Classifying motion states of AUV based on graph representation for multivariate time series

Chen Feng [a,*], Shuang Gao [a], Simin Chen [a], Zhongke Gao [b], Celso Grebogi [c]

[a] *College of Electronic Engineering, Ocean University of China, Qingdao, China*

[b] *The School of Electrical and Information Engineering, Tianjin University, Tianjin, China*

[c] *Institute for Complex Systems and Mathematical Biology, King's College, University of Aberdeen, Aberdeen, UK*

Abstract:

Motion state monitoring and recognition are the important issues to be dealt to improve the reliability of Autonomous Underwater Vehicle (AUV). In this work, we transform the motion state classification into Multivariate Time Series Classification (MTSC). By combining two kinds of MTSC methods, including the methods based on feature representation transformation and Deep Neural Network (DNN), we propose a new classification method for Multivariate Time Series (MTS). Firstly, multivariate monitoring data of AUV are fused to construct the complex networks as the graphs to represent the motion states of AUV. And then, Graph Convolutional Neural Network (GCNN) is used to extract the features of the graphs and classify the graphs. For validating the effectiveness of our method, navigational sea experiments are carried out with the measured data of three types of motion states of AUV. The experimental results show that the graphical representation based on complex networks can effectively describe the motion states of AUV. Compared with Support Vector Machine (SVM), the graphical features are extracted automatically by GCNN to get a higher accuracy of classification of the motion states of AUV. The experiments also show that the classification accuracy of our method is higher than that of other two DNNs.

Key words: AUV; MTSC; Complex network; GCNN

1. Introduction

Over the years, various Autonomous Underwater Vehicles (AUV) have been widely used in scientific and technological fields due to their autonomy and flexibility (Simetti and Casalino, 2016; Bian et al., 2012). However, working in the vast, harsh and challenging marine environment, AUVs face unanticipated events and faults, which lead to loss of vehicles and scientific data (Strutt, 2006; Nicholls et al., 2006). Therefore, it is important to improve the reliability of the AUV systems to adopt appropriate safety policies in the event of an emergency or failure (Dearden and Ernits, 2013), and one of the important problems to improve the reliability of AUV is to monitor and recognize its motion state (Zhang et al., 2015).

At present, there are two methods to monitor the motion states of AUV. One is to model the motion states of AUV and recognize its state according to the residual of predicted value and measured value; the other is to extract features implied in monitoring data by signal processing methods to classify the motion states of AUV. The main types of motion state of AUV modelling methods are based either on 6-Degree-Of-Freedom (DOF) methods or on Neural Networks (NN) modelling. In the DOF-based methods, the hydrodynamic coefficients need to be estimated. Empirical method (Nahon, 1996), Computational Fluid Dynamics (CFD) method (Kaya et al.,

* Corresponding author.
E-mail addresses: fccjg@ouc.edu.cn (Chen Feng).

2020; Liu et al., 2020) and experimental method (Randeni et al., 2018; Eng et al., 2016; Sajedi and Bozorg, 2019) are usually used for the estimation. Compared with DOF-based modelling, NN-based modelling is a parameter-free modelling method, which has the ability of approximating any nonlinear mapping with any degree of accuracy (Sun et al., 2016; Wu et al., 2019). The use of motion state model to monitor the motion states of AUV depends on the prediction accuracy of the model. In the methods based on motion state features, wavelet (Wang et al., 2016), fractal (Yu et al., 2020), and time-frequency domain decomposition (Lv et al., 2021; Liu et al., 2017) methods are used independently or in combination to extract the features that can best describe the motion states of AUV. Then, the classifiers are used to classify the motion states of AUV. The precise extraction of the latent features in the signal is the key to recognize the motion states of AUV.

The AUV state monitoring data comes from the time series output from the sensors. Therefore, the motion states of AUV can be classified from the perspective of Multivariate Time Series Classification (MTSC) to realize the motion state recognition. Based on this idea, we propose a new method for the motion state monitoring of AUV based on MTSC.

In the present MTSC methods, Multivariate Time Series (MTS) are usually transformed into new latent feature representation before being used to train a classifier. Then the features are extracted manually and fed into an off-the-shelf classifier such as Random Forest or Support Vector Machine (SVM) for the classification. For example, MTS are transformed into shapelets representation in the generalized Random Shapelet Forest (gRFS) method (Karlsson et al., 2016). In the WEASEL+MUSE method, bag of words is extracted from MTS using the Symbolic Fourier Approximation (SFA) (Schäfer and Leser, 2017). When Multiple Representation Sequence Learner (MrSEQL) is used to classify MTS, the data is transformed into the symbolic space via either Symbolic Aggregate Approximation (SAX) or SFA (Nguyen et al., 2019).

With the rise of deep learning methods, Deep Neural Network (DNN) is also used in MTSC. DNN is an end-to-end deep learning architecture, which can remove the bias due to manually designed features, thus enabling the network to learn the most distinguishing and useful features for the classification task. Based on the traditional deep Convolutional Neural Network (CNN), Multi Channel Deep Convolutional Neural Network (MCDCNN) (Zheng et al., 2014), Multi-scale Convolutional Neural Network (MCNN) (Cui et al., 2016), and Time-CNN (Zhao et al., 2017) were proposed and validated on MTS datasets. Moreover, Multi Layer Perceptron (MLP), deep Residual Network (ResNet) (Wang et al., 2017), time LeNet (t-LeNet) (Guennec et al., 2016), and Time Warping Invariant Echo State Network (TWIESN) (Tanisaro and Heidemann, 2016) were also proposed to classify MTS.

Comparing the above two kinds of methods, in the methods based on feature representation transformation, the process of transforming MTS into another representation plays the role of feature enhancement; the classification features are selected manually, therefore, they are not necessarily appropriate for a specific time series data classification problem. In the DNN-based methods, MTS is fed into the network directly to extract the features and to classify the motion states of AUV, therefore, subjective influences are reduced. If the constructed DNN is applied to the classified data, the latent features in MTS can be extracted comprehensively. From this point of view, (Ji et al., 2021) has proposed a DNN named Sequence Convolutional Neural Network (SeqCNN), which can extract global features and local features from state data of a small quadrotor AUV 'Haizhe' and classify the different fault states. However, in our experiments, we found that compared with the methods based on feature representation transformation, the classification accuracy of the DNN-based method is not significantly improved for the classification of the motion states of AUV. Therefore, in this work, combining the advantages of the two methods, we propose a new MTSC method based on our previous work (Zheng et al., 2019). We transform MTS into a complex network, which is a new graph representation employed to describe characteristics of the motion states of AUV effectively. And then we use a kind of Graph Deep Neural Network (GDNN) to extract the features of complex networks to classify the motion states of AUV automatically. In our method, the latent features of the motion states of AUV cannot only be represented by complex networks, but can also be effectively extracted by GDNN automatically. The following is the specific thinking process of our proposed method.

