Received February 26, 2020, accepted March 16, 2020, date of publication March 23, 2020, date of current version April 7, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2982772

A CNN-Based Hybrid Model for Tropical Cyclone Intensity Estimation in Meteorological Industry

WEI TIAN[1,2,3], (Member, IEEE), WEI HUANG[1], LEI YI[1], LIGUANG WU[4], AND CHAO WANG[5]

[1]School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China
[2]Jiangsu Collaborative Innovation Center of Atmospheric Environment and Equipment Technology(CICAEET), Nanjing University of Information Science and Technology, Nanjing 210044, China
[3]Engineering Research Center of Digital Forensics, Ministry of Education, Nanjing 210044, China
[4]Department of Atmospheric and Oceanic Sciences, Fudan University, Shanghai 200438, China
[5]School of Atmospheric Sciences, Nanjing University of Information Science and Technology, Nanjing 210044, China

Corresponding authors: Wei Tian (tw@nuist.edu.cn) and Wei Huang (hw19940831@gmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 41730961, Grant 41661144039, Grant 41875027, Grant 41922033, Grant 41675072, and Grant 61702277, and in part by the Natural Science Foundation of Jiangsu Province under Grant BK20181412.

ABSTRACT Accurate estimation of tropical cyclone (TC) intensity is the key to understanding and forecasting the behavior of TC and is crucial for initialization in forecast models and disaster management in the meteorological industry. TC intensity estimation is a challenge because it requires domain knowledge to manually extract TC cloud structure features and form various sets of parameters obtained from satellites. In this paper, a novel hybrid model is proposed based on convolutional neural networks (CNNs) for TC intensity estimation with satellite remote sensing. According to the intensity of TCs, we divide them into three types and use three different models for intensity regression, respectively. The results show that the use of piecewise thinking can improve the model's fitting speed on small samples. A classification network is provided to classify unlabeled TC samples before TC regression, whose results would determine which regression network to estimate these samples. Finally, the estimation values are sent to the backpropagation (BP) neural network to fit the suitable intensity values. Experimental results demonstrate that our model achieves high accuracy and low root-mean-square error (RMSE up to 8.91 kts) by just using inferred images.

INDEX TERMS Convolutional neural networks, hybrid model, tropical cyclone intensity estimation, infrared imagery.

I. INTRODUCTION

The meteorological industry covers many fields, such as transportation, agriculture and disaster prevention and mitigation. In the context of global warming, it is needed to strengthen the monitoring and analysis of meteorological conditions, early warning and forecast, extreme event detection and disaster analysis. Tropical cyclone (TC, also called as typhoon, hurricane, or cyclone) is a kind of terrible weather systems, which forms and develops on the warm tropical or subtropical ocean and often makes landfall. A TC could cause significant damage to properties and lives if it makes landfall [1]–[4].

As one of the most crucial parameters for TC forecasting and disaster management, the intensity of TC is defined as the maximum sustained surface wind near the TC center [5], [6]. Therefore, pursuing a precise estimation of TC intensity is a significant job for weather forecast and meteorological science research.

Aircraft observation was an early method of obtaining TC intensity; however, it was stopped because of the high cost [7]. Meanwhile, buoys in the ocean make it difficult to detect wind speed near the center of a TC. Therefore, the surveillances with satellites could serve as TC information's fundamental sources due to their worldwide scanning and excellent temporal frequency of geostationary satellites. Although the satellite remote sensing cannot directly measure TC intensity, it can be accessed as an agent for indirectly estimating TC intensity [8]–[11].

In meteorology, the development of TC intensity estimation now relies on constructing TC structure features and various parameter sets obtained from satellites. The Dvorak technique for TC intensity estimation has served for nearly 30 years [12]. However, it is still difficult to identify

The associate editor coordinating the review of this manuscript and approving it for publication was Hong-Ning Dai.

informative features for the diverse TCs in basins and different life stages, even for experienced meteorologists. In artificial intelligence, Ritesh Pradhan *et al.* [13] successfully applied CNN to the classification of TC category and found similar information to the Dvorak TC classification method through a visual model. Besides, Chen *et al.* [14] also proposed a rotation convolutional neural network to regress of TC intensity. However, the accuracy they obtained was not imprecise, leaving room for the improvement of the accuracy of TC classification and regression.

The purpose of this study is to duplicate the achievement with CNNs in deep learning to achieve the TC intensity from infrared satellite images with high accuracy. An alternative to current models in meteorology relying on human-constructed features is provided in this study. The contributions of this study are listed as follows:

- For the case where the training sample is small, we proposed the idea of piecewise regression and used other models to correct the negative impact caused by piecewise regression.
- By analyzing the output distribution of the model, we explain the principle of the models operation.
- The accuracy of TC intensity estimation is further improved, surpassing traditional meteorological methods.

The paper is organized as follows. In Section 2, the related works are introduced. Then, a mathematical model of a convolutional neural network with the introduction of the structure of each layer is described in Section 3. In Section 4, the CNN-based hybrid model, the specific parameters, and methods used to optimize the model are proposed. In section 5, the dataset used in the experiment is described in detail; besides, the processes and results of training, testing, and verification are analyzed. Moreover, the output distribution of the model is discussed in Section 6. Finally, the conclusions are drawn in the last section.

II. RELATED WORK

With the important influence of TC on social and economic activities, estimation of the intensity of TC would contribute to the prevention of disaster events and scientific perspective. In the last three decades, there are a number of skills utilizing satellite images for estimating TC intensity through applying Dvorak and its derivatives [15]–[17].

Several modified Dvorak techniques have been evolved over the last three decades. In the method of ADT [18], [19], computer-based algorithms are employed to identify cloud features, resulting in reducing subjectivity. Besides, ADT is approximately the maiden one utilizing linear regression [20] to estimate TC intensity. Low earth orbit satellites instead of geosynchronous satellites are utilized by AMSU to provide observations only when the satellites run across TC. SATCON [21] is a combination of different methods, such as ADT and AMSU. It could be taken as some heuristic blending guides relying on ocean region. Various models and parameters are used by DAVT techniques for each basin of TCs. In general, these methods are somewhat subjective, and it takes a lot of time and financial resources to train-related personnel.

A couple of researchers have made efforts to utilize CNN to pre-train their model on ImageNet; then, they have refined the model by the cross-entry loss on the TC classification dataset [22]. In 2018, Pradhan *et al.* divided TCs into eight categories with RMSE of 10.18 kts; then, the deep visualization toolbox [23] was employed to visualize the high activation degree as regularization optimization's result. The experimental results illustrated that the characteristics of varying levels' TCs could be extracted by CNN. Boyo Chen *et al.* [14], [24] presented a rotation-blended CNNs for the TC intensity regression model based on water vapor, passive micro-wave rain rate and infrared data. Combinido *et al.* [25] migrated the pre-trained VGG19 on ImageNets to the TC intensity estimation and obtained the RMSE of 13.23 kts only from the infrared cloud image of the TC obtained from the stationary satellite. Wimmers *et al.* [26], [27] mixed and analyzed passive microwave images from different frequency bands for different levels of TCs. They have provided the best frequency band for estimating TC passive microwave images with different intensities. The final RMSE of the model is 14.3 kts. The TC intensity estimation values obtained by this model are more global and stable compared to Dvorak; its enhanced techniques have been invoked in business for 30 years [28].

