model, it becomes clear that it contains all the necessary information about the dynamics of the vessel. As a result, the task of data collection actually moves from the field of software integration to the field of image analysis. The main task is to find the desired indicators on the screen, determine their location and correctly read the shown values. The possibility of such an approach is explained by the features of modern bridge simulators. Their interfaces usually have a stable structure: the devices are located in fixed areas of the screen, and their appearance during operation practically does not change. At the same time, the values of the indicators change over time in accordance with the models of ship movement. It is this stability that makes it possible to systematically analyze the information displayed on the screen.

### 6.3 Navigation simulator interface as a structured visual field

The most common approach is the use of internal digital registers and telemetry flows, which are analyzed after the simulation is complete. The trajectory, speed, trajectory and interaction with the navigation objects are removed from the respective data files to evaluate the quality of the maneuvers and compare the scenarios. This approach provides very detailed data and is related to the architecture and registration format of a particular simulation platform (**Fig. 6.1**). Therefore, portability between systems is very limited.


HDG 24.0° MAG 21.9° COG 24.0° SOG 24.3 SET 24.0° DRIFT 24.3 ROT 0 RAD ----

270° 1.5 kt 34° 21 m/s

Visual Controls 24.0° 0.0°

0.00 24.30 0.00

0.0

RPM 098

12:00:00 8.7 m

Ready

**Fig. 6.1** Interface of the navigation simulator, selected during the experiment

The second group focuses on analyzing operator behavior in the simulator. Behavioral markers, non-technical skill scores, eye-tracking data, and cognitive load actions are used to investigate decision-making and situational awareness. In this sense, the dynamics of the ship are perceived as a reflection of the operator's behavior and not as an independent object of quantitative research. The trajectory of a ship is rarely systematically analyzed.

The third study group deals with the mathematical and physical modeling of ship movements using hydrodynamic and hydrodynamic computational equations to refine the predictions. The data from these studies are generated within a mathematical model and require a different external method of data collection.

A significant deviation can be observed in all three methodological study groups: either internal direct access to the simulator data is assumed, or the analysis is limited to the operator's behavior. None of the existing approaches consider the visual layer of the simulator as an independent and universal data source that can provide reconstructed navigation settings without any software integration. In this context, the research results obtained on one simulation platform cannot easily be reproduced on another platform. Comparative studies of different systems require either rarely occurring standard export interfaces or specialized integration tools. The lack of an independent data collection method limits the scope of simulator research and its application in cases where internal access to the software is limited or unavailable.

As mentioned, the simulator navigation indicators are placed in fixed and predictable areas of the screen. Each indicator has a stable graphic format. It can be a digital display, a dial, an indicator in the form of a strip or a map element. The way information is updated on the screen is an important feature. Since data updates occur sequentially and predictably, automatic screen reading of information and systematic data collection becomes possible.

Consequently, the interface can be thought of as a two-dimensional space with coordinates ($x$, $y$) in which each visual element occupies a specific region $Z_i$. Suppose that $I(x, y, t)$ denotes a function describing the visual state of the interface at time $t$. $Z_i$ zones correspond to $p_i(t)$ parameters, each of which represents a specific aspect of the ship's condition. Therefore, the internal state of the simulator is described by the state vector $S(t)$, which includes variables of different types – kinematic, dynamic and ecological. The display does not display the status vector completely, but only displays a subset of the variables required for navigation and control. The main purpose of the mapping method is

$$F : I(x,y,t) \rightarrow P(t),$$

where $P(t)$ – a subset of the internal state $S(t)$ that can be recovered from the visual representation. This mapping is not active: some internal variables do not have a direct graphical counterpart, and some interface elements combine several parameters. Therefore, reconstruction is limited to indicators with stable, unambiguous visual representations.

For a method to work stably, it doesn't have to depend exactly on how the interface is displayed. This applies to the screen resolution, scale or graphic settings. Coordinate normalization is used for this. In other words, the surface elements are not described by exact pixels, but by the relative position on the screen. For example, if the area is in the top left corner of the screen, it will stay there even after the resolution has changed (e.g. from 1280 × 1024 to 1920 × 1080).