Under the coupling action of wave and ocean current, the motion states of AUV has nonlinear characteristics, and complex network is an effective tool to describe the nonlinearity (Gao et al., 2018; Gao et al., 2020; Myers et al., 2019). Complex network can also be regarded as a kind of new representation of time series. In the transformation from time series to complex networks, the time series on a small time scale is defined as network nodes in the coarse granularity procedure, which is similar to the coarse-grained process of time series in shapelet, SAX and SFA methods. In addition, the edge is used to represent the transition relationship between states on a small time scale of AUV. Compared with the original time series, complex networks can comprehensively better represent the features of time series on different time scales. In the early work, our group has proposed a method of AUV motion sate representation by constructing complex networks. We used the complex networks constructed from heading angles to describe the motion states of AUV on the horizontal plane. The experimental results indicated that the topological statistics of complex networks constructed from heading angles at different depths could accurately describe the motion states of AUV at different depths (Zheng et al., 2019). In that work, we only constructed the complex networks of one-dimensional heading angle time series to represent the motion states of AUV on the horizontal plane. In this study, we further focus on the motion states description of the vertical and roll planes of the AUV. Therefore, we use pitch and roll angles, which are two closely coupled time series (Prestero, 2001), to construct the complex networks for describing the motion states of the vertical and roll planes.

In the subsequent classification of the motion states of AUV, we use two methods. Same as in the gRFS, MrSEQL, SFA and SAX methods, we first train SVM by manually selecting topological statistics of complex networks for the classification. We then feed adjacency matrix and features of nodes into the Graph Convolutional Neural Network (GCNN) to classify the complex networks. GCNN is a kind of GDNN, which can achieve a better classification effect without manually extracting the features of the complex network topology (Errica et al., 2020; Song et al., 2018).

To verify the feasibility of our method, AUV navigational data collected in sea trials are used for experimental verification. The experimental results show that the topological statistics of two-variable complex network constructed by pitch and roll angles can effectively describe the normal and abnormal motion states of AUV near the surface and underwater, and the classification accuracy of GCNN is better than that of SVM. Meanwhile, we also use MLP and MCDCNN to classify the motion states of AUV. The results show that the classification accuracy of our method is also better than that of MLP and MCDCNN methods.

The main contributions of this paper are as follows:

(1) From the perspective of MTSC, by combining the advantages of the two kinds of MTSC methods, we propose a new AUV motion state classification method based on AUV state monitoring data. In

this method, the MTS output from the sensors are transformed into complex networks as graphs to represent the characteristics of motion states, and the GCNN is used for graph classification to realize AUV motion state classification.

(2) Based on complex networks, we propose a new feature representation transformation method for MTS, which describes the motion states of AUV on multi time scales. In the complex network, nodes are defined by coarse graining the time series on a small time scale, and the edges are defined based on the sequential connection relationship, which describe the correlation of the time series on a larger time scale.

The remaining of the paper is organized as follows. Section 2 gives an overview of the experimental platform and datasets. Complex networks construction method, SVM model, and GCNN model are introduced in section 3. Section 4 focuses on the experiments and validates the performance of the proposed methodology using AUV monitoring data for different motion states. Finally, the main conclusions of the study are presented in section 5.

2. Experimental platform and Datasets

2.1 Specifications of Sailfish-324 AUV

The experimental data was collected by the Sailfish-324 AUV developed by the Underwater Vehicle Laboratory of Ocean University of China (Fig. 1). The specifications of the AUV are shown in Table 1.

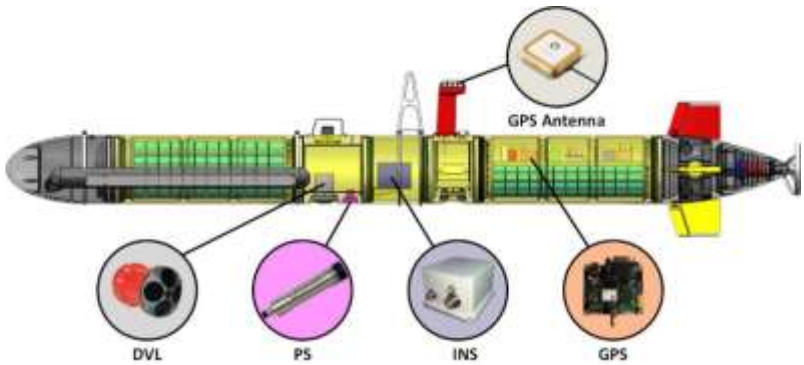


Fig. 1 Sailfish-324 AUV

Table1: Specifications of the Sailfish-324 AUV

Length	3.8 m
Diameter	32.4 cm
Displacement	260 Kg
Maximum speed	5 knots
Endurance	8h

Sailfish-324 is a single thruster and cruciform of rudders AUV. The main thruster is the EC60 brushless DC motor of the Maxon company, and the motor controller is APM090-30 of the Copley company. Sailfish-324 is equipped with multiple sensors including Global Position System (GPS), Inertial Navigation System (INS), Doppler Velocity Log (DVL), and pressure sensor, as shown in

Fig. 1. The data from all sensors and equipment are sent to onboard CPU for Ego-motion estimation. The core processor in our Sailfish is a single board computer with two cores and 777 Mhz main frequency. MOOS (Mission Oriented Operating Suite)-Ivp is the software to achieve autonomy and data acquisition. The sensors and equipment for navigation are connected to the processor by serial port. Sailfish-324 AUV has been tested several times in sea trials (Fig. 2).

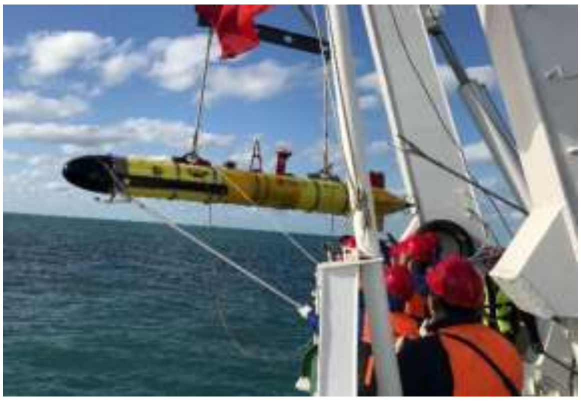


Fig. 2 Sea trials scenario of Sailfish-324 AUV

2.2 Datasets

The body-fixed coordinate system and world-fixed reference frame are used in the research on the motion law of AUV. The body-fixed coordinate system with origin O is a moving reference frame that is fixed to AUV. Ox axis points to the bow compartment, Oy axis points to the starboard, and the Oz axis points to the bottom. And [ξ, η, ζ] is a position vector with respect to the world-fixed reference frame whose origin is E. In this frame, the Euler angles are the heading angle ψ, pitch angle θ and roll angle φ (Fig. 3).