A CNN-based hybrid model was proposed to enhance the accuracy of TC intensity estimation; it involves regression, classification, and BP model. The parameters of the deep architecture are optimized, and the accuracy of the model is tested through a series of error measurement criteria in order to examine the validity of the proposed model. The experiment demonstrates that the proposed architecture is superior to the existing technologies in the task of estimating the intensity of TCs based on infrared images, even when a single infrared band is used.

III. INTERNAL CONVOLUTIONAL NEURAL NETWORK

The convolutional neural network is also called "Shift-Invariant Artificial Neural Networks" and is a special type of neural network. It has the capability to perform a shift-invariant classification of input information according to its hierarchical structure. Meanwhile, it can automatically extract features in images; therefore, it is suitable for computer vision applications. The use of a convolution kernel avoids the use of the one-to-one connection between all pixel units and reduces the number of parameters in the network through feature sharing [29], [30]. The convolution output at layer l is given in Eq. (1), where K is the size of the convolution kernel, w_{ab} is the weight of weighted kernel at position (a, b), and B is the bias for layer l.

$$x_l^{ij} = \sum_{K-1}^{a=0} \sum_{K-1}^{b=0} w^{ab} y_{l-1}^{(i+a)(i+b)} + B^l \tag{1}$$

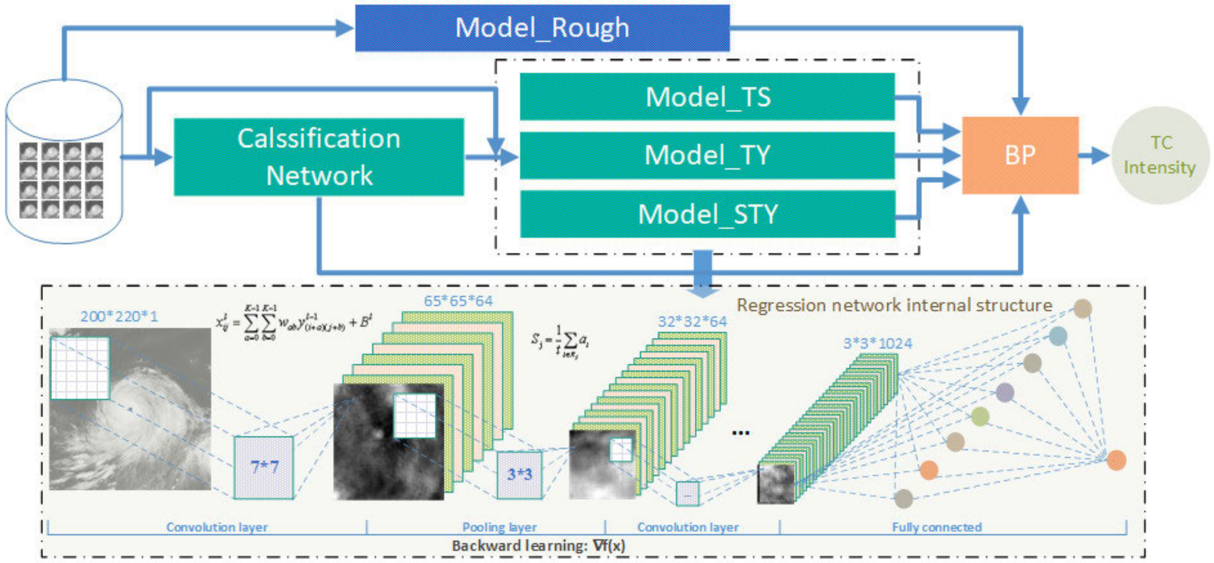


FIGURE 1. **Hybrid model overall architecture and internal regression model structure.**

The size(f), stride(s) and padding (p) of the convolution the kernel would affect the size of the feature map in the next layer; the size of the output after convolution would be rounded down. The calculation of the feature map n_H^L is given by Eq. (2) in the next layer.

$$n_H^L = \left\lfloor \frac{n_H^{L-1} + 2p^L - f^L}{s^L} + 1 \right\rfloor \tag{2}$$

Pooling can be understood as downsampling and used to improve the model computing speed and robustness with the advantages of parameter compression and overfitting reduction. Pooling includes max pooling, average pooling and so on. The input layer is divided into different areas by the max pooling with non-overlapping rectangular boxes; the maximum value of each rectangular box is taken as the output layer. The calculation method of pooling is given in Eq. (3), where R_j represents the divided rectangular bax in the j^{th} feature map, i represents the index in this rectangular box, and S_j indicates the pooling result of this rectangular box. The size of the feature map after pooling needs to be rounded up. This is different from the convolution.

$$S_j = \max_{i\in R_j} a_j \tag{3}$$

The need to constantly adapt to the new distribution is caused by the changes in the input distribution of the layer; it is called covariate drift [31]. The concept of covariate shifting can be extended to various parts of the entire learning system. During the training process, the distribution of each layer changes as the parameters of the previous layer change. This phenomenon is called the internal covariate offset [32] and can be solved by normalizing the layer input. Santurkar *et al.* [33] believed that the activation function would lose nonlinear factors due to batch normalization, resulting in affecting the data representation ability and reducing the role of the neural network. The calculation formula is given in Eq. (4). The mean and standard-deviation are calculated per-dimension over the mini-batches, and γ and β are learnable.

$$y = \frac{x - \mathrm{E}[x]}{\sqrt{\mathrm{Var}[x] + \epsilon}} * \gamma + \beta \tag{4}$$

Dropout [34] is to traverse the neurons in each layer of the convolutional neural network; then, set the neuron retention probability is set through the convolutional neural network in this layer; therefore, the neural network would not be biased to a certain neuron to reduce the overfitting of the neural network.

IV. TROPICAL CYCLONE INTENSITY ESTIMATION HYBRID MODEL

Targeted research on one aspect can penetrate the field faster. It is feasible to estimate the intensity of TCs at the different categories by coupling multiple models because it is difficult to use a single model to regress globally to achieve a high-precision output with small scale samples.

A. PROPOSED MODEL STRUCTURE

Based on the analysis of the output distribution of different models, a hybrid model for the estimation of TC intensity is proposed in this paper. The model structure is shown in Fig. 1. The figure contains the architecture of the hybrid model and the structure of the regression network inside the architecture. The upper part of the picture is the model architecture, and the structure of the regression network is shown in the dashed box (explained in the next section). The architecture shows the execution order and data flow of each sub-model, and the content in the dotted box shows the convolution implementation process of the regression network.