This property of invariance is important to use the method on different platforms. If the two simulation systems have a similar interface structure – as is often the case with commercial bridge simulators developed according to similar principles, then the zone configuration configured for one system can be transferred to another with minimal changes. This makes the method more universal. Unlike approaches that rely on direct access to internal simulator data, it is not rigidly tied to a specific software system.

At the same time, it is important to understand that the display of $F$ depends on the configuration of the interface. This is a practical limitation of the method. If the interface structure changes noticeably – for example, after updating the software, when the indicator panels move or change location – the zones must be defined again. However, this approach immediately poses a practical question: how can these visual elements be systematically interpreted and converted into quantitative parameters suitable for scientific analysis? By itself, comparing the state of the interface with navigation variables gives only a general conceptual idea. In order to turn this idea into a real research tool, it is necessary to define a specific work procedure. It should allow to find the desired interface zones, interpret their graphic content and combine the obtained values into consistent sets of time-varying data.

## 6.4 Method architecture and patent protection

The proposed method is based on a simple sequence of actions, which includes four main stages: image capture, determination of the desired zone on the screen, interpretation of parameters and formation of time series (**Fig. 6.2**).

Stage 1. Image capture. The image from the simulator is fixed at equal time intervals $\Delta t$. The capture frequency is selected to match the refresh rate of the interface simulator indicators. In the experimental part, the interval $\Delta t = 1$ second (that is, the frequency of 1 Hz) was used as the base value. At the same time, the very architecture of the method allows working with a higher frequency of – up to 10 Hz, if technical conditions allow it. The resulting frames are used as downstream processing outputs.

Stage 2. Localization of zones. Before starting work, the analyst performs a preliminary configuration of the system. For each navigation indicator to be tracked, the screen sets its own $Z_i$ zone. It is determined using normalized coordinates in the screen space. Once configured, the configuration is saved and used in all subsequent sessions. The setup process itself can be considered semi-automatic. First, the researcher manually marks the required areas on the screen, and then the system automatically applies this markup during further data processing.

Stage 3. Parameter interpretation. After the zones are defined, the system analyzes their content and converts it into numerical values. The method of processing depends on the type of indicator.

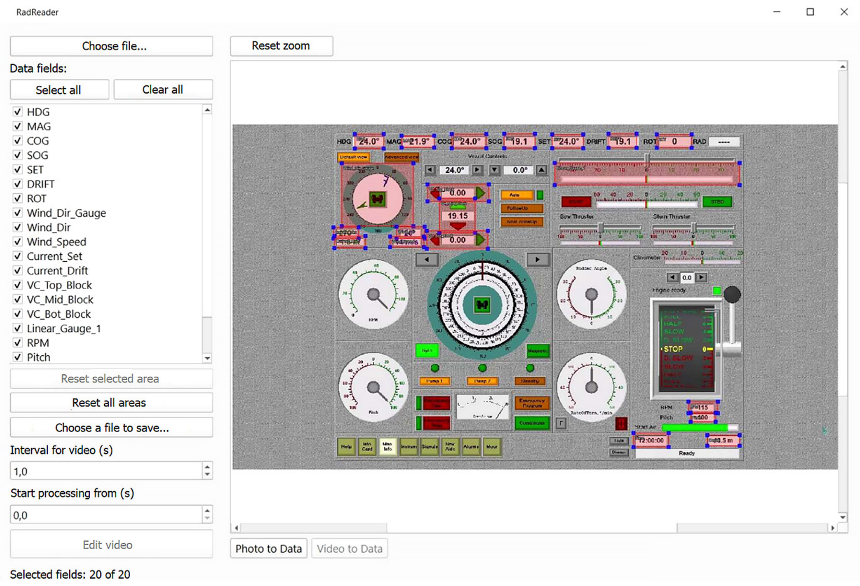
If the indicator shows the numbers – such as heading (HDG), speed relative to ground (SOG) or screw rotation (RPM) – optical character recognition is used. It simply reads the numbers displayed on the screen.

