A Novel Densely Connected Convolutional Neural Network for Sea-State Estimation Using Ship Motion Data

Xu Cheng, *Student Member, IEEE*, Guoyuan Li, *Senior Member, IEEE*, André Listou Ellefsen, Shengyong Chen, *Senior Member, IEEE*, Hans Petter Hildre, and Houxiang Zhang, *Senior Member, IEEE*

***Abstract*—The sea-state estimation is a fundamental problem in the development of autonomous ships. Traditional methods such as wave buoy, satellites, and wave radars are limited by locations, clouds, and costs, respectively. Model-based methods are prone to incorrect estimations due to their high dependence on mathematical models of ships. As previous data-driven studies for sea-state estimation consider only wave height and use the motion data from dynamic positioning (DP) vessels, this article introduces a new, deep neural network (SSENET) to estimate sea state in light of both wave height and wave direction and extends the generality of sensor data from ship motion with forward speed. SSENET is built on the basis of stacked convolutional neural network (CNN) blocks with dense connections between different blocks, channel attention modules, and a feature attention module. The dense connections build shortcut paths between input and all subsequent convolutional blocks, which can make full use of all the hierarchical features from the original time-series sensor data. The channel attention modules aim to enhance the features extracted by each convolution block. The feature attention module focuses on combining the feature fusion of hierarchical features in an adaptive manner. Benchmark experiments show the competitive performance against the state-of-the-art approaches. Applying the SSENET on two data sets of a zigzag motion for comparative studies shows the effectiveness of the proposed method.**

***Index Terms*—Autonomous ships, densely connected convolutional neural network (CNN), sea-state estimation, ship intelligence, time-series classification.**

I. Introduction

SHIP intelligence aims to make the marine and offshore industries more efficient, innovative, and adaptable to future operations. In fact, ship intelligence has been listed as an important part of the digital agenda, one of the pillars of the European growth strategy [1]. In recent years, interest in the development and employment of autonomous ships has increased. Autonomous ships use intelligence to make decisions that increase the control precision, lower the fuel consumption, and extend the operational window [2]. Autonomous ships face greater challenges than autonomous cars, mainly because of the more complicated environment at sea. Wind and waves are the most vexing aspects of this environment. Therefore, it is significant to develop a real-time and reliable method to estimate the sea state to aid onboard decisions.

Traditional technologies to estimate the sea state usually include manual observations, wave buoys, X-band wave radar, and meteorological remote sensing satellites or weather forecasts [3]. The advantage of manual observations is that the data have high persistence and do not depend on external sensors. However, manual observations themselves can be very subjective. The advantage of the wave buoy is that the observation data are highly reliable and comparable. Nevertheless, its shortcomings include weak ability to resist wind and waves, human installation and placement, and wide application in close coast. The cost of X-band wave radar is usually high and requires regular calibration and maintenance; meteorological remote sensing satellites are very susceptible to clouds; and the weather forecast is often subjected to a time delay of several hours.

In recent years, in order to overcome the shortcomings of the traditional sea-state estimation technologies, several researchers have made extensive explorations on the identification of environmental conditions based on onboard measurements [4]. A ship can be considered as a large wave buoy, and hence, it is essentially equipped with an environmental condition estimation system [5]. The use of ship motion data to identify sea states usually involves model-based and model-free methods. The model-based methods are mainly designed to utilize domain knowledge to establish a mathematical model of ship motions [6]. However, the drawback of these methods is that because they rely on mathematical models and corresponding assumptions, they are prone to

Manuscript received August 12, 2019; revised October 18, 2019; accepted December 23, 2019. Date of publication January 17, 2020; date of current version August 11, 2020. This work was supported in part by the Project "Digital Twins for Vessel Life Cycle Service" under Project 280703 and in part by the National Natural Science Foundation of China under Grant U1509207. The work of Xu Cheng was supported by the Chinese Scholarship Council. The Associate Editor coordinating the review process was Peter Liu. *(Corresponding author: Guoyuan Li.)*

Xu Cheng is with the School of Computer Science and Technology, Tianjin University of Technology, Tianjin 300384, China, and also with the Department of Ocean Operations and Civil Engineering, Norwegian University of Science and Technology, 6009 Trondheim, Norway.

Guoyuan Li, André Listou Ellefsen, Hans Petter Hildre, and Houxiang Zhang are with the Department of Ocean Operations and Civil Engineering, Norwegian University of Science and Technology, 6009 Trondheim, Norway (e-mail: guoyuan.li@ntnu.no).

Shengyong Chen is with the School of Computer Science and Technology, Tianjin University of Technology, Tianjin 300384, China.

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIM.2020.2967115

incorrect identifications due to the randomness of waves. On the other hand, the model-free methods are employing conventional machine learning or deep learning techniques to extract temporal and frequency features. The advantage of these techniques is that they do not depend on prior domain knowledge and they are easier to generalize. In other words, they can be applied to several vessels. Nevertheless, to the best of authors' knowledge, the previous model-free methods only considered the onboard measurement of dynamic positioning (DP) motion and the height of the waves without considering the direction of the waves [3], [7].

DP motion, used in [3] and [7], represents a special kind of maneuvering, which involves maintaining a fixed location or performing a very slow tracking task [8]. The use of this special maneuvering to estimate sea state lacks generality because most ships do not have a DP system and those that do are generally moving forward when in operation. Meanwhile, wave direction is as important as wave height, which can be used as a control variable for vessels. Thus, it is necessary to estimate sea state, including wave height and direction using the measurements of general ship maneuvering. To the best of authors' knowledge, this article is the first consideration of zigzag for sea-state estimation. Being different from the DP system, which is installed only in some ships, zigzag is an element of the basic maneuverability of modern ships. Thus, the use of motion data of zigzag is more common, and the trained model on this kind of motion data can be applied to more ships. There are several challenges to estimating the sea state using motion data of zigzag: first, it is necessary to select the proper variables for sea-state estimation. Second, the environmental influence relative to the ship hull itself is changing even though the external environment comes from a single direction. For the DP vessels which are maintaining a fixed position, the influence of wave is static, which makes it harder to learn the environmental information purely based on the data of zigzag motion than that with DP motion. Third, the sensor data of the ship motion in different sea states might be very similar, which makes it very hard to determine the sea state from current motion data alone. Considering both wave height and direction makes this even more difficult.

In order to utilize the sensor data of ship motion for sea-state estimation in light of both wave height and wave direction, it is necessary to extract fine-grained features. The most common solution, which employs deep learning techniques, is to use stacked convolutional neural networks (CNNs) sequentially to produce hierarchical representations. The convolutional operation can be viewed in order to extract features over the time-series sequence [9]. However, conventional connections alone cannot deal with sea states properly. Inspired by [10], we propose a densely connected CNN (SSENET) with two attention mechanisms for sea-state estimation. Through the dense connections, the SSENET can fully make use of all the hierarchical features from original time-series sensor data and all the features extracted by all convolutional blocks. The channel attention mechanism is adopted to enhance the features extracted by each convolutional block [11]. To fuse and select task-friendly features, a feature attention module is designed before the classification layer. The two characteristics enable the proposed network to obtain competitive results on benchmark data sets and ship motion data sets.