TABLE 1. Regression model parameters.

Name	(Input, output)	Kernel_size(padding)	MaxPool(padding)	BatchNorm\Dropout\ReLU
Conv1	(1,64)	$7 \times 7/3(0)$	$3 \times 3/2(0)$	64 0.5 True
Conv2	(64,256)	$5 \times 5/1(0)$	$3 \times 3/2(0)$	256 0.5 True
Conv3	(256,400)	$3 \times 3/1(1)$	$2 \times 2/1(0)$	400 0.5 True
Conv4	(400,576)	$3 \times 3/1(1)$	$3 \times 3/2(0)$	576 0.5 True
Conv5	(576,1024)	$3 \times 3/1(0)$		1024 0.5 True
fc1	(9216,1152)			ReLU
fc2	(1152,144)			ReLU
fc3	(144,1)			

Specifically, the hybrid model consists of the classification model, the regression model and the BP model. The regression model is further divided into a coarse-grained model (Model_Rough) (trained by all datasets), and three fine-grained models, including Model_TS, Model_TY and Model_STY (trained by well-classed sub-datasets). The classification model is used to classify TC samples of different intensities into three categories (strong, medium and weak). Three different cyclone intensities are estimated by Model_TS, Model_TY and Model_STY, respectively. The output of the classification model, two fine-grained Models and Model_Rough are taken as inputs by the BP model to obtain TC intensity. The output of the two fine-grained models comes from the output of the fine-grained regression model and neighboring model corresponding to the classification results.

It can be seen from the input of the BP model that the input comes from two parts, that is, the classification-regression model composing of the classification model and two fine-grained regression models and Model_Rough. The classification-regression model provides two types of output for BP, namely, a classification result and two of three regression results.

The number of fine-grained regression models is not completely fixed and can be determined according to the number of datasets, the complexity of the model and the accuracy of the classification model.

B. STRUCTURE AND PARAMETERS OF THE SUBMODEL

The regression model in the dashed box in Fig. 1 is used for the regression of TCs. The parameters of each layer are illustrated in Table 1. The model input size is 200×200. It can be known from the first row of the table that the sample passes through 64 feature maps. Among them, the size of the feature map is 7×7; the stride is 3; a matrix of size $65 \times 65 \times 64$ is obtained. Then, a pooling operation with a size of 3×3 and a stride of 2 to obtain a matrix of size $32 \times 32 \times 64$ is performed on the matrix. They are sent to the second layer to continue calculation after batch normalization, random inactivation and nonlinear activation function.

In the hybrid model, we set all three fine-grained classification networks and coarse-grained regression networks as the parameters described above. The classification model is almost the same as the structure of the regression model.

TABLE 2. Comparison scheme of BP model under different inputs and structures, the schemes were compared by means of controlling variables.

hidden layer	input size	Scheme	activation funcation
64	3	Scheme I	relu
	4		
	5		
128,64	3	Scheme II	
	4		
	5		
256,128,64	3	Scheme III	
	4		
	5		
64	5	Scheme IV	
128,64	4		
256,128,64	4		

In the final fully connected layer, Softmax is used to output the categories compared to the regression model.

The BP model with four input neurons and a 5-layer network structure are selected to fit the final intensity. The number of these five hidden layers is 256, 128, 64. Relu activation function is used between layers to obtain non-linear factors, except for the last layer. These parameters are obtained through different scheme experiments in Table 2.

C. OPTIMIZATION

In this subsection, the optimization methods used in the experiment would be proposed and described.

1) WEIGHT INITIALIZATION STRATEGY

The weights are initialized to a constant of 0.1 by Xavier [35], leading to speeding up the training of the model. Meanwhile, the problems of gradient explosion and non-linear relationship loss caused by initialization can be effectively solved.

2) ADAM OPTIMIZATION

Adam is an excellent first-order optimization algorithm that can replace traditional stochastic gradient descent by combining the advantages of AdaGrad and RMSProp [36], [37]. Besides, the neural network weights based on training data can be iteratively updated. It takes up less memory while performing efficient calculations; moreover, it can handle noise samples and adaptability to sparse gradients well. The learning rate is set to a fixed value of 0.0001 in this paper.

TABLE 3. **TCs classification.**

dataset	name	level	wind speed(m/s)	classifification
Data_TS	TD	6-7	10.8-17.1	Model_TS
	TS	8-9	17.2-24.4	
Data_TY	STS	10-11	24.5-32.6	Model_TY
	TY	12-13	32.7-41.4	
Data_STY	STY	14-15	41.5-50.9	Model_STY
	SuperTY	16 or above	>51.0	

TABLE 4. **Partitioned dataset.**

Total data set	46,919
Data_TS	26,397
Data_TY	11,676
Data_STY	6,846
Verification_set	2000

V. EXPERIMENT

In this section, the source, partition and some preprocessing methods of the dataset would be introduced; then, the training of each sub-model in our hybrid model would be proposed; finally, integration tests would be conducted.

A. DATASET

In this study, a TC remote sensing benchmark dataset is used to estimate the intensity of the Pacific Northwest and the Atlantic Ocean in 'TCIR' from 2003 to 2016 [14]. 46, 919 TC infrared images are selected as samples for TC intensity estimation; A frame of infrared images is a record of TCs per 6 hours. Each frame image has its own attributes, such as wind speed, size, minimum sea-level pressure and TC center location; For each frame, there are 201 × 201 datapoints; the resolution is 7/100 degree lat/lon; due to some problems of the satellite itself, there may be some default values in the infrared image, which is not friendly to the final regression intensity value.

1) DATA CLASSIFICATION

According to the "Tropical Cyclone Rating" standard (GB/T 19201-2006), the dataset is divided into three sub-datasets based on wind speed, including Data_TS, Data_TY and Data_STY, as shown in Table 3. The sub-datasets are used to train, test and validate the corresponding fine-grained models, such as training, testing, and validating Model_TS with Data_TS. The validation dataset does not participate in training.

It should be noted that TCs in the southern hemisphere and some frames with some default values were dropped; a total of 46, 919 samples were obtained. The sample number of the sub-dataset of each class is displayed in Table 4. The number of samples decreases as the intensity of the TCs increases.

2) DUPLICATION AND ROTATION

With the increase of TC intensity, the number of samples corresponding to the intensity decreases sharply. Therefore, the sample is extended by duplicating and rotating in order to ensure that the model has the same probability distribution for each mini-batch during training. The rotation invariance of TCs ensures that the rotated cyclone can also act as a duplication.

3) DATA PREPROCESSING

In the data preprocessing phase, we need to delete the samples with more default values and fill the samples with less default values with '0'. We define "more default values" as the percentage of all pixels with the default value greater than 10%. Then, the samples are normalized and converted into Tensor. Finally, the data are regularized with a mean value of 0.5 and a variance of 0.5.

