

# Detecting Tropical Cyclogenesis Using Broad Learning System from Satellite Passive Microwave Observations



Sheng Wang and Xiaofeng Yang

## 1 Introduction

Tropical cyclone (TC), as one of the most violent phenomena of air-sea interaction, often brings disastrous storm surges and flooding and causes significant damage to human life, agriculture, forestry, fisheries, and infrastructure. Therefore, the knowledge of TC track, intensity, structure, and evolution is required to guide severe weather forecasting and risk assessment. Generally, the formation of TCs needs the support of dynamic environmental conditions and thermodynamically favorable environmental conditions [19]. Because only a small percentage of convective disturbances are developing into TCs, it is still challenging to predict the TC formation accurately.

Since the Dvorak Technique (DT) was proposed and developed [8, 21, 28], it has been widely used in TCs intensity estimation [14, 22, 25, 29] and TC formation prediction [6, 20, 32, 33]. However, DT is based on the infrared technique, whose observations may be obscured by significant convection or cirrus clouds. In contrast, microwave radiation images can capture the strong convective areas and cloud organization. Therefore, it is potential to predict the formation of TCs with microwave remote sensing data.

With the advancements in high-performance computing, machine learning methods based on big datasets are widely used in tropical cyclogenesis detection of TC

---

S. Wang · X. Yang

State Key Laboratory of Remote Sensing Science, Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100101, China

S. Wang

State Key Laboratory on Internet of Things for Smart City and Department of Civil and Environmental Engineering, University of Macau, Macau 999078, China

X. Yang (✉)

Key Laboratory of Earth Observation of Hainan Province, Sanya 572029, China

e-mail: [yangxf@radi.ac.cn](mailto:yangxf@radi.ac.cn)

© The Author(s) 2023

X. Li and F. Wang (eds.), *Artificial Intelligence Oceanography*,  
[https://doi.org/10.1007/978-981-19-6375-9\\_7](https://doi.org/10.1007/978-981-19-6375-9_7)

formation. Based on the decision tree method, a series of classification rules are constructed to predict future tropical cyclone (TC) genesis events, and the overall prediction accuracy is 81.72% [35]. Using the dataset established with WindSat wind products, [23] established a classification model for tropical cyclogenesis detection. The validation shows that the model produced a positive detection rate of approximately 95.3% and a false alarm rate of 28.5%. This study confirmed the potential of microwave remote sensing observation in detecting typhoon formation [23]. Recently, based on the internal structure information of tropical cyclones obtained by satellite remote sensing, [27] effectively improved the prediction accuracy of the rapid enhancement process of tropical cyclones and reduced the false alarm rate using the machine learning method. Moreover, [13] compared different machine learning algorithms' TC formation detection performance. Their results prove that the machine learning method performs better than the traditional linear discriminant analysis.

However, with the continuous accumulation of remote sensing data, traditional machine learning methods cannot deal with massive data perfectly. Fortunately, powerful deep learning has demonstrated its more significant superiority over traditional physical or statistical-based algorithms for image information extraction [18]. In ocean remote sensing applications, the deep learning methods are used in hurricane intensity estimation [7, 24], sea ice concentration prediction [5, 9, 11], sea surface temperature estimation [1, 30] and other fields [10, 26, 37]. A deep learning approach has been proposed to identify tropical cyclones (TCs) and their precursors based on twenty-year simulated outgoing longwave radiation (OLR) calculated with a cloud-resolving global atmospheric simulation [19]. In the Northwest Pacific in the period from July to November, the probability of detection (POD) of the model is 79.9–89.1%, and the false alarm ratio (FAR) is 32.8–53.4%. In addition, this study reveals that the detection performance is correlated with the amount of training data and TC lifetimes.

Although deep learning is increasingly widely used in ocean remote sensing [18], the disadvantages of deep learning are also evident. It requires high computing power and a long training time. Moreover, most deep learning models do not have incremental learning capacity, which means the model needs to be retrained if updating the dataset. In ocean remote sensing, the satellite-based data increases every day, so the size of datasets is expected to expand further to improve the generalization ability and identification accuracy of models. Therefore, the defect of no incremental learning is not friendly to storage resources or model update times. Fortunately, the capacity of incremental learning of the Broad Learning System (BLS) [3] makes it have the potential to be applied in the field of ocean remote sensing. Meanwhile, the BLS is a time-cost-friendly learning strategy due to its flatted network. These advantages can compensate for the disadvantage of its accuracy compared with deep learning, so it has been widely used as soon as it is proposed. Recently, it has successfully been applied in seismic attenuation modeling [16], model updating [17], hyperspectral imagery classification [34], and crack detection [36].

In this chapter, we proposed a tropical cyclogenesis detection algorithm based on Special Sensor Microwave Imager (SSM/I) brightness temperature data. The proposed model based on BLS has three unique features: low hardware requirements,

fast computation speed, and incremental learning ability. In Sect. 2, the dataset used in this study is presented. In Sect. 3, the details of BLS are introduced. The experimental results are shown in Sect. 4, and the conclusion is given in Sect. 5.

## 2 Data Description

The dataset used in this chapter is extracted from the brightness temperature (TB) observations acquired by SSM/I. This series of instruments is carried onboard Defense Meteorological Satellite Program (DMSP) near-polar orbiting satellites. The SSM/I is a conically scanning sensor that measures the natural microwave emission from the Earth in the spectral band from 19 GHz to 85 GHz with different polarization (See Table 1). The parameters derived from these radiometer observations include surface wind speed, atmospheric water vapor, cloud liquid water, and rain rate [31]. Comparing the feature of TB images in different channels/polarizations, the 37 GHz H-polarization (37H) channel is selected due to its clear description of the features of disturbances and tropical cyclones.