In summary, the contributions of this article include the following.

1) A new deep neural network (SSENET) is proposed that is equipped with dense connections between convolutional blocks and with two feature attention mechanisms. Through these two design considerations, the network is able to select hierarchical features for sea-state estimation that reflect both wave height and wave direction adaptively using the measurement of zigzag motion.
2) The network is extensively evaluated on 12 benchmark data sets and two ship motion data sets. The network obtains competitive performance compared to the state-of-the-art baseline models and other attention mechanisms on these data sets.

The rest of this article is organized as follows. Section II is a brief review of the previous work, mainly including the introduction of sea-state estimation and time-series classification. Section III introduces the architecture of the proposed model. The proposed method is examined on both benchmark data sets and ship motion data sets in Section IV. Section V presents the conclusion and discussion.

II. Related work

A. Onboard Measurement-Based Sea-State Estimation

The sea-state estimation based on ship motion data slowly emerged from the 1970s [12]. Onboard measurements based on sea-state estimation usually fall into two categories: model-based approaches and data-driven approaches. Model-based approaches deduce the information of the sea state by the combination of wave-induced measurements and a mathematical model [6]. Most of the work on the model-based approaches address the frequency domain and/or the time domain. In the analysis of the frequency domain, the response spectrum of ship motion is combined with response amplitude operators (RAOs), which reveals how waves are transformed into ship response, so that the estimation of the wave spectrum is given [5], [13]–[17]. Unlike the analysis in the frequency domain, the analysis of the time domain is formulating the estimation of sea state directly in the time domain. Pascoal and Soares [18] proposed a Kalman filter-based method, which relies on the accurate RAOs for the estimation of wave height and wave direction in the time domain only. Nielsen *et al.* [19] also compute sea state directly in the time domain based on the measured response and corresponding theory regarding both wave height and direction. As stated in the literature, it is reasonable to expect this will provide a good estimation of the sea state [20]. However, such estimates depend on the reliability of the RAOs [6].

Data-driven approaches are employing machine learning or deep learning techniques to extract temporal and frequency features. Even though machine learning or deep learning techniques have been widely used in other areas, they have rarely been applied to sea-state estimation. Tu *et al.* [7] proposed a multilayer classifier for sea-state estimation in terms of wave height working on salient features extracted from the time domain and frequency domain of the motion data of DP vessels. Although this method does not rely on accurate mathematical models, it requires a lot of human involvement. To reduce the influence of artificial features, Cheng *et al.* [3] proposed a deep learning-based end-to-end model for the sea-state estimation using the DP motion data. While data-driven approaches have had good results, these approaches have not considered wave direction. Moreover, previous research has used the data of DP motion when the ship keeps a certain location. The use of DP motion data for sea-state estimation is limited because ships are in motion most of the time. Therefore, it is necessary to propose an approach that accounts for ship motion.

B. Time-Series Classification

In the literature, several algorithms have been developed over the years for the time-series classification. Most of them include distance-based methods, feature-based methods, and deep learning-based methods. The distance-based methods have proven successful in classifying multivariate time-series data [21]. The feature-based methods heavily depend on the extracted features that represent the local or global patterns of time series. Baydogan *et al.* [22] proposed a bag-of-features framework (TSBF), which can extract the interval features with different time scales. Schäfer [23], [24] proposed the bag-of-symbolic Fourier approximation symbols (BOSSs) model, which combines symbolic Fourier approximation (SFA) and word bag model. The hidden-state conditional random field and hidden-unit logic model are both successful feature-based methods, which produce state-of-the-art results when used on different benchmark data sets [25]. Significant effort has been made to exploit approaches based on deep learning as a way to overcome the limitations of feature engineering. Zheng *et al.* [26] proposed a multichannel model for multivariate time-series classification. Wang *et al.* [9] proposed several baseline models for time-series classification, such as the fully convolutional network (FCN) and the residual neural network (ResNet). Karim *et al.* [27], [28] proposed the Long Short-Term Memory (LSTM)-FCN, Attentional LSTM (ALSTM)-FCN, Multivariate LSTM (MLSTM)-FCN, and Attentional MLSTM (AMLSTM)-FCN models, in which they combined with CNN and recurrent neural network (RNN) to establish an end-to-end model. Fawaz *et al.* [29] who provided an overview of most deep learning approaches for time-series classification found that the ResNet [9], which adds an identity skip connection to bypass the nonlinear transformations, obtains the best results regardless of the size of the data set.

III. Densely Connected Convolutional Networks for Sea-State Estimation

A. New Concept of Sea State

Sea state is the general condition of wave and wind on the open sea at a certain location and moment [30]. Most of the studies generally define the worldwide sea state by wave height, as shown in Table I. Data-driven approaches often label the sea state based on Table I [3], [7]. However, this labeling approach ignores the information of wave direction. This article considers both wave height and direction for the following reasons: first, in marine operations, heading waves need to be used to reduce the sloshing caused by the waves. In order to make better use of heading waves, we need to know the direction of the waves as much as possible to facilitate decision-making. Second, the wave height and direction are two very important parameters for the wave spectrum. Only wave height is focused on by the previous data-driven studies. Thus, this article is the first attempt to estimate the wave direction based on deep learning techniques in addition to the wave height. As described in Fig. 1, the open sea is divided into eight parts. The first six sea states are used, as described in Table I. The reason for using the six sea states is that the sum of the first six sea states is almost 96%. The first two sea states are very similar, and hence, are merged. Therefore, there are five wave heights coming from eight directions, that is, 40 different new sea states are created for the new concept of the sea state.

TABLE I
Code of Sea State [30]

Sea State	Description	Wave height (m)	World wide probability (%)
0	Calm (glassy)	0	—
1	Calm (ripples)	0-0.1	11.2486
2	Smooth	0.1-0.5	—
3	Slight	0.5-1.25	31.6851
4	Moderate	1.25-2.5	40.1944
5	Rough	2.5-4.0	12.8005
6	Very rough	4.0-6.0	3.0253
7	High	6.0-9.0	0.9263
8	Very High	9.0-14.0	0.1190
9	Extreme	> 14.0	0.0009

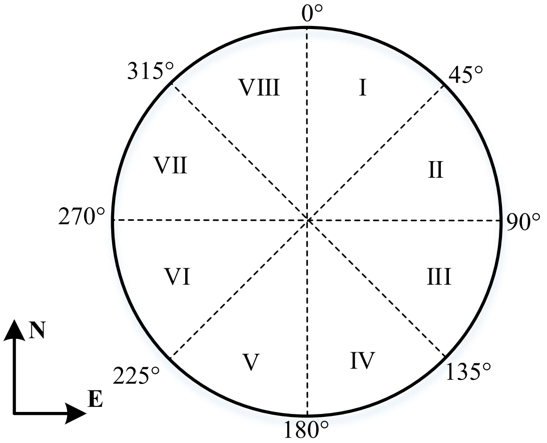

Fig. 1. Definition of the wave direction in the world coordinate.