B. TRAINING AND TESTING

The training of the model is independent during the training phase. After the classification model is trained and the regression model is completed, the dataset is sent to the trained model for verification; then, the result is sent to the BP model for training to complete the entire model training.

1) CLASSIFICATIONS MODEL

It can be seen from Fig. 8 that the accuracy of the classification model is significant for the error of the model regression. Some well-known models such as GoogleNet [38], ResNet [39] and DenseNet [40] are trained in this study; then, these models are compared with ours to find the model with the best accuracy.

During the training, a cross-validation function is used as the loss function. Finally, our classification model is selected as the classifier for the hybrid model.

2) REGRESSION MODEL

The fine-grained regression model based on the convolutional neural network is focused during the training. The data of the classifications are sent to the corresponding regression models for training and testing; the parameters of different regression models are fine-tuned separately.

In Fig. 2, the loss of training and testing of the model at 100 epochs is illustrated. The x-axis scale is the number of training iterations; the y-axis scale is the loss value.

As shown in Fig. 2, the loss obtained by the Model_Rough is between the loss of Mode_TY and Model_STY; besides, with the increasing of the training times, the training of TCs achieves better performance compared to Model_Rough. The training error of the strong typhoon model is close to that of Model_Rough. The loss of Model_TS reaches 6.58 knots; the loss of Model_SY reaches 8.40 knots; the loss of Model_STY reaches 12.51 knots; the loss of Model_Rough is 10.21 knots.

3) BP MODEL

After the classification model and the regression model complete the training, the BP model training leverages the output of the classification model and regression model as the input of the BP model. To find the optimal parameters and model structure, we list several sets of experiments in Table 2.

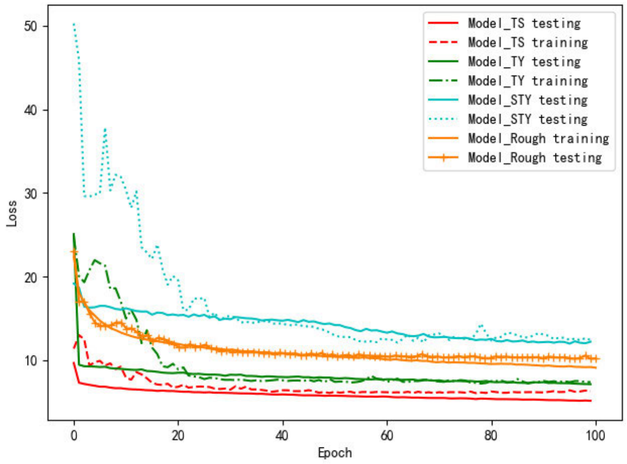
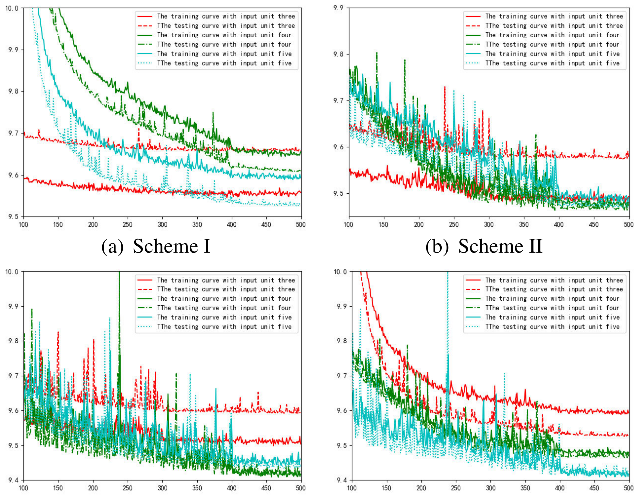


FIGURE 2. Regression model training.



(a) Scheme I (b) Scheme II

(c) Scheme III (d) Scheme IV

FIGURE 3. Comparison of training of different structures and input BP models.

The training curves of different inputs and different hidden layers of the BP model are displayed in Fig. 3. The lines would be caused to overlap and the training curve would be unclear because the loss in the first half of the training times is large and the loss training in the second half are roughly the same. Therefore, the data after 100 training epochs are selected for drawing the graph. It can be seen from Figs. 3(a), 3(b), and 3(c) that the three hidden layers effectively fit the inputs. It can be found in Fig. 3(d) that the fitting result of the BP model with an input layer of four neurons is the best. The four inputs of the BP model are the outputs of the coarse-grained model, the classification model and the two fine-grained regression models.

Moreover, the BP model with four inputs is further researched. The improvement of the performance of the model is not obvious as the number of hidden layers increases. Therefore, four inputs and three hidden layers are selected as the input and structure of the BP model.

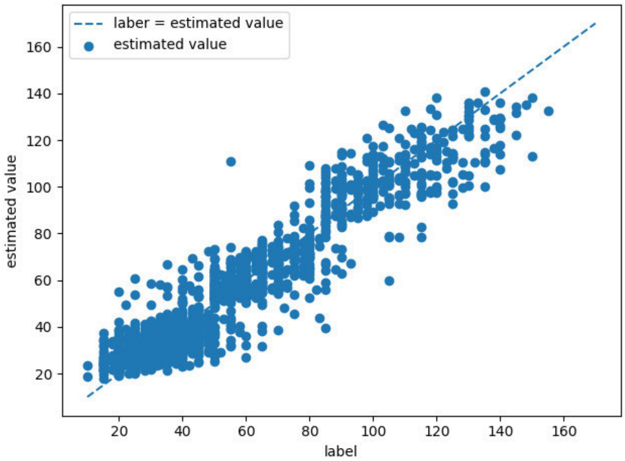


FIGURE 4. Validation estimation distribution.

