Estimates of wave field in the vicinity of a floating body using its motions and machine learning techniques

Akinturk, Ayhan; Zaman, Hasanat; Seo, Dong Cheol; He, Moqin; Mak, Lawrence

Publisher's version / Version de l'éditeur:
https://doi.org/10.23919/OCEANS44145.2021.9706106
OCEANS 2021: San Diego – Porto, 2021-09-20

Estimates of Wave Field in the Vicinity of a Floating Body Using Its Motions and Machine Learning Techniques

Ayhan Akinturk
Ocean Coastal and River Engineering Research Centre
National Research Council
St. John's, NL Canada
Ayhan.Akinturk@nrc-cnrc.gc.ca

Hasanat Zaman
Ocean Coastal and River Engineering Research Centre
National Research Council
St. John's, NL Canada
Hasanat.Zaman@nrc-cnrc.gc.ca

Dong Cheol Seo
Ocean Coastal and River Engineering Research Centre
National Research Council
St. John's, NL Canada
Dong.Seo@nrc-cnrc.gc.ca

Moqin He
Ocean Coastal and River Engineering Research Centre
National Research Council
St. John's, NL Canada
Moqin.He@nrc-cnrc.gc.ca

Lawrence Mak
Ocean Coastal and River Engineering Research Centre
National Research Council
St. John's, NL Canada
Lawrence.Mak@nrc-cnrc.gc.ca

***Abstract* — Real time estimation of actual and imminent wave fields in the vicinity of ships and floating structures are important for safe and efficient operations. According to the relevant literature, three different approaches have been used practically: the use of wave buoys, ship radars (e.g. X-band and K-band radars) / satellite and ship motions. The present study employs artificial neural networks and machine learning type modelling to extend the previous research on neighboring wave field estimation to consider the relative heading between the ship and the waves. For the different cases evaluated, the proposed machine learning model showed very good prediction for relative heading. Some of the results from this study are presented in this paper.**

Keywords — Wave field estimation, ship motions, machine learning, artificial neural networks and short time FFT

I. INTRODUCTION

Real time estimation of actual and imminent wave fields in the vicinity of ships and floating structures are important for safe and efficient operations. Historically, three main approaches have been cited in the open literatures with varying degrees of accuracies: the use of wave buoys, ship radars (e.g. X-band and K-band radars) / satellite and ship motions. A few publications from the open literature on these topics are given in [1], [2], [3] on wave rider buoys, [4] to [8] on using ship radars and [9] to [13], on hydrodynamic and spectral modeling using ship motions. It is not the intention here to present an exhaustive list and a detailed review on each topic but rather highlight a few papers focused on the machine learning model. In [16] Anagnostopoulos predicted propulsion power using Big Data techniques. Sclavounos and Ma used Support Vector Machine model to forecast the sea state elevations and vessel responses using past time records [17]. A Grey Online Sequential Extreme Learning Machine is implemented to model ship roll predictions by Yin et al. in [18].

The present work reported here builds upon the previous study mentioned above ([14] and [15]), in which machine learning model was developed to estimate the wave field using ship motions for varying advance speeds and sea states, and extends it to estimate relative wave headings. Short Time Fourier Transform (STFT) of selected ship motions is utilized as input to the machine learning tool for heading estimation. As a case study, an approximately 25 m long research vessel is used. The paper reports the performance of the relative heading estimation using simulated data, which were generated with a ship motion predictions computer code.

In the next sections, machine learning approach is introduced briefly, followed by a description of the artificial neural network model used in this study. Also, the results obtained for a near shore research vessel in head seas with various forward speeds and sea states are reported.

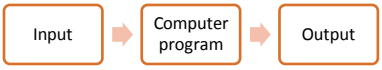
II. ACTUAL NEIGHBORING WAVE FIELD ESTIMATION

The present study assumes that a ship or a floating body is instrumented so that its motions can be monitored in real time. In this study two methodologies are used to estimate wave field and relative heading: machine learning technique and STFT.

A. Machine Learning

Machine learning has become popular particularly in the last decade. In essence, similar to any other prediction algorithm the

main focus is to provide accurate state estimations from the given dataset/measurements.



Compared with model-based approaches, machine learning has the ability to learn from the data and improve/adjust its coefficients as it encounters new cases without an adjustment from the outside/user. It works in two modes in general: supervised and unsupervised. In supervised learning mode, the machine learning is trained using a set of known inputs and associated outputs. During the training, coefficients in the machine learning model are optimized to reduce the error between the provided known outputs and predicted values (Figure 1).

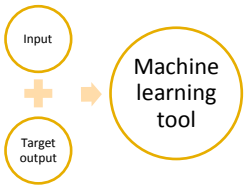


Figure 1 Machine learning tool updates its model coefficients during training

Once the training is completed the machine learning tool will give estimation for new data or measurements based on the trained model (Figure 2).

Figure 2 Normal operation mode

In unsupervised learning mode, there is no need to have known output for the training process. In this mode the machine learning tool usually relies on deep learning algorithms to arrive at clustering of the data, where it tries to unearth any hidden structure the data may have. The objective is to find commonalities in the data and group them based on those commonalities.

B. Short Time Fourier Transform

If a ship instrumented with motion sensors, for which there are several readily available affordable products / off-the-shelf solutions, real time motion data can be easily obtained. Applying the STFT technique to the collected motion data and utilizing the processed information in a machine learning frame work, ship heading with respect to the wave field are estimated.

In the STFT method, short time series of the ship motions, such as the one shown between two vertical lines in Figure 3, are processed segment by segment. Two consequent segments may or may not share the sampled data depending on the segment length and processing interval.

Each segment of measured motion data is then processed by using STFT, to obtain the motion amplitude spectrum shown in Figure 4.

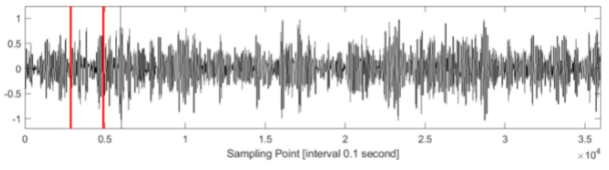


Figure 3 Segment selection

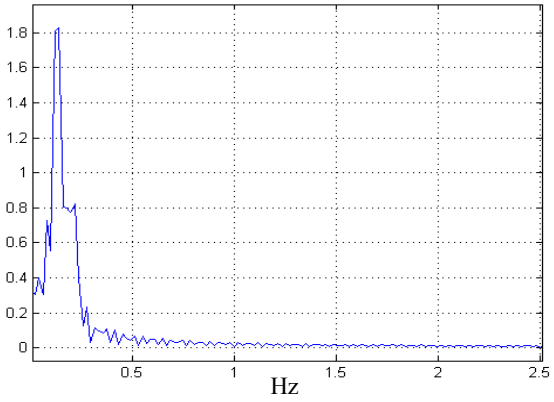


Figure 4 Segment processing

III. METHODOLOGY - PREDICTION OF HEADING

As mentioned above, the objective of this study is to predict the neighboring wave field and relative heading between the ship and the waves using ship motions. Wave field prediction using machine learning methodology has been published in earlier papers. In this paper, the focus is to estimate the heading.