Here, the motion data of zigzag will be used for sea-state estimation. It is easy to estimate wave height alone. However, estimating both wave height and direction simultaneously is difficult. To illustrate this problem, we take one of the onboard measurements to explain why the estimation of both wave height and direction is challenging, as shown in Figs. 2 and 3. Fig. 2 shows the changes in heave velocity in five different wave heights, while Fig. 3 represents the variations under the same wave heights (Hs = 0.1 m) but in different wave directions in zigzag motion. As shown in Figs. 2 and 3, it is

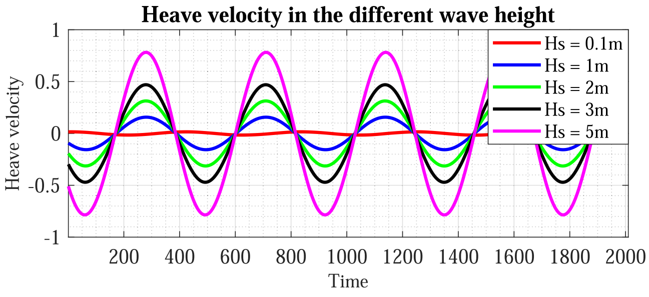

Fig. 2. Heave velocity in different wave heights.

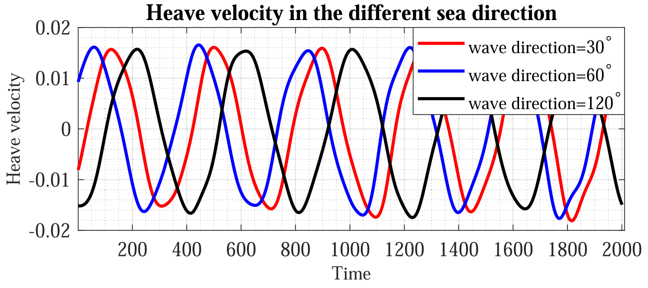

Fig. 3. Heave velocity in different wave directions.

easy to observe the distinctive feature when considering the weave height alone. However, when the wave direction is significantly different and the wave height is the same, phase shifts are small. In particular, the phase shift is smaller when the wave direction is almost the same. Thus, the designed network should be equipped with a powerful capability to represent fine-grained features.

B. Network Structure

Inspired by the powerful capability of feature extraction of ResNet, we think that it is possible to improve the performance of ResNet by adopting the idea of dense CNN [10]. The proposed network is mainly used for sea-state estimation (which is why it is named as SSENET), which takes the advantage of the structure of ResNet in time-series classification and ensures the maximum information flow between layers. The proposed SSENET consists of four parts that are depicted in Fig. 4: data processing, convolutional blocks and dense connections, channel attention, and feature attention. For the sake of simplicity, we only show three convolutional blocks. The model begins with the processed ship motion data and generates hierarchical features by convolution blocks and dense connections (denoted by the colored lines). The dense connections enable SSENET to form hierarchical features flexibly because each block receives additional input from all preceding blocks and passes its own feature maps to all subsequent blocks. To select task-friendly features, two attention mechanisms are presented to reweigh these hierarchical feature maps. A final representation is then built for sea-state classification.

C. Data Processing

With the development of vessels, all kinds of onboard measurements could be collected. In this article, we only consider the onboard 9-degree-of-freedom (DOF) measurements (surge velocity, sway velocity, heave velocity, roll angle, roll velocity, pitch angle, pitch velocity, yaw angle, and yaw velocity), which can be obtained from the inertial measurement unit (IMU) as initial input. The data processing mainly focuses on data cleaning, phase correction, and feature selection. It is necessary to clean the noise and redundant information to minimize its effect on further analysis and modeling [31]. To get rid of the noise, median filtering methods are employed in this article. For the roll angel, yaw angel, and pitch angle, the physical definition creates some jumping phenomena. The algorithm developed in our previous article is utilized [31]. A mutual information-based feature selection method is employed to select the most influential sensor data to sea state [32].

D. Convolutional Block and Dense Connection

The convolutional block consists of three basic 1-D CNNs and three channel attention blocks, as depicted by C-Attention in Fig. 4. The activation function is the rectified linear unit (ReLU) [33] for each basic 1-D CNN, and the feature extracted by CNN will be processed by a batch normalization (BN) [34] layer. The convolution operation is done by the kernel with the preset size.

The 1-D CNN operation is

$$\begin{aligned} s &= W \otimes X + b \\ s &= BN(s) \\ s &= ReLU(s) \end{aligned} \tag{1}$$

where X represents the input, W and b stand for the trainable weights and bias, respectively, and $\otimes$ is the convolution operator. After 1-D CNN operation, the feature maps are enhanced by channel attention. Assuming the function of the channel attention module is $C_l(\cdot)$, the output can be represented by $y = C_l(s)$. The final convolutional block is built by stacking three 1-D CNNs and channel attention modules with the preset filter sizes.

Each block can receive the feature maps from all preceding blocks

$$X_l = F_l(W, [x_0, x_1, \ldots, x_{l-1}]) \tag{2}$$

where $[x_0, x_1, \ldots, x_{l-1}]$ is the concatenation of the feature extracted in blocks $0, \ldots, l-1$. W is the learnable parameter. $F_l(\cdot)$ is the composite function of each block.

E. Channel Attention Module

Channel attention, which is defined in [11], is utilized to exploit the latent relationship of features in channels. The design of this module is focusing on the meaningful part of a given input image. We adopt the idea of channel attention for time-series classification. There are two channel features, $\mathrm{C_{avg}}$ and $\mathrm{C_{max}}$, which utilize global average pooling and global max pooling, respectively. As described in [11], both $\mathrm{C_{avg}}$ and $\mathrm{C_{max}}$ can gather more vital information than each of them separately. It is better to use both of them to infer channel-wise features. Both features are forwarded to a shared one hidden layer multilayer perceptron (MLP) to produce the feature map. After the shared MLP, the feature vector is obtained by the element

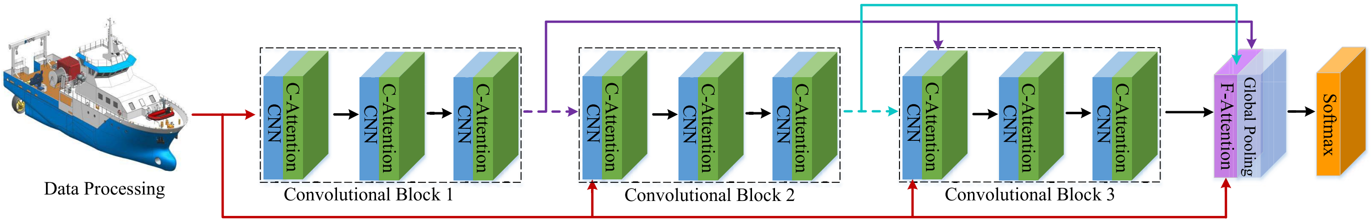

Fig. 4. Illustration of the proposed SSENET.