If the indicator has the form of a scale or dial – for example, the rate of rotation (ROT) indicator or the angle of translation of the steering wheel – geometric analysis is used. In this case, the system determines the position of the arrow or marker on the scale and then calculates the corresponding value.

Stage 4. Formation of time series. The parameter values obtained at each time point $t_k$ are combined into one set. It can be written as a vector

$$P(t_k)=\{p_1(t_k),p_2(t_k),\ldots,p_n(t_k)\}.$$

Each element of this vector corresponds to a specific navigation or control parameter that is read from the simulator interface. Such sets are formed for each moment in time. If to place them one after the other, a multidimensional time series is formed. It shows how the state of the vessel has changed throughout the simulation.



☑ HDG
☑ MAG
☑ COG
☑ SOG
☑ SET
☑ DRIFT
☑ ROT
☑ Wind_Dir_Gauge
☑ Wind_Dir
☑ Wind_Speed
☑ Current_Set
☑ Current_Drift
☑ VC_Top_Block
☑ VC_Mid_Block
☑ VC_Bot_Block
☑ Linear_Gauge_1
☑ RPM
☑ Pitch

Interval for video (s): 1,0

Start processing from (s): 0,0

Selected fields: 20 of 20

**Fig. 6.2** Localization of information zones for reconstructing navigation parameters

**Table 6.1** shows the main parameters that were obtained during the experiments. For each of them, it is indicated exactly how it is displayed on the simulator interface and how it is used in further analysis.

**Table 6.1 Parameters obtained by the method**

| Parameter | Symbol | Display Type | Analytical role |
|---|---|---|---|
| Course | HDG | Numeric/Dial | Main trajectory parameter |
| Heading over ground | COG | Numeric | Indicator of ecological drift |
| Speed over ground | SOG | Numeric | Kinematic state variable |
| Turn rate | ROT | Numeric/Dash | Indicator of maneuver dynamics |
| Propeller revolutions | RPM | Numeric/Dial | Propellant state variable |
| Rudder angle | Rudder | Dial | Controlling input variable |
| Wind direction | Wind Dir | Numeric/Gauges | Environmental disturbances |
| Wind speed | Wind Spd | Numeric | External influences |
| Water depth | Depth | Numeric | Bathymetric context |

One parameter value shows almost nothing by itself. It is much more important to see how these values change during the maneuver. Forming time series – is not just about combining data. It allows to move from individual measurements to the analysis of how the system behaves over time. It is this kind of sequence that helps to detect reaction delays, oscillations, stabilization processes and the relationship between the control actions and the ship's reaction.

Therefore, it is very important that all parameters are linked to the same timeline. For example, changing the course, shifting the steering wheel and changing the engine speed must correspond to the same moment in time. Only in such a case can the causal relationships between the control actions and the ship's response be correctly analyzed. In the proposed method, synchronization is ensured by the fact that all parameters are read from the same image frame. Due to this, all elements of the vector $P(t_k)$ correspond to the same moment in time during the simulation.

At the same time, the discrete nature of time series creates certain limitations. Events that occur within the same time interval $\Delta t$ may not be fixed separately. However, for maneuvering analysis, this is usually not a significant problem, since such processes unfold quite slowly and their characteristic time is measured in tens of seconds. Therefore, a sampling rate of 1 Hz is sufficient in most cases.

If it is necessary to investigate fast transients in more detail, the method allows the use of a higher sampling rate. In practice, its maximum value is limited by the speed of updating the simulator interface and the computing capabilities of the image processing system.

Data recovery from the visual display inevitably introduces some error compared to the reference values stored in the internal simulator model. Unlike direct access to telemetry, the proposed approach recovers parameter values through graphical interpretation, and accuracy depends on display resolution, indicator format, and graphical display stability.

