Highlights

Nitrogen Prediction in the Great Barrier Reef using Finite Element Analysis with Deep Neural Networks

Mohammad Jahanbakht,Wei Xiang,Barbara Robson,Mostafa Rahimi Azghadi

- Finite element analysis is incorporated in deep neural networks to form a new FE-DNN model.
- Nitrogen distribution in the wide Great Barrier Reef is forecasted using the proposed FE-DNN.
- The required stiffness matrices are numerically calculated.
- The resulting next-frame predicting model exhibits high resolution and high accuracy.
- FE-DNN is applicable to other environmental models that are governed by partial differential equations.

Nitrogen Prediction in the Great Barrier Reef using Finite Element Analysis with Deep Neural Networks

Mohammad Jahanbakht[a], Wei Xiang[b,*], Barbara Robson[c] and Mostafa Rahimi Azghadi[a,*]

[a]College of Science and Engineering, James Cook University, 1 James Cook Dr, Douglas 4811, QLD, Australia
[b]School of Engineering and Mathematical Sciences, La Trobe University, Plenty Rd, Bundoora 3086, VIC, Australia
[c]Australian Institute of Marine Science and AIMS@JCU, PMB No.3, Townsville MC, Townsville 4810, QLD, Australia

ARTICLE INFO

Keywords:
Machine Learning
Deep Neural Networks
Finite Element Analysis
Partial Differential Equation
Total Nitrogen Forecasting
Next-frame Prediction
Great Barrier Reef
Physics-Informed Neural Network
eReefs Modelling Suite

ABSTRACT

The corals of the Great Barrier Reef (GBR) in Australia are under pressure from contaminants including nitrogen entering the sea. To provide decision support in reaching target water quality outcomes, development of a nitrogen forecasting model may be useful. Here, we propose a new technique that considers the whole GBR as a frame and treats forecasting of nitrogen as a next-frame prediction task, to produce spatial maps of nitrogen over the whole GBR at forecast time-steps. To achieve this, we design an innovative Deep Neural Network (DNN) inspired by the Finite Element (FE) analysis concept. In our proposed method, the GBR area is meshed into small elements with pre-calculated stiffness matrices first. Next, both the stiffness matrices and the nitrogen values of each element are fed into the designed DNN for element-wise nitrogen prediction. The final result is then gained by attaching separate outputs of each element. Unlike other next-frame prediction models, our FE-DNN model generates accurate forecasts with unblurred prediction frames. We demonstrate that our model is the first to provide nitrogen forecasts for the entire GBR with low Mean Square Error (MSE), while generating a high-resolution prediction frame. The proposed model is applicable to other environmental modelling applications that are governed by Partial Differential Equations (PDE), e.g., sea temperature prediction and sediment distribution forecasting. Nonetheless, no knowledge of the underlying PDEs is required to use our DNN-based model. Our method can produce accurate forecasting predictions by leveraging existing hindcasting simulation models.

1. Introduction

The Great Barrier Reef (GBR) is the world's largest coral reef system, located off the east coast of Queensland, Australia. This world heritage site is facing severe threats that challenge its resilience, including extreme weather events and climate change, agricultural pollutants, coastal activities, surface runoff associated with the catchment areas, etc. Among these threats, land and agricultural activities are the main sources of pollutants from GBR catchments (Steven et al., 2019).

Nutrients, fine sediments, and pesticides are considered to be the primary land-based pollutants that significantly reduce ocean water quality (Waterhouse et al., 2020). According to the Australian and Queensland Government's long-term sustainability plan for the GBR (Reef 2050 Plan) (Reef-2050, 2021), excess nitrogen is particularly challenging in the GBR. High rainfall, flash floods, numerous short river basins, and the close proximity of the reef to the Wet Tropics of Queensland mean nutrients are flushed to the reef lagoon quickly.

Accordingly, the total nitrogen is amongst the most commonly measured and monitored water quality variables worldwide. In coastal and marine waters, nitrogen is usually considered the primary limiting nutrient. In other words, there is a strong consensus that it is the limited supply of nitrogen that limits marine ecosystem productivity in most cases, although phosphorus, silica, and iron may co-limit productivity in some situations (Howarth and Marino, 2006). When the total nitrogen increases, the growth and productivity of marine algae and other photosynthesising organisms increases, often to the detriment of marine ecosystems. This process is known as *eutrophication* and there is an extensive literature assessing its prevalence, causes, and management (Smith et al., 1999).

There is extensive evidence that the coastal waters of the GBR have been subject to some degree of eutrophication due to changes in its catchment land use since European settlement (Kroon et al., 2012; Bell et al., 2014; McCloskey et al., 2021) and that this has had a negative effect on GBR ecosystems (De'ath and Fabricius, 2010; MacNeil et al., 2019), though the offshore GBR and much of the midshelf remain oligotrophic (i.e., has low nitrogen and phosphorus concentrations) in absolute terms (McKinnon et al., 2017).

Management of nitrogen loads to the GBR in order to improve GBR water quality has been the focus of major investments by state and federal governments, not-for-profit organizations and farmers for many years (Kroon et al., 2016; Coggan et al., 2021; Waltham et al., 2021). Towards this end, a greater focus on experimentation, evaluation, and modelling to understand future nitrogen scenarios could

This work is funded by the Australian Government Research Training Program Scholarship.
*Corresponding authors.
Email addresses: mohammad.jahanbakht@my.jcu.edu.au (Mohammad Jahanbakht); w.xiang@latrobe.edu.au (Wei Xiang); b.robson@aims.gov.au (Barbara Robson); mostafa.rahimiazghadi@jcu.edu.au (Mostafa Rahimi Azghadi)
ORCID(s): 0000-0002-3609-9677 (Mohammad Jahanbakht); 0000-0002-0608-065X (Wei Xiang); 0000-0002-1811-3527 (Barbara Robson); 0000-0001-7975-3985 (Mostafa Rahimi Azghadi)

further support water quality programs (Najafzadeh et al., 2019). In particular, predictive models can be used to forecast and manage the high risk areas in the coral reef ecosystems (Waterhouse et al., 2020).

However, implementing an accurate nitrogen predictor for the vast areas of the GBR is a challenging task. Nitrogen values in the GBR form a big frame (matrix) that vary with both spatial coordinate (x, y) and the time. One technique to handle this giant time-varying frame is to transform it into a timeseries by averaging all nitrogen values on each day.

This technique has been employed by many predictive models for a variety of target parameters, e.g., physical, chemical, and biochemical characteristics of water (Najafzadeh and Niazmardi, 2021), water quality index (Najafzadeh et al., 2021), nitrogen uptake in crops (Sharifi, 2020), marine environment salinity, O_2, NO_3, phosphorus, silicon, chlorophyll, and alkalinity (Wen et al., 2021), etc. The employed timeseries forecasting models in these published works range from decision tree and multivariate regression in statistical models to support vector regression in shallow neural networks, and further to the Long Short-Term Memory (LSTM) in deep neural networks. For example, one of the most recent models that has used this averaging technique is the fuzzy partitioning LSTM model introduced by Wen et al. (2021). In this model, the data attributes are partitioned by fuzzy c-means before feeding to an LSTM network for supervised learning. This architecture makes the model ready for high-speed distributed learning, as well as inference.

As opposed to the above technique, there is a second approach to design a next-frame predictor. In this approach, nitrogen values of each day across the GBR form a frame. The goal is to forecast future frames from the historical frames. This approach is referred to as *next-frame prediction* in parlance (Zhou et al., 2020). While time series forecasting could be applied to predict a value for each pixel separately, next frame forecasting has the great advantage of incorporating both spatial and time-series information rather than considering the history of each pixel in isolation. This provides a much richer source of information for each prediction.

It is worth mentioning that next-frame prediction is a type of forecasting problem, which is different from simulation problems widely carried out by hydrodynamic models (Huang et al., 2021). Standalone hydrodynamic models cannot forecast unless future boundary conditions can be reliably predicted, except by coupling with a data-driven surrogate model. For this reason, hydrodynamic models for water quality forecasting are rarely reported in the literature, are mainly timeseries forecasting models, and typically have high errors (Khan et al., 2020).

To the best of our knowledge, all existing data-driven next-frame predictors in the literature treat each frame as a whole. In other words, they simply stack up historical 2D frames, making a 3D matrix, and then feed the resulting 3D matrix to their Deep Neural Network (DNN) models to output a 2D prediction frame. Some of the commonly used DNNs are recurrent neural networks (Wang et al., 2019), 3D Convolutional Neural Network (Conv3D) (Mathieu et al., 2016), Convolutional Long Short-Term Memory (ConvLSTM) (Hong et al., 2018; Guen and Thome, 2020), etc.

One of the most successful next-frame predicting models in the literature is PhyDNet proposed by Guen and Thome (2020). PhyDNet disentangles physical knowledge described by partial differential equations from data, before feeding it to the ConvLSTM model. The experiments with sea surface temperature data showed the ability of PhyDNet to outperform state-of-the-art methods. In ensuing sections, we will apply PhyDNet to our nitrogen distribution dataset for comparison. We will show that the main disadvantage of these next-frame predictors is their low coefficient of determination (R^2). In other words, frames predicted by these models are blurred (i.e., reduced R^2) to reduce their overall prediction error (as measured by the mean squared error (MSE)). To address this problem, we propose a new DNN inspired by the Finite Element analysis (FE-DNN). By dividing the GBR study area into small elements, and by introducing the so-called stiffness matrices concept from the finite element analysis into the proposed FE-DNN model, prediction accuracy is increased, while the details of data variations are preserved.

To investigate the performance of the proposed FE-DNN model, we employ it to forecast nitrogen distribution frames in the GBR from hindcast distributions provided by an existing Partial Differential Equations (PDE) based simulation model. This distribution follows a complicated set of PDEs (Baird et al., 2020). The *eReefs* modelling suite (Steven et al., 2019) provides plenty of simulated nitrogen distribution data based on biogeochemical transformations and the spatial distribution of total nitrogen across the GBR but does not forecast future values. In addition, there are some sparsely collected nitrogen measurements across the GBR, which are useful in understanding and predicting nitrogen distribution in the GBR. These criteria make nitrogen prediction a good case study for FE-DNN implementation.

The rest of this article is organized as follows. In Section 2, nitrogen in the GBR will be defined, and challenges in high-resolution nitrogen prediction will be discussed. Section 3 will describe FE-DNN as our proposed solution to the problem of next-frame nitrogen prediction in the GBR. This data is introduced in Subsection 4.1. We will then evaluate the accuracy of the FE-DNN model for nitrogen distribution forecasting in the rest of Section 4, where a detailed investigation of both the computational complexity and the ablation properties of our model is also provided. The paper is concluded in Section 5.