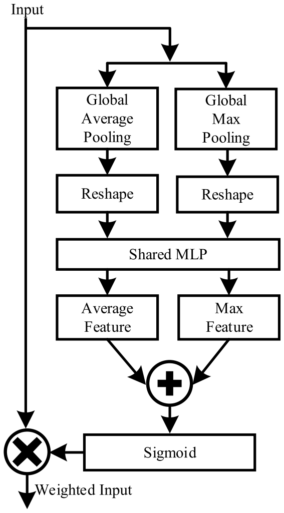

Fig. 5. Channel attention module.

summation. Finally, the final weighted input can be computed using the sigmoid transformation. The whole process for the channel attention module is shown in Fig. 5.

In short, the weights of channel attention and weighted inputs are computed as follows:

$$\begin{aligned} \alpha_{C_{\text{att}}} &= \text{Sigmoid}(\mathbf{MLP}(\text{C}_{\text{avg}}) + \mathbf{MLP}(\text{C}_{\text{max}})) \\ X_{\text{weighted}} &= \alpha_{C_{\text{att}}} \otimes \mathbf{X} \end{aligned} \tag{3}$$

where X is the original input, $\otimes$ means the element-wise multiply, and C_{avg} and C_{max} share the weights of **MLP**. The reweighted features will be sent to the next and all subsequent 1-D CNNs.

F. Feature Attention Module

Through dense connections, the downstream layers of the network can access the features generated by the upstream layer. However, how to utilize these features (some are redundant) effectively remains a key issue. Therefore, we propose a feature attention mechanism to effectively make use of these features for classification. The feature attention module includes two operations: feature ensemble and feature weight, which is depicted in Fig. 6. It is noteworthy that those parts except feature ensemble are called feature weight. The feature ensemble is designed to recombine the features from different convolutional blocks. These new features are employed as the input to generate attention weights to reweight the features adaptively. The whole processing will be realized during the training processing.

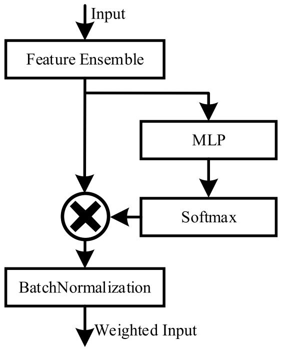

Fig. 6. Feature attention module.

Assuming that the feature from different convolutional blocks are $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_L$, the new feature can be represented as $X_{\text{ensemble}} = [\mathbf{X}_1; \mathbf{X}_2; \ldots; \mathbf{X}_L]$.

The attention weight α and the weighted inputs can be computed as follows:

$$\begin{aligned} \boldsymbol{\alpha} &= \text{softmax}(\mathbf{MLP}(X_{\text{ensemble}})) \\ X_{\text{weighted}} &= \alpha \otimes \text{X}_{\text{ensemble}}. \end{aligned} \tag{4}$$

After the attention calculation, the weighted input will be fed into a softmax classification layer that corresponds to 40 sea states.

IV. Experiment

All experiments were performed on a server equipped with an Intel Xeon processor, 128-GB RAM and Nvidia Tesla K80 and 24-GB RAM. The software environment used is Anaconda[1] Python 3.6, and all the layers are implemented by Keras 2,[2] using TensorFlow[3] as the backend.

A. Data Set

1) *Benchmark Data Set:* The proposed SSENET is evaluated on 12 public data sets, which are used for multivariate time-series classification [28]. The detailed information of the 12 public data sets is shown in Table II. These data sets contain several domains, and the number of classes and the number of variables differ greatly. Moreover, these data sets have been preprocessed and split into training and testing data sets. Instead of rerunning other methods on these data sets,

[1] https://anaconda.org/
[2] https://keras.io
[3] https://www.tensorflow.org/

TABLE II
INFORMATION OF BENCHMARK DATA SETS

Datasets	Classes	Variables	Length	Domain	Train/Test
AREM	7	7	480	Activity Recognition	50%/50%
HAR	6	9	128	Activity Recognition	71%/29%
Daily Sport	19	45	125	Activity Recognition	50%/50%
Gesture Phase	5	18	214	Gesture Recognition	50%/50%
EEG	2	13	117	EEG Recognition	50%/50%
EEG2	2	64	256	EEG Recognition	20%/80%
HT Sensor	2	11	5396	Food Recognition	50%/50%
Movement AAL	2	4	119	Movement Recognition	50%/50%
Occupancy	2	5	3758	Occupancy Recognition	35%/65%
Ozone	2	72	291	Weather Recognition	50%/50%
Action 3d	20	570	100	Activity Recognition	50%/50%
Activity	16	570	337	Activity Recognition	50%/50%

we just duplicated the results from four state-of-the-art methods [27], [28] reported by their respective authors in their publications to ensure a fair comparison.

2) *Ship Motion Data Set:* Two data sets of zigzag motion are collected from the digital ship of the Norwegian University of Science and Technology (NTNU)'s research vessel, R/V Gunnerus [35]. The reasons to collect two data sets are to eliminate the influence of the specific data set and to study the performance of SSENET in different data sets. The difference of simulation setting for the two data sets comes from the wave directions and peak-to-peak period. The first data set, with a peak-to-peak period 10 s, contains the following wave directions: 30°, 60°, 120°, 150°, 210°, 250°, 300°, and 330°. The second one, with a peak-to-peak period 15 s, includes: 20°, 50°, 110°, 160°, 200°, 240°, 280°, and 350°. It is also noteworthy that the same zigzag command would be executed in the 40 sea states. The two data sets are split into training and testing by 80%–20%.

B. Benchmark Comparison

To illustrate the feasibility of our proposed model, we first compare our model with state-of-art methods. In these tests, the number of convolutional blocks is set to 2. In this article, we adopt the idea of setting the number of filters in FCN [27]. Those hyperparameters can also be optimized by the parameter tuning algorithms [36]. The numbers of filters in the first block are 128, 256, and 128, while the numbers of filters in the second block are 256, 512, and 256. The kernel sizes in the two blocks are 8, 5, and 3. All the networks in this section are trained in two steps. They are trained initially using the Adam optimizer [37]. The initial and final learning rates are set to 1eminus3 and 1eminus4, respectively. The learning rate changes every 50 epochs using a factor of $(1/\sqrt[3]{2})$. The mini-batch is set to 128 in the first-step training. The second step is to perform fine-tuning of the network trained in the first step on the whole original data set. The fine-tuning process is repeated five times. The mini-batch and learning rate are first set to 32 and 1eminus3 and then reduced by half at the end of each iteration.