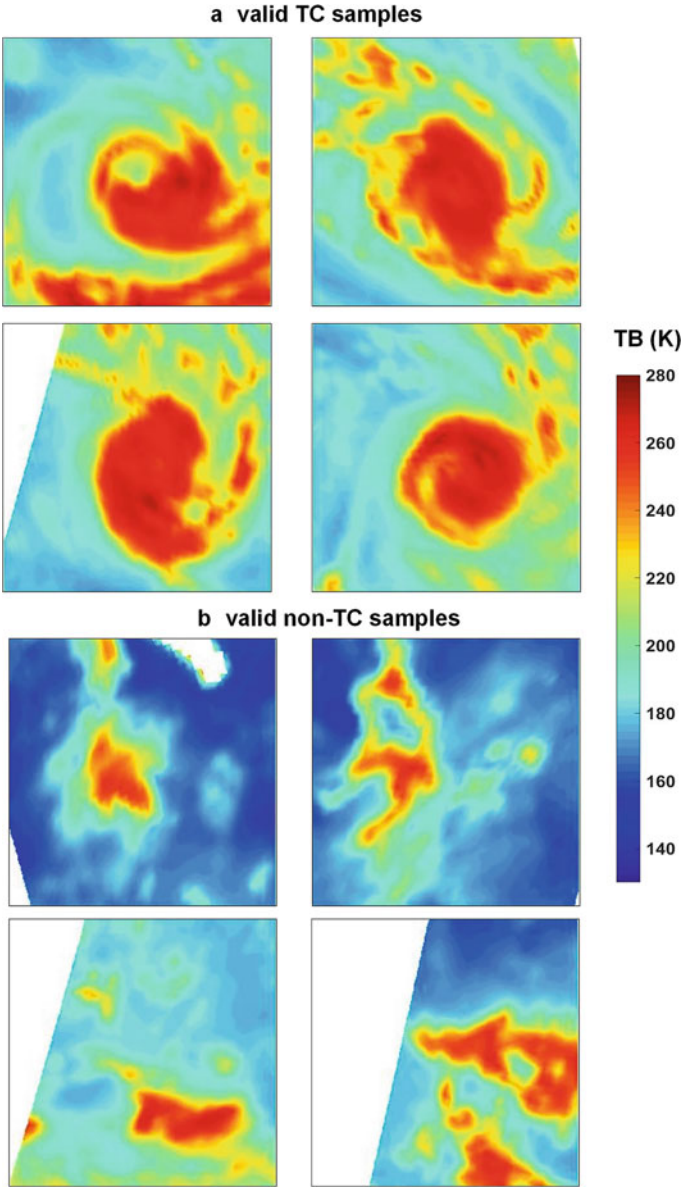
To collect the sample images covered TCs or non-developed disturbances (non-TC), the TC best tracks and tropical cloud cluster (TCC) tracks during 2005–2009 are used as auxiliary data. This information can be obtained from the International Best Track Archive for Climate Stewardship dataset (IBTrACS) [15] and Global Tropical Cloud Cluster dataset [12], respectively. The time resolution of these two datasets is three hours. Note that not all the best track records in the TC evolution period are used, but those during the TC formation period are selected. Specifically, the time when the TC maximum wind speed reaches 25 knots for the first time is defined as the starting time. Then, the 72 h after this time is defined as the TC formation period [23]. For the TCC tracks, only the records that have not developed into TCs are selected. The preprocessing steps for extracting the TC and non-TC images are described as follow:

- (1) For each TC/non-TC track record, determine the matching SSM/I TB data within the absolute time difference of 1.5 h.
- (2) Take the track record as the image center position, and extract the sub-images with the size of  $8^{\circ} \times 8^{\circ}$  from the SSM/I TB observations.
- (3) The sub-images with more than 60% non-empty pixels are retained as qualified samples (see Fig. 1); otherwise, the invalid data will be excluded (see Fig. 2).

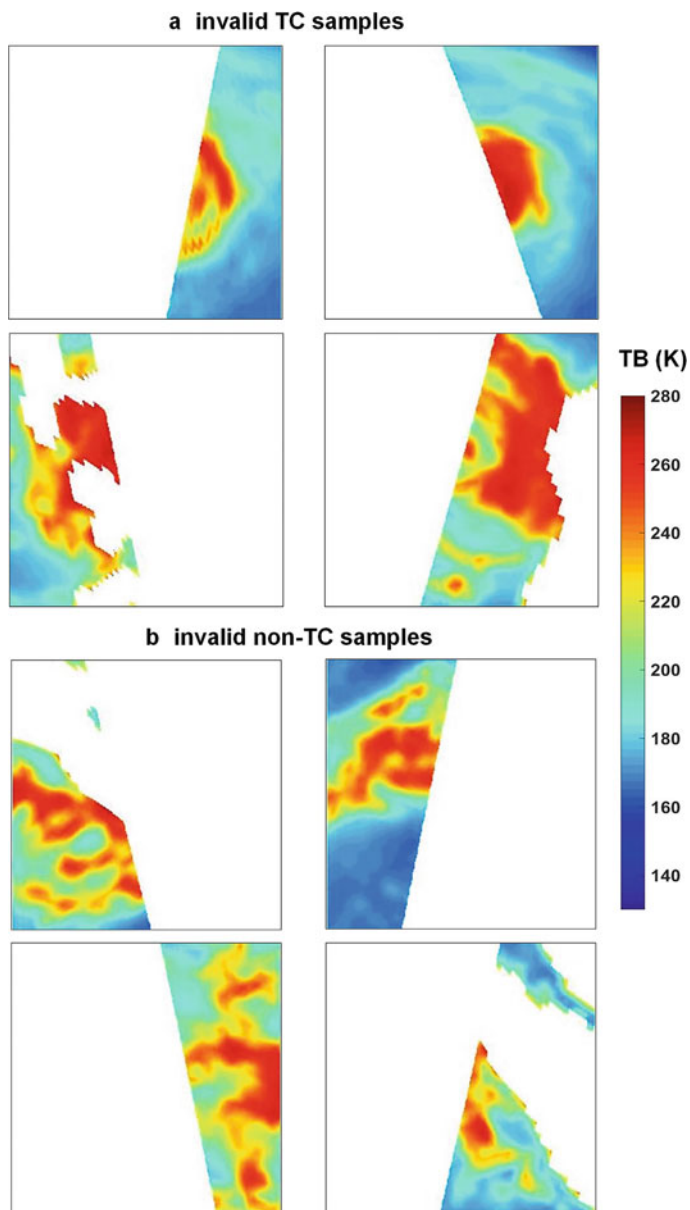
**Table 1** Channel characteristics of SSM/I