In traditional naval architecture, ship motions can be determined by a wave field and ship parameters such as hull shape, speed and heading (Figure 5). There are well established methods and software available to this extent.

Figure 5 Model for ship motion computations

Whereas in this study, the model is reversed as shown in Figure 6, in that a ship can be treated as a wave buoy.

Figure 6 Model for near wave field prediction

A. Methodology used inthis study

Oscillatory motions of a ship are mainly affected by waves, so the prediction model needs to take account every ship motion. Usually the ship motion parameters include displacements, velocities, and accelerations of all six degrees of freedoms. Hence, a total of 18 different measurements form the basis of the data.

In this study, the machine learning algorithm for heading estimation was developed though the following steps:

1) Step 1: Generate ship motion data

The ship motion data were generated by SHIPMO3D™ which is a frequency domain panel method code that can estimate the hydrodynamic coefficients such as added mass and wave excitation, and also simulate the ship responses in prescribed wave fields. In this study the motion data for the machine learning tool were generated from the motion simulations. A typical fisheries research vessel about 25 m long was used as the test vessel. It is noted that the data presented in this paper are based on simulated data, however, work is underway to validate the proposed methodology using the sea trial data of the same ship. The SHIPMO3D simulations were carried out for a nominal JONSWAP spectrum for sea-state 3 with the ship sailing at 5 knots. The simulations were conducted for a duration of one hour. The motion data were also repeated for various heading angles from 0 degree (following seas) to 180 degrees (head seas) at 15 degree increments.

2) Step 2: Preprocess the motion data using STFT for the machine learning tool

Using the motion data obtained from ship motion simulations, the following data processing was implemented.

- Select a segment size
- Select a motion parameter, e.g. pitch motion
- Slide the segment (Figure 3) to sweep the whole time series for the selected motion parameter
- For each segment,
  - calculate standard deviation of the associated motion parameter
  - Perform STFT to determine the spectral peak amplitude and the corresponding frequency (Figure 7)
- Repeat this preprocess for all 18 motion parameters

At the end of this preprocessing stage a total of 54 new input parameters (i.e. standard deviation, peak and peak frequency for each motion segment) can be obtained from the raw motion data. Note that as part of the current study, several parameters (i.e. subset of entire parameters) were examined, however, these 54 parameters showed the best performance based on the research results carried out so far, and thus are reported in this paper.

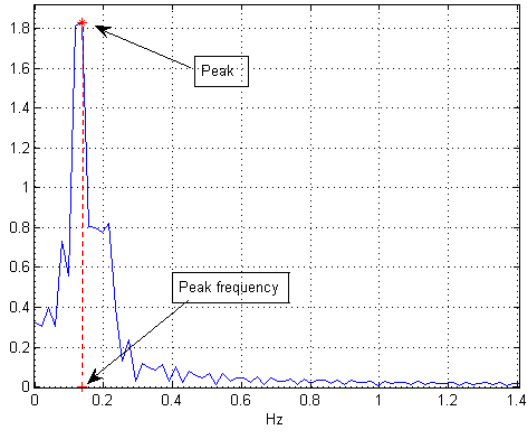


Figure 7 Spectral parameters

3) Step 3: Training - use the newly obtained 54 parameters and the associated known headings to train the machine learning tool

Figure 8 shows the machine learning model used during training. In this study, the machine learning algorithm was implemented using MATLAB machine learning toolbox. In the figure,

$$x(t) = \begin{bmatrix} Stdev's & Peaks & Peak\ freq's \\ 18\ columns & 18\ columns & 18\ columns \end{bmatrix}$$

$\rightarrow$ a total of 54 columns

$y(t) = \{heading\ angles\}$ $\rightarrow$ 1 column

Both x(t) and y(t) have number of rows equal to the number of segments in each time series record.

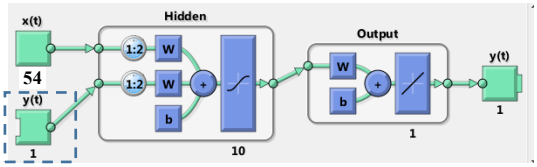


Figure 8 Time series Neural Network model during training

Developing the tool to a level ready to be used as an estimation tool, the above mentioned three stages are used: training, validation and testing. These three stages together are referred as "training phase" in this paper. The main difference between the training phase and the phase of the tool, in which it is used for a heading angle estimation tool, is that in the former heading angle is explicitly used as one of the inputs, which is y(t) inside the square drawn with broken lines in Figure 8. However, note that this y(t), i.e. the input, is only used to estimate the error between the targets and the outputs to optimize the neural network model during the training. Whereas in the latter, i.e. Figure 9 and Figure 10, the input does not include heading angle.

The target output is heading angle, i.e. y(t). In the training mode, the 70% of data are used for training, 15% for the validation and the remaining 15% for testing the machine learning tool. These 15 %'s were sampled from the x(t) values. During the training, motion data for the following heading angles are used: 0, 15, 30, 45, 60, 90, 105, 120, 135, 150 and 180 degrees.

The training was conducted using Levenberg-Marquardt method. During the training two different neural network models were used: Configuration A - a model with 10 hidden neurons with number of delays of 2; and Configuration B - a model with 20 hidden neurons with number of delays of 2.

4) Step 4: the machine learning tool as an estimation tool after the training

In this phase, the heading angle estimation performance is examined for when it is used as an estimation tool. Note that the motion data used in this phase were not used in the training phase. These new data were for heading angles of 75 and 165

degrees. In this phase, machine learning model has only x(t) as input unlike the training phase, which also has y(t). As shown in Figure 9 and Figure 10, two models of neural network were used: a model with 10 hidden neurons with number of delays of 2 and a model with 20 hidden neurons with number of delays of 2. In these models, the problem is formulated as follows:

$$y(t) = f(x(t-1), \ldots, x(t-d))$$

where "d" is number of delays. Again, y(t) is the heading angle.

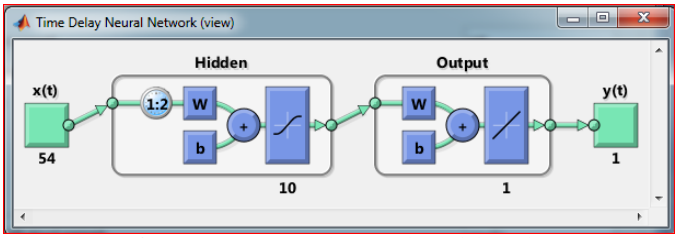


Figure 9 Configuration A: Neural network model used in training: 10 hidden neurons with a time delay of 2.