To verify the performance of the proposed model and the training algorithm, the testing results of the proposed model with and without fine-tuning are provided, as shown in Table III. From the results without fine-tuning, it is observed that there is 3.01% improvement on EEG2, 2.76% on Gesture Phase, 0.17% on Human Activity Recoginition (HAR), 4.9% on Occupancy, and 14.83% on Action 3-D. It is interesting that our model can achieve better results on data sets, which consists of a testing set bigger than the training set. This means that our model can extract more features with the help of its deep and flexible architecture. From the results with fine-tuning, SSENET achieves the highest average accuracy and wins on most of the benchmark data sets. Our network is able to achieve good results on most data sets. Specifically, our network obtained almost 14.82% improvement in Action 3-D, 7% in EEG2, and 5.1% in Activity Recognition system based on Multisensor (AREM). By comparing the result with and without model fine-tuning, the fine-tune is very helpful on some data sets, such as AREM with 7.9% improvement, EEG with 11.9%, EEG2 with 4.2%, Gesture Phase with 0.92%, HAR with 0.84%, humidity and temperature (HT) Sensor with 9.5%, Ozone with 4.8%, and Action 3-D with 1.87%. The other data sets got the same result whether the fine-tuning is performed or not. From the benchmark tests, we can also know that the SSENET can be applied to several tasks in different domains.

C. Data Analysis and Feature Selection

In the literature, only a few parameters would be used for sea-state estimation both in model-based methods and model-free methods. In [7], surge velocity, sway velocity, roll angle, and yaw angle are selected. In [4] and [5], heave velocity, roll, and pitch angle are chosen. In [20], heave velocity, pitch angle, roll angle, and sway velocity are utilized. While in [16], sway velocity, heave velocity, pitch angle, and yaw angle are employed. In the conventional model-based methods, these variables are selected because they are wave-induced responses, which can infer from the mathematical models. In this article, we use a mutual information-based variable selection method to explain why these variables are important from a data perspective. The input parameters are the 9-DOF onboard measurements, and the 40 classes are the output. From Fig. 7, we can obviously know that the heave velocity, pitch angle, pitch velocity, and yaw angle are the first four most important variables to the sea state. In order to keep the same number of variables with other sea state methods, four variables are chosen as the input of our proposed model SSENET.

D. Baseline Comparison

We compare our model with six baselines as follows.

1) *MLP:* Three stacked fully connected layers are used in MLP with 500 neurons in each layer and the ReLU used as the activation function. The dropout layer with a dropping rate of 0.8 is utilized between layers [38].

TABLE III

ACCURACY COMPARISON WITH THE STATE-OF-THE-ART TIME-SERIES CLASSIFICATION METHODS (%)

Datasets	**LSTM-FCN**[27]	**MLSTM-FCN**[28]	**ALSTM-FCN**[27]	**MALSTM-FCN**[28]	**Other methods** [28]	**SSENET**	**SSENET(FT)**[*]
AREM	89.74	92.31	82.05	92.31	76.92 [DTW]	89.74	**97.44**
Daily Sport	99.65	99.65	99.63	**99.72**	98.42 [DTW]	99.61	99.61
EEG	60.94	**65.63**	64.06	64.07	62.5 [RF]	57.81	**65.63**
EEG2	90.67	91	90.67	91.33	77.5 [RF]	94.17	**98.33**
Gesture Phase	50.51	53.53	52.53	53.05	40.91 [DTW]	55.05	**55.56**
HAR	96	96.71	95.49	96.71	81.57 [RF]	96.87	**97.69**
HT Sensor	68	78	72	80	72 [DTW]	76	**84**
Movement AAL	73.25	**79.63**	70.06	78.34	65.61 [SVM-Poly]	77.71	77.71
Occupancy	71.05	76.31	71.05	72.37	67.11 [DTW]	**80.26**	**80.26**
Ozone	67.63	81.5	79.19	79.78	75.14 [DTW]	79.77	**83.82**
Action 3d	71.72	75.42	72.73	74.74	70.71 [DTW]	88.55	**90.24**
Activity	53.13	61.88	55.63	58.75	**66.25** [DTW]	65.00	65.63
Accuracy	74.36	79.30	75.42	78.43	71.22	80.05	**82.99**
No. of wins	0	2	0	1	1	1	**9**
Ranking	5	2	4	3	—	—	**1**

[*] FT means the models have been fine-tuned.

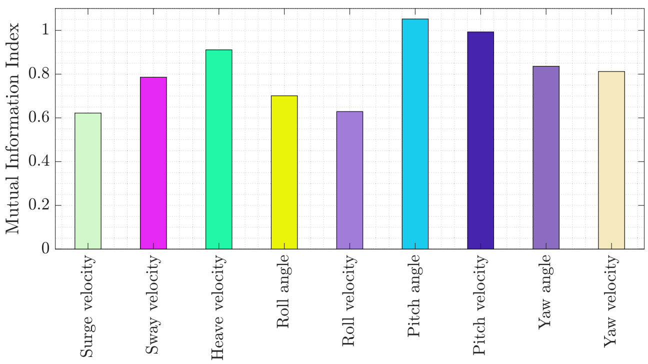

Fig. 7. Results of mutual information.

TABLE IV

ACCURACY COMPARISON WITH BASELINES ON SHIP MOTION DATA

Methods	**Type**	**dataset 1**	**dataset 2**	**average**
MLP [38]	Simple NN	75%	77.12%	76.06%
LSTM	Simple NN	75%	79.81%	77.41%
CNN	Pure CNN	75.58%	78.65%	77.12%
FCN [28]	Pure CNN	75.38%	81.35%	78.37%
ResNet [29]	Pure CNN	78.27%	80.96%	79.62%
MLSTM-FCN [28]	Combined NN	75.19%	80.77%	77.98%
SeaStateNet [3]	Combined NN	80.38%	84.42%	82.40%
SSENET	Pure CNN	**89.81%**	**93.46%**	**91.64%**

2) *CNN:* Two 1-D convolutional layers are employed with the sigmoid activation function, and average pooling is used between layers. We choose the best network for comparison from four different sets of filters $\{64, 128, 256, 512\}$.
3) *FCN:* The FCN has the same settings as in MLSTM-FCN [28].
4) *MLSTM-FCN:* The same settings as for MLSTM-FCN are adopted from [28].
5) *LSTM:* Five different LSTMs are trained with different numbers of hidden units $\{8, 16, 32, 64, 128\}$.
6) *ResNet:* We use the same settings as in ResNet from [29].
7) *SeaStateNet:* SeaStateNet is specifically designed for sea-state estimation [3]. SeaStateNet is composed of three parallel parts: LSTM part, CNN part, and fast Fourier transform (FFT) part.