Band/GHz	Polarization	Spatial resolution/km × km
9.35	V/H	69 × 43
23.235	V	50 × 40
37.0	V/H	37 × 28
85.5	V/H	15 × 13

\* H-horizontal polarization, V-vertical polarization



**Fig. 1** Qualified samples: **a** valid TC samples and **b** valid non-TC samples



**Fig. 2** Unqualified samples: **a** invalid TC samples and **b** invalid non-TC samples

Following the above steps, 880 TC samples and 6268 non-TC samples were obtained from the SSM/I observations in 2005–2009. Due to the significant number difference between the two samples, only 2506 non-TC samples in 2005–2006 and 880 TC samples in 2005–2009 were selected to form the final dataset. Each sample is in the size of  $224 \times 224$  pixels with RGB channels. Finally, These two datasets are randomly divided into the training set and the testing set in the ratio of 4:1, respectively.

### 3 Broad Learning System for Tropical Cyclogenesis Detection

Once the dataset is established, the tropical cyclogenesis detection can be executed with the broad learning system. In contrast to deep learning methods, the BLS provides a time-cost-friendly learning strategy due to its flatted network. The main structure of BLS consists of the input layer, node layer, and output layer. Specifically, the node layer includes the feature nodes and enhancement nodes. Generally, the input data is mapped to feature nodes with random weights. Then, the feature nodes are further mapped to enhancements with new random weights. Finally, the final weights of BLS can be trained by estimating the output data with these feature nodes and enhancement nodes. Figure 3 shows the architecture of this study, the definition of variables are:  $X$  is the input data,  $F$  is the feature node, and  $E$  is the enhancement node.  $Y$  is the respective classification labels of the input data  $X$ . The details of BLS are presented as follows.

#### 3.1 Broad Learning Model

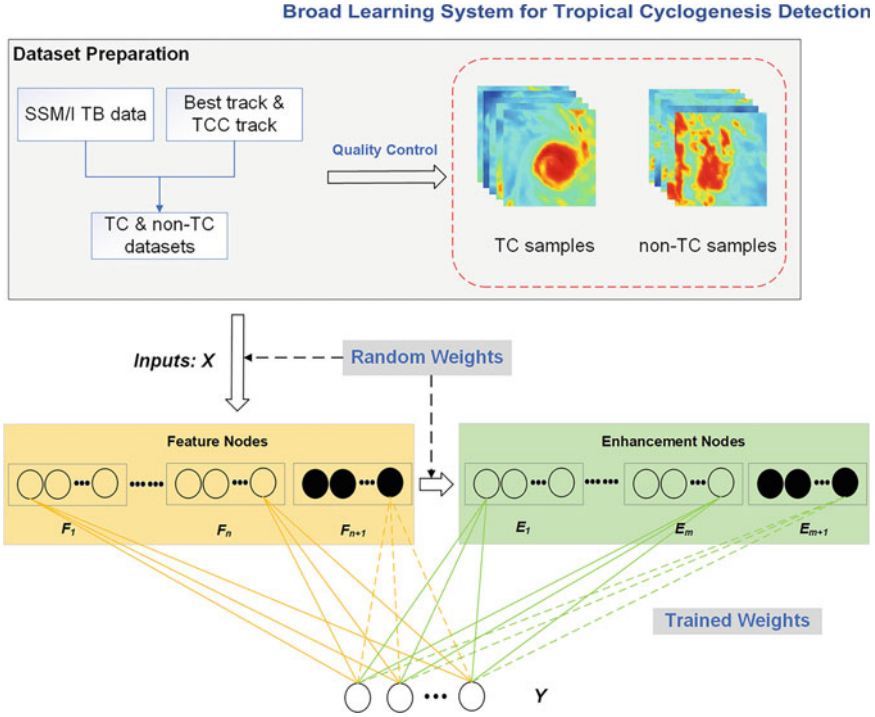
Assume that the input data is  $X$ , so the feature vector  $F$  mapped with random weight can be described as

$$F_i = \phi(XW_{e_i} + b_{e_i}), i = 1, \dots, n \quad (1)$$

where  $F_i$  is the  $i$ -th feature node,  $W_e$  and  $b_e$  are the random weights and biases with the proper dimensions, respectively. Denote  $F^n \equiv [F_1, \dots, F_n]$ , which is the concatenation of all the first  $n$  groups of mapped features. Then, the enhancement nodes can be given by:

$$E_m = \xi(F^n W_{h_m} + b_{h_m}) \quad (2)$$

where  $E_m$  is the  $m$ -th enhancement node,  $W_h$  and  $b_h$  are the random weights and biases with the proper dimensions, respectively. Similarly, the concatenation of all the first  $m$  groups of enhancement nodes are denoted as  $E^m \equiv [E_1, \dots, E_m]$ . Therefore, the broad model can be represented as the equation of the form



**Fig. 3** The architecture of BLS

$$\begin{aligned}
 Y &= [F_1, \dots, F_n \mid \xi(F^n W_{h_1} + b_{h_1}), \dots, \xi(F^n W_{h_m} + b_{h_m})] W^m \\
 &= [F_1, \dots, F_n \mid E_1, \dots, E_m] W^m \\
 &= [F^n \mid E^m] W^m
 \end{aligned} \tag{3}$$

where  $W^m = [F^n \mid E^m]^+ Y$  are the connecting weights for the broad structure to be computed and  $[F^n \mid E^m]^+$  is the pseudo-inverse of  $[F^n \mid E^m]$ . In a flatted network, pseudo-inverse can be considered a very convenient approach to solving the output-layer weights of a neural network. However, a straightforward solution is too expensive, especially when the training samples and input patterns suffer from high volume, high velocity, and/or high variety [4]. Under this situation of the expensive cost for directly computing the pseudo-inverse, the solution can be approximated by ridge regression:

$$A^+ = [F^n \mid E^m]^+ = (\lambda I + [F^n \mid E^m][F^n \mid E^m]^T)^{-1} [F^n \mid E^m]^T \tag{4}$$

where  $\lambda$  is the regularization parameter. Finally, the model weights are given by

$$W = A^+ Y \tag{5}$$

During the BLS computation, it should be noted that the number of enhancement nodes is a hyperparameter ( $N_3$ ), and the number of feature nodes is the combination of two hyperparameters: the number of feature windows ( $N_1$ ) and the number of nodes in each feature window ( $N_2$ ). Here, the Bayesian optimization method is used to find the optimal model hyperparameters, and it can be easily executed with the Hyperopt package [2].