2. Background and Problem Definition

The GBR is recognized by UNESCO as a World Heritage Area of "Outstanding Universal Value" due to its great cultural and natural significance and unmatched biodiversity. As stated in the previous section, reduced water quality since European settlement has been identified as a key threat to

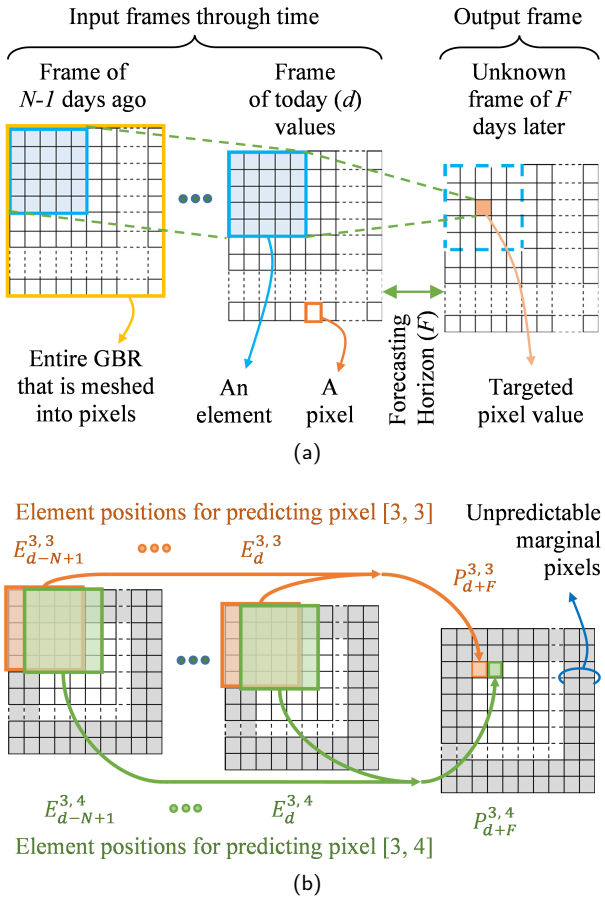


Figure 1: (a) The study area frame is meshed into small pixels that hold the temporal TN values of each of the past days until today. The pixels of each of the N historical frames are grouped into 5×5 elements, which are then used to predict the TN value of a target pixel F days after today. (b) The elements are then swept across the study area to predict all the target pixels one-by-one, while leaving a narrow 2-pixel margin of unpredictable TN values.

the health and resilience of GBR ecosystems (De'ath and Fabricius, 2010). While climate change is the single greatest threat to the world heritage status of the GBR, water quality adds cumulative pressure, reduces the resilience of reef ecosystems to climate change (MacNeil et al., 2019), and may be more readily subject to improvement through local management actions. To support such actions, it is important to be able to monitor and predict water quality on the scale of the whole GBR.

In this regard, land-sourced pollutants from farms and runoff in GBR catchments cause major damage to coral reefs. Among all the primary pollutants, the greatest water quality risks to the GBR are from nitrogen discharges (Reef-2050, 2021). Accordingly, the key component of the Australia's Reef 2050 Plan is to manage all nitrogen forms in GBR waters, including nitrogen oxides (NOx), ammonia ions (NH_4^+), detrital particulates, dissolved organic nitrogen, living biological forms of nitrogen, etc. (Baird et al., 2020).

There is continual recycling between all the above forms of nitrogen. Therefore, the Total Nitrogen (TN) is defined as the sum of all the nitrogen forms and employed in this paper as the parameter of interest.

TN distribution in the GBR can be simulated through a complicated set of PDEs which are solved by the *eReefs* modelling suite. The *eReefs* marine models are process-based simulation models that predict (in hindcast) the past spatial and temporal distribution of physical and water quality variables in space and time as a function of environmental drivers including river discharges, meteorological conditions, and global ocean currents.

This suite of models includes components that simulate hydrodynamics, sediment dynamics, biogeochemical transformations of water quality variables, and optical conditions in the water by numerical integration of a set of PDEs (Steven et al., 2019). The models are used to supplement sparse *in situ* water quality observations to support monitoring of the Great Barrier Reef, and have also been used to project how water quality might change under alternative land management scenarios, to support policy decisions for GBR catchments. Hence, they provide two types of predictions:

- *Hindcasting* (i.e., prediction of past conditions); and
- *Projection* (i.e., counter-factual scenario analysis).

However, *eReefs* does not currently provide forecasting predictions (i.e., prediction of conditions at a specific point in future time). Prediction of water quality in lakes, rivers and marine ecosystems has long been a focus of research and pragmatic modelling efforts. This began with simple empirical and physics-based models of phosphorus and nitrogen dynamics in aquatic systems and has steadily progressed over time to more and more complex coupled hydrodynamic-biogeochemical-ecosystem modelling systems. The range and development of these models has been reviewed by Ren and Yang (2000), Robson (2014), and Lindemann et al. (2017). Hydrodynamic water quality models in current use are generally complex, computationally intensive, and have limited application in forecasting contexts.

More recently, machine learning approaches have been adopted in water quality modelling. For example, Ahmed et al. (2019) applied the Adaptive Neuro-Fuzzy Inference System, Radial Basis Function Neural Network, and Multi-Layer Perceptron Neural Network to forecast time-series of three water quality parameters in a river basin as a function of a range of other water quality observations, while Haghiabi et al. (2018) compared the performance of an artificial neural network, a support vector machine, and a group method of data handling in a similar context. Most machine learning applications to date have been limited to forecasting time-series of water quality at one or a few discrete locations. Other works have focused on hybrid approaches such as using machine learning models as surrogate or meta-models for physics-based water quality models, or the use of machine learning to support data assimilation to imrpove the

performance of physics-based models. One example of this approach is the work of Margvelashvili et al. (2013), who used error-subspace emulators to assimilate remote sensing ocean color data into the eReefs marine sediment dynamics models.

In contrast to all previous works, we employ the DNN to forecast TN distribution in the wide GBR. Based on the *eReefs* findings, the TN in nearshore regions of the GBR is due mostly to river discharges. On the other hand, in the midshelf we can also see marine sources for TN. These various nitrogen sources in the wide coverage of the GBR, make the TN distribution a challenge task to predict. In the current study, we show how hindcast results from a process-based environmental model can be used to train an FE-DNN model to provide forecast predictions. Measured TN values in the GBR are scarce, making data-hungry DNN training unfeasible. To address this, a physics-informed neural network will be designed.

3. Proposed Model

As discussed in Section 1, there is no high-resolution model in the literature that is able to forecast TN distribution over the GBR. In our proposed model, shown in Fig. 1a, we solve the TN forecasting problem by meshing the GBR study area into small overlapping *elements*. To elaborate, each day in N days of the input frames consists of a frame of TN values of all the meshes. The TN value for each mesh is termed a pixel, which represents the average TN in a 16 km^2 mesh area. Several pixels are then grouped into a matrix to create a square element.

The historical element-wise TN values, until the present day, are used to predict a pixel value for F days later, where F is known as the forecasting horizon. As illustrated in Fig. 1b, the element is slid across the entire frame so that all possible pixel values can be predicted. The element size shown in Fig. 1 is 5×5. However, this size is a hyperparameter in our model that needs to be optimized. As shown, there is a narrow margin of pixels where TN values cannot be predicted. The size of this unpredictable margin is equal to half of the element size, e.g., 2 pixels for 5×5 elements and 3 pixels for 7×7 elements.

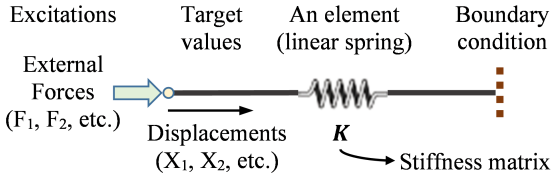
It is worth noting that classic image processing techniques for filling the marginal pixels (e.g., padding, flipping) are not suitable for our TN distribution prediction problem. This is mainly because every pixel in the GBR is highly dependent on its neighbors' historical TN values, which are chaotic, asymmetric, and highly dynamic (Skerratt et al., 2019). Therefore, we cannot simply fill these pixels by techniques such as constant padding or symmetrical flipping. Alternative solutions are needed to predict these marginal pixels, e.g., linear regression, convolutional neural network, LSTM, etc.

3.1. FE-DNN model

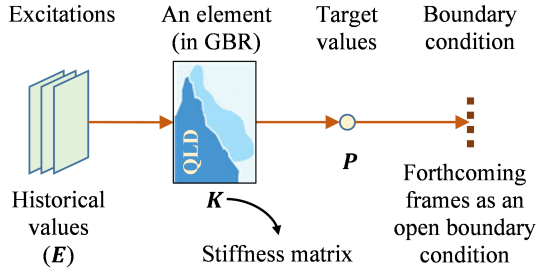
To accurately predict the output TN frame from the timed input frames, we design a novel DNN. This network is inspired by the concept of Finite Element Analysis (FEA),

Table 1
Conceptual comparison between FE-DNN and FEA.

Finite Element Analysis	
Abstract of the governing equation: $\boldsymbol{X} = f(\boldsymbol{F}, \boldsymbol{K})$	Processing technique: Recursive approach in linear algebra



FE-DNN	
Abstract of the governing equation: $\boldsymbol{P} = \boldsymbol{W}\, f(\boldsymbol{E}, \boldsymbol{K})$	Processing technique: Gradient descent in deep learning



and thus is dubbed FE-DNN. FEA is a well-known numerical method for solving boundary value problems in engineering. This method is extensively used in mathematical physics simulations, i.e., current transient response to the current transient inputs. However, FEA cannot be readily applied to forecasting scenarios for obtaining future response to the historical inputs, unless future inputs (boundary conditions) can be accurately estimated. To address this problem, our proposed FE-DNN integrates the concept of FEA with modern machine learning techniques to predict unknown future values.

Table 1 provides a conceptual comparison between FE-DNN and FEA, by using the well-known linear spring problem. FEA starts by dividing a study area into small elements (Huang et al., 2020). In each element, the governing equation in FEA (i.e., $\boldsymbol{X} = f(\boldsymbol{F}, \boldsymbol{K})$) takes both the stiffness matrix $\boldsymbol{K}$ and the excitations as input, and yields target values. Similarly, our DNN can be expressed as a system of linear equations (i.e., $\boldsymbol{P} = \boldsymbol{W}\, f(\boldsymbol{E}, \boldsymbol{K})$), where $\boldsymbol{P}$ represents the unknown output pixel values in the predicted frame, $\boldsymbol{W}$ is the known weights of the neurons, $\boldsymbol{E}$ is the known input frames, and $\boldsymbol{K}$ is the known stiffness matrix[1]. Additionally, the recursive approach to solve the FEA is similar to the recursive gradient descent approach to solve the system of linear equations in FE-DNN.