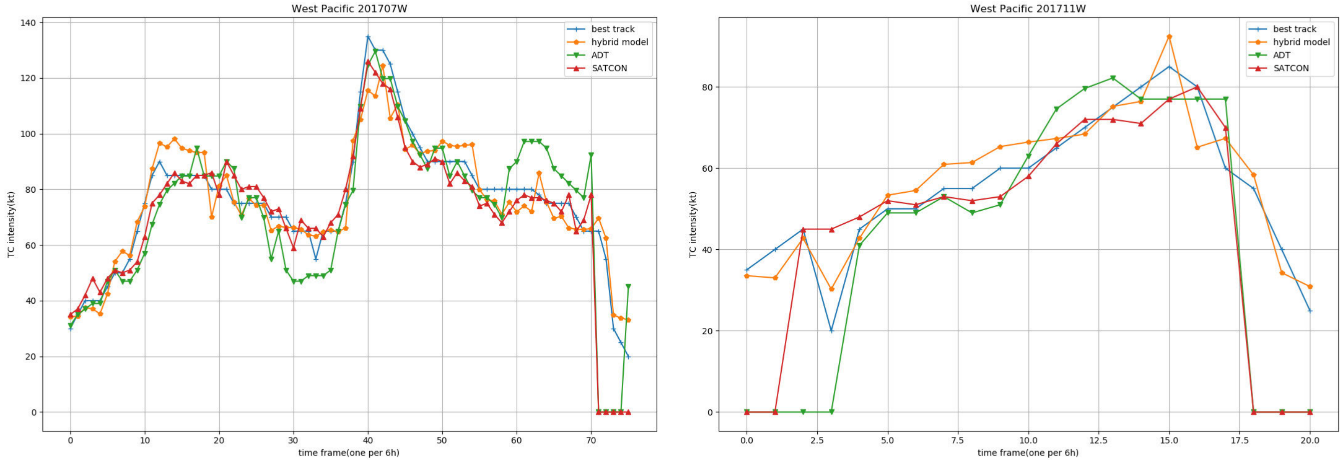
C. MODEL INTEGRATION

After the training of each model, TC intensity is estimated according to the model structure of Fig. 1. The coarse-grained regression model and the classification-regression model jointly provide inputs for the BP model. In the BP model, a coarse-grained model is used to correct the output of the classification-regression model in order to optimize the result of our hybrid model. The RMSE accuracy reaches 8.91 knots. The output distribution of the BP model is shown in Fig. 4; the outputs are concentrated near the straight line with slope equal to one and intercept equal to zero.

D. MODEL VALIDATION

In order to verify the applicability and effectiveness of the model, we randomly selected two TCs with a complete life cycle from 33 TCs in the Western Pacific in 2017 to evaluate the performance of the model. These data are new and do not participate in the training process of the model. As shown in Figure 5, it is a line chart of real-time regression of TCs with different technologies over the entire life cycle; the x-axis represents the time series with an interval of 6 hours, and the y-axis represents the intensity of the TC; lines of different styles and colors represent different types of technologies. We used ADT and SATCON technology to compare with our hybrid model. The best track data comes from the Joint Typhoon Warning Center (JWTC). The ADT and SATCON data are from the Meteorological Satellite Research Cooperation Institute. The sampling rate of the ADT and SATCON data is half an hour. We selected the intensity value at the hour to match the best track data.

As is shown in Figure 5, We can observe that ADT performs better when the TC is extremely strong, and SATCON can also rely on ADT to perform well in such condition by heuristic rules; the performances of the hybrid model and SATCON are similar; The estimation of the intensity of TCs that is small is abandoned. It is noticeable that both ADT and SATCON derive from the Dvorak-derived technology; As a heuristic rule, SATCON consists of multiple technologies



(a) NORU (b) NESAT

FIGURE 5. Line graphs of the intensity estimates of the complete life cycle of TCs under different methods. On the left (a) is TC No.7 (NORU) in the Western Pacific in 2017, and on the right (b) is TC No. 11 (NESAT).

with certain complexity; The method based on deep learning has greater advantages than meteorological methods in real-time and objectivity.

VI. DISCUSSION

In this section, the difficulties of capturing the features of the cyclones at three different categories of ‘‘weak, medium and strong’’ and the output distribution of different models are compared.

A. THE DIFFICULTY OF CAPTURING FEATURES

In Fig. 2, the training performance of the Model_STY is not as good as that of Model_Rough because the training and testing errors of the models become larger as the intensity of TC increases. This is not able to determine that the ability of Model_STY to grasp the TC characteristics is not as good as that of Model_Rough; besides, it can not be concluded that the features of strong TCs are more difficult to capture than those of weak TCs.

TC is a severe weather system. Approximately 20 TCs are generated each year and flow into coastal countries in the Pacific Northwest. Although satellite data have been used for nearly 13 years since 2003 for TC intensity estimation, the size of the dataset is still not large enough. Moreover, the number of samples decreases as TC intensity becomes strong. Although TC samples are rotated and replicated, the number of strong TC samples and weak TC samples is uneven. Therefore, the number of training samples of the other two fine-grained models is kept to be consistent with the number of samples of Data_STY in order to verify the difficulty of capturing the characteristics of TCs of different categories, as shown in Fig. 6.

The Model_STY exhibits better results under the same sample size, while the other two models display different degrees of over-fitting. It is concluded that it is easier for the characteristics of the Model_STY to be captured compared to the other two fine- grained models. This is also consistent with the perception that the characteristics of the strong TC’s eye area and convective cloud system are more stable on the cloud images. We believe that the model would be more accurate and achieve better performance with the increase of the size of Data_STY, which improves the accuracy of the hybrid model.

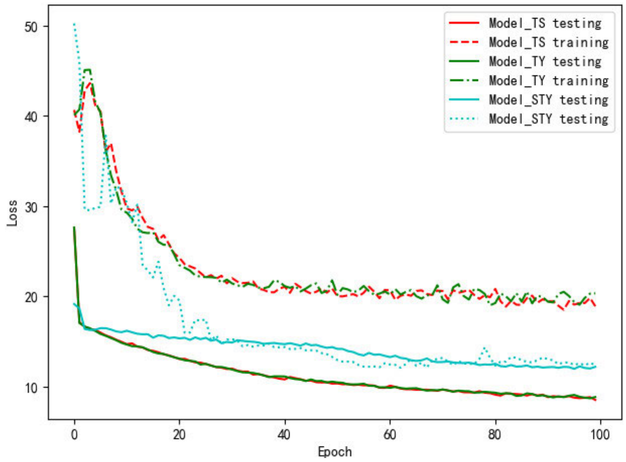
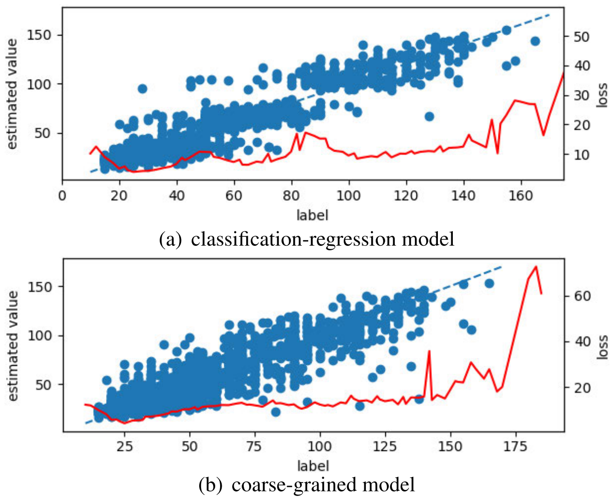


FIGURE 6. Training loss graph of different fine-grained models under the same samples.

During model training, the loss caused by strong TCs would be reduced by a large number of weak TCs, leading to reducing the overall loss. The strong TCs are tested with a coarse-grained model; the accuracy is 13.05 knots, which is consistent with our assumption.