### 3.2 Incremental Learning of BLS

When a deep learning model works not well, the number of convolutional kernels or the number of convolutional layers will increase. This will lead to expensive computation and long time costing. However, BLS usually uses incremental learning to address the low model accuracy caused by insufficient mapping nodes. Generally, the incremental learning part can improve the model performance. There are two ways to expand the broad structure: (1) increment of enhancement nodes and feature nodes, and (2) adding input data.

#### 3.2.1 Increment of the Feature Nodes and Enhancement Nodes

Assume that the initial BLS has  $n$  feature nodes and  $m$  enhancement nodes that is  $A = [F^n \mid E^m]$ . In the adding process, the  $(n+1)$ -th feature node is given by:

$$F_{n+1} = \phi(XW_{e_{n+1}} + b_{e_{n+1}}) \quad (6)$$

So that the corresponding enhancement node to this feature node is given by:

$$E_{ex_m} = [\xi(F_{n+1}W_{ex_1} + b_{ex_1}), \dots, \xi(F_{n+1}W_{ex_m} + b_{ex_m})] \quad (7)$$

Then, there are additional  $p$  enhancement nodes added to the BLS structure, and the  $(m+1)$ -th enhancement node is given by

$$E_{m+1} = \xi(F^n W_{h_{m+1}} + b_{h_{m+1}}) \quad (8)$$

Therefore, the final node layer matrix is combined as

$$A' = [A \mid F_{n+1} \mid E_{ex_m} \mid E_{m+1}] \quad (9)$$

Then, the pseudo-inverse of  $A'$  is computed with

$$(A')^+ = \begin{bmatrix} A^+ - DB^T \\ B^T \end{bmatrix} \quad (10)$$



where  $D = (A)^+ [F_{n+1} \mid E_{ex_m} \mid E_{m+1}]$ ,

$$B^T = \begin{cases} C^+ & C \neq 0 \\ (1 + D^T D)^{-1} B^T A^+ & C = 0 \end{cases} \quad (11)$$

and  $C = [F_{n+1} \mid E_{ex_m} \mid E_{m+1}] - AD$ . Finally, the new weights are

$$W' = \begin{bmatrix} W - DB^T Y \\ B^T Y \end{bmatrix} \quad (12)$$

As seen in Eq. 12, the updated weights consist of the initial and the new parts. There is no need to re-calculate the pseudo-inverse for the whole nodes but only compute the added nodes.

### 3.2.2 Increment of the Input Data

The increment of feature nodes and enhancement nodes mentioned above are for the fixed dataset. However, in most learning models, the input dataset is the core factor influencing the prediction accuracy. As for deep learning, once some new data is added to the training dataset, the existing model needs to be retrained. This is time-consuming and reduces the timeliness of model updating and application. Fortunately, there is no need to retrain the whole model for BLS after adding input data. The BLS will train only the added ones.

Denote  $X_a$  as the new inputs, the respective increment of mapped feature nodes and enhancement nodes are:

$$F_x^n = [\phi(X_a W_{e_1} + b_{e_1}), \dots, \phi(X_a W_{e_n} + b_{e_n})] \quad (13)$$

$$E_m^x = [\xi(F_x^n W_{h_1} + b_{h_1}), \dots, \xi(F_x^n W_{h_m} + b_{h_m})] \quad (14)$$

where the  $W_{e_i}$ ,  $W_{h_i}$  and  $b_{e_i}$ ,  $b_{h_i}$  are randomly generated during the initial BLS. Hence, the updating matrix is

$$A' = \begin{bmatrix} A \\ A_x^T \end{bmatrix} \quad (15)$$

where  $A_x^T = [F_x^n \mid E_m^x]$ . The associated updating pseudo-inverse could be deduced as follows:

$$W_n^x = W + (Y_a - A_x^T W)B \quad (16)$$

where  $Y_a$  are the respective labels of additional  $X_a$ . Similar to the two increment processes mentioned above, only the pseudo-inverse associated with new inputs is calculated. It greatly improves the update speed of the BLS model.

4 Results

Before training the model, one should determine the hyperparameters first. For the basic BLS, the hyperparameters include the number of feature windows ( $N_1$ ), the number of nodes in each feature window ( $N_2$ ), and the number of enhancement nodes ( $N_3$ ). To compare the training time and model accuracy, the classical ResNet50 model is also applied to the same dataset. For this ResNet50 model, the initial learning rate is  $10^{-3}$ , the multiplicative factor of learning rate decay is 0.5, the batch size is 16, and the number of epochs is 20. Before training the ResNet50 network, all input images were resized to  $224 \times 224$ . To fairly compare the training time of these two networks, the training tasks were operated on a computer with an Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz and 64GB RAM.

4.1 Basic BLS Results

Using the Hyperopt, the hyperparameters are optimized as:  $N_1 = 5, N_2 = 24, N_3 = 2332$ . The training and testing accuracies and training time of these two methods are shown in Table 2. The testing accuracy of BLS is 86.83%, which is slightly lower than the 91.88% of ResNet50. On the other hand, though the ResNet50 is operated with an accelerating GPU, the training time of 2090.45 s is still 20 times than 60.52 s of BLS. Therefore, BLS has obvious advantages in computational efficiency, but it is not as accurate as the deep learning network because it is insensitive to the image features. Furthermore, we compared the hit rate (HR) and false alarm rate (FAR) of these two models. Table 3 lists the HR and FAR for the training and testing processes. The testing HR and FAR are 81.14 and 11.18%, respectively. Compared to the 79.9–89.1% of HR and 32.8–53.4% of FAR of existing deep learning research [19], our results are competitive. But it should be noted that the size of the dataset used in [19] is 50000 TCs and 500000 non-TCs, which is significantly larger than our dataset.