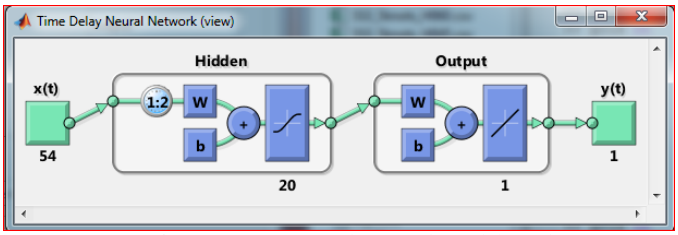


Figure 10 Configuration B: Neural network model used in training: 20 hidden neurons with a time delay of 2.

IV. RESULTS OF SIMULATIONS

In implementing the methodology, two of the main questions addressed were the size of the segments and the configuration of the neural network. The effects of these two parameters on the success of the machine learning tool in estimating the heading angle from ship motions had been investigated by a series of simulations by varying them as follows:

- Two neural network models as specified above, i.e. Configurations A and B
- Segment (Figure 3) sizes: 6 seconds, 30 seconds, 60 seconds, 90 seconds, 120 seconds, 300 seconds, 600 seconds, 900 seconds and 1200 seconds

Table 1 summarizes the combinations used in the investigation.

Next a selection of the results obtained. In the figures presented, "Target" is the actual heading angle, "Output" and "Response" are the prediction results given by the machine learning tool. "Error" is the difference between "Output" and "Target". The unit of measurement for heading angle in these cases is degrees. Summaries of the results are also presented at the end of this section.

A. Case 1: Neural network model is in Configuration A (Figure 9), segment size is 6 seconds

This case corresponds to a segment size of 6 seconds and a Neural Network configuration presented in Figure 9. Figure 11 shows the performance of the machine learning tool during training, validation and testing. Overall the estimates, are very good as shown in the top pane. The bottom pane shows the error in degrees between the targets and the outputs. Figure 12, error histogram, shows the magnitude of error is concentrated around zero. The larger errors in Figure 11 correspond to the transition regions in the data from one heading angle to another. Note that one might expect such sharp changes can only happen when the ship is maneuvering or exposed to severe yaw effects. However, this was not the case here. Hence, it is assumed that the performance in estimating the heading angle is satisfactory.

Table 1 Summary of cases

Case No	Neural Network Configuration	Segment size (s) (see Figure 3)
1	Config A (Figure 9)	6
2	Config A	30
3	Config A	60
4	Config A	90
5	Config A	120
6	Config A	300
7	Config A	600
8	Config A	1200
9	Config B (Figure 10)	6
10	Config B	30
11	Config B	60
12	Config B	90
13	Config B	120
14	Config B	300
15	Config B	600
16	Config B	1200

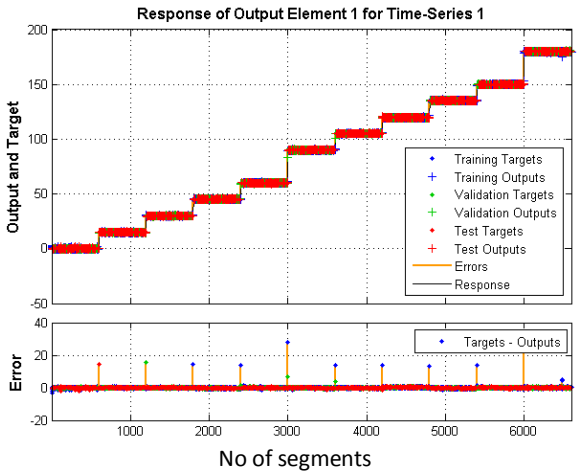


Figure 11 Case 1 – Performance of the machine learning tool during training, validation and testing

Figure 13 shows the comparisons for the regression analysis between the targets and the outputs for each phase: training, validation and testing. The fits seem to be quite reasonable.

In the next step, additional motion data for heading angles 75 and 165 degrees were used in order to confirm if the proposed method can provide a proper estimation even for not trained conditions.

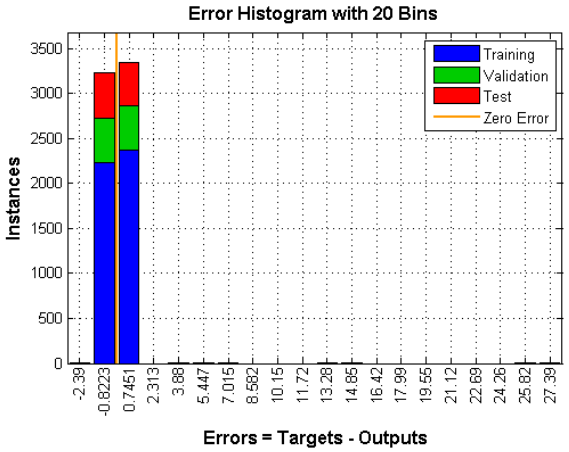


Figure 12 Case 1 – Error histogram

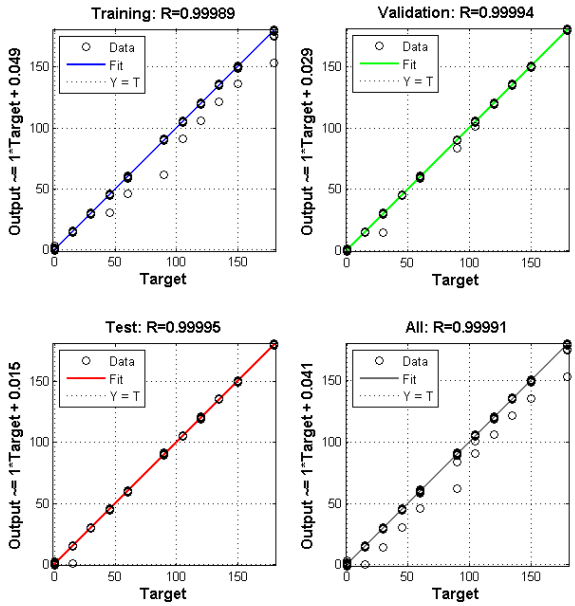


Figure 13 Case 1 – Regressions for training, validation and testing

Figure 14 and Figure 15 show the corresponding performance. From the error histogram in Figure 15, the error range is usually between +/- 5 degrees. The largest error occurs in the instantaneous transition of heading angle from 75 degrees to 165 degrees. It is concluded that the machine learning tool performed satisfactory for the case simulated.