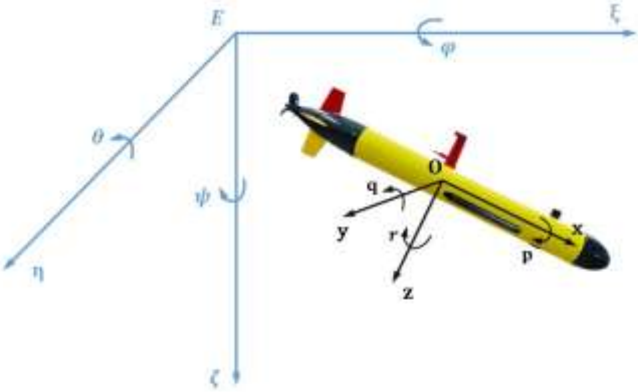


Fig. 3 Coordinate system of AUV

In this work, the pitch and roll angles in the world-fixed frame are used to describe the motion states of the vertical and roll planes. We use the motion monitoring data under three classes of working conditions acquired by AUV during sea trials. The class 1 is the motion state of straight sailing near the water surface, the class 2 is the motion state of straight sailing at fixed underwater depth, and the class 3 is the motion state of control divergent caused by the influence of unknown ocean currents during straight sailing at fixed underwater depth. The motion state of class 1 and 2 are normal, and that of class 3 is abnormal. The physical background and data characteristics of these three classes of data is as follows.

Because the Sailfish-324 is designed to have positive buoyancy, when the AUV sails at a fixed depth, it needs to constantly adjust the rudder angle to change its torque on the Oy axis for overcoming its positive buoyancy. Therefore, under the normal motion states, AUV constantly changes the pitch angle during fixed depth sailing, so that the depth of AUV can achieve the dynamic balance around the desired depth, and then the pitch angle fluctuates around the desired 0º pitch angle. AUV cannot control the roll Angle.

When the AUV sails at the fixed depth near the water surface, it is not only affected by the torque, but also by the wave force. The wave force includes mainly the first order wave force and the second order wave force. Among them, the second-order wave force is the force pointing to the water surface with non-zero mean value and irregular vertical upward direction, which affects the heave and pitch of AUV and makes the pitch and roll angles to change rapidly. Therefore, there are many types of AUV motion state on small time scales.

With the increase of the diving depth of AUV, the second-order wave force is sharply attenuated, the external influence is reduced, and the types of AUV motion state on small time scales decreases correspondingly (Fossen and Thor, 1994).

When AUV is sailing at the fixed depth in deep water, sometimes it is affected by the current from all directions and diverge in control. At this time, the pitch angle could not be controlled to fluctuate near the desired pitch angle, and a large amplitude change would occur, making the pitch angle in a fixed mode of rising or falling on a small time scale. Therefore, the types of AUV motion state on small time scales would continue to decrease.

3. Method

3.1 Overview of the proposed method and compared methods

In this work, our method involves transforming the pitch and roll angles into complex networks and constructing Graph Convolutional Neural Network (GCNN) to classify motion states of AUV. At the same time, we also use other three classification methods to compare with our method to verify the effectiveness of our method. The process is shown in Fig. 4.

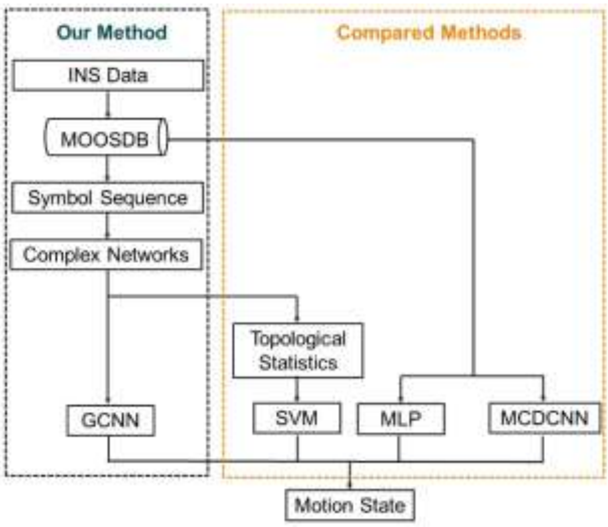


Fig. 4 Overview of our method and compared methods

When an AUV is in operation, INS provides the pitch and roll angles, which describe the motion states of the sailing AUV, and all the data are input into MOOSDB (Mission Oriented Operating Suite Database). In our method, subsequently, the data is converted into a symbol sequence via the symbolization method. By defining the combination of symbol sequences in small windows as nodes and determining edges, complex networks under different motion states are constructed. The complex network is fed into the GCNN to classify and recognize the motion states. Comparing with our method, the topological statistics of the complex networks of various motion states are calculated and input into the SVM to train the model for the recognition of the motion states. The pitch and roll angles are directly fed into the MLP and MCDCNN to train the models and classify the motion states.

In the MLP and MCDCNN methods, the hidden distinguishing features are learned from the original time series in an end-to-end manner for state classification. Although there is no need for data preprocessing in those methods, the detection performance of the classifier is easily reduced by ignoring different characteristics of data in different physical backgrounds. This is because the classification detection performance is not only affected by the detection method adopted, but more importantly by the quality of input data (Wang et al., 2017). In the GCNN and SVM methods, the characteristics of the data are considered before classification, and the original time series data are transformed into a new feature representation space in the form of a complex network, which can provide higher quality training data for the classification model. However, in the further feature extraction of complex networks, SVM only manually extracts the global topological statistics of complex networks as classification features, while GCNN obtains the local and global topological features of complex networks in an end-to-end manner by describing the features of data on the different time scales.

3.2 Complex networks construction

Figure 5 shows the process of the complex network construction.