Let $p_i^*(t)$ denote the reconstructed value of parameter *I* at time *t*, and $p_i(t)$ – its reference value within the simulator model. The instant recovery error is defined as follows

$$\varepsilon_i(t)=p_i^*(t)-p_i(t).$$

For the estimation over a given time interval, the mean absolute error (MAE) is used as the main indicator of accuracy

$$MAE_i=\frac{1}{N}\Sigma_k\left|p_i^*(t_k)-p_i(t_k)\right|,$$

where *N* – the number of discrete observation steps in the selected interval. This indicator is insensitive to the sign, which is suitable for estimating errors in the context of navigation, where both positive and negative deviations are equally significant.

The method described in this chapter is currently undergoing a patenting procedure. It was followed by two applications to the Ukrainian National Office for Intellectual Property and Innovation (UANIPIO):

– invention application No. a202600852;

– utility model application No. u202600851.

Both applications relate to the same technical solution. It is a method of automatically obtaining the navigation and control parameters of the vessel by analyzing the image of the navigation simulator interface, without referring to its internal telemetry data. Since the patent examination procedure has not yet been completed, the text uses the standard wording: "patent application filed, examination continues". Before possible commercial use of the method, it is recommended to clarify the current status of both applications on the official register of UANIPIO. The fact that two applications were submitted is related to the nature of the decision. On the one hand, the proposed approach contains new conceptual elements, so it is perceived as an invention. On the other hand, the method has quite obvious practical applicability, so it can be designed as a useful model.

### 6.5 Experimental verification and results

The method was validated on a fully functional bridge navigation simulator. Its interface shows a complete set of navigation and technical parameters that reflect the state of the vessel during simulation.

At the top of the interface are the main navigation indicators: heading (HDG), magnetic heading (MAG), heading relative to the ground (COG), speed (SOG), drift parameters and turning speed (ROT).

In the central part there is a circular course indicator that shows the orientation of the vessel in space.

At the bottom of the interface there are indicators of the speed of rotation of the propeller shaft (rpm), the angle of translation of the steering wheel, the speed of rotation and the engine controls.

The experimental program included several maneuvering scenarios. In these scenarios, the ship's course changed according to a given plan, while the engine operating parameters changed.

For each scenario, the parameter values recovered by the proposed method were compared with the reference values. These reference data were read directly from the simulator interface under controlled conditions. Such a comparison made it possible to calculate the indicators of the average absolute error.

Before starting the experiments, the zones corresponding to the different indicators were adjusted for the simulator interface.

For all the parameters shown in **Table 6.1**, as well as for some additional environmental variables displayed on the screen, separate observation areas were defined.

This installation was performed once. After that, the configuration was automatically used in all subsequent experimental sessions.

The coordinates of the zones were kept in a normalized form. Due to this, their position did not depend on the resolution of the screen. In tests where the resolution changed between different sessions, the configuration of the zones did not have to be adjusted again. This confirms the practical efficiency of the coordinate normalization approach.

**Table 6.2** summarizes the results of the assessment of the accuracy of the main navigation parameters obtained during a series of experimental tests.

A course recovery error not exceeding 1° corresponds to the level of accuracy usually required when studying the maneuverability of a vessel and analyzing its trajectory. The error of determining the engine speed is less than 2% of the nominal value. This shows that the thrust state of the engine is reconstructed quite reliably over the entire range of modes used in the experimental scenarios.

**Table 6.2 Reconstruction accuracy of basic navigation parameters**

| Parameter | MAE | Rating |
|---|---|---|
| Heading (HDG) | ≤ 1.0° | Sufficient for maneuver analysis |
| Propeller RPM | < 2% nominal | Sufficient for motor setup research |
| Rotation Rate (ROT) | < 0.5 deg/min | Suitable for transient analysis |
| Speed Over Ground (SOG) | < 0.2 kt | Suitable for trajectory research |

**Fig. 6.3** is a smoothed time series of vessel heading (HDG) and yaw rate (ROT) during the active phase of the maneuver. The timeline covers approximately 700 seconds and includes the stage of starting the maneuver, the turn itself and further stabilization of the movement.