B. Case 9: Neural network model is in Configuration B (Figure 10), segment size is 6 seconds

In this case, Neural Network model is changed to Configuration B, which increased the hidden neurons from 10 to 20. Figure 16 shows the performance of the machine learning tool during training, validation and testing. Similar to the previous case, the performance is satisfactory. The error histogram (Figure 17) reveals that the magnitude of the error is concentrated around zero. The larger errors in the bottom pane of Figure 16 correspond to the transition in the data from one heading angle to another.

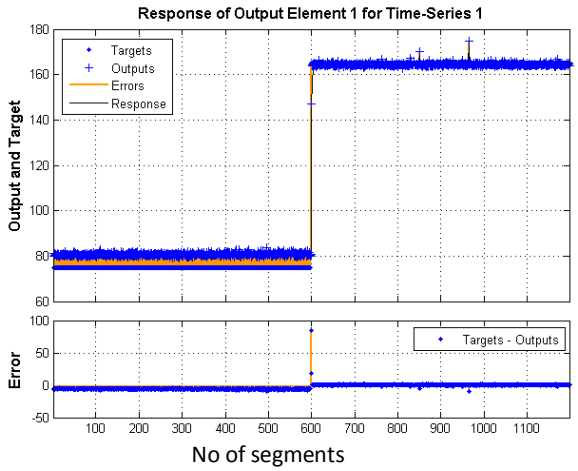


Figure 14 Case 1 – Performance after the training – two heading angles used: 75 and 165 degrees

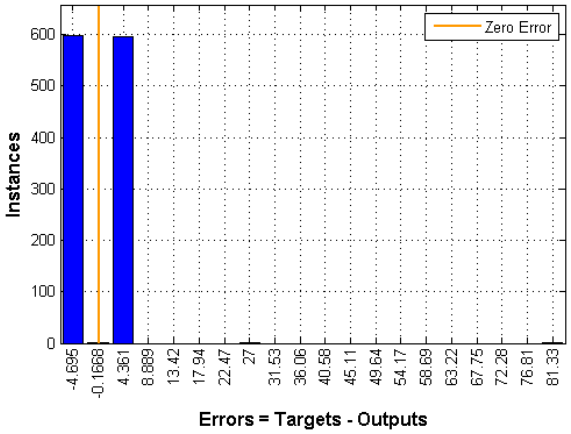


Figure 15 Case 1 – Error histogram after the training – two heading angles used: 75 and 165 degrees

Figure 18 shows the comparisons for the regression analysis between the targets and outputs for each phase: training, validation and testing. The correlations are good.

The performance for the untested heading conditions (i.e. heading angles 75 and 165 degrees) is shown in Figure 19 and Figure 20 . In the figure the estimation errors were quite low, hence the performance was assumed to be satisfactory for the data considered. Similar to the previous case above, the largest error was observed in the instantaneous transition of heading angle from 75 degrees to 165 degrees.

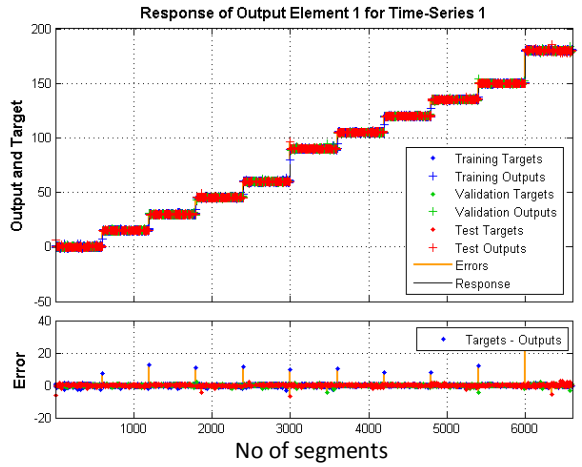


Figure 16 Case 9 – Performance of the machine learning tool during training, validation and testing

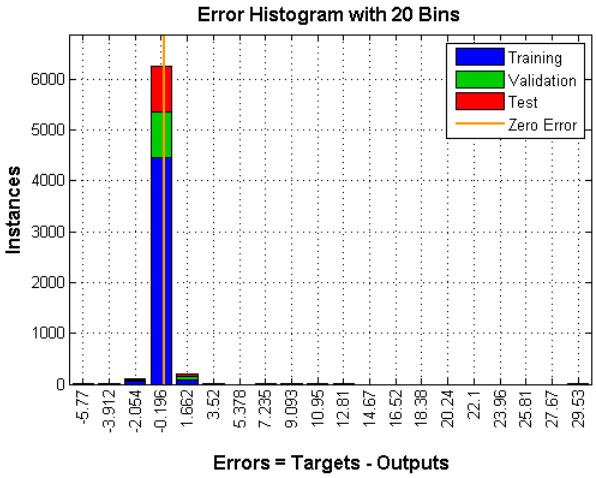


Figure 17 Case 9 – Error histogram

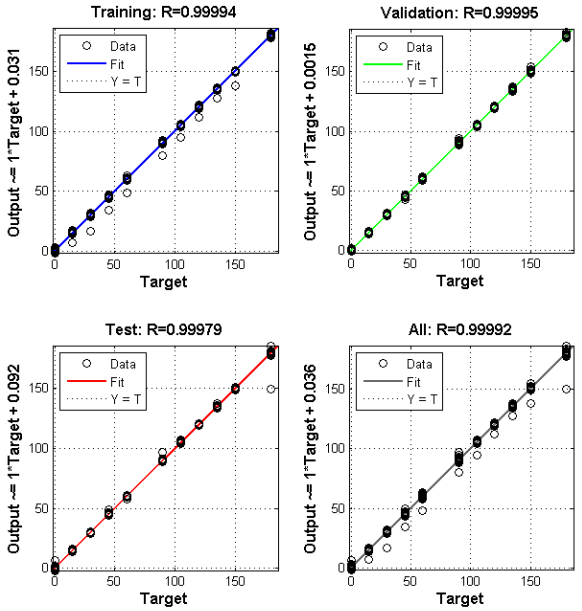


Figure 18 Case 9 – regressions for training, validation and testing

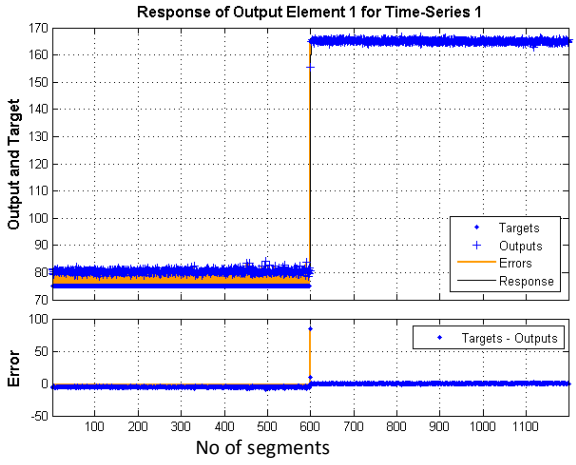


Figure 19 Case 9 – Error histogram after the training – two heading angles used: 75 and 165 degrees