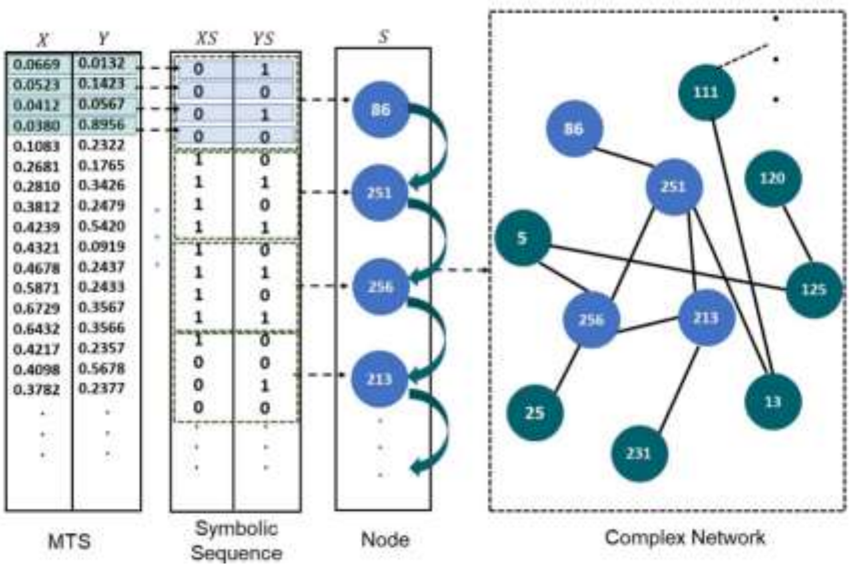


Fig. 5 Process of complex network construction

Defining nodes and determining edges rules are two key problems in complex network construction. Time series $X = (x_1, x_2, x_3, \dots, x_L)$ is symbolized to $XS = (xs_1, xs_2, xs_3, \dots, xs_{L-1})$ by

$$xs_{i-1} = \begin{cases} 0, & x_i - x_{i-1} < 0 \\ 1, & x_i - x_{i-1} \geq 0 \end{cases} . \quad (1)$$

Similarly, the time series $Y = (y_1, y_2, y_3, \dots, y_L\ )$ is symbolized to $YS = (ys_1, ys_2, ys_3, \dots, ys_{L-1}\ )$. For each pair of combination (xs_i, ys_i), there are 4 states: $(0{,}0)$, $(0{,}1)$, $(1{,}0)$ and $(1{,}1)$. Therefore, for w pairs of combination $\begin{pmatrix} xs_i, ys_i \\ \vdots \\ xs_{i+w-1}, ys_{i+w-1} \end{pmatrix}$, there are $N = 4^w$ states:

$$S = \{s_1, s_2, \dots, s_N\}. \quad (2)$$

Each state s_i is denoted as a node i, and the nodes are connected in chronological order to construct the complex network. The complex network describes the transformation of states under w time-length windows. The properties of a complex network can be expressed via its adjacency matrix $A \in \mathbb{R}^{N \times N}$, whose element A_{ij} takes the value 1 if an edge connects the node i to the node j and 0 otherwise.

Three topological statistics, including average degree connectivity, average harmonic centrality and degree connectivity of each complex network, have been computed to describe the global topological features of complex networks (Costa et al., 2007). Average degree connectivity of a node i is defined as:

$$k_{nn,i} = \frac{1}{D_i} \sum_{j \in Ne(i)} D_j\ , \quad (3)$$

where $D_i = \sum_{j=1}^{N} A_{ij}$ is the degree of the node i and $Ne(i)$ are the neighbors of node i .

Harmonic centrality of a node i is the sum of the reciprocal of the shortest path distances from all other nodes to i, and is defined as:

$$H_i = \sum_{i \neq j} \frac{1}{d(i,j)}, \quad (4)$$

where $d(i,j)$ is the shortest path distances from node i to node j .

The above two statistical measures describe the local topological characteristics of the nodes in the complex network. Therefore, it is necessary to average the values of each node to represent the topological characteristics of the whole network.

The degree assortativity measures the similarity of connections in the graph with respect to the node degree, and is defined as:

$$r = \frac{\frac{1}{M}\sum_{j>i} D_i D_j A_{ij} - [\frac{1}{M}\sum_{j>i} \frac{1}{2}(D_i + D_j) A_{ij}]}{\frac{1}{M}\sum_{j>i} \frac{1}{2}(D_i^2 + D_j^2) A_{ij} - [\frac{1}{M}\sum_{j>i} \frac{1}{2}(D_i + D_j) A_{ij}]^2}\ , \quad (5)$$

where M is the total number of edges of the complex network.

3.3 SVM

Support Vector Machine (SVM) model is a classification method. Its main steps are to construct an appropriate kernel function, find the optimal classification hyperplane in the sample feature space, make the intervals between all kinds of samples in the maximum training set to realize the effective classification of features. We use Radial Basis Function (RBF) as kernel function to train SVM model and to classify the topological characteristics of complex network under different motion states of AUV. The SVM approach is not the focus of this paper, though details can be found in

(Vapnik, 1995).

3.4 GCNN

The GCNN architecture proposed by Zhang et al. is shown in Fig. 6. The architecture consists of three graph convolutional and the pooling layers, node features are aggregated in the readout layer. Then graph feature representations are input to MLP layer to classify the graphs (Zhang et al., 2018).

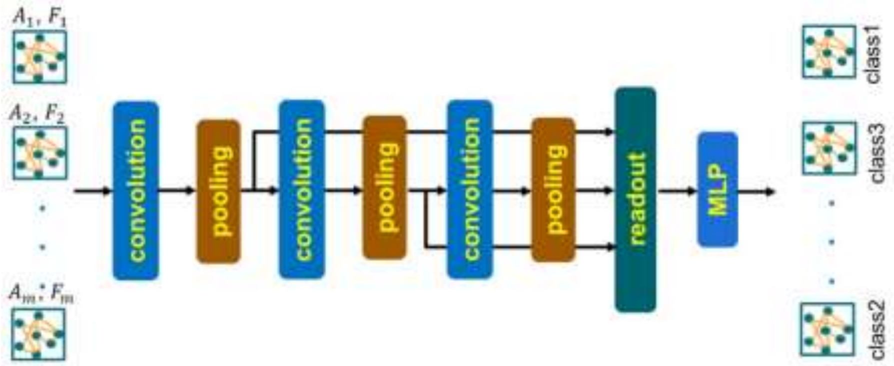


Fig. 6 GCNN architecture

Convolution layer: Graph convolution proposed by Kipf & Welling (Kipf and Welling, 2017) is used with the following layer-wise propagation rule:

$$h^{(l+1)} = \sigma_1(\widetilde{D}^{-\frac{1}{2}}\tilde{A}\widetilde{D}^{-\frac{1}{2}}h^{(l)}\Theta), \quad (6)$$

where $h^{(l)}$ is the node representation of the l^{th} layer. $\tilde{A} = A + I$ is the adjacency matrix with self-connections, and $\widetilde{D}$ is the degree matrix of $\tilde{A}$. $\Theta \in \mathbb{R}^{m\times m'}$ is the convolution weight with input feature dimension m and output feature dimension m'. $h^{(0)} = F$, F is the node feature matrix in which each row vector is a vector of the features of each node. If the features of each node are not considered, F is the unit matrix. The Rectified Linear Unit function $ReLU(\cdot) = max\,(0,\cdot)$ is used as an activation function $\sigma_1(.)$ (Hinton et al., 2012).