In this section, the hyperparameters of SSENET are the same as in Section IV-B, but it is trained without fine-tuning. The settings of hyperparameters and training algorithm for SSENET for all the following experiments are the same as in this section and will not be further elaborated. The proposed SSENET is compared with the baseline methods on the two ship motion data sets. To fully test our model, we present the best performance of each method in Table IV. Among these methods, CNN, FCN, ResNet, and SSENET are pure CNNs. LSTM-FCN and SeaStateNet belong to different combinations of neural networks. LSTM-FCN consists of two parts: LSTM and FCN which work parallelly, and SeaStateNet is composed of three parts: LSTM part, CNN part, and FFT part, which also work parallelly. In terms of accuracy, our proposed network clearly outperforms all the baseline methods on both data sets. The worst performance occurs in MLP, and the performance of LSTM is relatively better with the capability of learning the periodic features. SSENET shows 10.50% and 9.67% improvement compared to the SeaStateNet on data set 1 and data set 2, and 19.4% and 15.7% improvement compared to the MLSTM-FCN on both data sets. From the results, we can also see that the SeaStateNet is better than MLSTM-FCN. The reason might be that the SeaStateNet has one more FFT part, which can extract features in the frequency domain. For these pure convolution neural networks, ResNet performs better than CNN and FCN, which reveals the advantage of its complex structure to extract features from this kind of ship motion data. Compared to ResNet, the SSENET shows 12.85% and 13.37% improvement on the data set 1 and data set 2. The reasonable explanation could be that the proposed SSENET is based on the ResNet with additional dense connection and attention modules, which can improve the performance significantly.

E. Ablation Study

To conduct the ablation study, four variants are compared.

1) *SSENET-Attention:* The two-attention mechanisms: C-Attention and F-Attention as shown in Fig. 4 are removed.
2) *SSENET-C-Attention:* There is no C-Attention in SSENET.

TABLE V
ABLATION STUDY

Methods	dataset 1	dataset 2	average
SSENET-Attention	81.92%	86.15%	84.04%
SSENET-C-Attention	85.77%	88.85%	87.31%
SSENET-F-Attention	87.88%	90.77%	89.33%
SSENET-Connection	87.88%	90.58%	89.23%
SSENET	**89.81%**	**93.46%**	**91.64%**

TABLE VI
COMPARISON STUDY OF ATTENTION MODULES

Methods	dataset 1	dataset 2	average
CBAM [11]	81.92%	82.69%	82.31%
Global_text [39]	85.77%	81.92%	83.85%
SE [40]	87.88%	83.27%	85.58%
SSENET	**89.81%**	**93.46%**	**91.64%**

3) *SSENET-F-Attention:* To validate the F-Attention mechanism, we remove it from SSENET directly.
4) *SSENET-Connection:* This variant is constructed using the stacked convolutional blocks, that is, the variant does not consider the dense connections.

Each variant was tested on the two data sets. To present an equal measurement, we present the best performance of the four variants. From Table V, we observe that: 1) the biggest accuracy drop happens when there are no attention modules; 2) in terms of average accuracy, the accuracy drops by 4.33% when the C-Attention module is removed. However, the accuracy only drops by 2.31% when there is no F-Attention module. This means that the C-Attention module is more important than the F-Attention module; and 3) the full combination of the two attention modules show superiority against the variants SSENET-Attention (without attention modules), SSENET-C-Attention (C-Attention is removed), and SSENET-F-Attention (F-Attention is removed), which demonstrates the importance of the proposed two attention mechanisms.

To illustrate the importance of dense connections, we further compare the network with and without the dense connections. As we can see from Table V, the accuracy drops by 1.93% and 2.88% in data set 1 and data set 2, respectively. In terms of average accuracy, the accuracy drops by 2.41%. From the results, the dense connection can improve the accuracy by considering the fusion of hierarchical features. Furthermore, despite the usage of dense connection, the computational cost does not increase significantly.

F. Comparison With Other Attention Mechanisms

To further verify the importance of the proposed attention mechanisms, the proposed network is compared with three attention mechanisms. As shown in Fig. 4, the number pertaining to the C-Attention module is much greater than that pertaining to the F-Attention module. The C-Attention module is replaced by one attention module at each comparison. From Table VI, we observe that the Squeeze-and-Excitation (SE) obtained the best performance. It is noteworthy that the C-Attention module is one part of convolutional block attention module (CBAM). The CBAM is applying the C-Attention module and the spatial attention module sequentially, so that it can learn where to focus and on what in the channel and spatial axes [11]. While this article only considers the idea of channel attention, it is interesting that CBAM is not as good as C-Attention, even though it is better than C-Attention in the original article for computer vision applications. The reason may be that CBAM is good at extracting more influential features in the 3-D image data than in 2-D time-series data. In the practice of SE, only the average-pooled features are exploited, missing the importance of max-pooled features which is verified by the experimental results.

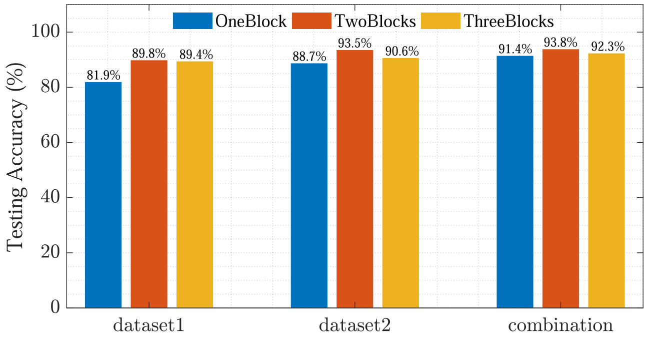

Fig. 8. Influence of the number of convolutional blocks.

G. Sensitivity Analysis of Network Structure

The sensitivity analysis of the network structure focuses on the influence of the number of convolutional blocks and the number of 1-D CNNs in each convolutional block. To study the influence of the number of convolutional blocks, three different networks are created. The first network contains one convolutional block, as shown in Fig. 4, with the filter numbers $\{128, 256, \text{and } 128\}$. The second network has two convolution blocks, which is tested in previous experiments. The third network includes three convolutional blocks with the number of filters $\{128, 256, 128, 256, 512, 256, 128, 256, \text{and } 128\}$. These networks are compared using three data sets: data set 1 and data set 2 as described above, and a third data set that combines data set 1 and data set 2. The use of the third data set makes it possible to train more parameters as the depth of the networks increases. We trained all the networks several times, and the networks with the best performance are chosen for comparison. Fig. 8 represents the influence of the number of convolutional blocks. As we expect, the network just with one convolutional block has the worst performance, and the network containing two convolutional blocks obtains the best accuracy in the three data sets. Another finding is that the accuracy of the three networks in the data set 1 and data set 2 is consistent with the previous experiment. However, higher accuracy is obtained in the combination of the two data sets. The explanation is that the bigger data set can provide more information resulting in better accuracy.

To investigate the influence of the number of 1-D CNNs in each convolutional block, four networks are established with one to four 1-D CNNs, respectively. In this comparison, the number of convolutional blocks is set to 2. The corresponding settings for the numbers of filters in each 1-D CNN are: $\{128\}, \{128, 256\}, \{128, 256, 128\}, \text{ and } \{128, 256, 256, 128\}$.