Table 2 The results for BLS and ResNet50

Model	Training accuracy (%)	Testing accuracy (%)	Training time
BLS	99.96	86.83	60.52 (CPU)
ResNet50	98.30	91.88	2090.45 (GPU)

**Table 3** HR and FAR of BLS and results of [19]

Prediction accuracy (%)	BLS (%)	Matsuoka et al. [19] (%)
HR	81.14	79.9–89.1
FAR	11.18	32.8–53.4

**Table 4** Prediction accuracy with different hyperparameters

Feature nodes (N <sub>1</sub> , N <sub>2</sub> )	Enhancement nodes	Training time (s)	Testing accuracy (%)	HR (%)	FAR(%)
5, 24	2332	58.68	86.83	81.14	11.18
5, 24	1000	57.59	86.24	72.57	8.98
5, 24	1500	59.78	85.35	75.43	11.18
5, 24	2000	58.28	84.62	74.29	11.78
5, 24	2500	58.82	85.50	74.88	10.78
5, 24	3000	59.68	85.50	74.86	10.78
6, 24	3000	68.01	84.47	77.71	13.17
7, 24	3000	73.34	86.24	78.28	10.98
8, 24	3000	77.02	85.50	78.86	12.18

In contrast to the massive number of hyperparameters, the BLS only has primary hyperparameters to influence the prediction accuracy. To know the prediction performance of BLS influenced by different combinations of these three hyperparameters, we trained the same dataset several times, and the results are listed in Table 4. The combination of the number of feature nodes and enhancement nodes is essential for the accuracy of the model. It shows that the more nodes, the longer the training time, but the changing trend of model testing accuracy, HR, and FAR is inconsistent. However, the optimized hyperparameter combination has the highest HR because the optimization process takes the HR as the selection standard. Therefore, It is indispensable to determine these parameters using the optimization algorithm.

**4.2 Incremental Learning Results**

The capacity of incremental learning is the prominent superior feature of BLS to most traditional deep learning models. For most deep networks, the structures are fixed once the training process is finished. In contrast, the BLS can be updated by adding new nodes or updating the dataset. There is no need to retrain the whole network, which significantly saves the costing time of model updating. The dataset size is 3386, and we set the initial size as 1386 with the adding input patterns of 400. First, we trained the initial model with the initial samples. Then, the incremental learning method was operated to add corresponding input patterns each time until

**Table 5** Prediction accuracy and CPU time of incremental learning

Number of inputs	Training accuracy (%)	Testing accuracy (%)	Training time (s)
707	100.00	26.33	31.53
708–1107	27.46	61.83	28.93
1108–1507	53.35	74.11	28.79
1508–1907	63.14	74.11	28.43
1908–2307	69.53	74.11	28.56
2308–2707	74.03	74.11	29.09

all training samples were input. During these incremental learning steps, the value of hyperparameters is still as  $N_1 = 5$ ,  $N_2 = 24$ ,  $N_3 = 2332$ . The results are shown in Table 5 and note that the testing dataset is unchanged during the incremental learning. The results in the table show that with the size of inputs increasing, the training accuracy grows. For the initial process, the small size of the dataset leads to unreasonable accuracies. Because the test dataset for each incremental learning is the same, the testing accuracy tends to be stable.

**4.3 Case Study: Hurricane Wilma (2005)**

Once the model training/testing processes are finished, the model hyperparameters ( $N_1$ ,  $N_2$ , and  $N_3$ ) and the weights of nodes (feature nodes and enhancement nodes) are fixed. Based on the trained BLS, the prediction task of specific TC cases can be executed. Here, we select Hurricane Wilma (2005) as the study case to validate the effectiveness of the proposed model. Wilma was an extremely intense hurricane over the northwestern Caribbean Sea. It had the all-time lowest central pressure for an Atlantic basin hurricane. According to the statistics, twenty-three deaths have been directly attributed to Wilma, and the total economic losses reached 16 billion to 20 billion dollars. As the best tracks in Table 6 shown, Wilma developed into a tropical storm (TS) from tropical depression (TD) at 06:00 UTC 17 October, and then strengthened into a hurricane (HU) at 12:00 UTC 18 October. We planned to collect the samples from 18:00 UTC 15 October to 15:00 UTC 18 October during the data preparation. However, there were only four qualified samples were retained (see Fig. 4). The corresponding best track records of Fig. 4a–d are 00:00 UTC 17 October, 12:00 UTC 17 October, 00:00 UTC 18 October, and 12:00 UTC 18 October, respectively.

Table 7 lists the prediction results of these four samples. It shows that Fig. 4b is incorrectly classified into a non-TC label, and the remaining three samples are correctly identified. Figure 4a is the first sample captured by SSM/I during Wilma’s formation time, and its correct classification means the proposed model can detect tropical cyclogenesis as early as possible. Figure 4c, d are the observations during

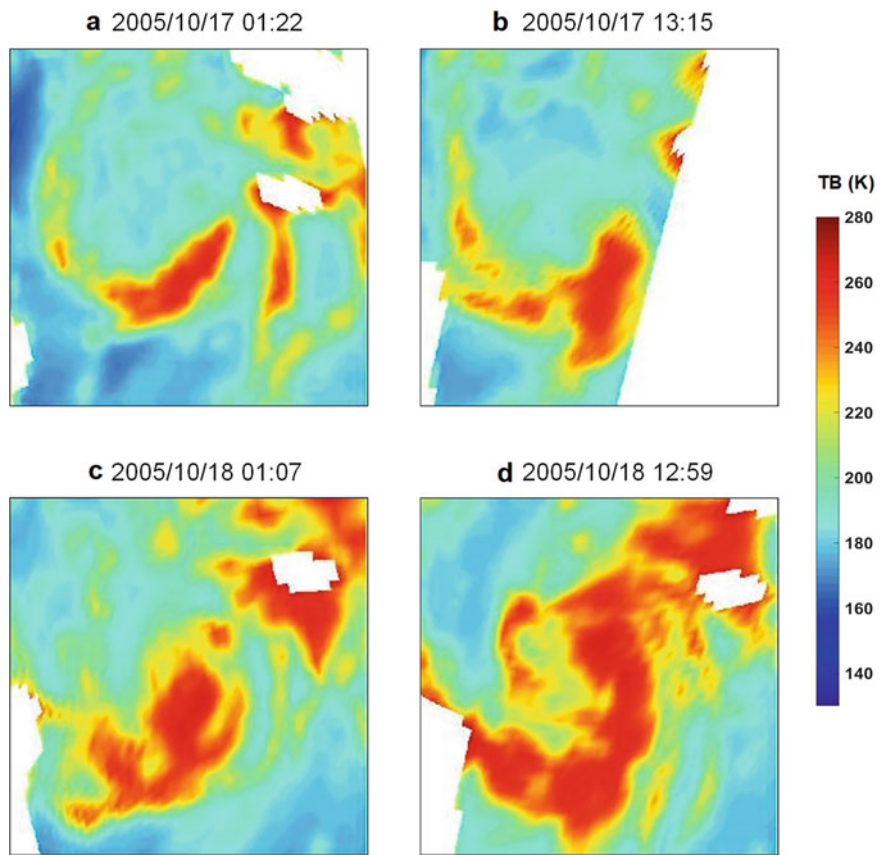
**Table 6** The best track information during the formation time of Wilma (2005)