Pooling layer: In the down-sampling of graph data, a subset of nodes is adaptively selected to form a new but smaller graph. Self-Attention Graph Pooling (SAGPool) (Lee et al., 2019; Gao and Ji, 2019) is used as an attention mechanism to select the important nodes:

$$Z = \sigma_2(\mathrm{GCN}(F, A)), \quad (7)$$

where $\sigma_2(.)$ is the $tanh$ activation function. The attention score Z is calculated from Graph Convolutional Network (GCN) layer and the top k nodes are selected to form the smaller graph.

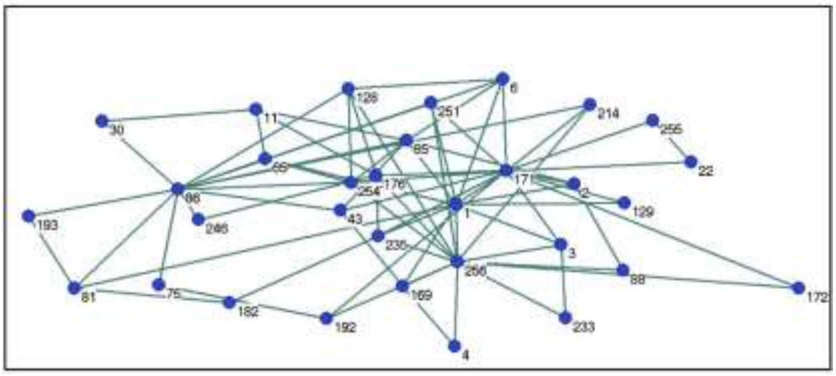
Readout layer: A readout layer, which aggregates node features to make a fixed size representation summarizing output feature, is as follows: (Xu et al., 2018)

$$g = \frac{1}{N}\sum_{i=1}^{N} F_i. \quad (8)$$

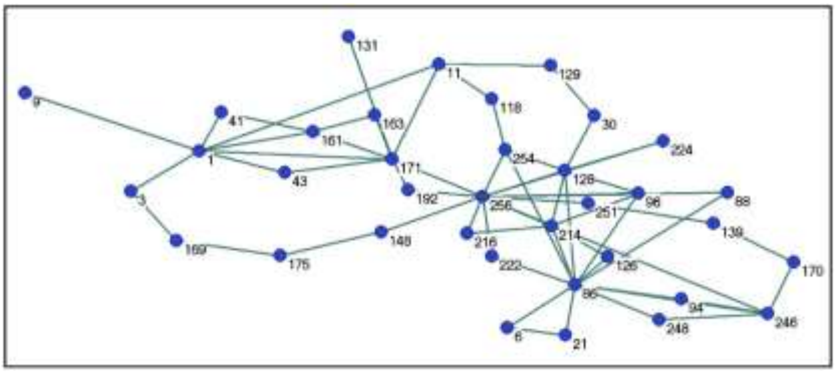
4. Experiments

4.1 Complex networks construction

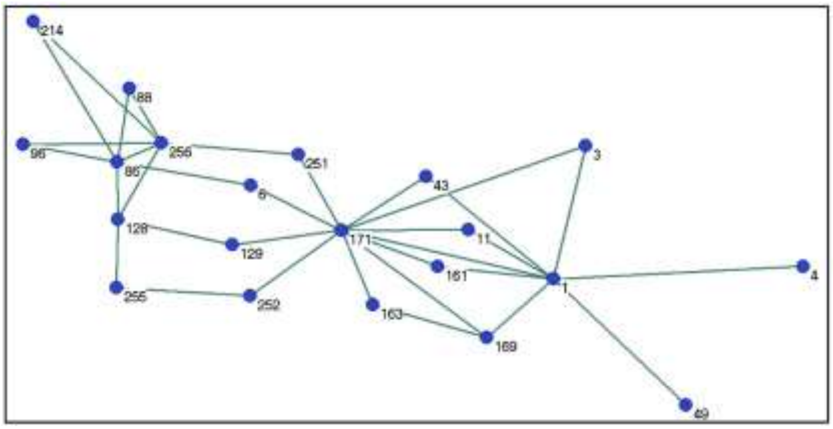
The time series data of pitch and roll angles are divided into subsets of time series data with length of 500 to construct the complex networks. We choose $w = 4$, therefore there are 126 states on small time scales. Each state is defined as a node, and the number of nodes in each complex network can be up to 126. We randomly select three complex networks under three kinds of motion states, as shown in Fig. 7.



(a)



(b)



(c)

Fig. 7 Complex networks of (a) class 1, (b) class 2, and (c) class 3

Figure 7 shows the complex network size (number of the nodes), node composition and topological connection for different motion states.

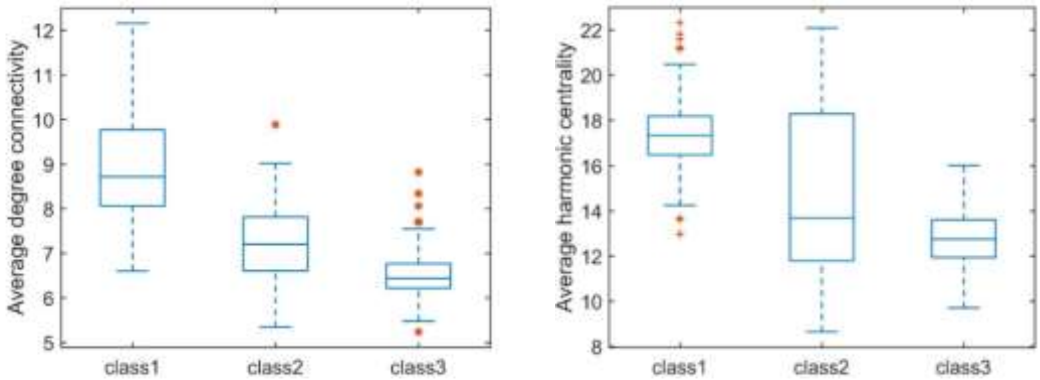
Because each node represents the AUV motion state on a small time scale, the size of the complex network is the number of the types of AUV motion state on a small time scale. The average sizes of complex networks of class 1, 2 and 3 are 35, 32 and 27, respectively. Compared with the normal motion states, the size of complex network under the abnormal motion state is smaller, which is consistent with the motion state analysis of AUV under different working conditions, as in Section

2.2. In terms of node composition, states s_1, s_{86}, s_{171} and s_{256} appear in all networks. But, other states appear at different frequencies in complex networks under different motion states, which reflects the difference in the distribution of AUV motion states on small time scales under the different motion states. In addition, the topological statistics of the three classes of network nodes are also different. For example, in the network of class 1, the degree of each node is relatively large and the network connectivity is strong, while in the network of class 3, the degree of each node is small and the network connectivity is weak, which reflects the difference in the transformation of AUV motion states on small time scales in different situations.