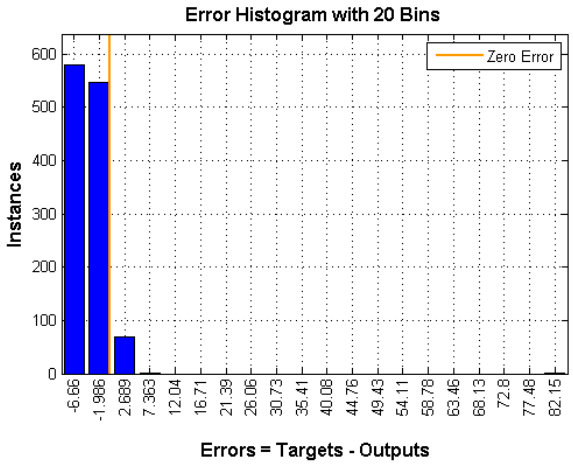


Figure 20 Case 9 – Error histogram after the training – Two cases used: 75 and 165 degrees heading angles

C. Case 5: Neural network model is in Configuration A (Figure 9), segment size is 120 seconds

The next case consists of a Neural Network model of Configuration A together with a segment size of 120 seconds, which is longer than Case 1 (6 seconds). Figure 21 shows the similar performances during training, validation and testing. Similar to the previous two cases, the performance is satisfactory. The error histogram (Figure 22) reveals that the magnitude of the error is concentrated around -1.0. The larger errors in the bottom pane of Figure 21 are observed during the transition in the data from one heading angle to another similar to the previous cases.

Figure 23 shows the comparisons for the regression analysis between the targets and outputs (i.e. actual and predicted values) for each phase: training, validation and testing. The correlation coefficients – R values for the fits are good.

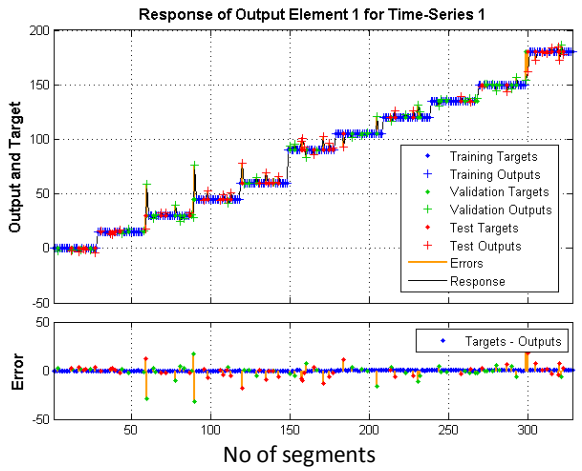


Figure 21 Case 5 – Performance of the machine learning tool during training, validation and testing

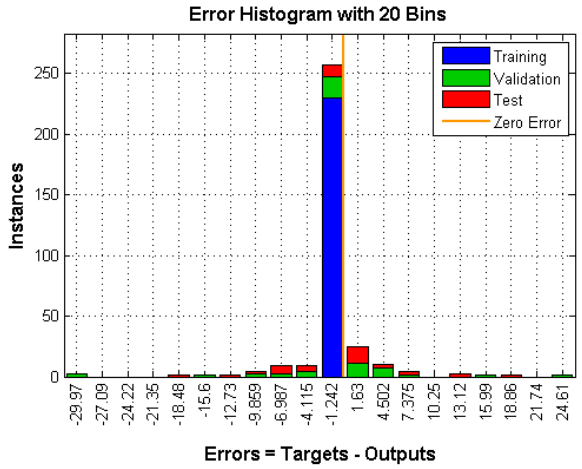


Figure 22 Case 5 – Error histogram

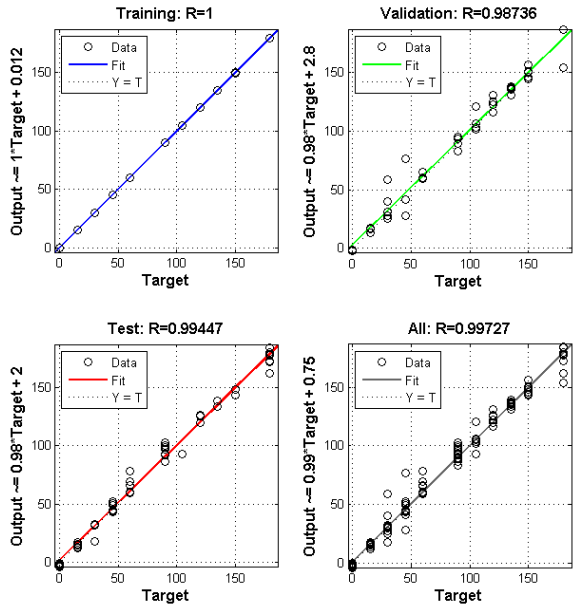


Figure 23 Case 5 – regressions for training, validation and testing

As same as the previous cases, the machine learning model was tested with the extra motion data for heading angles of 75 and 165 degrees. Figure 24 and Figure 25 show the performance in this mode. In the figure, the estimation errors are reasonably low, hence the performance was assumed to be satisfactory for the data considered. Similar to the previous case above, the largest error occurs in the instantaneous transition of heading angle from 75 degrees to 165 degrees.

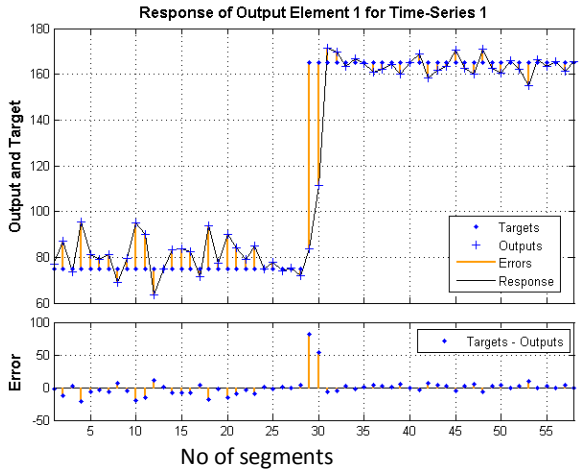


Figure 24 Case 5 – Performance after the training – two heading angles used: 75 and 165 degrees