Time (UTC)	Status	Longitude	Latitude	Wind speed
10-15 18:00	TD	78.50° W	17.60° N	25
10-15 21:00	TD	78.66° W	17.61° N	25
10-16 00:00	TD	78.80° W	17.60° N	25
10-16 03:00	TD	78.91° W	17.55° N	27
10-16 06:00	TD	79.00° W	17.50° N	30
10-16 09:00	TD	79.10° W	17.49° N	30
10-16 12:00	TD	79.20° W	17.50° N	30
10-16 15:00	TD	79.30° W	17.51° N	30
10-16 18:00	TD	79.40° W	17.50° N	30
10-16 21:00	TD	79.52° W	17.49° N	30
10-17 00:00	TD	17.60° W	17.40° N	30
10-17 03:00	TD	79.61° W	17.18° N	32
10-17 06:00	TS	79.60° W	16.90° N	35
10-17 09:00	TS	79.64° W	16.59° N	37
10-17 12:00	TS	79.70° W	16.30° N	40
10-17 15:00	TS	79.75° W	16.12° N	42
10-17 18:00	TS	79.80° W	16.00° N	45
10-17 21:00	TS	79.86° W	15.89° N	50
10-18 00:00	TS	79.90° W	15.80° N	55
10-18 03:00	TS	79.88° W	15.70° N	57
10-18 06:00	TS	79.90° W	15.70° N	60
10-18 09:00	TS	80.04° W	15.91° N	62
10-18 12:00	HU	80.30° W	16.20° N	65
10-18 15:00	HU	80.68° W	16.44° N	70

\*TD-Tropical Depression, TS-Tropical Storm, HU-Hurricane

Wilma's mature period, and its TC structure is relatively stable and complete. So it is predictable to obtain the correct results. However, the negative result of Fig. 4b proves that the quality of samples brings uncertainty and error to the model prediction. Specifically, compared with the other three samples, Fig. 4b loses nearly half of the TC system information, which reduces the number of effective pixels and damages spiral or TB distribution characteristics of TCs. This could be the main reason for the poor prediction.

However, not all samples with missing information will be incorrectly identified, but the core TC system information loss will lead to misclassification. To verify this conclusion, we select the tropical storm Hilda (2009) as the test case. Figure 5 shows Hilda's four qualified TC samples, and Table 8 lists the corresponding prediction results. It shows that all the samples are correctly classified though parts of information lost in Fig. 5. In particular, the size of the missing part in Fig. 5a, b is similar to



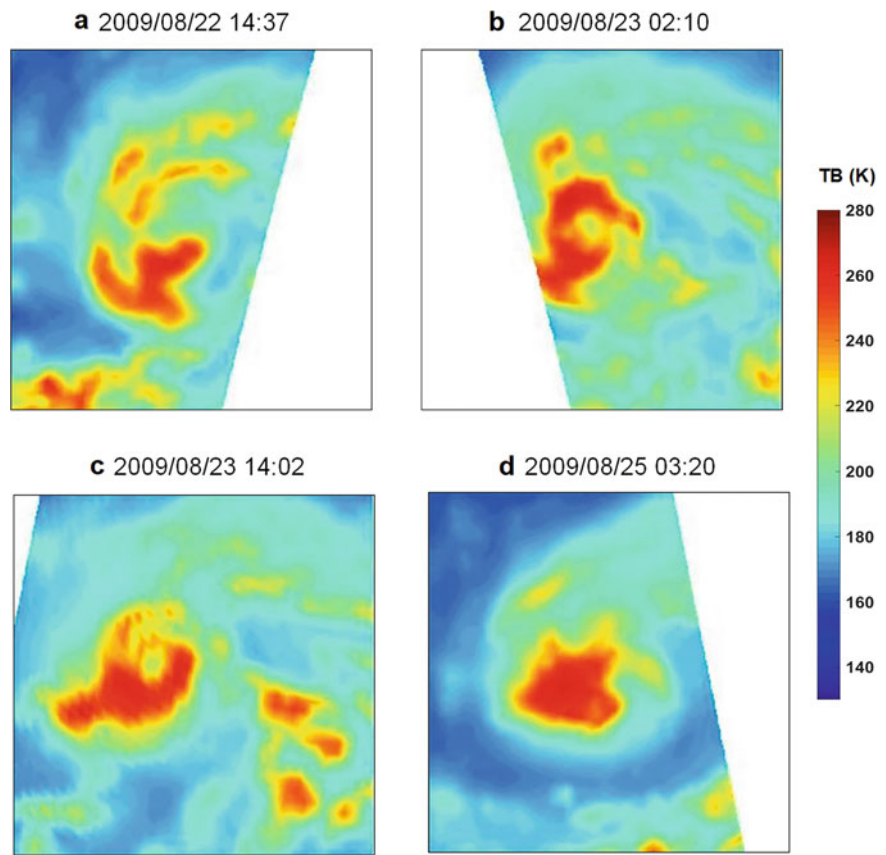
**Fig. 4** The bright temperature images during the formation of Wilma (2005)

**Table 7** Prediction results for the four samples of Wilma (2005)

Time of samples (UTC)	Label	Prediction label	Result	Operating time (s)
0–17 01:22	1	1	True	<0.01
10–17 13:15	1	0	False	<0.01
10–18 01:07	1	1	True	<0.01
10–18 12:59	1	1	True	<0.01

\* Label 1-TC, Label 0-non-TCC

that in Fig. 4b, but there is less TC system information in the missing part. The TC structure and TB distribution pattern are not contaminated significantly. Therefore, these two samples can still be correctly identified. All in all, the proposed model can well detect tropical cyclogenesis with high-quality data.