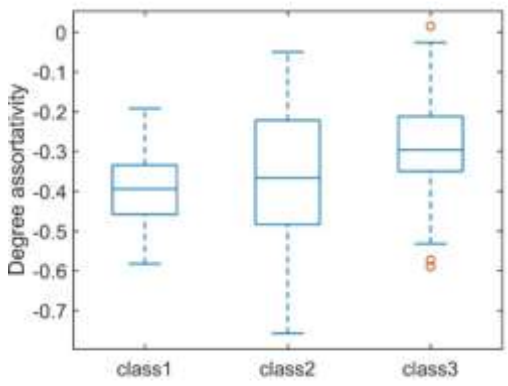
The above differences indicate that the local and global topological features of the complex network transformed from the pitch and roll angle time series can describe the difference of AUV motion states under the different conditions. Based on these differences of these networks, we used SVM and GCNN methods to carry out experiments to classify different AUV motion states.

4.2 Classifying motion states of AUV based on SVM

The SVM model classifies objects according to their features. For the complex networks, topological statistics can describe their topological features. Therefore, we first compute three topological statistics, including average degree connectivity (F1), average harmonic centrality (F2), and degree assortativity (F3) of each complex network as the classification features. Combined with our construction method of complex network, these topological statistics also have specific physical significance. The degree connectivity describes the transition relationship between each state on a small time scale and its adjacent states. The harmonic centrality describes the importance of each state on a small time scale in the process of state transition. The degree assortativity describes the similarity between each state on a small time scale. In order to observe the distribution difference of various topological statistics, the statistical results are displayed in Fig. 8 in the form of block plots.



(a) (b)



(c)

Fig. 8 Box plots of (a) average degree connectivity (F1), (b) average harmonic centrality (F2), and (c) degree assortativity (F3) of complex networks under different motion states

We then use Gaussian radial basis function as kernel function to construct SVM. These three features are combined to train the SVM model; the obtained classification accuracy is shown in Table 2.

Table 2: Classification accuracy based on different topological feature using SVM

	F1	F2	F3	F1&F2	F2&F3	F1&F3	F1&F2&F3
Accuracy	0.608	0.808	0.540	0.834	0.810	0.817	**0.843**

Figure 8 shows that in the complex network under different motion states, the feature values of the average degree connectivity (F1) and the average harmonic centrality (F2) are different, which can better be used to classify the motion states of AUV. Table 2 shows that when each feature is used separately for classification, the classification effect of average harmonic centrality (F2) is the best. The classification accuracy of pairwise feature combination is higher than that of average harmonic centrality (F2) alone. When all three features are used for classification, the classification accuracy is the highest.

We also add other topological statistics, such as average clustering coefficient and average closeness centrality into the classification features, but these statistics do not improve the classification accuracy.

4.3 Classifying motion states of AUV based on GCNN

We further use GCNN model to classify the motion states. As shown in expression (7), the input of GCNN is the adjacency matrix A and the node feature matrix F of each complex network. The adjacency matrix describes the characteristics of graph topological structures, while the node features describe the characteristics of node itself. For each complex network, the adjacency matrix is unchanged. Different node features can be selected to form vectors and experiments can be carried out to find the node feature vector that can obtain the best classification effect. In the experiment, the node feature vectors are set as follows:

Case 1: Do not consider the node feature;

Case 2: select the degree connectivity and the harmonic centrality of each node as feature vector;

Case 3: select the state on a time small scale represented by the node as feature vector;

Case 4: select the degree connectivity, the harmonic centrality, and the state of each node on a small time scale as feature vector.

The classification accuracy of the above four cases is shown in Table 3.

Table 3: Classification accuracy based on different node feature using GCNN

	Case 1	Case 2	Case 3	Case 4
Accuracy	0.410	0.852	**0.969**	0.907

In case 1, GCNN only classifies complex networks based on graph topology, so the classification accuracy is very low. In case 2, when the degree connectivity and harmonic centrality are added, the accuracy is improved to 0.852, but these characteristics only describe the connection relationship between nodes rather than the nature of the nodes themselves. Therefore, the accuracy is similar to the highest accuracy of SVM method, i.e., 0.843. In Case 3, classification accuracy is the highest because the state is the characteristics of the node itself, which is determined by the complex network construction method in Section 3.1. In case 4, the combination of connectivity, harmonic centrality and state are used as the feature vector. Although the accuracy is higher than that in case 2, it is still lower than that in case 3. This shows that in order to improve the classification accuracy, it is not the number of node features contained in the feature vector, but to select the features that can essentially describe the characteristics of nodes.

By comparing Table 2 and Table 3, the GCNN model performs much better than the SVM model when classifying the complex networks under different motion states. This is for two reasons:

(1) **From the perspective of classification model method**: Compared with SVM, which needs to manually extract feature data for classification, GCNN realizes end-to-end learning and classification, which makes the exploration of graphic features more in-depth.

(2) **From the point of view of the complex network construction method:** When constructing complex network to describe the different motion states, the motion states on small time scales are symbolized and defined as nodes, and the edges are defined according to the transitions of the states on small time scales. Therefore, the complex network essentially describes the state transition on a small time scale within a period of time. When the AUV is in different motion states, the state transitions on small time scales are not the same, thus the constructed complex networks under different motion states have different topological structures and node combinations. So not only the topological connection between nodes in the graph (the transition of states on small time scales), but also the states represented by the nodes themselves should be considered. When classifying the complex networks, GCNN model explores the node features and the graph topological features, while SVM model only uses the topological features of the graph.

4.4 Comparison with other methods

We also use MLP and MCDCNN to classify the motion states to compare with our method. The results are shown in Table 4.

Table 4: Classification accuracy using MLP, MCDCNN and our method

	MLP	MCDCNN	our method
Accuracy	0.925	0.835	**0.969**

The results show that the classification accuracy of our method is better than that of other deep learning methods. The reason is that in MLP and MCDCNN methods, the original motion state monitoring data of AUV is directly used for state classification, but in our method, the original data is first transformed into a new graph representation, and then the graph is classified. The process of graph representation is the process of feature enhancement of the original data.

5. Conclusion

In order to monitor the motion states of AUV and to improve its reliability based on the roll and pitch angles, we adopted a complex network method to describe the motion states of AUV and then trained the SVM and GCNN models to classify the complex network under different motion states. We also fed the roll and pitch angles directly to two kinds of DNN, including the MLP and MCDCNN, to classify the motion states. Three classes of motion state monitoring data were used for the experiments to show the effectiveness of our proposed method. We drew the following conclusions from the experiments.