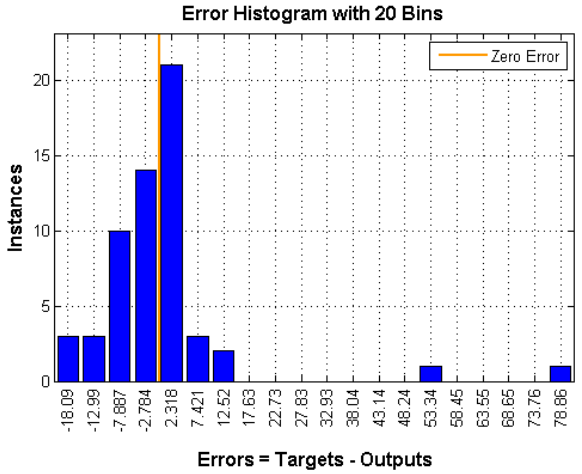


Figure 25 Case 5 – Error histogram after the training – two heading angles used: 75 and 165 degrees

D. Case 13: Neural network model is in Configuration B (Figure 10), segment size is 120 seconds

As a follow up case, the Neural Network model was changed to Configuration B while keeping segment size at 120 seconds. Figure 26 shows the performance of the machine learning tool during training, validation and testing. Similar to the previous cases, the performance seems reasonable. As the error histogram in Figure 27 shows that the magnitude of the error is concentrated around -1.0. The larger errors in the bottom pane of Figure 26 corresponds to the transition in the data from one heading angle to another. Figure 28 shows the comparisons for the regression analysis between the targets and the outputs for

each phase: training, validation and testing. The fits as given are good.

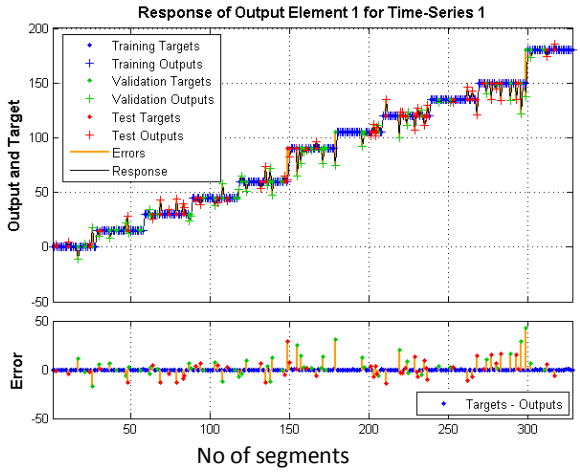


Figure 26 Case 13 – Performance of the machine learning tool during training, validation and testing

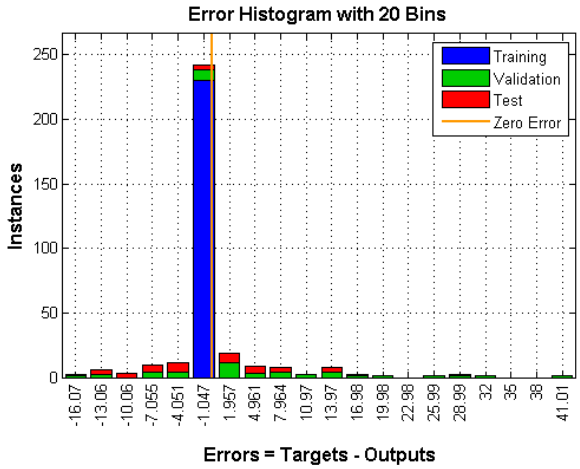


Figure 27 Case 13 – Error histogram

Similar to the other cases, the tool is again tested with the extra motion data for heading angles of 75 and 165 degrees. Figure 29 and Figure 30 show the prediction performances. It seems the estimation performance is not satisfactory unlike the previous cases however, it can still be considered as reasonable. Similar to the previous cases above, the largest error occurs in the transition periods.

E. Summaries of the all cases

In this section, summaries of the results from all the cases mentioned in Table 1 are presented. Table 2 shows the mean square error (MSE) and the correlation coefficient (R) values for all the cases.

Based on Table 2 and also the figures from Figure 31 to Figure 34, segment sizes of 90 and 120 seconds gave the largest MSE and lowest R values. Based on these figures, it seems MSE would peak around segment sizes of 90 to 120 seconds and level off after 300 seconds (five minutes). For these segment sizes, for which MSE peaks, computed R values are the lowest among the all cases considered.

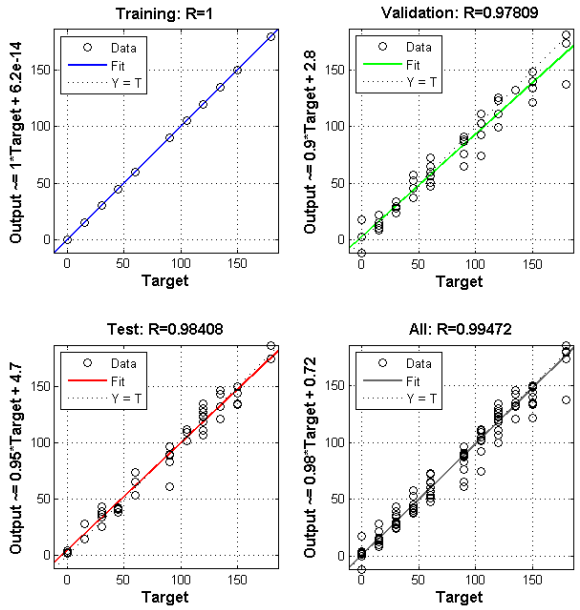


Figure 28 Case 13 – regressions for training, validation and testing

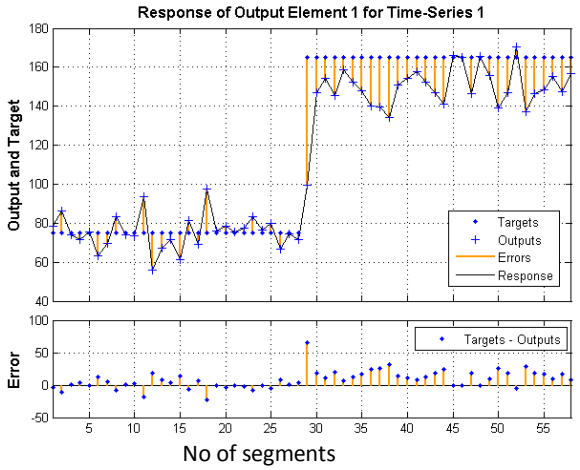


Figure 29 Case 13 – Performance of the machine learning tool during training, validation and testing

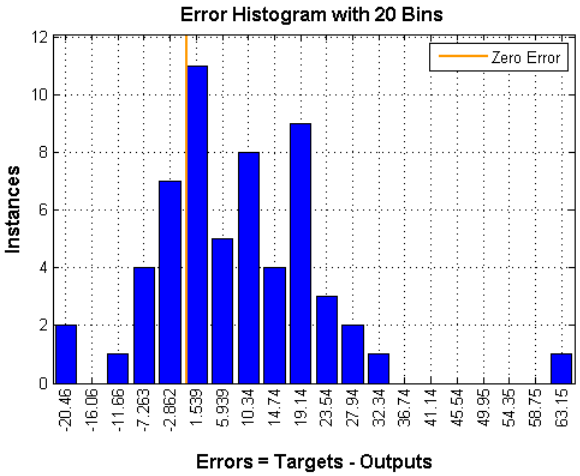


Figure 30 Case 13 – Error histogram after the training – two heading angles used: 75 and 165 degrees