[1] During the training phase in our supervised learning, $\boldsymbol{P}$ is known and $\boldsymbol{W}$ is unknown.

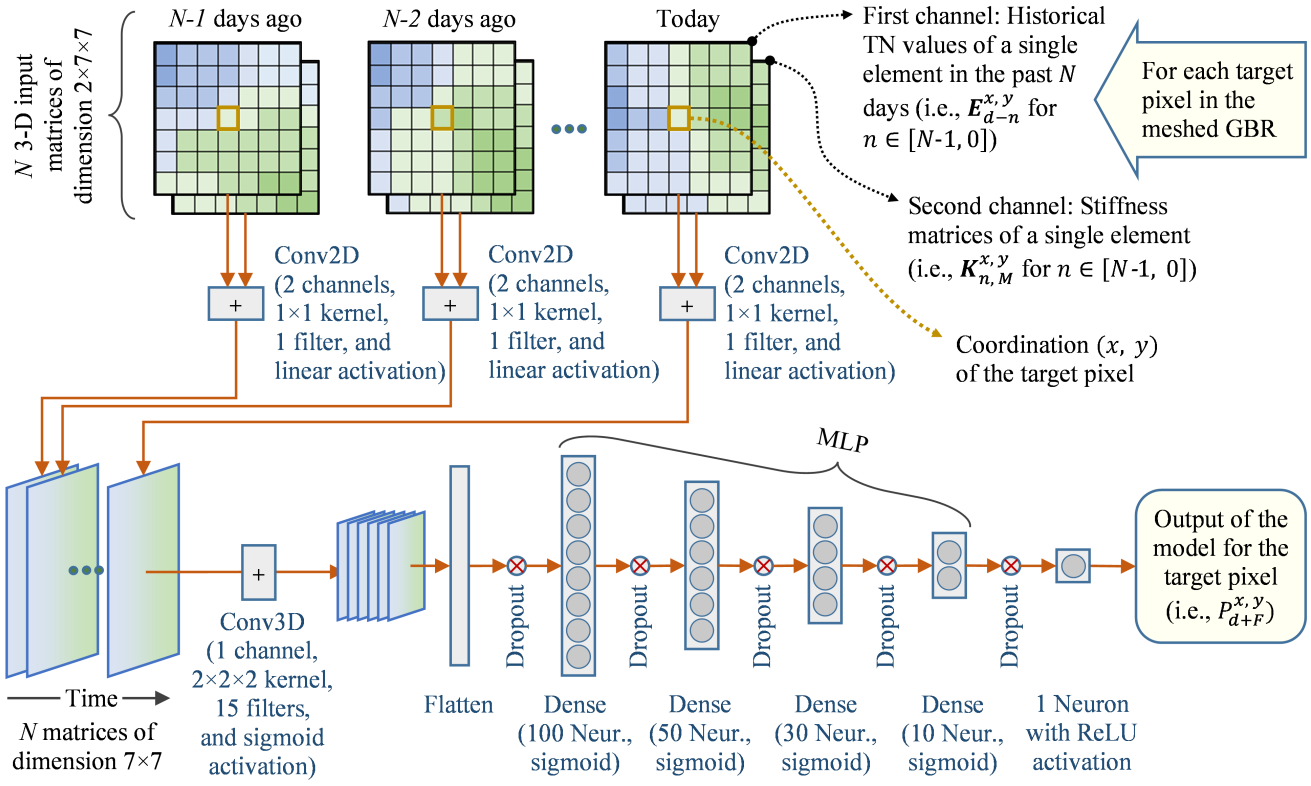


Figure 2: The architecture of the proposed FE-DNN network to predict a TN frame in F days later, where d represents today. Here, N input elements (as shown in Fig. 1) surrounding a target pixel with coordinates of (x, y) as well as N stiffness matrices (calculated for month M by (3)) are required to predict the TN value at the target pixel $P^{x,y}_{d+F}$. The output of the model for the target pixel is fed into Fig. 5. The shown kernel sizes, filter numbers, activation functions, etc. are optimized for this study, and they can be different in other next-frame prediction applications.

The stiffness matrix $\boldsymbol{K}$ in FEA for the spring problem represents the elastic behavior of the underlying material. We will discuss $\boldsymbol{K}$ in more detail in Section 3.2. However, in contrast to FEA that multiplies the inversed stiffness matrix into the excitation, we feed $\boldsymbol{K}$ as a separate input to our DNN (i.e., $f(\boldsymbol{E}, \boldsymbol{K})$). This way, the FE-DNN will learn the behavior of the stiffness matrix, in conjunction with input TN values variations across the historical frames.

Fig. 2 illustrates the architecture of the proposed FE-DNN model. It takes N elements along with N stiffness matrices as input. These inputs are fed into their N corresponding 2D Convolutional layers (Conv2D). The Conv2D represents $f(\boldsymbol{E}, \boldsymbol{K})$ in Table 1, which merges the stiffness matrices with their relevant TN elements. The resulting merged matrices then form a 3D matrix and fed to a Conv3D layer. Finally, the outputs of the Conv3D layer are flatten to enter a Multilayer Perceptron (MLP) with four dense layers.

Except for the last dense layer that uses the Rectified Linear Unit (ReLU) activation function, the rest of the MLP dense layers use sigmoid. Our experiments show that using ReLU in all dense layers enforces the lower TN bound to be 0. This makes the model lazy in truly learning the complex transformations in GBR nitrogen distribution. As a result, the model cannot capture TN variations, which results in a lower $\boldsymbol{R}^2$. In other words, a model with ReLU activation functions cannot capture TN variations in the vast GBR as good as it could do with sigmoid.

The total number of trainable weight parameters in the proposed FE-DNN model is 154,136. To avoid overfitting, while training these weights, five dropout points with a 20% dropping ratio are placed in layer intervals of Fig. 2. All the layers are equipped with the Ridge regularization of $L2 = 0.01$. Meanwhile, the learning rate is set to 0.001 in an effort to both improve convergence of weight learning and avoid overfitting.

To better understand how the proposed FE-DNN model works, a flowchart is presented in Fig. 3. This flowchart covers both the training and inference phases. The workflow starts by meshing the study area into pixels, and completes by saving the trained model or yielding the forecasted results. Data flow in this diagram has a main loop to sweep the input elements' locations to predict the TN values in every pixel of the output frame. This loop is marked by a dashed-line inside the figure.

3.2. Stiffness matrices

In FEA, stiffness matrix calculation is a pre-processing step of numerical modelling. The stiffness matrix can be defined as an approximate solution to the underlying PDEs,

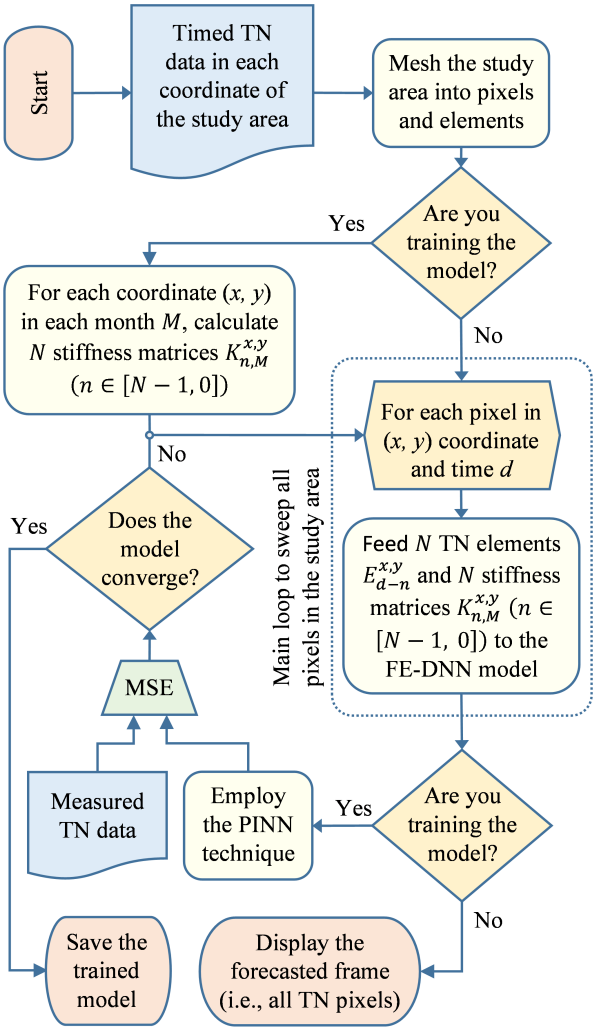


Figure 3: Flowchart of the FE-DNN workflow from input data to forecasted results.

which represent the elastic deformation of matters in accordance with both their own properties and the constant external perturbations (Huang et al., 2020).

Accordingly, calculating the stiffness matrix requires obtaining a solution to the complex underlying PDEs of the system under consideration. However, in our proposed FE-DNN method, instead of finding an approximate solution to these complex PDEs to achieve the required stiffness matrices, we use existing training data to extract the variation of the output pixel in response to the changes in the input frames. While the resulting matrix resembles the definition of the stiffness matrix in FEA, its calculation requires no knowledge of the underlying PDEs.

To elaborate, consider the linear spring problem in Table 1, with two external forces F_1, F_2, and one displacement value X_1. The spring constant is $k = 2.0$, which results in the stiffness matrix $\boldsymbol{K} = [2.0, -2.0]$. We simulate this problem for F_1, F_2, and X_1 in Fig. 4a, where F_1 and F_2 are sine functions in the presence of random Gaussian noise, and X_1 is the target displacement. The simulation is conducted for 2 seconds with 20 sample points, which form our training dataset. By dividing this time into 4 segments with 5 samples per segment, one can numerically calculate the stiffness matrix of each segment r, $\boldsymbol{K}_r$, as

$$\boldsymbol{K}_r \boldsymbol{X} = \boldsymbol{F} \;\Rightarrow\; \boldsymbol{K}_r = \frac{\boldsymbol{F}}{\boldsymbol{X}} \xrightarrow[\text{matrix algebra}]{\text{In the form of}} \boldsymbol{K}_r = \boldsymbol{F}\,\boldsymbol{X}^T \left(\boldsymbol{X}\,\boldsymbol{X}^T\right)^{-1}, \quad (1)$$

where $\boldsymbol{F}$ is a 2×5 matrix of five F_1 and F_2 samples, and $\boldsymbol{X}$ is a 1×5 matrix of five X_1 samples. The final stiffness matrix $\boldsymbol{K}$ of the linear spring problem can then be calculated by averaging $\boldsymbol{K}_r$ as follows

$$\boldsymbol{K} = \frac{1}{4} \sum_{r=1}^{4} \boldsymbol{K}_r \,. \quad (2)$$

The true versus calculated values of $\boldsymbol{K}$ are presented in Fig. 4a. As can be seen from the figure, the result has about 97% accuracy across the entire dynamic range.