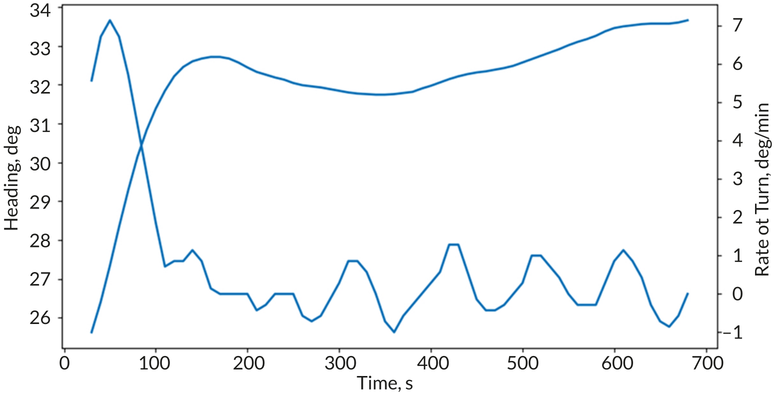


**Fig. 6.3** Time series of vessel heading (HDG) and angular rate of rotation (ROT) on the maneuver interval
*Source: Imagined with AI*

The turn rate (ROT) time series shows a well-defined maximum that appears before a noticeable change in course. This corresponds to the real dynamics of the ship's movement: first, the angular speed of the turn increases, and then the course itself begins to change rapidly. When the ROT peak occurs earlier than the maximum rate of change of course dHDG/dt is a characteristic feature of a real maneuver and at the same time is an indirect check of the correctness of the performed data reconstruction.

There are no noticeable phase shifts or artificial fluctuations in the reconstructed time series, and this is especially important, because if such distortions occurred, it could lead to a misinterpretation of cause-and-effect relationships. For example, it might seem that the change of course occurs before the controlling influence, although in reality this is not the case. And it is possible to see that the reconstruction method does not make such mistakes.

The time series of propeller revolutions (**Fig. 6.4**) at the same interval shows a gradual decrease in rotation frequency, which corresponds to the controlled transition of the vessel to a lower mode of movement. The synchronicity of the change of revolutions with the dynamics of the course and the angle of translation of the steering wheel allows analyzing the maneuver simultaneously from a kinematic and energy point of view. In other words, it becomes possible to consider the relationship between the mode of propulsion and the vessel's response to maneuver within a single coherent data set.

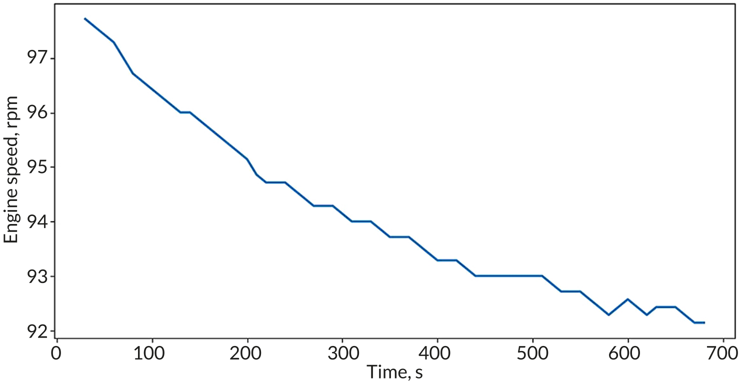


**Fig. 6.4** Time series of engine speed (RPM) during the maneuver interval
*Source: Imagined with AI*

As a result of the experiments, it became clear that the proposed method allows to reconstruct navigation parameters quite stably and with reasonable accuracy, using only the image of the simulator interface. It can be applied to the analysis of maneuvering dynamics, the study of transient modes of motion and the comparison of different scenarios. At the same time, no connection to the internal software architecture of the simulator is required.