Table 2 Summary of the results for simulations

			Training						After Training	
			Training		Validation		Test		Normal use	
Case No	Neural Network Configuration	Segment size (s)	MSE	R	MSE	R	MSE	R	MSE	R
1	Config A	6	0.68	0.999	0.4	0.999	0.33	0.999	23.07	0.998
2	Config A	30	1.78	0.999	3.49	0.999	6.87	0.999	87.33	0.979
3	Config A	60	5.46	0.999	39.02	0.994	27.03	0.996	115.78	0.976
4	Config A	90	0	0.999	39.8	0.994	42.75	0.993	243.3	0.945
5	Config A	120	0	0.999	75.01	0.987	39.92	0.994	214.05	0.946
6	Config A	300	0.52	0.999	0.64	0.999	0.64	0.999	23.37	0.998
7	Config A	600	0.18	0.999	1.26	0.999	0.69	0.999	25.69	0.998
8	Config A	1200	0.4	0.999	0.49	0.999	1.54	0.999	16.32	0.997
9	Config B	6	0.37	0.999	0.3	0.999	1.35	0.999	21.21	0.998
10	Config B	30	0.13	0.999	18.92	0.997	3.75	0.999	46.07	0.99
11	Config B	60	0	1	23.76	0.996	26.38	0.996	128.13	0.971
12	Config B	90	0	1	48.33	0.993	57.29	0.991	149.71	0.964
13	Config B	120	0	0.999	144.18	0.978	80.88	0.984	253.78	0.958
14	Config B	300	0.61	0.999	0.45	0.999	0.49	0.999	26.65	0.998
15	Config B	600	0.14	0.999	0.53	0.999	1.48	0.999	29.47	0.998
16	Config B	1200	0.69	0.999	0.68	0.999	1.71	0.999	38.61	0.996

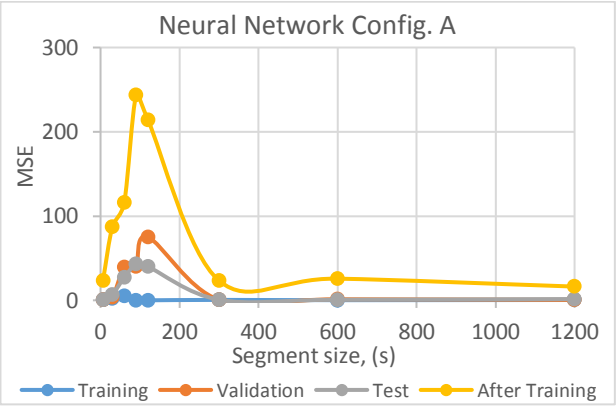


Figure 31 Resulted MSE values for the four phases for neural Network Configuration A

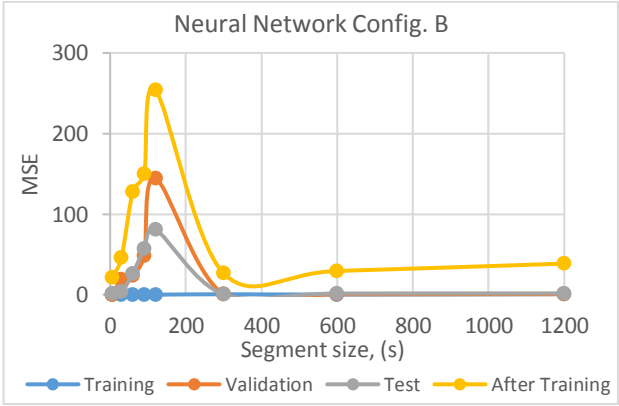


Figure 32 Resulted MSE values for the four phases for neural Network Configuration B

The cases, in which the machine learning tool is in the "after training" i.e. for the untrained wave headings such as 75 and 165 degrees, MSE values seem to increase compared to the other cases, i.e. wave headings aligned with the training conditions, regardless of the segment size or network configurations (Figure 31 and Figure 32. The same trend repeats itself for R only in reverse such that "after training" mode gives lowest values among the four phases of evaluation (Figure 33 and Figure 34).

Figure 35 and Figure 36 show a comparison of the effects of the number of hidden layers on the estimates. The comparisons given in Figure 35 are for the tests performed during training. Figure 36 is for after training estimates. Both trends in the figures follow the same pattern as before. For segment sizes larger than 300 seconds both perform similar. For segment sizes less than 300 seconds, the peaks are quite different for the tests performed under training.

Based on these results, a segment size of 300 seconds or more seems reasonable to use when MSE and R of the estimates

are considered. This can be connected with the point that STFT requires sufficient sampling duration to capture the frequency characteristics of ship motions.

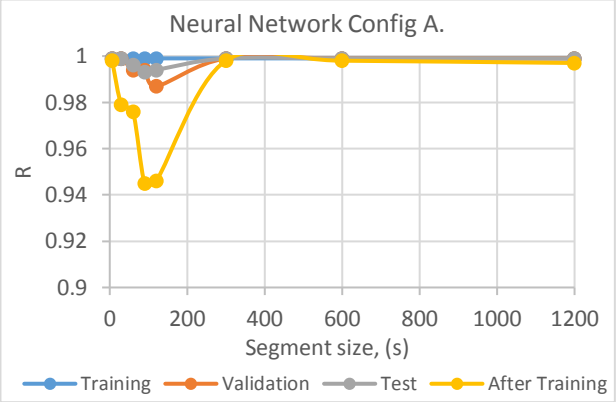


Figure 33 Resulted R values for the four phases for neural Network Configuration A

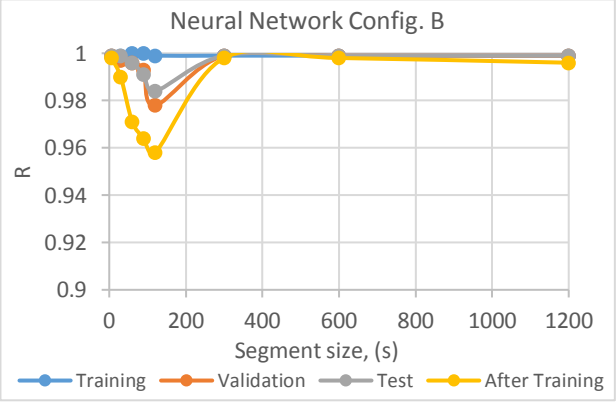


Figure 34 Resulted R values for the four phases for neural Network Configuration B