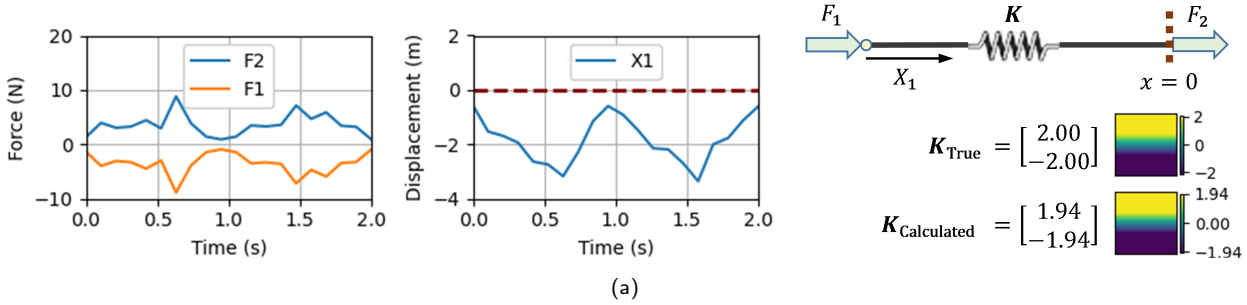
Considering the TN stiffness matrix in the GBR, applying (1) and (2) to our TN prediction problem requires the following adjustments:

1. Each pixel in the targeted output frame is influenced by the past N days, so we will have N stiffness matrices for each pixel.
2. We can split the wide GBR TN values in time, by calculating the stiffness matrices for each month of the year. In this way, the high-dynamics of the GBR will be better captured.
3. As illustrated in Fig. 2, in our proposed method $\boldsymbol{K}$ is calculated and provided to the model as a separate input. This is unlike FEA, where the stiffness matrix $\boldsymbol{K}$ is mathematically multiplied by its relevant element.
4. By assigning a 1×1 kernel and 1 filter to all the Conv2D layers in Fig. 2, these layers implement $\boldsymbol{E} + \boldsymbol{K}$. However, given the negative values in $\boldsymbol{K}$, the output of the Conv2D layers are $\boldsymbol{E} - \boldsymbol{K}$. Therefore, in our proposed model, unlike the original FEA implementation where the stiffness matrix is calculated as $\boldsymbol{K} = \boldsymbol{F}/\boldsymbol{X}$, it is calculated by subtracting the known outputs of the model ($\boldsymbol{P}$) from element $\boldsymbol{E}$, i.e. $\boldsymbol{K} = \boldsymbol{E} - \boldsymbol{P}$.

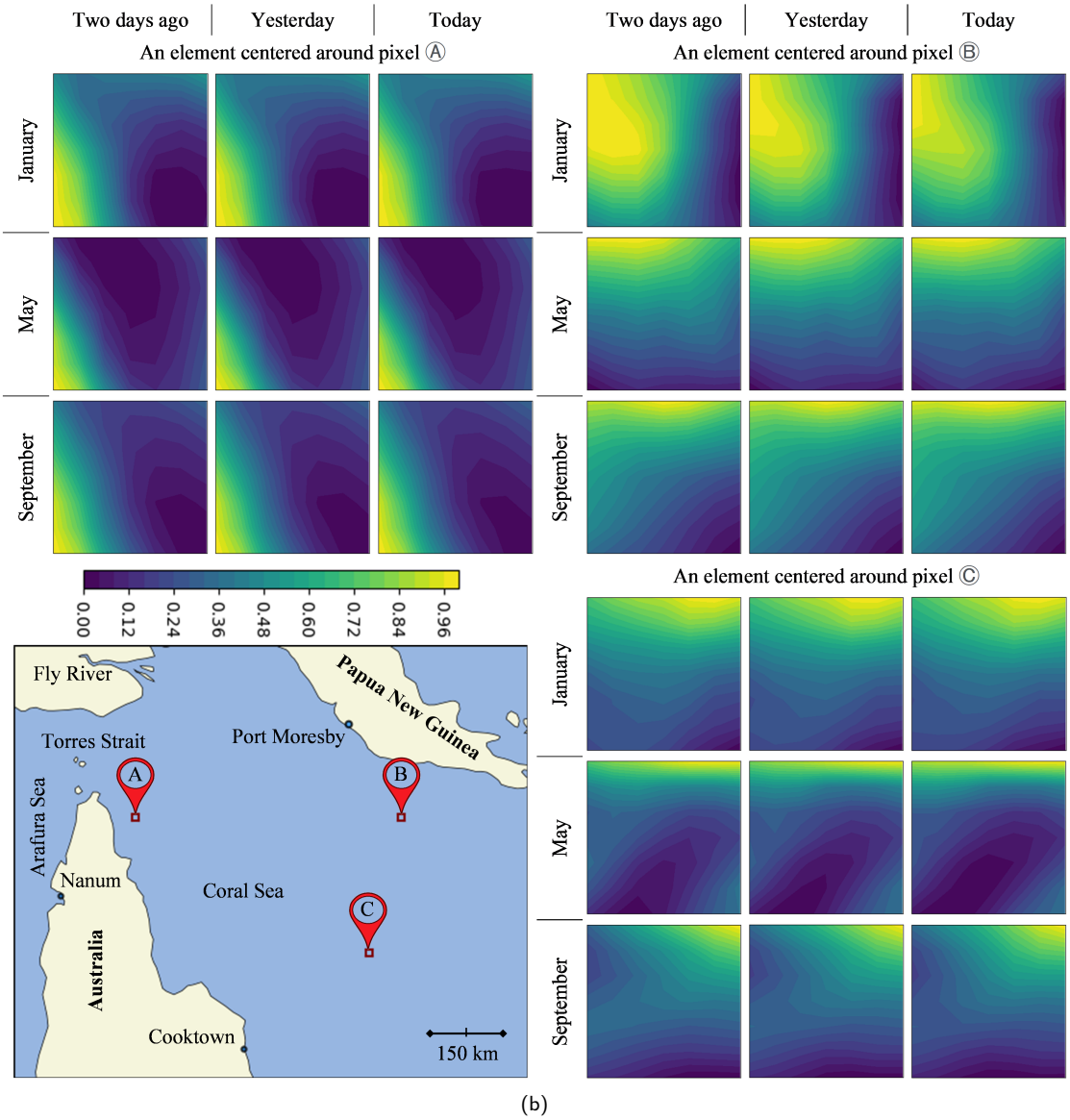
Given the above adjustments, we reformulate (1) and (2) to better address the requirements of our GBR application. As also shown in Fig. 1, consider the situation, where N input elements $\boldsymbol{E}$ around the target pixel at coordinate (x, y) are used to calculate its value $\boldsymbol{P}$ in F days after today $P^{x,y}_{d+F}$, in month $\boldsymbol{M}$. The stiffness matrix $\boldsymbol{K}$ will be

$$\boldsymbol{K}^{x,y}_{n,M} = \frac{1}{N_y} \frac{1}{N_d} \sum_{y} \sum_{d} \left(\boldsymbol{E}^{x,y}_{d-n} - P^{x,y}_{d+F}\, \boldsymbol{J} \right) \quad \forall n \in [0, N-1], \quad (3)$$

where y sweeps the years of the training dataset, N_y is the number of training years, d sweeps the days of month $\boldsymbol{M}$,



(a)



(b)

Figure 4: (a) Stiffness matrix calculation for the linear spring problem in Table 1 in the presence of a white Gaussian noise. (b) The stiffness matrices in normalized logarithmic scale of three typical geolocations Ⓐ at (11.18°S, 143.25°E), Ⓑ at (11.18°S, 148.17°E), and Ⓒ at (13.70°S, 147.57°E) in three months of January, May, and September, where the element size is 7×7.

N_d is the total number of days in month M, and $\boldsymbol{J}$ is an all-one matrix of the same size as our elements. As can be seen from (3), we will have N stiffness matrices for a given coordinate (x, y) in a given month M of the year, with a given forecasting horizon F. In other words, $\boldsymbol{K}^{x,y}_{N-1,M}$, $\boldsymbol{K}^{x,y}_{N-2,M}$, ..., $\boldsymbol{K}^{x,y}_{0,M}$ represent the average variations of a pixel in month M, in response to the element-wise TN variations in the last N days. It is worth mentioning that the stiffness matrix $\boldsymbol{K}$ in (3) depends on the month but is independent of both the day and year.

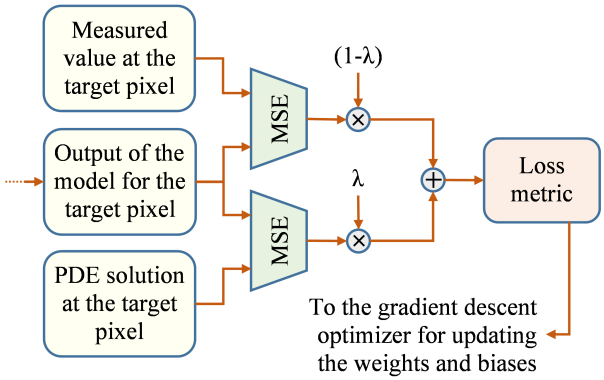


Figure 5: PINN loss function applied to our proposed FE-DNN model.

To better illustrate these calculations, the normalized stiffness matrices for three random geolocations $(x,\ y)$ in the GBR are plotted in Fig. 4b. These plots are made for the months of January, May, and September. The element size is set to be 7×7, and the forecasting horizon (F) is one day. The number of input frames (N) is equal to 3, resulting in three stiffness matrices per month. These stiffness matrices are labeled as *two days ago*, *yesterday*, and *today*, denoted by $\boldsymbol{K}^{x,\,y}_{2,\,M}$, $\boldsymbol{K}^{x,\,y}_{1,\,M}$, and $\boldsymbol{K}^{x,\,y}_{0,\,M}$, respectively.

3.3. Physics-informed neural network

To train the proposed FE-DNN model, one would require a large quantity of observational TN data in the GBR. However, the existing sparse TN measurements in GBR are insufficient for our data-hungry DNN. To overcome this problem, a novel method termed Physics-Informed Neural Network (PINN) (Zhu et al., 2021) is employed. The use of PINN enables us to merge scarce observational data with readily available *eReefs* simulation results, and use both types of data to train our neural network.