(1) In the process of constructing a complex network from motion state monitoring data of AUV, the definition of nodes representing motion states on a small time scale is similar to the coarsening process of MTS on a small time scale in the traditional feature representation transformation based methods. However, the edge connection in complex networks can further describe the transformation relationship of motion states on a larger time scale. Compared with the traditional feature representation transformation based methods, complex network can describe the motion state characteristics of AUV on multiple time scales. Therefore, the new feature representation transformation method, in which complex network is used as graph representation, is more suitable for representing the motion states of AUV.

(2) In the process of classifying the complex network describing the motion states of AUV, compared with the SVM classification model, which needs to manually extract the topological features of complex network, DCNN can automatically scrutinize the topological features more completely. Therefore, the classification accuracy of DCNN is higher than that of SVM, which means that DCNN is more suitable for the classification of the complex network constructed from motion monitoring data of AUV.

(3) The transformation of motion monitoring data of AUV into complex network is actually the feature preprocessing of the original data. Therefore, compared with the MLP and MCDCNN methods, the accuracy of our method is also better.

The complex network constructed in this work is an undirected and unweighted network, and the direction and times of state transition are not considered. In a further work, we will use additional classes of motion state data, build the weighted and directed complex network, and modify the structure of GCNN to further improve the accuracy of motion state classification and recognition of AUV.

Acknowledgement

This work is supported by Natural Science Foundation of Shandong Province (ZR2020MF079) and China Scholarship Council (CSC).

References

Simetti, E. and Casalino, G., 2016. Manipulation and transportation with cooperative underwater vehicle manipulator systems. IEEE J. Ocean. Eng., 42, 782-799. https://doi.org/10.1109/JOE.2016.2618182.

Bian, X., Yan, Z., Chen, T., Yu, D. and Zhao, Y., 2012. Mission management and control of BSA-AUV for ocean survey. 55, 161-174. https://doi.org/10.1016/j.oceaneng.2012.06.022.

Strutt, J. E., 2006. Report of the inquiry into the loss of Autosub2 under the Fimbulisen. NOCS Research and Consultancy Report. [Online]. Available: http://eprints.soton.ac.uk/41098.

Nicholls, K. W., Abrahamsen, E. P., Buck, J. J. H., Dodd, P. A., Goldblatt, C., Griffiths, G., et al., 2006. Measurements beneath an Antarctic ice shelf using an autonomous underwater vehicle. Geophys. Res. Lett., 33, L08612. https://doi.org/10.1029/2006GL025998.

Dearden, R. and Ernits, J., 2013. Automated fault diagnosis for an autonomous underwater vehicle. IEEE J. Ocean. Eng., 38, 484-499. https://doi.org/10.1109/JOE.2012.2227540.

Zhang, M., Liu, X., Yin, B. and Liu, W., 2015. Adaptive terminal sliding mode based thruster fault tolerant control for underwater vehicle in time-varying ocean currents. J. Frankl. Inst., 352, 4935-4961. https://doi.org/10.1016/j.jfranklin.2015.08.009.

Nahon, M., 1996. A simplified dynamics model for autonomous underwater vehicles. Proceedings of Symposium on Autonomous Underwater Vehicle Technology. In: Proceedings of Symposium on Autonomous Underwater Vehicle Technology, of Conference Monterey, CA, USA. 373-379. https://doi.org/10.1109/AUV.1996.532437.

Kaya, K. D., Goren, A., Yilmaz, S. and Bayramolu, K., 2020. Determination of operating parameters of an AUV following a preplanned trajectory using hydrodynamic analysis data. Ocean. Eng., 217. https://doi.org/10.1016/j.oceaneng.2020.107708.

Liu, Y., Yang, Y., Zhang, H. and Zhang, L., 2020. Computational fluid dynamics prediction of the dynamic behavior of autonomous underwater vehicles. IEEE J. Ocean. Eng., 45, 724-739. https://doi.org/10.1109/JOE.2019.2916119.

Randeni, P. S., Forrest, A. L., Cossu, R., Leong, Z. Q., Ranmuthugala, D. and Schmidt, V., 2018. Parameter identification of a nonlinear model: replicating the motion response of an autonomous underwater vehicle for dynamic environments. Nonlinear Dyn., 91, 1229–1247. https://doi.org/10.1007/s11071-017-3941-z.

Eng, Y. H., Teo, K. M., Chitre, M. and Ng, K. M., 2016. Online system Identification of an autonomous underwater vehicle via in-field experiments. IEEE J. Ocean. Eng., 41, 5-17. https://doi.org/10.1109/JOE.2015.2403576.

Sajedi, Y. and Bozorg, M., 2019. Robust estimation of hydrodynamic coefficients of an AUV using Kalman and H∞ filters. Ocean Eng., 182, 386-394. https://doi.org/10.1016/j.oceaneng.2019.04.039.

Sun, Y., Li, Y., Zhang, G., Zhang, Y. and Wu, H., 2016. Actuator fault diagnosis of autonomous underwater vehicle based on improved elman neural network. J. Cent. South Univ., 23, 808-816. https://doi.org/10.1007/s11771-016-3127-8.

Wu, H., Song, S., You, K. and Wu, C., 2019. Depth control of model-free AUVs via reinforcement learning. IEEE Trans. Syst. Man Cybern. -Syst., 49, 2499-2510. https://doi.org/10.1109/TSMC.2017.2785794.

Wang, Y., Liu, W., Liu, X. and Zhang, M., 2016. Weak thruster fault detection for AUV based on stochastic resonance and wavelet reconstruction. J. Cent. South Univ., 23, 2883-2895. https://doi.org/10.1007/s11771-016-3352-1.

Yu, D., Deng, K., Gong, W., M., Z. and Z., C., 2020. Thruster fault feature extraction method for underwater vehicle.

In: Proceedings of the 32nd Chinese Control and Decision Conference (CCDC), of Conference Hefei, China. 2751-2757. https://doi.org/10.1109/CCDC49329.2020.9164316.

Lv, T., Zc, A., Feng, Y. and Mz, A., 2021. Fault feature extraction method based on optimized sparse decomposition algorithm for AUV with weak thruster fault. Ocean. Eng., 233. https://doi.org/10.1016/j.oceaneng.2021.109013.

Liu, W., Wang, Y., Yin, B., Liu, X. and Zhang, M., 2017. Thruster fault identification based on fractal feature and multiresolution wavelet decomposition for autonomous underwater vehicle. Proc. Inst. Mech. Eng. Part C-J. Eng. Mech. Eng. Sci., 231, 2528-2539. https://doi.org/10.1177/0954406216632280.