**Fig. 5** The bright temperature images during the formation of Hilda (2009)

**Table 8** Prediction results for the four samples of Hilda (2009)

Time of samples (UTC)	Label	Prediction label	Result	Operating time (s)
08-22 14:37	1	1	True	<0.01
08-22 02:10	1	1	True	<0.01
08-23 14:02	1	1	True	<0.01
08-25 03:20	1	1	True	<0.01

\* Label 1-TC, Label 0-non-TC

## 5 Conclusion

In this study, a tropical cyclogenesis detection model is proposed using BLS. In contrast to the deep network methods, the new model is a lightweight flatted network, leading to lower computation and shorter training time. Meanwhile, the capacity of incremental learning of BLS is consistent with the continuously updated and accumulated remote sensing data. Adding new input data does not need to retrain the whole updated dataset. Based on the dataset consisting of 3386 TB images, the testing accuracy, HT, and FAR of BLS are 86.83%, 81.14%, and 11.18% respectively. This study confirms the applicability of BLS in the binary classification problem in ocean remote sensing. It also proves the possibility of detection of TC formation from satellite microwave TB data.

Although the BLS has shown great power in the classification problem, two defects need to be addressed: (1) the BLS is insensitive to the features of images, which will lead to poor accuracy when the image features are complicated. Inspired by the powerful ability of convolution neural network (CNN) to capture and learn image features, we will add a feature extracting mode before our BLS to improve its image processing ability. (2) The size of the dataset is too small to support the learning requirements perfectly. There are three ways to expand the dataset: One is to add the TB data from other channels (e.g., 19H/V), the other is to obtain samples in a longer period, and the last is to utilize the TB observations from other microwave radiometers (e.g., Microwave Imager onboard FengYun series satellites).

## References

1. Aparna SG, D'Souza S, Arjun NB (2018) Prediction of daily sea surface temperature using artificial neural networks. *Int J Remote Sens* 39(11–12):4214–4231
2. Bergstra J, Yamins D, Cox D (2013) Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In: *International Conference on Machine Learning*, PMLR, pp 115–123
3. Chen CLP, Liu Z (2018) Broad learning system: An effective and efficient incremental learning system without the need for deep architecture. *IEEE Trans Neural Netw Learn Syst* 29(1):10–24. <https://doi.org/10.1109/TNNLS.2017.2716952>
4. Chen CLP, Zhang CY (2014) Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. *Inf Sci* 275:314–347. <https://doi.org/10.1016/j.ins.2014.01.015>
5. Chi J, Kim Hc (2017) Prediction of arctic sea ice concentration using a fully data driven deep neural network. *Remote Sens* 9(12). <https://doi.org/10.3390/rs9121305>
6. Cossuth JH, Knabb RD, Brown DP, Hart RE (2013) Tropical cyclone formation guidance using Pregenesis Dvorak Climatology. part i: Operational forecasting and predictive potential. *Weather Forecast* 28(1):100–118. <https://doi.org/10.1175/WAF-D-12-00073.1>
7. Dawood M, Asif A, Minhas FuAA (2020) Deep-PHURIE: deep learning based hurricane intensity estimation from infrared satellite imagery. *Neural Comput Appl* 32(13):9009–9017. <https://doi.org/10.1007/s00521-019-04410-7>