B. OPTIMIZING THE OUTPUT OF THE CLASSIFICATION -REGRESSION NETWORK

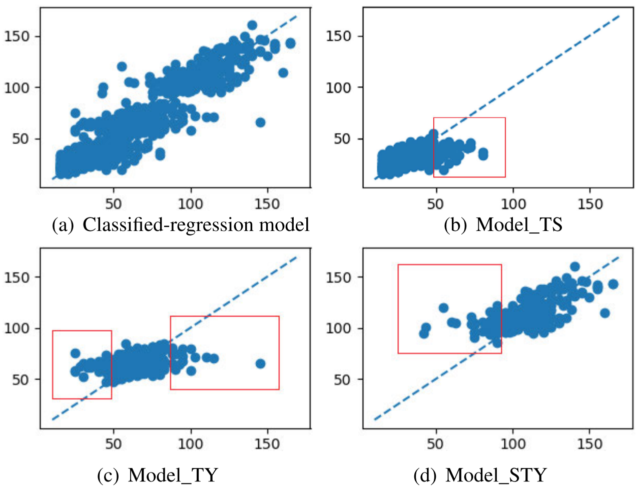
In this section, the output distribution of each model is analyzed; the principle of our hybrid model is explained; then,



(a) classification-regression model

(b) coarse-grained model

FIGURE 7. Distribution of coarse-grained and fine-grained models output and loss of corresponding labels.



(a) Classified-regression model

(b) Model_TS

(c) Model_TY

(d) Model_STY

FIGURE 8. Distribution of noise in classification-regression model among fine-grained models.

how to further improve the accuracy of the model is pointed out.

The output distribution of 2000 random samples under different models is illustrated in Fig. 7. The x-axis of the graph represents labels of samples, the y-axis on the left side represents the estimation values corresponding to the blue points, and the y-axis on the right side represents losses of labels corresponding to the red polyline. Fig. 7(a) is the output of the classification-regression model; Fig. 7(b) is the output of the Model_Rough.

It can be seen from the comparison of Fig. 7 that the red polyline representing losses of Fig. 7(b) is a relatively smooth while, but the overall loss is larger than that shown in Fig. 7(a). In Fig. 7(a), a good fitting relationship in some intervals on the coordinate is displayed; however, the bias in the interval of $40-60$ knots and $80-100$ knots are larger than that shown in Fig. 7(b). These intervals increase at the boundary of the sub-datasets.

Based on the above discussion, the model is expected to minimize the loss of corresponding labels. For instance, the final intensity value from the model shown in Fig. 7(b) is expected to be provided when the labels range from 40 to 60 knots. However, we do not know that the correct value for an unlabeled TC sample is obtained from which model; we hope to find the relationship between them through the BP model.

C. THE IMPACT OF CLASSIFICATION ACCURACY

As can be seen from Fig. 7, most of the output values of the model are distributed near the line and the other part is scattered. Then, we will analyze these scattered points.

Fig. 7(a) is divided into three parts that correspond to the output distributions of the three fine-grained models using the results of the classification model. The division results are illustrated in Fig. 8. The x-axis scale represents the labels and the y-axis scale represents the estimation values.

The discrete values in the classification-regression model with the points in Figs. 8(b), 8(c) and 8(d) are compared, indicating that those discrete points are mainly caused by the errors from the classification model. For example, those points in Fig. 8(b) are greater than 49 knots, indicating a classification error. Besides, it is also found that those estimation values are distributed below the corresponding labels. These misclassified points lead to some discrete points in Fig. 8(a)

It can be seen that the error in Fig. 8 is mainly caused by the classification error; the accuracy of the hybrid model can be further improved in the case of further improving the classification accuracy.

D. EDGE OUTPUT DISTRIBUTION OF THE SUBSET

To facilitate further discussion, we define the following concepts.

- The boundaries of the sub-datasets: a dataset with label values ranging from 0 to 170 knots is provided. The sub-dataset is divided according to labels. Data_TY label values range from 49 to 83 knots; 49 and 83 knots are called as the boundaries of the Data_TY.
- Edge-dataset: the dataset is further divided according to the boundary of the sub-dataset and the performance of the classification model. For example, an edge dataset with a label range of $49-54$ knots is provided; among them, 49 knots is the boundary of Data_TY; the classification network has poor performance on $49-54$ knots.
- Strong model and weak model: Model_TY is specifically used to classify cyclones with a label range of $49-83$ knots; the results of the classification model are not entirely correct. We can say that the sample of Data_TY is misclassified into a weak model for regression estimation when the sample of Data_TY is misclassified into Model_TS.

As can be seen from Fig. 8, the estimated values on the Edge-dataset are offset; the distribution of these points is generally

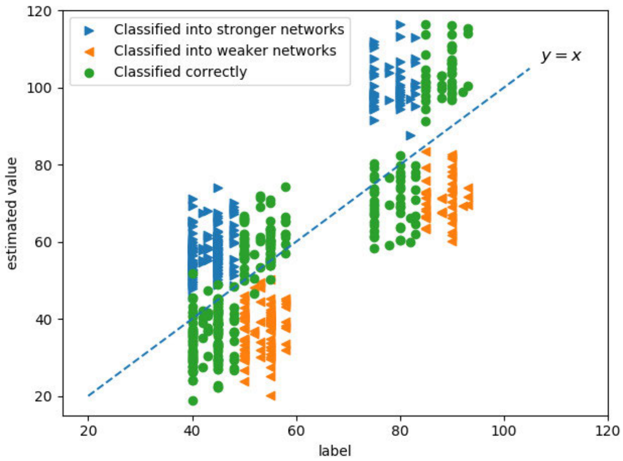


FIGURE 9. Estimation distribution of the same samples under correct and wrong classification.

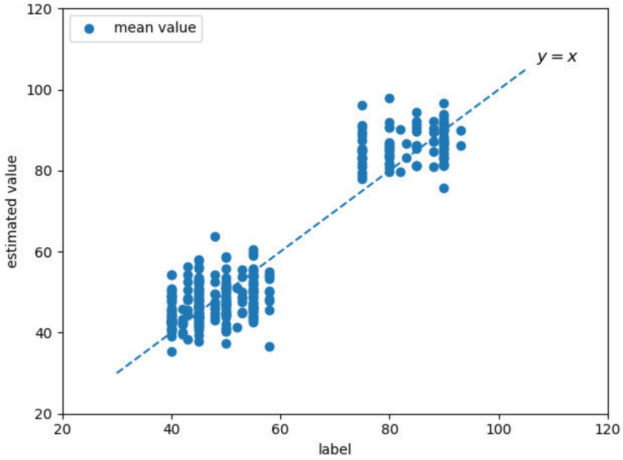


FIGURE 10. The distribution of the average of the prediction values obtained in the corresponding model and the values obtained in the adjacent model.

up or down. Put these points into the corresponding and adjacent models for estimation, and get the results as shown in Fig. 9.