Karlsson, I., Papapetrou, P. and Boström, H., 2016. Generalized random shapelet forests. Data Min. Knowl. Discov., 30, 1053–1085. https://doi.org/10.1007/s10618-016-0473-y.

Schäfer, P. and Leser, U., 2017. Multivariate time series classification with WEASEL+MUSE. Available: https://arxiv.org/abs/1711.11343.

Nguyen, T. L., Gsponer, S., Ilie, I., O'Reilly, M. and Ifrim, G., 2019. Interpretable time series classification using linear models and multi-resolution multi-domain symbolic representations. Data Min. Knowl. Discov., 33, 1183–1222. https://doi.org/10.1007/s10618-019-00633-3.

Zheng, Y., Liu, Q., Chen, E., Ge, Y. and Zhao, J. L., 2014. Time series classification using multi-channels deep convolutional neural networks. In: International conference on web-age information management, of Conference Springer, Cham. 298–310. https://doi.org/10.1007/978-3-319-08010-9_33.

Cui, Z., Chen, W. and Chen, Y., 2016. Multi-scale convolutional neural networks for time series classification. Available: https://arxiv.org/abs/1603.06995.

Zhao, B., Lu, H., Chen, S., Liu, J. and Wu, D., 2017. Convolutional neural networks for time series classification. J. Syst. Eng. Electron. , 28, 162–169. https://doi.org/10.21629/JSEE.2017.01.18.

Wang, Z., Yan, W. and Oates, T., 2017. Time series classification from scratch with deep neural networks: a strong baseline. In: International joint conference on neural networks (IJCNN, of Conference Anchorage, AK, USA. 1578–1585. https://doi.org/10.1109/IJCNN.2017.7966039.

Guennec, A. L., Malinowski, S. and Tavenard, R., 2016. Data augmentation for time series classification using convolutional neural networks. In: ECML/PKDD workshop on advanced analytics and learning on temporal data, of Conference Riva Del Garda, Italy. https://halshs.archives-ouvertes.fr/halshs-01357973.

Tanisaro, P. and Heidemann, G., 2016. Time series classification using time warping invariant echo state networks. In: 15th IEEE International Conference on Machine Learning and Applications (ICMLA), of Conference Anaheim, CA, USA. 831–836. https://doi.org/10.1109/ICMLA.2016.0149.

Ji, D., Yao, X., Li, S., Tang, Y. and Yian, Y., 2021. Model-free fault diagnosis for autonomous underwater vehicles using sequence convolutional neural network. Ocean. Eng., 232, 108874. https://doi.org/10.1016/j.oceaneng.2021.108874.

Zheng, X., Feng, C., Li, T. and He, B., 2019. Analysis of Autonomous Underwater Vehicle (AUV) navigational states based on complex networks. Ocean. Eng., 187, 106141. https://doi.org/10.1016/j.oceaneng.2019.106141.

Gao, Z., Dang, W., Mu, C., Yang, Y., Li, S. and Grebogi, C., 2018. A novel multiplex network-based sensor information fusion model and its application to industrial multiphase flow system. IEEE Trans. Ind. Inform., 14, 3982-3988. https://doi.org/10.1109/TII.2017.2785384.

Gao, Z., Liu, M., Dang, W. and Cai, Q., 2020. A novel complex network-based deep learning method for characterizing gas-liquid two-phase flow. Pet. Sci., 18, 259-268. https://doi.org/10.1007/s12182-020-00493-3.

Myers, A., Munch, E. and Khasawneh, F. A., 2019. Persistent homology of complex networks for dynamic state detection. Phys. Rev. E., 100, 022314. https://doi.org/10.1103/PhysRevE.100.022314.

Prestero, T., 2001. Verification of a six-degree of freedom simulation model for the REMUS autonomous underwater vehicle. Available: http://hdl.handle.net/1721.1/65068.

Errica, F., Podda, M., Bacciu, D. and Micheli, A., 2020. A fair comparison of graph neural networks for graph classification. In: Proceedings of the International Conference on Learning Representations (ICLR), of Conference Addis Ababa, Ethiopia. https://arxiv.org/abs/1912.09893.

Song, T., Zheng, W., Song, P. and Cui, Z., 2018. EEG Emotion Recognition Using Dynamical Graph Convolutional Neural Networks. IEEE Trans. Affect. Comput., 11, 532-541. https://doi.org/10.1109/TAFFC.2018.2817622.

Fossen and Thor, I., 1994. Guidance and Control of Ocean Vehicles, Chichester, John Wiley & Sons.

Wang, H., Gu, J. and Wang, S., 2017. An effective intrusion detection framework based on SVM with feature augmentation. Knowledge-Based Syst., 136, 130-139. https://doi.org/10.1016/j.knosys.2017.09.014.

Costa, L., Rodrigues, F. A., Travieso, G. and Boas, P. V., 2007. Characterization of complex networks: A survey of measurements. Adv. Phys., 56, 167-242. https://doi.org/10.1080/00018730601170527.

Vapnik, V., 1995. The nature of statistical learning theory, New York, Springer. https://doi.org/10.1007/978-1-4757-2440-0.

Zhang, M., Cui, Z., Neumann, M. and Chen, Y., 2018. An end-to-end deep learning architecture for graph classification. In: 32nd AAAI Conference on Artificial Intelligence, of Conference New Orleans, Louisiana, USA. 4438-4445. https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/viewPaper/17146.

Kipf, T. N. and Welling, M., 2017. Semi-supervised classification with graph convolutional networks. In: 5th International Conference on Learning Representations (ICLR), of Conference Toulon, France. https://doi.org/https://arxiv.org/abs/1609.02907.

Hinton, G., Deng, L., Yu, D., Dahl, G. E. and Kingsbury, B., 2012. Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. IEEE Signal Process. Mag., 29, 82-97. https://doi.org/10.1109/MSP.2012.2205597.

Lee, J., Lee, I. and Kang, J., 2019. Self-Attention graph pooling. In: Proceedings of the 36th International Conference on Machine Learning (PMLR), of Conference Long Beach, USA. 97. https://arxiv.org/abs/1904.08082v4.

Gao, H. and Ji, S., 2019. Graph U-nets. In: Proceedings of the 36th International Conference on Machine Learning (PMLR), of Conference Long Beach, USA. 97, 2083–2092. https://arxiv.org/abs/1905.05178.

Xu, K., Li, C., Tian, Y., Sonobe, T., Kawarabayashi, K. I. and Jegelka, S., 2018. Representation learning on graphs with jumping knowledge networks. In: Proceedings of the 35th International Conference on Machine Learning (PMLR), of Conference Stockholm, Sweden. 5453-5462. https://arxiv.org/abs/1806.03536.