Despite its name, the PINN is not a new neural network on its own right, but a technique in defining a physics-informed loss function, which mixes PDE solutions with measured values (i.e., the ground-truth). Therefore, it can be applied to almost any neural network corresponding to a physical model that can be described by underlying PDEs.

The PINN-inspired loss function that we develop for our FE-DNN model is illustrated in Fig. 5, using the Mean Squared Error (MSE) metric. The *output of the model* in this figure is the output of our FE-DNN network in Fig. 2. We use this output to calculate two loss functions as follows

$$L_{\text{Measured}} = \frac{1}{N_{\text{data}}} \sum (TN_{\text{Output}} - TN_{\text{Measured}})^2\,, \quad (4)$$

$$L_{\text{PDE}} = \frac{1}{N_{\text{data}}} \sum (TN_{\text{Output}} - TN_{\text{PDE}})^2\,, \quad (5)$$

where N_{data} is the total number of data points, TN_{Measured} is the observational TN values, and TN_{PDE} is the simulated TN values obtained from *eReefs*.

The loss functions in (4) and (5) are then combined together to create the following overall loss metric

$$L = (1-\lambda)\, L_{\text{Measured}} + \lambda\, L_{\text{PDE}}\,, \quad (6)$$

where λ is an adjustable hyperparameter. We then use the loss function in (6) to train our model and to recursively optimize the unknown weights of the FE-DNN network. To summarize, we overcome the observational data sparsity problem by integrating the PINN technique with the process of DNN training.

4. Results and Discussions

In this section, we will start by introducing the measured TN data, along with the PDE simulation results for TN in the GBR. We will then optimize the element size, before proceeding to the accuracy analysis, computational complexity, and ablation studies.

4.1. Data sources

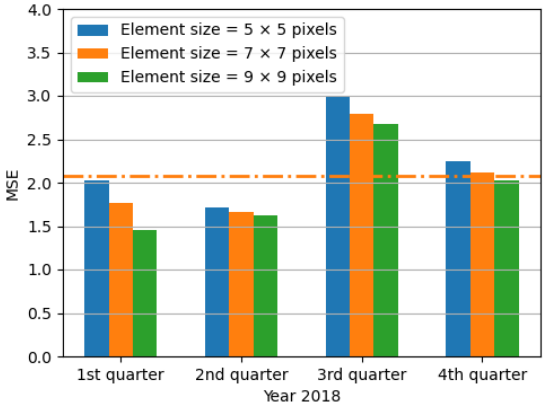
The proposed FE-DNN network is used to predict the TN distribution in the GBR. The observational TN values (i.e., TN_{Measured} in (4)) are gathered from the GBR Marine Park Authority Marine Monitoring Program (MMP), which is led by Australian Institute of Marine Science (AIMS) (AIMS-MMP, 2021). These measurements are sparsely gathered and thus are insufficient for training the proposed FE-DNN model. Hence, by integrating the PINN technique described in Section 3.3, the simulated data (from solving PDEs) are used to compensate for the scarcity of the measured data.

As discussed earlier in Section 2, to obtain the PDE solutions for the TN distribution in the GBR (i.e., TN_{PDE} in (5)), the *eReefs* modelling suite is employed. *eReefs* has a regional model on a 4 km grid (GBR4), which extends into the Coral Sea and covers the entire GBR area (NCI, 2021). However, this raw GBR4 biogeochemical model has another version, which is interpolated onto a regular grid. This version of the *eReefs* simulation data is downloadable from the AIMS website (AIMS-eReefs, 2021), and we have, therefore, used it in our study.

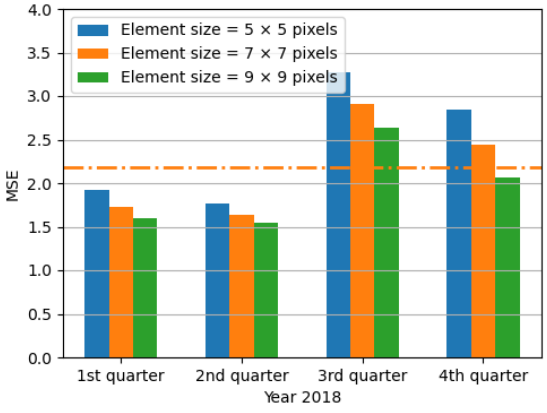
The *eReefs* simulation data from the AIMS website (AIMS-eReefs, 2021) is provided on a daily basis, from 2011 to 2018. We divide this time span into 2011 to 2017 for the training and validation dataset, and 2018 for the testing dataset. To increase the model training speed, we have spatially downsampled the dataset by a factor of 4, which has resulted in 16 km wide pixels, as mentioned in Section 3.

The TN concentrations within river deltas in the GBR are so high that using the full range of the data values for processing makes variations in other areas appear insignificant, even though there are significant and ecologically important variations in water quality (including sediment and nitrogen concentrations) throughout the nearshore regions and out to the Midshelf waters. To capture these variations, we use the logarithmic scale for model training as follows

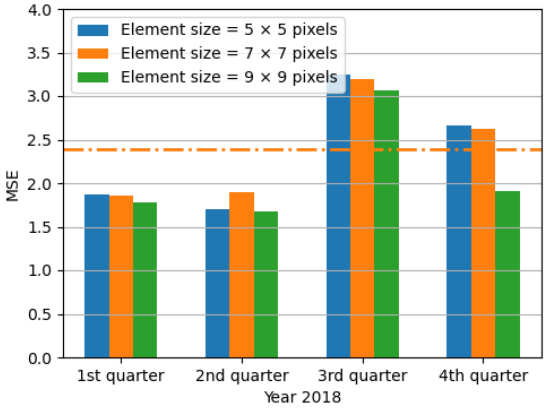
$$TN_{\log} = \log_{10}(TN + 1)\,. \quad (7)$$



(a)



(b)



(c)

Figure 6: Effect of element size on the MSE metric in (a) 3-day, (b) 5-day, and (c) 7-day forecasting horizons, where number of past days used for training is $N = 3$. The horizonal dashed lines show the average MSE for the 7×7 element size.

The added 1 in (7) is to avoid log(0). This equation is used both for TN scaling and for the stiffness matrix calculation. It is worth mentioning that the logarithmic scaling is not required in other next-frame prediction applications if the data is linearly distributed between its boundaries. To better understand the nature of these *eReefs* simulation outputs, the statistics of TN values are presented in Table 2. All the data in this table are in linear scale.

Table 2
Statistics of the values of TN_{PDE} from *eReefs* marine models.

Dataset	Year	Min	Max	Mean	STD
Train and Validation	2011	44.05	528.37	59.36	22.88
	2012	44.33	464.63	56.88	16.57
	2013	44.10	427.97	56.26	15.84
	2014	43.18	410.69	54.63	12.71
	2015	43.00	254.96	53.04	10.43
	2016	42.86	244.99	52.25	10.20
	2017	42.58	510.02	53.97	12.39
Test	2018	43.03	369.38	54.00	11.85

4.2. Prediction accuracy

As in the *eReefs* modelling suite, the physical unit of TN in this paper is $[mgN/m^3]$, which is the same as $[\mu gN/L]$. By contrast, TN measurements in the MMP are made in $[\mu mol/L]$. So, we need to convert the MMP values by multiplying them by 14.01, given that the molar mass of nitrogen is 14.01 g.

As stated earlier in Section 3.1, the element size is a hyperparameter that needs to be optimized. The effect of the element size on prediction accuracy is investigated in Fig. 6. The MSE values in this figure are averaged per quarter of the training year of 2018. In all cases of the 3-day, 5-day, and 7-day forecasting horizons, increasing the element size generally improves the performance. By contrast, a greater element size leads to more unpredictable marginal pixels, as shown in Fig. 1b. The horizontal dash lines in this figure indicate the average MSE values for the 7×7 element size. In the reported results of our FE-DNN, we have used the element size of 7×7.

Fig. 7 demonstrates our FE-DNN TN prediction results for a typical day of the first 8 months in 2018 (i.e., for the test dataset). The true values and the absolute differences between the true and predicted values are also plotted. All the predictions are made using 7×7 elements with $N = 3$, and all the absolute differences are multiplied by 1000. The results are in logarithmic scale, and they show 1-day, 3-day, 5-day, and 7-day forecasting horizons. This demonstrates the ability of our model to very closely predict TN values across the entire GBR area.

Due to the adoption of the element size of 7×7 and based on the illustration in Fig. 1, three marginal pixels are left unpredicted by the proposed FE-DNN model. These pixels are predicted in Fig. 7 by employing a simple linear regression model. The higher error values of the regression model are obvious in the surrounding margins of this figure, especially in the longer forecasting horizons.

For all $F = 1$ in Fig. 7a, $F = 3$ in Fig. 7b, $F = 5$ in Fig. 7c, and $F = 7$ in Fig. 7d, the 1^{st}, 2^{nd}, and 3^{rd} days of each month are fed to the model's input. The 4^{th}, 6^{th}, 8^{th}, or 10^{th} days of the months are forecast in those forecasting horizons, respectively. The absolute differences (i.e., the