The distribution of green dots in Fig. 9 indicates that the samples are placed in the correct model for estimation. The blue right-facing arrows illustrate that the samples are classified into the strong model while the orange left-facing arrows illustrate that the samples are classified into the weak model. In other words, the samples are misclassified into adjacent models by the classification model.

It is obvious that outputs of Edge-datasets are all distributed at one side of the straight line with slope equal to one and intercept equal to zero. The samples with small label values in the sub-dataset are located below the straight line in the corresponding model; besides, the predicted distribution of the samples with large label values are above the straight line. Samples input weak model with small label values in the data subset indicte a downward shift in the output distribution. In other words, the data distribution would be caused to shift downward due to the misclassification of samples with small label values in the data subset.

Based on the distribution of the output of the model, the average of the prediction values obtained in the corresponding model and the values obtained in the adjacent model are empirically computed, as shown in Eq. (5).

$$output_{actual} = \frac{output_{true} + output_{flase}}{2} \tag{5}$$

The distribution of the sample is shown in Fig. 10. The results are distributed on both sides of the straight line with slope equal to one and intercept equal to zero; the loss of the Edge-dataset is greatly optimized; the value is highly robust.

It can be seen that the final output of the edge value can be affected by the output of two fine-grained models adjacent to the edge-dataset. Therefore, the estimation values of the coarse-grained model and the classification-regression model are sent to the BP model for training; finally, the estimation values are obtained.

VII. CONCLUSION

In this paper, we applied the CNN-based hybrid model to satellite data feeds to objectively estimate the TC intensity in near real time. The CNN-based hybrid model consists of a coarse-grained model, a classification model, three fine-grained models and a BP model. In the classification model, the TC intensity level is provided to select the corresponding fine-grained model for intensity estimation. The errors caused by the classification are reduced by the coarse-grained model through the BP network, contributing to the improvement of the final accuracy of our hybrid model.

The model was trained through a large number of samples to obtain an optimized set of parameters. With these optimized parameters, the intensity of TCs can be quickly estimated. Compared with the latest meteorological method [21], our optimization results have only improved by 0.3 kts, but they have significantly improved their objectivity and real-time performance. Compared with the deep learning method [14], our method is improved by 1.52 kts.

Our hybrid model can be applied in many areas of future work. The model has prominent advantages in the prediction of continuous variables.

ACKONWLEGMENT

Thanks for the data support from Computational Learning Lab, Department of CSIE, National Taiwan University, Taiwan, (https://www.csie.ntu.edu.tw/htlin/program/TCIR/) and Cooperative Institute for Meteorological Satellite Studies (http://tropic.ssec.wisc.edu/real-time/satcon/).

REFERENCES

[1] Q. Zhang, L. Wu, and Q. Liu, "Tropical cyclone damages in China 1983–2006," *Bull. Amer. Meteorological Soc.*, vol. 90, no. 4, pp. 489–496, Apr. 2009.

[2] Y. Zuo, Y. Wu, G. Min, and L. Cui, "Learning-based network path planning for traffic engineering," *Future Gener. Comput. Syst.*, vol. 92, pp. 59–67, Mar. 2019.

[3] H. Wang, Y. Wu, G. Min, J. Xu, and P. Tang, "Data-driven dynamic resource scheduling for network slicing: A deep reinforcement learning approach," *Inf. Sci.*, vol. 498, pp. 106–116, Sep. 2019.

[4] Z. Yao, J. Ge, Y. Wu, and L. Jian, "A privacy preserved and credible network protocol," *J. Parallel Distrib. Comput.*, vol. 132, pp. 150–159, Oct. 2019.

[5] Y. Wang, Z.-Y. Ding, X. Lin, X.-Y. Shen, and Y. Fan, "Preliminary analysis on the intensity and structure of typhoon morakot (2009) during its landing process," *J. Tropical Meteorol.*, vol. 20, no. 1, pp. 35–44, 2014.

[6] L. Wu, W. Tian, Q. Liu, J. Cao, and J. A. Knaff, "Implications of the observed relationship between tropical cyclone size and intensity over the western North Pacific," *J. Climate*, vol. 28, no. 24, pp. 9501–9506, Dec. 2015.

[7] W. Tian, T. Ma, Y. Zheng, X. Wang, Y. Tian, A. Al-Dhelaan, and M. Al-Rodhaan, "Weighted curvature-preserving PDE image filtering method," *Comput. Math. Appl.*, vol. 70, no. 6, pp. 1336–1344, Sep. 2015.

[8] T. L. Olander and C. S. Velden, "Tropical cyclone convection and intensity analysis using differenced infrared and water vapor imagery," *Weather Forecasting*, vol. 24, no. 6, pp. 1558–1572, Dec. 2009.

[9] C. Velden, B. Harper, F. Wells, J. L. Beven, R. Zehr, T. Olander, M. Mayfield, C. Guard, M. Lander, R. Edson, L. Avila, A. Burton, M. Turk, A. Kikuchi, A. Christian, P. Caroff, and P. McCrone, "The Dvorak tropical cyclone intensity estimation technique: A satellite-based method that has endured for over 30 years," *Bull. Amer. Meteorol. Soc.*, vol. 87, no. 9, pp. 1195–1210, Sep. 2006.

[10] H. Wang, C. Guo, and S. Cheng, "LoC—A new financial loan management system based on smart contracts," *Future Gener. Comput. Syst.*, vol. 100, pp. 648–655, Nov. 2019.

[11] H. Wang, S. Ma, H.-N. Dai, M. Imran, and T. Wang, "Blockchain-based data privacy management with nudge theory in open banking," *Future Gener. Comput. Syst.*, early access, Oct. 4, 2019, doi: 10.1016/j.future.2019.09.010.

[12] X. Xu, X. Zhang, H. Gao, Y. Xue, L. Qi, and W. Dou, "BeCome: Blockchain-enabled computation offloading for IoT in mobile edge computing," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4187–4195, Jun. 2020.

[13] R. Pradhan, R. S. Aygun, M. Maskey, R. Ramachandran, and D. J. Cecil, "Tropical cyclone intensity estimation using a deep convolutional neural network," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 692–702, Feb. 2018.

[14] B. Chen, B.-F. Chen, and H.-T. Lin, "Rotation-blended CNNs on a new open dataset for tropical cyclone Image-to-intensity regression," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 90–99.

[15] X. Xu, Y. Chen, X. Zhang, Q. Liu, X. Liu, and L. Qi, "A blockchain-based computation offloading method for edge computing in 5G networks," *Softw., Pract. Exper.*, early access, Sep. 10, 2019, doi: 10.1002/spe.2749.

[16] V. F. Dvorak, "Tropical cyclone intensity analysis and forecasting from satellite imagery," *Monthly Weather Rev.*, vol. 103, no. 5, pp. 420–430, May 1975.