## 6.6 Conclusions

In this project, we took what is essentially a very simple idea and tested it in practice: what if, instead of digging into the simulator's inner workings, we simply read what it already displays on the screen? To do this, we used image recognition. We stopped viewing the interface merely as a picture for the navigator and began to see it as a reflection of the model's internal state. Course, rudder angle, RPM – all of this is already right before our eyes. We just need to interpret it correctly. As it turned out, the approach works.

The main motion parameters are reconstructed quite reliably even without access to telemetry. The time series does not fall apart, events occur in the correct order, and the ship's behavior can be tracked dynamically. According to the experimental results, the sampling rate reached approximately 10 Hz, and the error remained within about 1° for the heading and up to 2% for the propeller RPM. For practical tasks, such as maneuver analysis or trajectory estimation, this is more than sufficient.

If to look at the bigger picture, the difference from typical approaches is quite noticeable here. Usually, a researcher connects to the simulator's internal mechanisms and retrieves logs, telemetry, or service channels. This is convenient, but there's a catch. Everything is tightly tied to a specific system. A different simulator means a new integration. Here, the situation is different. We don't connect to anything; instead, we work with what's already been exposed externally. Therefore, the method is easier to port between platforms. Essentially, it's not the tool that changes, but only the interface.

Of course, there are limitations. It all depends on how stable the interface is. However, if the layout of the elements changes, the system must be reconfigured. There is also a dependency on image quality, as a lack of sharpness or low resolution can affect accuracy, although this is not usually a critical issue under real-world conditions.

An important point is that this approach does not replace traditional methods of data collection. In other words, direct access to telemetry remains the simplest method, but if such access is unavailable or restricted, the option of screen reading becomes a perfectly viable alternative. This makes it possible to capture a consistent data set without interfering with the system software.

Looking at the bigger picture, the idea itself is interesting. Image recognition is usually associated with monitoring and diagnostics, but can also be used in experimental studies. In ship simulators, the interface is generally viewed as something for the human user, although it is actually data-level and quite extensive.

And this opens up scope for development. For example, it is possible to remove the manual configuration and train the system to find the necessary elements on the screen itself; it is possible to increase the processing frequency.

Overall, the main result of this work is that navigation data does not necessarily have to be taken from inside the simulator. It can be reconstructed from what is already visible. This makes experimental work simpler, more flexible, and less dependent on a specific system.

## Conflict of interest

The authors declare that there is no conflict of interest in relation to this paper, as well as the published research results, including the financial aspects of conducting the research, obtaining and using its results, as well as any non-financial personal relationships.

## Financing

The study was performed without financial support.

## Use of artificial intelligence statement

The authors have used artificial intelligence technologies (ChatGPT, OpenAI) for language refinement and structuring of the manuscript. The authors bear full responsibility for the final manuscript. Generative AI tools are not credited and are not responsible for the final results.

Some figures in this manuscript were generated using artificial intelligence and are labeled accordingly.

## Authors' contributions

**Yevgeniy Kalinichenko**: Conceptualization, Methodology, Supervision, Writing – review and editing.

**Oleksandr Koliesnik**: Investigation, Data Curation, Writing – original draft.

**Oleg Safyan**: Formal Analysis, Validation, Writing – review and editing.

## References

1. Munim, Z. H., Krabbel, H., Haavardtun, P., Kim, T.-E., Bustgaard, M., Thorvaldsen, H. (2023). Scenario Design, Data Measurement, and Analysis Approaches in Maritime Simulator Training: A Systematic Review. Methodologies and Intelligent Systems for Technology Enhanced Learning, Workshops – 13th International Conference. Springer, 39–47. https://doi.org/10.1007/978-3-031-42134-1_4