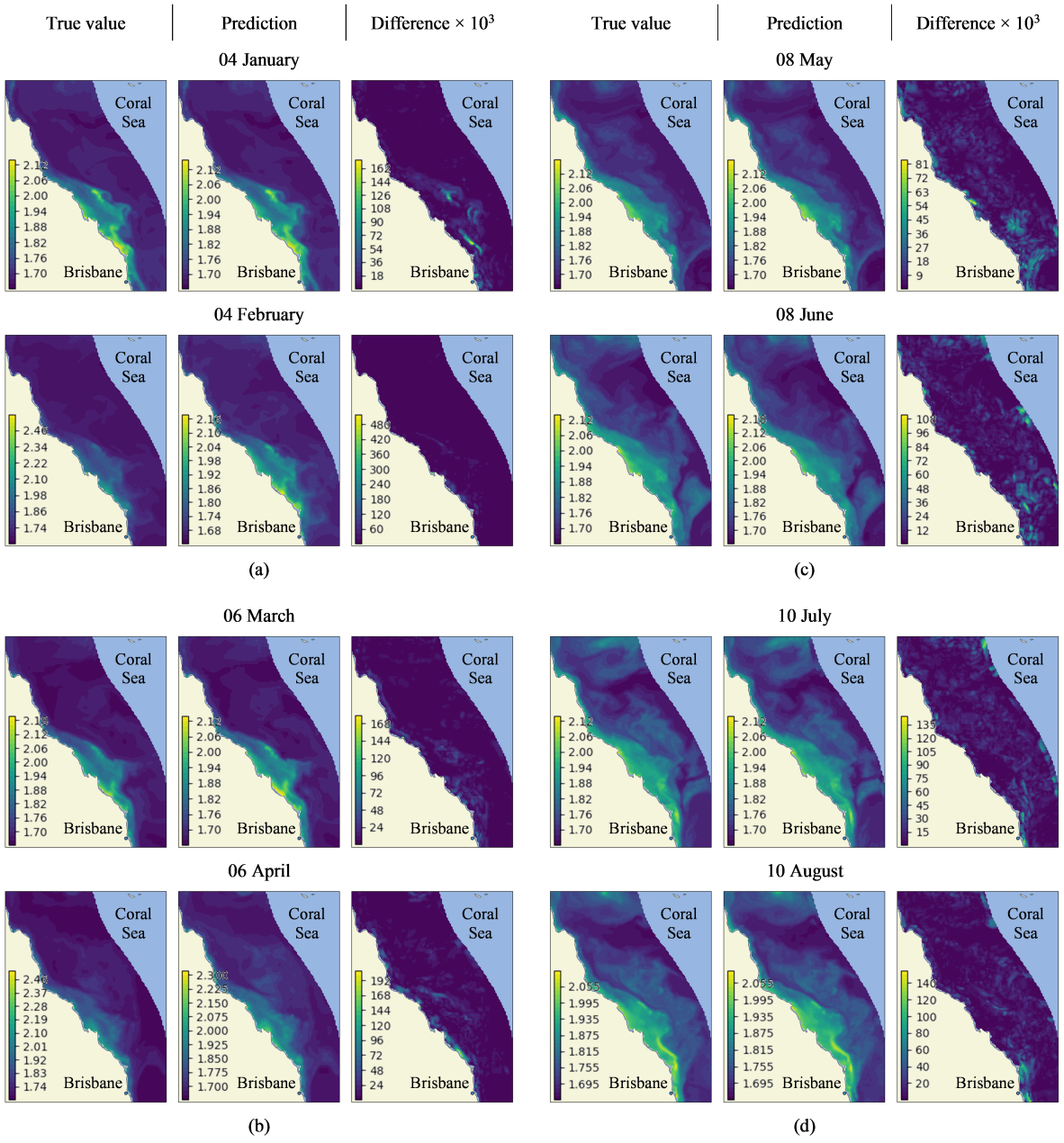


Figure 7: True values, predictions, and their absolute differences (multiplied by 10^3) for the 2018 test $TN_{\log}$ dataset in the GBR, for (a) 1-day (b) 3-day, (c) 5-day, and (d) 7-day forecasting horizons.

prediction errors) spread geographically, when increasing the forecasting horizon F, resulting in a larger MSE.

Fig. 7 shows that not only the proposed FE-DNN can result in very accurate prediction, it also generates unblurred output frames (i.e., a high coefficient of determination R^2), which are not achievable by conventional next-frame prediction methods. Both of these advantages are demonstrated in more detail in Table 3, where the MSE is used to measure the prediction accuracy of our proposed model, while R^2 indicates the high-resolution and unblurred prediction frames. Here, the MSE is as low as 3% of the test data on average, and it is almost constant for all studied F values. In addition, the R^2 values imply that, in all prediction cases, we have accurately captured around $98-99\%$ of the predicted TN variations throughout the GBR.

We always use a direct forecasting approach in the reported scenarios of this paper. This means that we separately train the model for each forecasting horizon. However, it is also possible to employ the direct-recursive forecasting approach. In other words, we can use a previously predicted frame as input to predict the next frame and so forth.

To investigate how well our model predicts future values compared with simple propagation of the historical input values, a study is conducted in Fig. 8, where the MSE is employed as a mathematical distance metric to measure the pairwise distances between our prediction, true future

Table 3
MSE and R^2 metrics in TN prediction of the test dataset in 2018, using elements of size 7×7 pixels.

Metric	Forecasting horizon	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
MSE	1 Day	1.21	0.83	1.75	1.11	0.98	1.62	2.25	3.50	2.04	1.33	1.55	1.55
	3 Days	1.76	1.22	2.30	1.91	1.15	1.92	2.84	3.03	2.51	1.81	2.24	2.31
	5 Days	1.74	1.33	2.11	1.55	1.31	2.07	2.66	3.33	2.73	2.15	2.46	2.74
	7 Days	2.04	2.00	1.54	1.97	1.49	2.23	2.47	4.27	2.84	2.40	2.29	3.18
R^2 (%)	1 Day	99.22	99.45	99.12	99.40	99.41	99.07	98.84	98.32	98.89	99.13	98.91	99.07
	3 Days	98.86	99.20	98.85	99.00	99.31	98.90	98.54	98.53	98.64	98.82	98.45	98.63
	5 Days	98.88	99.11	98.93	99.15	99.21	98.81	98.63	98.39	98.51	98.59	98.27	98.40
	7 Days	98.68	98.61	99.25	98.87	99.09	98.70	98.71	97.94	98.47	98.46	98.34	98.12

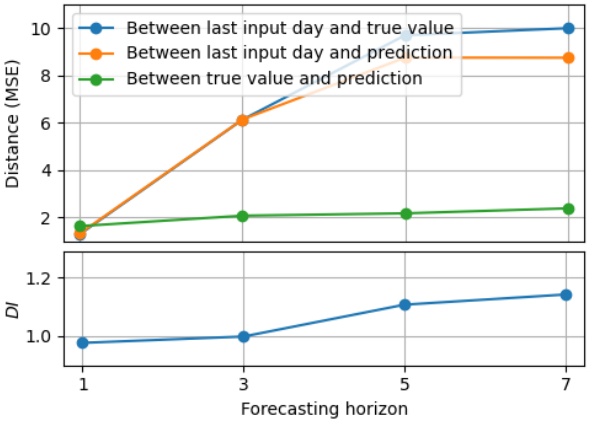


Figure 8: Pairwise mathematical distances between the input values of the last day, true values in F days later, and our prediction, using elements of size 7×7 pixels and $N = 3$, along with their corresponding DI.

values, and historical input values. To better comprehend this figure, a new Distance Index (DI) metric is defined as

$$DI = \frac{\text{dist}\,(\text{true future values},\ \text{historical input values})}{\text{dist}\,(\text{prediction},\ \text{historical input values})}, \quad (8)$$

where dist(·) stands for the MSE distance. As expected, the distance between future values (in F days later) and historical values (in the current day) increases with F. Besides, by increasing the forecasting horizon, our prediction broadens its distance from the input values. Also, the prediction keeps its constant distance with the true values, even for the case of a week ahead prediction, i.e., ($F = 7$). All these desirable distancing behaviours, keep the DI value close to 1.0 for all forecasting horizons.

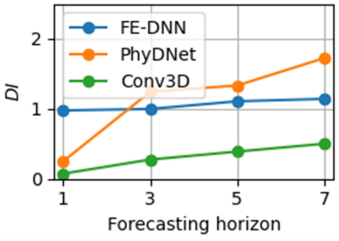
Finally in Table 4, the performance of the proposed FE-DNN model with 7×7 elements is compared with both the Conv3D model by Mathieu et al. (2016) and the ConvLSTM PhyDNet model by Guen and Thome (2020). At the time of writing, PhyDNet (Guen and Thome, 2020) is ranked as the best video predicting model in multiple categories (papers with code). All the comparisons in Table 4 are conducted for the 2018 test dataset. As can be seen in this table, neither of Conv3D nor PhyDNet can accurately predict the TN dynamics in the wide GBR. The MSE is greater than 10 for larger F values and the Mean Absolute Error (MAE) is always greater than one, which results in $R^2 < 90\%$ and $DI \neq 1$.

The FE-DNN error bias values in Table 4 are close to zero, indicating unbiased predictions. Another performance metric in Table 4 is the scatter index, which is calculated in percentage by dividing the root-MSE by the mean of the true values in each day and expressing the result as a percentage. The near-zero scatter index of FE-DNN indicates a low relative error with respect to the mean TN.

The reliability analysis in Table 4 calculates the percentage of relative absolute errors that are less than 0.2 (according to the Chinese Standards) (Saberi-Movahed et al., 2020). High reliability values show that this approach is consistently accurate. U95 is another performance metric in Table 4, which is a type of uncertainty metric (Saberi-Movahed et al., 2020). This metric considers the 95% confidence interval, and calculates the uncertainty range of models' predictions. The proposed FE-DNN model offers the lowest uncertainty values of the models evaluated, with 95% confidence.

We also conduct an F-test in Table 4 to analyze the variance of the forecasted TN. The F-test is a statistical test to find out whether the predictions and the true values have the same variance. The Null Hypothesis (H0) is that the variances are equal. P-values greater than 0.05 reject the H0, indicating that variances are not equal. Based on this test, our model perfectly catches the variations in TN, while other models cannot adapt to the rapid TN changes in the wide GBR. It is worth reminding from Section 1 that both

Table 4
Comparing the performance of the proposed FE-DNN model with two recently published works in the literature.

DNN Model	Forecasting horizon (F)				Forecasting horizon (F)				Forecasting horizon (F)			
	1	3	5	7	1	3	5	7	1	3	5	7
	MSE				MAE				R^2 (%)			
Conv3D by Mathieu et al. (2016)	25.58	31.16	38.83	41.02	3.07	3.40	3.79	3.80	88.3	85.2	81.5	80.5
PhyDNet by Guen and Thome (2020)	8.59	11.62	18.72	21.30	1.60	2.03	2.48	2.73	95.4	92.6	88.7	86.5
The Proposed FE-DNN model	1.64	2.08	2.18	2.39	0.75	0.84	0.87	0.91	99.1	98.8	98.7	98.6
	Bias				Scatter Index (%)				Reliability (%)			
Conv3D by Mathieu et al. (2016)	-0.56	-0.38	-0.22	0.16	6.4	7.9	9.1	9.5	97.0	96.5	95.3	94.7
PhyDNet by Guen and Thome (2020)	-0.41	-0.68	-0.26	-0.38	3.8	5.6	6.96	7.6	98.7	98.8	97.6	97.2
The Proposed FE-DNN model	-0.04	-0.01	0.14	0.23	2.3	2.6	2.7	2.8	99.97	99.96	99.95	99.95
	U95				F-test (p-value)				*DI*			
Conv3D by Mathieu et al. (2016)	0.28	0.29	0.29	0.29	0.34	0.36	0.40	0.47				
PhyDNet by Guen and Thome (2020)	0.26	0.27	0.27	0.27	0.30	0.33	0.41	0.44				
The Proposed FE-DNN model	0.26	0.26	0.26	0.26	2e-34	3e-25	9e-29	3e-20				

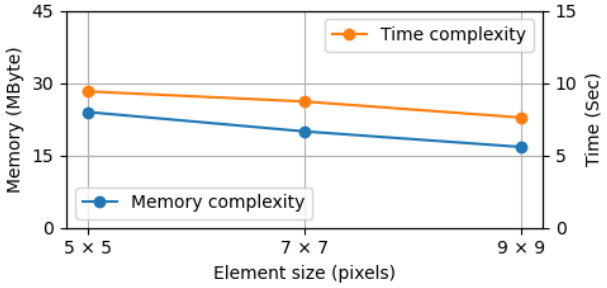


Figure 9: Time and memory complexities with respect to the element size for one-day forecasting horizon, with $N = 3$.

the Conv3D and PhyDNet next-frame predictors treat each frame as a whole. Consequently, they fall short against the proposed FE-DNN model that borrows the finite element concept from FEA, and uses the modified stiffness matrices to produce accurate predictions.