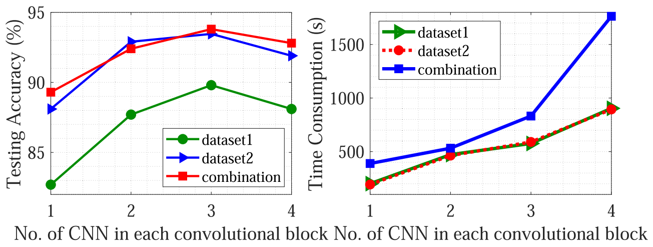

Fig. 9. Influence of the number of 1-D CNNs in each convolutional block.

The other settings for hyperparameters are the same as shown in Section IV-B. Those networks are also tested on the three data sets as mentioned above. Fig. 9 describes the influence of the numbers of 1-D CNNs in each convolutional block. The left panel shows the validation accuracy in the three data sets, and the right panel represents the training time of the four networks in the three data sets. It is obvious that the highest accuracy happens when there are three 1-D CNNs, and the accuracy is almost the same when there are two and four 1-D CNNs. In addition, the training time increases significantly when there are four 1-D CNNs. According to the above experiments, the best choice is to select three 1-D CNNs considering both accuracy and consumed time.

V. Discussion and Conclusion

This article introduces a new deep neural network to estimate sea state based on ship motion data of zigzag considering both wave height and direction. The network is built on the basis of stacked CNN blocks with dense connections between different blocks, channel attention modules, and a feature attention module. The dense connections build shortcut paths between input and all subsequent convolutional blocks. The channel attention module aims to enhance the features extracted by each convolution block. The feature attention module focuses on the feature fusion of hierarchical features jointly and adaptively.

There are some interesting findings in this article. One is that the proposed SSENET achieved a different accuracy in data set 1 and data set 2 even though there is not very much difference between the two data sets. The reason why this occurred is that the distribution of the training set and the test set differs somewhat when the two are split randomly from the original one. Superficially, the proposed SSENET is quite similar to ResNet. However, SSENET has concatenation instead of summation. From the experimental results, the seemingly small modification has resulted in different behaviors of the two networks. Thanks to the input concatenation, the feature extracted by any layers of SSENET can be accessible to by all subsequent layers. In this design, the features can be fully reused throughout the network and lead to a more network. Another possible explanation for the improved accuracy of the proposed SSENET is that the two attention modules may have enhanced the extracted features. However, the ablation study shows that it is still not easy to identify which attention module is more important even though there are more accuracy drops when the channel attention module is removed. The reason is that there are more channel attention modules than feature modules.

Future research will employ more tests to determine the importance of each part of SSENET. Furthermore, the hyperparameters should be optimized to find the best network structure. The third and most important point is that we need to integrate SSENET into a ship motion monitoring system to serve an autonomous ship.

References

[1] X. Cheng, G. Li, R. Skulstad, S. Chen, H. P. Hildre, and H. Zhang, “A neural-network-based sensitivity analysis approach for data-driven modeling of ship motion,” *IEEE J. Ocean. Eng.*, to be published.

[2] R. Jalonen, R. Tuominen, and M. Wahlström, “Remote and autonomous ships—The next steps: Safety and security,” Adv. Auton. Waterborne Appl. (AAWA), Rolls-Royce Holdings, London, U.K., Tech. Rep., 2016, pp. 56–73. [Online]. Available: https://www.rolls-royce.com/~/media/Files/R/Rolls-Royce/documents/customers/marine/ship-intel/aawa-whitepaper-210616.pdf

[3] X. Cheng, G. Li, R. Skulstad, S. Chen, H. P. Hildre, and H. Zhang, “Modeling and analysis of motion data from dynamically positioned vessels for sea state estimation,” in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 6644–6650.

[4] U. D. Nielsen, A. H. Brodtkorb, and A. J. Sørensen, “A brute-force spectral approach for wave estimation using measured vessel motions,” *Marine Struct.*, vol. 60, pp. 101–121, Jul. 2018.

[5] A. H. Brodtkorb, U. D. Nielsen, and A. J. Sørensen, “Sea state estimation using vessel response in dynamic positioning,” *Appl. Ocean Res.*, vol. 70, pp. 76–86, Jan. 2018.

[6] U. D. Nielsen, “A concise account of techniques available for shipboard sea state estimation,” *Ocean Eng.*, vol. 129, pp. 352–362, Jan. 2017.

[7] F. Tu, S. S. Ge, Y. S. Choo, and C. C. Hang, “Sea state identification based on vessel motion response learning via multi-layer classifiers,” *Ocean Eng.*, vol. 147, pp. 318–332, Jan. 2018.

[8] A. J. Sørensen, “A survey of dynamic positioning control systems,” *Annu. Rev. Control*, vol. 35, no. 1, pp. 123–136, Apr. 2011.

[9] Z. Wang, W. Yan, and T. Oates, “Time series classification from scratch with deep neural networks: A strong baseline,” in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 1578–1585.

[10] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.

[11] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, “CBAM: Convolutional block attention module,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.

[12] K. Lindemann, J. Odland, and J. Strengtheagen, “On the application of hull surveillance systems for increased safety and improved structural utilization in rough weather,” Dept. Mar. Transp. Technol., Soc. Nav. Architects Mar. Eng., Hoboken, NJ, USA, Tech. Rep. 3, 1977.

[13] T. Iseki and K. Ohtsu, “Bayesian estimation of directional wave spectra based on ship motions,” *Control Eng. Pract.*, vol. 8, no. 2, pp. 215–219, Feb. 2000.

[14] U. D. Nielsen, “Estimations of on-site directional wave spectra from measured ship responses,” *Marine Struct.*, vol. 19, no. 1, pp. 33–69, Jan. 2006.

[15] R. Pascoal, C. G. Soares, and A. J. Sørensen, “Ocean wave spectral estimation using vessel wave frequency motions,” *J. Offshore Mech. Arctic Eng.*, vol. 129, no. 2, pp. 90–96, May 2007.

[16] R. Pascoal, L. Perera, and C. G. Soares, “Estimation of directional sea spectra from ship motions in sea trials,” *Ocean Eng.*, vol. 132, pp. 126–137, Mar. 2017.

[17] U. D. Nielsen and A. H. Brodtkorb, “Ship motion-based wave estimation using a spectral residual-calculation,” in *Proc. OCEANS-MTS/IEEE Kobe Techno-Oceans (OTO)*, May 2018, pp. 1–9.

[18] R. Pascoal and C. G. Soares, “Kalman filtering of vessel motions for ocean wave directional spectrum estimation,” *Ocean Eng.*, vol. 36, nos. 6–7, pp. 477–488, May 2009.

[19] U. D. Nielsen, R. Galeazzi, and A. H. Brodtkorb, “Evaluation of shipboard wave estimation techniques through model-scale experiments,” in *Proc. OCEANS-Shanghai*, Apr. 2016, pp. 1–8.