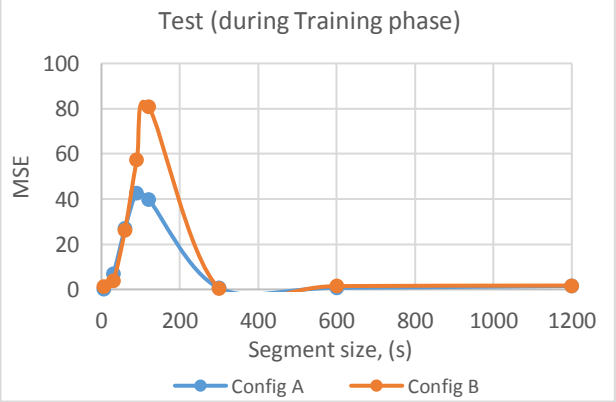


Figure 35 Comparison of MSE values between Configurations A and B for all cases included for tests during training phase

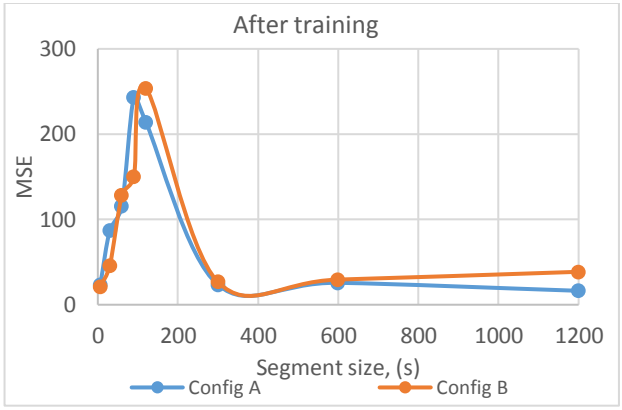


Figure 36 Comparison of MSE values between Configurations A and B for all cases included during after training phase

I. CONCLUSIONS

Machine learning concept combined with STFT method was implemented to estimate the relative heading between a ship and the encountered wave field. The method basically uses ship motions such as displacement, velocity and acceleration, however, STFT was used as a preprocessing tool to capture the motion characteristics in frequency domain which is well established approach in naval architecture. In this paper, different network structures and segment sizes are evaluated. The proposed methodology combining STFT and neural network showed satisfactory performance. The further validation is underway, especially to apply the proposed method with the data collected in full scale sea trials.

ACKNOWLEDGMENT

All personnel, who have contributed to this project in any form, deserve our sincere gratitude.

REFERENCES

[1] J. Allender, , T. Audunson, S. F. Barstow, S. Bjerken, H. E. Krogstad, P. Steinbakke, L. Vartdal and C. G. L. E. Borgman,(1989) "The WADIC project: A comprehensive field evaluation of directional wave instrumentation." Ocean Eng., 16, 505-536.

[2] S. Van Essen, K. Ewans and J. McConichie (2018) "Wave buoy performance in short and long waves, evaluated using tests on a hexapod." ASME 2018 37th International Conference on OMAE.

[3] Q. Liu, T. Lewis, Y. Zhang and W. Sheng (2015) "Performance assessment of wave measurements of wave buoys." International Journal of Marine Energy, 12, 63-76.

[4] J. Giron-Sierra. (2010). "State-Of-The-Art of Wave Measurement for Ship Motion Prediction." IFAC Proceedings Volumes (IFAC - Papers Online). 295-300. 10.3182/20100915-3-DE-3008.00057.

[5] M. Belmont, J. M. K. Horwood, R. W. F. Thurley, J. Baker (2007) "Shallow Angle Wave Profiling Lidar" J. Atmospheric Ocean Technology, 24 (2007), pp. 1150-1156

[6] W. Alpers, D. Ross and C. Rufenach (1981) "On the Detectability of Ocean Surface Waves by Real and Synthetic Aperture Radar" J. Geophysical Research, 86 (no. C7), pp. 6481-6498

[7] W. Alpers, (2003) "Ocean Surface Wave Imaging from Seasat to Envisat" IEEE Intl. Symp. Geoscience and Remote Sensing, pp. 35-37

[8] X. Liu, W. Huang and E. W. Gill. (2016) "Wave Height Estimation from Shipborne X-Band Nautical Radar Images" Journal of Sensors, Volume 2016 |Article ID 1078053 | https://doi.org/10.1155/2016/1078053

[9] M. Hinostroza and C. Guedes Soares (2016). Parametric estimation of the directional wave spectrum from ship motions. The International Journal of Maritime Engineering. 158. A121-A130. 10.3940/rina.2016.a2.356.

[10] T. Iseki, and K. Ohtsu. (2000) “Bayesian Estimation of Wave Spectra Based on Ship Motions,” Control Eng. Pract., Vol 8, pp. 215–219.

[11] U. D. Nielsen, A. H. Brodtkorb, and A. J. Sorenson (2018), “A Brute-Force Spectral Approach for Wave Estimation Using Vessel Motions”, Marine Structures, 60 (2018) 101-121.

[12] R. Pascoal, L. P. Perera and G. Soares (2017), “Estimation of Directional Sea Spectra From Ship Motion in Sea Trial”, Ocean Engineering 132 (2017) 126-137.

[13] U. D. Nielsen (2006). “Estimations of directional wave spectra from measured ship responses.” Mar. Struct., Vol.19, pp.33–69

[14] A. Akinturk, H. Zaman, M. He, L. Mak and D. Seo, “Near real time estimation of neighboring actual and imminent wave fields in the vicinity of a floating body” Oceans Conference and Exposition, Oct 5-14, 2020, Paper no 200602-024.

[15] A. Akinturk, H. Zaman, D. Seo, L. Mak and M. He, “Near Real Time Estimates of Wave Field in the Vicinity of a Floating Body Using Machine Learning Techniques” IEEE AUV Conference, St. John’s, Canada, September 30 – October 2, 2020, Paper no 200530-017.

[16] Anagnostopoulos, A. (2018), “Big Data Techniques for Ship Performance Study”, Proceedings of the 28th International Ocean and Polar Engineering Conference, Sapporo, Japan, June 10-15, 2018.

[17] Sclavounos, P.D. and Ma, Y (2018), “Artificial Intelligence Machine Learning in Marine Hydrodynamics”, Proceeding of 37th International Conference on Ocean, Offshore and Arctic Engineering, Madrid, Spain, June 17-22, 2018, Paper No. OMAE2018-77599.

[18] Yin, J.C.; Zou, Z.J.; Xu, F. and Wang, N.N. (2014), “Online Ship Roll Motion Prediction Based on Grey Sequential Extreme Learning Machine”, Neurocomputing 129 (2014) 168-174.