8. Dvorak VF (1975) Tropical cyclone intensity analysis and forecasting from satellite imagery. *Mon Weather Rev* 103(5):420–430
9. Gao Y, Gao F, Dong J, Wang S (2019) Transferred deep learning for sea ice change detection from synthetic aperture radar images. *IEEE Geosci Remote Sens Lett* 16(10):1655–1659. <https://doi.org/10.1109/LGRS.2019.2906279>
10. Ham YG, Kim JH, Luo JJ (2019) Deep learning for multi-year ENSO forecasts. *Nature* 573(7775):568+. <https://doi.org/10.1038/s41586-019-1559-7>
11. Han Y, Gao Y, Zhang Y, Wang J, Yang S (2019) Hyperspectral sea ice image classification based on the spectral-spatial-joint feature with deep learning. *Remote Sens* 11(18). <https://doi.org/10.3390/rs11182170>
12. Hennon CC, Helms CN, Knapp KR, Bowen AR (2011) An objective algorithm for detecting and tracking tropical cloud clusters: Implications for tropical cyclogenesis prediction. *J Atmos Oceanic Tech* 28(8):1007–1018. <https://doi.org/10.1175/2010JTECHA1522.1>
13. Kim M, Park MS, Im J, Park S, Lee MI (2019) Machine learning approaches for detecting tropical cyclone formation using satellite data. *Remote Sens* 11(10). <https://doi.org/10.3390/rs11101195>
14. Knaff JA, Brown DP, Courtney J, Gallina GM, Beven JL II (2010) An evaluation of Dvorak technique-based tropical cyclone intensity estimates. *Weather Forecast* 25(5):1362–1379. <https://doi.org/10.1175/2010WAF2222375.1>
15. Knapp KR, Kruk MC, Levinson DH, Diamond HJ, Neumann CJ (2010) The international best track archive for climate stewardship (IBTrACS) unifying tropical cyclone data. *Bul Am Meteorol Soc* 91(3):363+. <https://doi.org/10.1175/2009BAMS2755.1>
16. Kuok SC, Yuen KV (2020) Broad learning for nonparametric spatial modeling with application to seismic attenuation. *Comput-Aided Civil Infrastruct Eng* 35(3):203–218
17. Kuok SC, Yuen KV (2020) Multi-resolution broad learning for model updating using incomplete modal data. *Struct Control Health Monit* 27(8):e2571
18. Li X, Liu B, Zheng G, Ren Y, Zhang S, Liu Y, Gao L, Liu Y, Zhang B, Wang F (2020) Deep-learning-based information mining from ocean remote-sensing imagery. *Natl Sci Rev* 7(10):1584–1605. <https://doi.org/10.1093/nsr/nwaa047>
19. Matsuoka D, Nakano M, Sugiyama D, Uchida S (2018) Deep learning approach for detecting tropical cyclones and their precursors in the simulation by a cloud-resolving global nonhydrostatic atmospheric model. *Progress Earth Planet Sci* 5. <https://doi.org/10.1186/s40645-018-0245-y>
20. Nakano M, Kubota H, Miyakawa T, Nasuno T, Satoh M (2017) Genesis of super cyclone pam (2015): Modulation of low-frequency large-scale circulations and the madden-julian oscillation by sea surface temperature anomalies. *Mon Weather Rev* 145(8):3143–3159. <https://doi.org/10.1175/MWR-D-16-0208.1>
21. Olander TL, Velden CS (2007) The advanced dvorak technique: Continued development of an objective scheme to estimate tropical cyclone intensity using geostationary infrared satellite imagery. *Weather Forecast* 22(2):287–298. <https://doi.org/10.1175/WAF975.1>
22. Olander TL, Velden CS (2019) The advanced dvorak technique (ADT) for estimating tropical cyclone intensity: Update and new capabilities. *Weather Forecast* 34(4):905–922. <https://doi.org/10.1175/WAF-D-19-0007.1>
23. Park MS, Kim M, Lee MI, Im J, Park S (2016) Detection of tropical cyclone genesis via quantitative satellite ocean surface wind pattern and intensity analyses using decision trees. *Remote Sens Environ* 183:205–214. <https://doi.org/10.1016/j.rse.2016.06.006>
24. Pradhan R, Aygun RS, Maskey M, Ramachandran R, Cecil DJ (2018) Tropical cyclone intensity estimation using a deep convolutional neural network. *IEEE Trans Image Process* 27(2):692–702. <https://doi.org/10.1109/TIP.2017.2766358>
25. Rozoff CM, Velden CS, Kaplan J, Kossin JP, Wimmers AJ (2015) Improvements in the probabilistic prediction of tropical cyclone rapid intensification with passive microwave observations. *Weather Forecast* 30(4):1016–1038. <https://doi.org/10.1175/WAF-D-14-00109.1>
26. Scher S, Messori G (2019) Weather and climate forecasting with neural networks: using general circulation models (GCMs) with different complexity as a study ground. *Geosci Model Dev* 12(7):2797–2809. <https://doi.org/10.5194/gmd-12-2797-2019>

27. Su H, Wu L, Jiang JH, Pai R, Liu A, Zhai AJ, Tavallali P, DeMaria M (2020) Applying satellite observations of tropical cyclone internal structures to rapid intensification forecast with machine learning. *Geophys Res Lett* 47(17). <https://doi.org/10.1029/2020GL089102>
28. Velden C, Olander T, Zehr R (1998) Development of an objective scheme to estimate tropical cyclone intensity from digital geostationary satellite infrared imagery. *Weather Forecast* 13(1):172–186. [https://doi.org/10.1175/1520-0434\(1998\)013<0172:DOAOST>2.0.CO;2](https://doi.org/10.1175/1520-0434(1998)013<0172:DOAOST>2.0.CO;2)
29. Velden C, Harper B, Wells F, Beven JL II, Zehr R, Olander T, Mayfield M, Guard CC, Lander M, Edson R, Avila L, Burton A, Turk M, Caroff A, Christian A, Caroff P, McCrone P, (2006) The Dvorak tropical cyclone intensity estimation technique. *Bull Am Meteorol Soc* 87(9):1195–1210. <https://doi.org/10.1175/BAMS-87-9-1195>
30. Wei L, Guan L, Qu L, Guo D (2020) Prediction of sea surface temperature in the China Seas based on long short-term memory neural networks. *Remote Sens* 12(17). <https://doi.org/10.3390/rs12172697>
31. Wentz F (1997) A well-calibrated ocean algorithm for special sensor Microwave/Imager. *J Geophys Res-Oceans* 102(C4):8703–8718. <https://doi.org/10.1029/96JC01751>
32. Xiang B, Lin SJ, Zhao M, Zhang S, Vecchi G, Li T, Jiang X, Harris L, Chen JH (2015) Beyond weather time-scale prediction for Hurricane Sandy and super typhoon Haiyan in a global climate model. *Mon Weather Rev* 143(2):524–535. <https://doi.org/10.1175/MWR-D-14-00227.1>
33. Yamaguchi M, Koide N (2017) Tropical cyclone genesis guidance using the early stage dvorak analysis and global ensembles. *Weather Forecast* 32(6):2133–2141. <https://doi.org/10.1175/WAF-D-17-0056.1>
34. Yi K, Wang X, Cheng Y, Chen C (2018) Hyperspectral imagery classification based on semi-supervised broad learning system. *Remote Sens* 10(5):685
35. Zhang W, Fu B, Peng MS, Li T (2015) Discriminating developing versus nondeveloping tropical disturbances in the Western North Pacific through Decision Tree Analysis. *Weather Forecast* 30(2):446–454. <https://doi.org/10.1175/WAF-D-14-00023.1>
36. Zhang Y, Yuen KV (2021) Crack detection using fusion features-based broad learning system and image processing. *Comput-Aided Civil Infrastruct Eng*. <https://doi.org/10.1111/mice.12753>
37. Zheng G, Li X, Zhang RH, Liu B (2020) Purely satellite data-driven deep learning forecast of complicated tropical instability waves. *Sci Adv* 6(29). <https://doi.org/10.1126/sciadv.aba1482>

**Open Access** This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (<http://creativecommons.org/licenses/by-nc-nd/4.0/>), which permits any noncommercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if you modified the licensed material. You do not have permission under this license to share adapted material derived from this chapter or parts of it.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