[17] E. A. Ritchie, K. M. Wood, O. G. Rodríguez-Herrera, M. F. Piñeros, and J. S. Tyo, "Satellite-derived tropical cyclone intensity in the north pacific ocean using the deviation-angle variance technique," *Weather Forecasting*, vol. 29, no. 3, pp. 505–516, Jun. 2014.

[18] H. Wang, S. Ma, and H.-N. Dai, "A rhombic dodecahedron topology for human-centric banking big data," *IEEE Trans. Comput. Social Syst.*, vol. 6, no. 5, pp. 1095–1105, Oct. 2019.

[19] T. L. Olander and C. S. Velden, "The advanced dvorak technique: Continued development of an objective scheme to estimate tropical cyclone intensity using geostationary infrared satellite imagery," *Weather Forecasting*, vol. 22, no. 2, pp. 287–298, Apr. 2007.

[20] L. Qi, Y. Chen, Y. Yuan, S. Fu, X. Zhang, and X. Xu, "A QoS-aware virtual machine scheduling method for energy conservation in cloud-based cyber-physical systems," *World Wide Web*, vol. 23, no. 2, pp. 1275–1297, Mar. 2020.

[21] D. Herndon, C. Velden, and J. Hawkins, "Update on satellite-based consensus (SATCON) approach to TC intensity estimation," in *Proc. 30th AMS Conf. Hurricanes Tropical Meteorol.*, Ponte Vedra Beach, FL, USA, 2012.

[22] J. Miller, M. Maskey, and T. Berendes, "Using deep learning for tropical cyclone intensity estimation," in *Proc. AGU Fall Meeting Abstr.*, vol. 2017, Dec. 2017, p. IN11E-05.

[23] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson, "Understanding neural networks through deep visualization," 2015, *arXiv:1506.06579*. [Online]. Available: http://arxiv.org/abs/1506.06579

[24] X. Xu, C. He, Z. Xu, L. Qi, S. Wan, and M. Z. A. Bhuiyan, "Joint optimization of offloading utility and privacy for edge computing enabled IoT," *IEEE Internet Things J.*, early access, Sep. 26, 2019, doi: 10.1109/JIOT.2019.2944007.

[25] J. S. Combinido, J. R. Mendoza, and J. Aborot, "A convolutional neural network approach for estimating tropical cyclone intensity using satellite-based infrared images," in *Proc. 24th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2018, pp. 1474–1480.

[26] A. Wimmers, C. Velden, and J. H. Cossuth, "Using deep learning to estimate tropical cyclone intensity from satellite passive microwave imagery," *Monthly Weather Rev.*, vol. 147, no. 6, pp. 2261–2282, Jun. 2019.

[27] X. Xu, R. Mo, F. Dai, W. Lin, S. Wan, and W. Dou, "Dynamic resource provisioning with fault tolerance for data-intensive meteorological workflows in cloud," *IEEE Trans. Ind. Informat.*, early access, Dec. 2019, doi: 10.1109/TII.2019.2959258.

[28] X. Xu, S. Fu, L. Qi, X. Zhang, Q. Liu, Q. He, and S. Li, "An IoT-oriented data placement method with privacy preservation in cloud environment," *J. Netw. Comput. Appl.*, vol. 124, pp. 148–157, Dec. 2018.

[29] I. Hadji and R. P. Wildes, "What do we understand about convolutional networks?" 2018, *arXiv:1803.08834*. [Online]. Available: http://arxiv.org/abs/1803.08834

[30] X. Xu, Q. Liu, Y. Luo, K. Peng, X. Zhang, S. Meng, and L. Qi, "A computation offloading method over big data for IoT-enabled cloud-edge computing," *Future Gener. Comput. Syst.*, vol. 95, pp. 522–533, Jun. 2019.

[31] H. Shimodaira, "Improving predictive inference under covariate shift by weighting the log-likelihood function," *J. Stat. Planning Inference*, vol. 90, no. 2, pp. 227–244, Oct. 2000.

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

[33] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, "How does batch normalization help optimization?" in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 2483–2493.

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

[35] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.

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

[37] S. J. Reddi, S. Kale, and S. Kumar, "On the convergence of Adam and beyond," 2019, *arXiv:1904.09237*. [Online]. Available: http://arxiv.org/abs/1904.09237

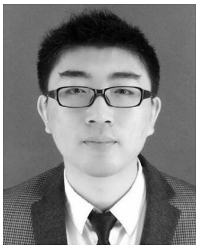
[38] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.

[39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[40] G. Pleiss, D. Chen, G. Huang, T. Li, L. van der Maaten, and K. Q. Weinberger, "Memory-efficient implementation of DenseNets," 2017, *arXiv:1707.06990*. [Online]. Available: http://arxiv.org/abs/1707.06990

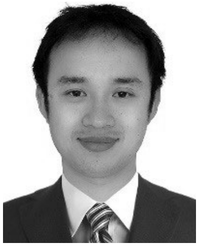
WEI TIAN (Member, IEEE) received the M.S. degree in systems analysis and integration and the Ph.D. degree in meteorology from the Nanjing University of Information Science and Technology, China, in 2007 and 2016, respectively. He worked as a Research Scholar with Michigan State University, USA, from April 2017 to May 2018. He is currently an Associate Professor with the School of Computer and Software, Nanjing University of Information Science and Technology. His research interests include computer software, big data, tropical cyclone, remote sensing image processing, and deep learning.

WEI HUANG is currnetly pursuing the M.S. degree in software engineering with the Nanjing University of Information Science and Technology, China. His research interests include tropical cyclone and deep learning.



LEI YI is currnetly pursuing the M.S. degree in software engineering with the Nanjing University of Information Science and Technology, China. His research interests include radar rainfall and deep learning.

LIGUANG WU received the Ph.D. degree from the University of Hawaii, in 1999. He is currently a Professor with Fudan University. He worked as an Atmospheric Infrared Sounder (AIRS) data specialist at DAAC/NASA and an Aassociate Research Scientist with GSFC/NASA and the University of Maryland at Baltimore County, from 2000 to 2008. He has published more than 60 articles in refereed journals. His research interests include the formation, intensity and structure change and track of tropical cyclones, and climate change of tropical cyclone activity. He is currently an Associate Editor of Advances in Atmospheric Sciences, an Editor of Acta Meteorologica Sinica, *Journal of Meteorological Science and Advances in Meteorological Science and Technology*.



CHAO WANG received the M.S. degree in atmospheric sciences and the Ph.D. degree in meteorology from the Nanjing University of Information Science and Technology, China, in 2009 and 2016, respectively. He worked as a Research Scholar with the University of Hawaii, USA, from August 2017 to August 2018. He is currently a Lecturer with the School of Atmospheric Sciences, Nanjing University of Information Science and Technology. His research interests include tropical cyclone and climate change.