4.3. Computational complexity

The proposed FE-DNN model was implemented using Keras APIs of TensorFlow in Python. The model was trained on a windows machine with Intel® Core i7-7700HQ CPU, NVIDIA® GeForce 1050 GPU, and 16 GB RAM.

The computational complexity is analyzed in terms of computational resource requirements, as well as the running times (Vaz et al., 2017). In this regard, we evaluate our model's demand when varying the input element size. This analysis for one-day TN prediction in the entire GBR is illustrated in Fig. 9. The results are obtained during inference, where the weights and biases are fixed. The memory in this figure refers to local RAM consumption, not the GPU memory in use. It also excludes the TN input and output data, and only includes the model variables.

The small reduction in resource demand when increasing the element size is due to an increase in the number of unpredictable marginal pixels. This results in a need for predicting fewer pixels by the model. Overall, the plots reveal almost a constant demand for both the simulation time and the memory, which suggest our model is efficient.

The time and memory demands of the proposed FE-DNN model with 7×7 elements are compared with those of the TensorFlow implementation of Conv3D (Mathieu et al., 2016) and the PyTorch implementation of PhyDNet (Guen and Thome, 2020) in Table 5. FE-DNN has a similar memory footprint to the contribution of Mathieu et al. (2016), because both of them are based on the CONV3D neural networks. By contrast, the PhyDNet model demands larger

Table 5
Comparing time and memory complexities of the proposed FE-DNN with two recently published works in the literature.

DNN Model	Time	Memory (MByte)
Conv3D by Mathieu et al. (2016)	1.0 ms	17.06
PhyDNet by Guen and Thome (2020)	13.0 ms	943.3
The Proposed FE-DNN model	8.7 s	19.97

Table 6
Evaluating the elimination of selected blocks on accuracy of the one-day TN forecasting.

Ablation study	MSE	MSE increment
Including all the blocks	1.64	0%
Excluding stiffness matrix	1.94	18%
Excluding the Conv3D layer	1.85	12%
Excluding the first dense layer (with 100 neurons)	1.76	7%
Excluding both the first and second dense layers	1.98	21%

RAM, as it is based on the ConvLSTM neural networks. One-day TN prediction in the entire GBR takes longer in the FE-DNN model. This is due to the fact that our model sweeps the study area pixel-by-pixel, while the other two models digest the whole input TN frame at once.

4.4. Ablation study

In this subsection, an ablation study is conducted to better understand the impacts of different blocks of our model shown in Fig. 2. The ablation study calculates the overall accuracy of the model, when leaving a target block out of the structure (Du, 2020). To elaborate, we quantify the importance of any desired block, simply by omitting it from the ensemble of the proposed FE-DNN.

The results of the ablation study are shown in Table 6, using 7×7 elements over the test dataset in 2018. The MSE in the first row is calculated in presence of all the blocks, and it is averaged over the 12 months. As expected, the MSE increases in subsequent rows, by removing functional parts from the structural body. The difference between the MSE values in the first row and any other row is an indicator of the significance of the excluded block in the prediction performance of our proposed model.

Comparing all the MSE values in this table reveals the importance of the stiffness matrix in the next-frame analysis. Removing this FEA-inspired parameter reduces the model accuracy by 18%. Moreover, the overall effect of the four dense layers of MLP in Fig. 2 seems to be more significant than removing the Conv3D layer.

4.5. Limitations

Similar to other DNN-based models, the proposed FE-DNN is subject to some limitations. These limits are applicable when using the FE-DNN in other geolocations or employing it to forecast other environmental parameters. Some of these limitations are listed bellow.

- Our FE-DNN technique does not rely on PDEs or their solutions. As a DNN, FE-DNN learns the behavior of the underlying system only by looking at the training data. So, the main limitation of the developed model is the availability of training data. It is worth noting that, access to suitable training data is one of the main limitations of any DNN model.

- The current FE-DNN model is trained to forecast the TN distribution in the GBR. Applying the developed model to another study area requires re-training the model with local data, which must be available for each pixel on a daily time-step to achieve comparable results. Besides, the proposed model can be retrained regularly to take note of future challenges such as climate change and how they impact water quality.

- The proposed FE-DNN is only suitable for any spatio-temporal data that are gathered or interpolated in regular spatial nodes in regular time intervals. For example, the model can be applied to remotely sensed observation forecasting, only if its data are regularly interpolated in both the time and spatial domains to fill gaps due to sun-glint, clouds, or other observational quality issues. Though, this limitation will be present for other similar forecasting models that require constant spatiotemporal training data.

- Computational time and memory resources are two important limitations of this model. To be able to train the model for the entire GBR in a decent time, we have downsampled the dataset into spatially 16-km wide pixels. In the absence of downsampling, the lack of computation resources would impose a significant problem. However, this problem can be addressed by using a more powerful computing unit.

- This approach supports short-term forecasting of sediment concentrations in the GBR, complementing the capabilities of simulation models that are used to provide longer-term projections to support management and policy decisions to protect the GBR from land-derived pollutants and climate change impacts. While short-term forecasting does not support long-term policy decisions, it can be used to support short-term operational decisions, such as where and when to conduct in situ monitoring and process studies to improve the value of monitoring data.

4.6. Future directions

Future research can involve actions either to address the limitations discussed in Section 4.5, or to enhance the

capabilities of the proposed model. Some of these actions are discussed below.

FEA is a numerical technique to solve large-scale PDEs, arising in engineering and mathematical physics. FEA can also deal with arbitrarily shaped regions, as long as a discrete representation of the region (i.e., the meshing) exists (Huang et al., 2020). Relying on the FEA concept, the proposed FE-DNN is applicable to almost any physical or environmental next-frame forecasting problem with ruling PDEs. While the next-frame prediction of TN is carried out in this contribution, and the next-frame prediction of sea surface temperature is conducted by Guen and Thome (2020), the proposed FE-DNN model can be applied, in future research, to many other environmental parameter such as heat transfer, water flow, small particle movements, etc.

Remote sensing data could be extremely useful in training our data-driven DNN-based model. While remote sensing TN data for the wide GBR is not available, ocean color algorithms have been developed to provide remote sensing observations of other water quality variables, including chlorophyll-a, total suspended sediments, Secchi depth, and benthic photosynthetically active radiation (Magno-Canto et al., 2019; Petus et al., 2019). There are some published works in the literature that have tried to estimate nitrogen distribution over wide areas using correlations with other remotely sensed environmental parameters. For example, Sarangi (2012) estimated nitrogen in southern Indian waters using remotely sensed sea surface temperature, and Wang et al. (2018) estimated nitrogen in the coastal regions of East China Sea using remotely sensed sea surface salinity and sea surface reflectance. Having said that, an accurate algorithm for retrieving TN levels at oceanic scales in optically diverse waters from remote sensing observations has yet to be developed.

Finally, training of computationally expensive DNN solutions like the FE-DNN requires a variety of hardware resources. This demand for computational resources can be handled by existing parallel processing techniques, e.g., the shared-memory multiprocessors or the Distributed Computing Systems (DCS) (Jahanbakht et al., 2021). GPUs, FPGAs, and multi-core CPUs are few examples of the shared-memory parallelization techniques. DCS, on the other hand, consist of a network of cooperating computers that offer high-performance data processing. Using an on-premise DCS or a cloud-based distributed computing service like Amazon AWS, Microsoft Azure, etc. can be the next step for FE-DNN implementation research.

5. Conclusion

Inspired by the well-known FEA, we proposed the FE-DNN model for next-frame prediction of physical parameters in wide spatial coordinates. Our model is applicable to any environmental modelling scenarios, which are governed by underlying PDEs. We applied our novel model to the problem of TN distribution prediction in the GBR. To the best of our knowledge, our study is the first to use a data-driven machine learning approach for nitrogen prediction in the GBR. One challenge in training our DNN-based model is the scarcity of observational TN data in the GBR. To address this problem, we employed the PINN technique to merge the large amounts of simulated data with the sparse measurement data. This enabled us to successfully train our proposed FE-DNN model for TN forecasting. The performed analyses revealed that our next-frame predictor model achieves a very high accuracy with a low prediction MSE, while yielding high-resolution prediction frames with very high R^2 values. The calculated R^2 metric was more than 98%, resulting in unblurred TN prediction frames in the entire GBR. We believe that our model and this study can be beneficial and support internally significant water quality programs like the Australia's Reef 2050 Plan. This can help improve ecosystem recovery and resilience by informed decision making based on accurate prediction modelling. Furthermore, it can be adopted by existing hindcasting simulators to provide accurate forecasting predictions.

Software and Data Availability

The observational TN values are gathered from the GBR Marine Park Authority MMP, which is led by Australian institute of marine science (AIMS-MMP, 2021). The PDE solutions for the TN distribution in the GBR are obtained from the *eReefs* modelling suite. To elaborate, the *eReefs* regional GBR4 biogeochemical simulation data are downloaded from the AIMS website (AIMS-eReefs, 2021). The proposed FE-DNN model is implemented by Keras APIs of TensorFlow 2.5.0 in Python 3.8.

Acknowledgment

The *eReefs* model simulations were produced as part of the *eReefs* project (eReefs.info), a collaboration between the Science Industry Endowment Fund (SIEF), the Commonwealth Scientific Industrial Research Organisation (CSIRO), the Australian Institute of Marine Science (AIMS), the Bureau of Meteorology (BOM), and the Great Barrier Reef Foundation (GBRF), with support from BHP Billinton Mitsubishi Alliance, the Australian and Queensland governments, and with observations obtained through the Integrated Marine Observing System (IMOS). The marine monitoring program is funded by the GBR Marine Park Authority, and data provided by AIMS.

References

Ahmed, A.N., Othman, F.B., Afan, H.A., Ibrahim, R.K., Fai, C.M., Hossain, M.S., Ehteram, M., Elshafie, A., 2019. Machine learning methods for better water quality prediction. Journal of Hydrology 578, 124084.