[20] N. Montazeri, U. D. Nielsen, and J. Juncher Jensen, “Estimation of wind sea and swell using shipboard measurements—A refined parametric modelling approach,” *Appl. Ocean Res.*, vol. 54, pp. 73–86, Jan. 2016.

[21] C. Orsenigo and C. Vercellis, "Combining discrete SVM and fixed cardinality warping distances for multivariate time series classification," *Pattern Recognit.*, vol. 43, no. 11, pp. 3787–3794, Nov. 2010.

[22] M. G. Baydogan, G. Runger, and E. Tuv, "A bag-of-features framework to classify time series," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2796–2802, Nov. 2013.

[23] P. Schäfer, "The BOSS is concerned with time series classification in the presence of noise," *Data Mining Knowl. Discovery*, vol. 29, no. 6, pp. 1505–1530, Nov. 2015.

[24] P. Schäfer, "Scalable time series classification," *Data Mining Knowl. Discovery*, vol. 30, no. 5, pp. 1273–1298, Sep. 2016.

[25] W. Pei, H. Dibeklioglu, D. M. J. Tax, and L. Van Der Maaten, "Multivariate time-series classification using the hidden-unit logistic model," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 4, pp. 920–931, Apr. 2018.

[26] Y. Zheng, Q. Liu, E. Chen, Y. Ge, and J. L. Zhao, "Exploiting multi-channels deep convolutional neural networks for multivariate time series classification," *Front. Comput. Sci.*, vol. 10, no. 1, pp. 96–112, Feb. 2016.

[27] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "LSTM fully convolutional networks for time series classification," *IEEE Access*, vol. 6, pp. 1662–1669, 2018.

[28] F. Karim, S. Majumdar, H. Darabi, and S. Harford, "Multivariate LSTM-FCNs for time series classification," 2018, *arXiv:1801.04503*. [Online]. Available: https://arxiv.org/abs/1801.04503

[29] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Müller, "Deep learning for time series classification: A review," *Data Mining Knowl. Discovery*, vol. 33, no. 4, pp. 917–963, Jul. 2019.

[30] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. Hoboken, NJ, USA: Wiley, 2011.

[31] G. Li, H. Zhang, B. Kawan, H. Wang, O. L. Osen, and A. Styve, "Analysis and modeling of sensor data for ship motion prediction," in *Proc. OCEANS-Shanghai*, Apr. 2016, pp. 1–7.

[32] J. R. Vergara and P. A. Estévez, "A review of feature selection methods based on mutual information," *Neural Comput. Appl.*, vol. 24, no. 1, pp. 175–186, Jan. 2014.

[33] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 807–814.

[34] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*. [Online]. Available: https://arxiv.org/abs/1502.03167

[35] S. Skjong, M. Rindarøy, L. T. Kyllingstad, V. Æsøy, and E. Pedersen, "Virtual prototyping of maritime systems and operations: Applications of distributed co-simulations," *J. Mar. Sci. Technol.*, vol. 23, no. 4, pp. 835–853, Dec. 2018.

[36] H. Jin, Q. Song, and X. Hu, "Efficient neural architecture search with network morphism," 2018, *arXiv:1806.10282*. [Online]. Available: https://arxiv.org/abs/1806.10282

[37] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980

[38] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[39] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "GCNet: Non-local networks meet squeeze-excitation networks and beyond," 2019, *arXiv:1904.11492*. [Online]. Available: https://arxiv.org/abs/1904.11492

[40] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

Xu Cheng (Student Member, IEEE) received the master's degree in computer science and technology from the Zhejiang University of Technology, Hangzhou, China, in 2015. He is currently pursuing the joint Ph.D. degree with the Norwegian University of Science and Technology (NTNU), Trondheim, Norway, as part of the Mechatronics Laboratory, Department of Ocean Operations and Civil Engineering, and with the Tianjin University of Technology, Tianjin, China.

His current research interests include sea state estimation, data analysis, neural network, and ship motion modeling.

Guoyuan Li (Senior Member, IEEE) received the Ph.D. degree from the Department of Informatics, Institute of Technical Aspects of Multimodal Systems (TAMS), University of Hamburg, Hamburg, Germany, in 2013.

Since 2014, he has been with the Mechatronics Laboratory, Department of Ocean Operations and Civil Engineering, Norwegian University of Science and Technology (NTNU), Trondheim, Norway. In 2018, he became an Associate Professor of ship intelligence. He has authored or coauthored more than 50 articles. His research interests include path planning, ship motion prediction, maneuvering control, artificial intelligence, optimization algorithms, and locomotion control of bioinspired robots.

André Listou Ellefsen received the master's degree in subsea technology from the Norwegian University of Science and Technology (NTNU), Trondheim, Norway, in 2016, where he is currently pursuing the Ph.D. degree with the Mechatronics Laboratory, Department of Ocean Operations and Civil Engineering.

His current research interests include artificial intelligence, deep learning, decision support, predictive maintenance, and digital twins.

Shengyong Chen (Senior Member, IEEE) received the Ph.D. degree in computer vision from the City University of Hong Kong, Hong Kong, in 2003.

He was with the University of Hamburg, Hamburg, Germany, from 2006 to 2007. He is currently a Professor with the Tianjin University of Technology, Tianjin, China. He has authored or coauthored over 100 scientific articles in international journals. His research interests include computer vision, robotics, and image analysis.

Dr. Chen is a fellow of IET and a Senior Member of CCF. He received a Fellowship from the Alexander von Humboldt Foundation of Germany and the National Outstanding Youth Foundation Award of China in 2013.

Hans Petter Hildre is currently a Professor and the Head of the Department of Ocean Operations and Civil Engineering, Norwegian University of Science and Technology (NTNU), Trondheim, Norway. He is also the Centre Director of the Centre for Research Driven Innovation (SFI-MOVE) within marine operations. This is cooperation between NTNU, SINTEF, Trondheim, Norway, University Sao Paulo, Butanta, Brazil, and 15 companies at the west coast of Norway. He is the Head of research in National Program Global Center of Expertise Blue Maritime, the project leader in several research projects, and a member of the board in five companies. He holds number of patents. His area of interest is product design and system architecture design.

Houxiang Zhang (Senior Member, IEEE) received the Ph.D. degree in mechanical and electronic engineering in 2003 and the Habilitation degree in informatics from the University of Hamburg, Hamburg, Germany, in 2011.

From 2004, he worked as a Post-Doctoral Fellow with the Department of Informatics, Faculty of Mathematics, Informatics and Natural Sciences, Institute of Technical Aspects of Multimodal Systems (TAMS), University of Hamburg. He joined the Norwegian University of Science and Technology (NTNU), Trondheim, Norway (before 2016, Aalesund University College), in April 2011, where he is currently a Professor of robotics and cybernetics. The focus of his research lies on two areas. One is on biological robots and modular robotics. The second focus is on virtual prototyping and maritime mechatronics. He has authored or coauthored over 130 journal and conference articles and book chapters.