2. Sellberg, C. (2016). Simulators in bridge operations training and assessment: a systematic review and qualitative synthesis. WMU Journal of Maritime Affairs, 16 (2), 247–263. https://doi.org/10.1007/s13437-016-0114-8
3. Tusher, H. M., Munim, Z. H., Nazir, S. (2023). An evaluation of maritime simulators from technical, instructional, and organizational perspectives: a hybrid multi-criteria decision-making approach. WMU Journal of Maritime Affairs, 23 (2), 165–194. https://doi.org/10.1007/s13437-023-00318-1
4. Hetherington, C., Flin, R., Mearns, K. (2006). Safety in shipping: The human element. Journal of Safety Research, 37 (4), 401–411. https://doi.org/10.1016/j.jsr.2006.04.007
5. Saeed, F., Bury, A., Bonsall, S., Riahi, R. (2018). A proposed Evidential Reasoning (ER) Methodology for Quantitative Assessment of Non-Technical Skills (NTS) Amongst Merchant Navy Deck Officers in a Ship's Bridge Simulator Environment. TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation, 12 (3), 597–608. https://doi.org/10.12716/1001.12.03.20
6. Atik, O. (2019). Eye tracking for assessment of situational awareness in bridge resource management training. Journal of Eye Movement Research, 12 (3). https://doi.org/10.16910/jemr.12.3.7
7. Ronca, V., Uflaz, E., Turan, O., Bantan, H., MacKinnon, S. N., Lommi, A. et al. (2023). Neurophysiological Assessment of An Innovative Maritime Safety System in Terms of Ship Operators' Mental Workload, Stress, and Attention in the Full Mission Bridge Simulator. Brain Sciences, 13 (9), 1319. https://doi.org/10.3390/brainsci13091319
8. Frette, V., Kleppe, G., Christensen, K., Garrido, P. L., Marro, J., de los Santos, F. (2011). Analysis of ship maneuvering data from simulators. AIP Conference Proceedings, 1332, 276. https://doi.org/10.1063/1.3577668
9. Li, H., Zhao, N., Zhou, J., Chen, X., Wang, C. (2024). Uncertainty Analysis and Maneuver Simulation of Standard Ship Model. Journal of Marine Science and Engineering, 12 (7), 1230. https://doi.org/10.3390/jmse12071230
10. Pires da Silva, P., Sutulo, S., Guedes Soares, C. (2023). Sensitivity Analysis of Ship Manoeuvring Mathematical Models. Journal of Marine Science and Engineering, 11 (2), 416. https://doi.org/10.3390/jmse11020416
11. He, H., Lataire, E., Zwijnsvoorde, T. V., Delefortrie, G. (2023). A Ship Manoeuvring Desktop Simulator for Developing and Validating Automatic Control Algorithms. TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation, 17 (3), 607–616. https://doi.org/10.12716/1001.17.03.12

12. Yasukawa, H., Yoshimura, Y. (2015). Introduction of MMG standard method for ship maneuvering predictions. Journal of Marine Science and Technology, 20 (1), 37–52. https://doi.org/10.1007/s00773-014-0293-y
13. Lee, J., Nam, B., Lee, J.-H., Kim, Y. (2021). Development of Enhanced Two-Time-Scale Model for Simulation of Ship Maneuvering in Ocean Waves. Journal of Marine Science and Engineering, 9 (7), 700. https://doi.org/10.3390/jmse9070700
14. Kim, D., Tezdogan, T., Incecik, A. (2022). A high-fidelity CFD-based model for the prediction of ship manoeuvrability in currents. Ocean Engineering, 256, 111492. https://doi.org/10.1016/j.oceaneng.2022.111492
15. Smith, R. (2007). An Overview of the Tesseract OCR Engine. Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), 2, 629–633. https://doi.org/10.1109/icdar.2007.4376991
16. Szeliski, R. (2011). Computer Vision: Algorithms and applications. Texts in Computer Science. London: Springer. https://doi.org/10.1007/978-1-84882-935-0
17. Wuest, T., Weimer, D., Irgens, C., Thoben, K.-D. (2016). Machine learning in manufacturing: advantages, challenges, and applications. Production & Manufacturing Research, 4 (1), 23–45. https://doi.org/10.1080/21693277.2016.1192517
18. Sami Ur Rehman, M., Shafiq, M. T., Ullah, F. (2022). Automated Computer Vision-Based Construction Progress Monitoring: A Systematic Review. Buildings, 12 (7), 1037. https://doi.org/10.3390/buildings12071037