AIMS-eReefs, 2021. A mirror eReefs model data, with custom aggregations performed by the eAtlas team. www.aims.gov.au.

AIMS-MMP, 2021. Australian institute of marine science: water quality particulate and dissolved nutrient data. www.aims.gov.au.

Baird, M.E., Wild-Allen, K.A., Parslow, J., Mongin, M., Robson, B., Skerratt, J., Rizwi, F., Soja-Woźniak, M., Jones, E., Herzfeld, M., 2020.

CSIRO Environmental Modelling Suite (EMS): scientific description of the optical and biogeochemical models (vB3p0). Geoscientific Model Development 13, 4503–4553.

Bell, P.R.F., Elmetri, I., Lapointe, B.E., 2014. Evidence of large-scale chronic eutrophication in the Great Barrier Reef: quantification of chlorophyll a thresholds for sustaining coral reef communities. Ambio 43, 361–376.

Coggan, A., Thorburn, P., Fielke, S., Hay, R., Smart, J.C.R., 2021. Motivators and barriers to adoption of improved land management practices: a focus on practice change for water quality improvement in Great Barrier Reef catchments. Marine Pollution Bulletin 170, 112628.

De'ath, G., Fabricius, K., 2010. Water quality as a regional driver of coral biodiversity and macroalgae on the Great Barrier Reef. Ecological Applications 20, 840–850.

Du, L., 2020. How much deep learning does neural style transfer really need? an ablation study, in: Proc. Winter Conference on Applications of Computer Vision (WACV), Snowmass, CO, USA. pp. 3139–3148.

Guen, V.L., Thome, N., 2020. Disentangling physical dynamics from unknown factors for unsupervised video prediction, in: Proc. Computer Vision and Pattern Recognition (CVPR), Virtual. pp. 11474–11484.

Haghiabi, A.H., Nasrolahi, A.H., Parsaie, A., 2018. Water quality prediction using machine learning methods. Water Quality Research Journal 53, 3–13.

Hong, S., Kim, S., Joh, M., Song, S.K., 2018. Psique: next sequence prediction of satellite images using a convolutional sequence-to-sequence network. Computing Research Repository 1711.10644, 1–5.

Howarth, R.W., Marino, R., 2006. Nitrogen as the limiting nutrient for eutrophication in coastal marine ecosystems: evolving views over three decades. Limnology and oceanography 51, 364–376.

Huang, R., Lu, M., Peyton, A., Yin, W., 2020. A novel perturbed matrix inversion based method for the acceleration of finite element analysis in crack-scanning eddy current NDT. IEEE Access 8, 12438–12444.

Huang, W., Ma, W., Liu, X., Peng, W., Zhang, J., 2021. Numerical study of hydrodynamics and water quality in Qinhuangdao coastal waters, China: implication for pollutant loadings management. Environmental Modeling & Assessment 26, 63–76.

Jahanbakht, M., Xiang, W., Hanzo, L., Rahimi Azghadi, M., 2021. Internet of Underwater Things and big marine data analytics – a comprehensive survey. IEEE Communications Surveys and Tutorials 23, 904–956.

Khan, U., Cook, F.J., Laugesen, R., Hasan, M.M., Plastow, K., Amirthanathan, G.E., Bari, M.A., Tuteja, N.K., 2020. Development of catchment water quality models within a realtime status and forecast system for the Great Barrier Reef. Environmental Modelling & Software 132, 104790.

Kroon, F.J., Kuhnert, P.M., Henderson, B.L., Wilkinson, S.N., Kinsey-Henderson, A., Abbott, B., Brodie, J.E., Turner, R.D.R., 2012. River loads of suspended solids, nitrogen, phosphorus and herbicides delivered to the Great Barrier Reef lagoon. Marine pollution bulletin 65, 167–181.

Kroon, F.J., Thorburn, P., Schaffelke, B., Whitten, S., 2016. Towards protecting the Great Barrier Reef from land-based pollution. Global change biology 22, 1985–2002.

Lindemann, C., Aksnes, D.L., Flynn, K.J., Menden-Deuer, S., 2017. Modeling the plankton–enhancing the integration of biological knowledge and mechanistic understanding. Frontiers in Marine Science 4, 358.

MacNeil, M.A., Mellin, C., Matthews, S., Wolff, N.H., McClanahan, T.R., Devlin, M., Drovandi, C., Mengersen, K., Graham, N.A.J., 2019. Water quality mediates resilience on the Great Barrier Reef. Nature Ecology & Evolution 3, 620–627.

Magno-Canto, M.M., McKinna, L.I.W., Robson, B.J., Fabricius, K.E., 2019. Model for deriving benthic irradiance in the Great Barrier Reef from MODIS satellite imagery. Optics Express 27, 1350–1371.

Margvelashvili, N., Andrewartha, J., Herzfeld, M., Robson, B.J., Brando, V.E., 2013. Satellite data assimilation and estimation of a 3D coastal sediment transport model using error-subspace emulators. Environmental modelling & software 40, 191–201.

Mathieu, M., Couprie, C., LeCun, Y., 2016. Deep multi-scale video prediction beyond mean square error, in: Proc. 4th International Conference on Learning Representations (ICLR), San Juan, Puerto Rico. pp. 1–14.

McCloskey, G.L., Baheerathan, R., Dougall, C., Ellis, R., Bennett, F.R., Waters, D., Darr, S., Fentie, B., Hateley, L.R., Askildsen, M., 2021. Modelled estimates of fine sediment and particulate nutrients delivered from the Great Barrier Reef catchments. Marine pollution bulletin 165, 112163.

McKinnon, A.D., Duggan, S., Logan, M., Lønborg, C., 2017. Plankton respiration, production, and trophic state in tropical coastal and shelf waters adjacent to northern Australia. Frontiers in Marine Science 4, 346.

Najafzadeh, M., Ghaemi, A., Emamgholizadeh, S., 2019. Prediction of water quality parameters using evolutionary computing-based formulations. International Journal of Environmental Science and Technology 16, 6377–6396.

Najafzadeh, M., Homaei, F., Farhadi, H., 2021. Reliability assessment of water quality index based on guidelines of national sanitation foundation in natural streams: integration of remote sensing and data-driven models. Artificial Intelligence Review , 1–33.

Najafzadeh, M., Niazmardi, S., 2021. A novel multiple-kernel support vector regression algorithm for estimation of water quality parameters. Natural Resources Research , 1–15.

NCI, 2021. eReefs GBR4 BGC and sediments model data by Australian national computational infrastructure. www.nci.org.au.

Petus, C., Waterhouse, J., Lewis, S., Vacher, M., Tracey, D., Devlin, M., 2019. A flood of information: using Sentinel-3 water colour products to assure continuity in the monitoring of water quality trends in the Great Barrier Reef (Australia). Journal of environmental management 248, 109255.

Reef-2050, 2021. Australian and Queensland government: The Reef 2050 plan. www.environment.gov.au.

Ren, L., Yang, J., 2000. Nitrogen nutrients cycling in marine environment and its modeling research. Advances in Earth Science 15, 58–64.

Robson, B.J., 2014. State of the art in modelling of phosphorus in aquatic systems: review, criticisms and commentary. Environmental Modelling & Software 61, 339–359.

Saberi-Movahed, F., Najafzadeh, M., Mehrpooya, A., 2020. Receiving more accurate predictions for longitudinal dispersion coefficients in water pipelines: training group method of data handling using extreme learning machine conceptions. Water Resources Management 34, 529–561.

Sarangi, R.K., 2012. Remote-sensing-based estimation of surface nitrate and its variability in the southern peninsular Indian waters. International Journal of Oceanography 2011.

Sharifi, A., 2020. Using Sentinel-2 data to predict nitrogen uptake in maize crop. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 13, 2656–2662.

Skerratt, J.H., Mongin, M., Baird, M.E., Wild-Allen, K.A., Robson, B.J., Schaffelke, B., Davies, C.H., Richardson, A.J., Margvelashvili, N., Soja-Wozniak, M., Steven, A.D.L., 2019. Simulated nutrient and plankton dynamics in the Great Barrier Reef (2011–2016). Journal of Marine Systems 192, 51–74.

Smith, V.H., Tilman, G.D., Nekola, J.C., 1999. Eutrophication: impacts of excess nutrient inputs on freshwater, marine, and terrestrial ecosystems. Environmental pollution 100, 179–196.

Steven, A.D.L., Baird, M.E., Brinkman, R., Car, N.J., Cox, S.J., Herzfeld, M., Hodge, J., Jones, E., King, E., Margvelashvili, N., 2019. eReefs: an operational information system for managing the Great Barrier Reef. Journal of Operational Oceanography 12, 12–28.

Vaz, R., Shah, V., Sawhney, A., Deolekar, R., 2017. Automated big-O analysis of algorithms, in: Proc. International Conference on Nascent Technologies in Engineering (ICNTE), Mumbai, India. pp. 1–6.

Waltham, N.J., Wegscheidl, C., Volders, A., Smart, J.C.R., Hasan, S., Lédée, E., Waterhouse, J., 2021. Land use conversion to improve water quality in high DIN risk, low-lying sugarcane areas of the Great Barrier Reef catchments. Marine Pollution Bulletin 167, 112373.

Wang, D., Cui, Q., Gong, F., Wang, L., He, X., Bai, Y., 2018. Satellite retrieval of surface water nutrients in the coastal regions of the east China Sea. Remote Sensing 10.

Wang, Y., Zhang, J., Zhu, H., Long, M., Wang, J., Yu, P.S., 2019. Memory in memory: A predictive neural network for learning higher-order non-stationarity from spatiotemporal dynamics, in: Proc. Computer Vision and Pattern Recognition, Long Beach, CA, USA. pp. 9154–9162.

Waterhouse, J., Brodie, J., Tracey, D., Smith, R., VanderGragt, M., Collier, C., Petus, C., Baird, M., Kroon, F., Mann, R., 2020. Land use impacts on the Great Barrier Reef water quality and ecosystem condition. The Scientific Consensus Statement, State of Queensland .

Wen, J., Yang, J., Jiang, B., Song, H., Wang, H., 2021. Big data driven marine environment information forecasting: a time series prediction network. IEEE Transactions on Fuzzy Systems 29, 4–18.

Zhou, Y., Dong, H., Saddik, A.E., 2020. Deep learning in next-frame prediction: a benchmark review. IEEE Access 8, 69273–69283.

Zhu, Q., Liu, Z., Yan, J., 2021. Machine learning for metal additive manufacturing: Predicting temperature and melt pool fluid dynamics using physics-informed neural networks. Computational Mechanics 67, 619–635.