BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 04/10 pp32–43
Volume 6, Number 1, March 2023
DOI: 10.26599/BDMA.2021.9020017

Deep Convolutional Network Based Machine Intelligence Model for Satellite Cloud Image Classification

Kalyan Kumar Jena, Sourav Kumar Bhoi, Soumya Ranjan Nayak*, Ranjit Panigrahi, and Akash Kumar Bhoi

Abstract: As a huge number of satellites revolve around the earth, a great probability exists to observe and determine the change phenomena on the earth through the analysis of satellite images on a real-time basis. Therefore, classifying satellite images plays strong assistance in remote sensing communities for predicting tropical cyclones. In this article, a classification approach is proposed using Deep Convolutional Neural Network (DCNN), comprising numerous layers, which extract the features through a downsampling process for classifying satellite cloud images. DCNN is trained marvelously on cloud images with an impressive amount of prediction accuracy. Delivery time decreases for testing images, whereas prediction accuracy increases using an appropriate deep convolutional network with a huge number of training dataset instances. The satellite images are taken from the Meteorological & Oceanographic Satellite Data Archival Centre, the organization is responsible for availing satellite cloud images of India and its subcontinent. The proposed cloud image classification shows 94% prediction accuracy with the DCNN framework.

Key words: satellite images; satellite image classification; cyclone prediction; Deep Convolutional Neural Network (DCNN); features; layers; down-sampling process

1 Introduction

Cyclones contain rain and hazardous winds, which can cause much damage to nature, the environment, and human life. The destruction includes floods, fires, water-borne diseases, and communication system disruptions. Approximately 726 million people have been affected by storms from 1998 to 2017[1,2]. A recent study reveals that cyclone is the main cause for at least 10 000 casualties in Odisha in the year 1999[3]. Thus, detecting cyclones will help remote sensing establishments and provide ample scope to plan and tackle such a perilous situation.

Weather forecasting, specifically cyclone detection, often uses numerical threshold[4–6] and statistical techniques[7–9] as detection mechanisms. Many numerical models are often used for weather prediction; one is Weather Research and Forecasting (WRF), that predicts weather conditions on the basis of numerical algorithms. In the data assimilation system, the model covers a wide range of areas with a vast metrological application, which conducts simulation on the basis of the actual atmosphere for prediction. It offers an operational forecasting computationally efficient platform, because it uses parallel processing with two high computing cores. However, the efficiency decreases, as it is used in low-performance computing

• Kalyan Kumar Jena and Sourav Kumar Bhoi are with the Department of Computer Science and Engineering, Parala Maharaja Engineering College, Berhampur 761003, India. E-mail: kalyan.cse@pmec.ac.in; sourav.cse@pmec.ac.in.
• Soumya Ranjan Nayak is with Amity School of Engineering and Technology, Amity University, Uttar Pradesh 201303, India. E-mail: nayak.soumya17@gmail.com.
• Ranjit Panigrahi is with the Department of Computer Applications, Sikkim Manipal Institute of Technology, Sikkim Manipal University, Sikkim 737102, India. E-mail: ranjit.p@smit.smu.edu.in.
• Akash Kumar Bhoi is with the Directorate of Research, Sikkim Manipal University, Gangtok, Sikkim 737102, India. E-mail: akashkrbhoi@gmail.com.
* To whom correspondence should be addressed.
Manuscript received: 2021-07-15; revised: 2021-09-17; accepted: 2021-09-18

systems[10]. Apart from WRF, Numerical Weather Prediction (NWP) is a useful mathematical model for land bodies and water bodies to predict weather phenomena, which focus on current weather conditions. It mainly pays attention to the current observation of weather and predicts further atmospheric situations. However, a problem arises while solving a bunch of equations, which latterly increase time complexity at the time of prediction[11]. The Regional Atmospheric Modeling System is a set of programs that are used for climate research and weather forecast. It is based on NWP, which uses a complex calculation of corner description[12]. This system fails to give precise accuracy due to the less number of vectors[13]. Conditional Nonlinear Optimal Perturbation is a robust forecasting process that provides a higher degree of accuracy, but fails to classify more features in images[14]. Most numerical or threshold methods involve visible and near-infrared bands to detect the presence of cyclones in clouds. Threshold schemes are popular because of their speed of calculation. However, acceptable accuracy is needed, as these schemes also generate significant false positives. Threshold methods are most appropriate for satellite cloud image classification. Many supervised and unsupervised techniques play crucial roles in cloud image classification. Under statistical methods, modern deep learning schemes are widely used to detect cyclone activities, which generate high detection accuracy with little time complexity while classifying satellite cloud images[7, 15–17]. Deep learning methods are proven to be robust, as they provide the best results even with unstructured data[18]. Moreover, no ground knowledge is required to develop state-of-the-art deep learning models. Deep Convolutional Neural Network (DCNN) is built out of a densely interconnected input, output, and hidden layer[19].

Each layer belongs to convolutions which are responsible for detecting image features. Each layer of the detection features of an image is ascertained. The fully connected dense layer finally combines each layer's output to predict each class value. In the case of the satellite image process for cyclone detection, the input of the deep convolutional network is the high-resolution images of satellites, such as INSAT-3D and KALPANA[20, 21]. The network output is either a cyclonic image or a non-cyclonic image. CNN is trained to observe the feature of the cyclone and the hidden layer is responsible for showing the output. Each hidden layer carries a single-valued output. These hidden layer outputs are inputted into the next layer. The DCNN planning is outlined for classifying images. Therefore, the center of the model objectives is to build a model that can classify cyclone occurrences efficiently. It uses the deep convolutional approach, which has been trained and validated. The model checks each test image with high accuracy and less time complexity, or the testing is conducted in a real-time scenario. A model that achieves an accuracy of more than 90% can practically help the remote sensing work, which can be used for further evaluation or climate prediction.

This article discusses various related works for climate/weather prediction in Section 2. The proposed model is elaborately described in Section 3. The result is analyzed in Section 4. The conclusion is presented in Section 5.

2 Related Work

Deep learning plays a crucial role in weather forecasting, especially in cyclone prediction. Many state-of-the-art machine learning and deep learning based approaches have been proposed for accurate estimations of the presence of cyclones. Recently, a Dichotomous Logistic Regression (DLR) based on a fuzzy hypergraph model has been proposed using the deep convolutional network for classifying cyclone images. The model reveals an acceptable detection accuracy with less time complexity[7]. Landsat 8 OLI Satellite Image Classification[22] is another approach of CNN to identify natural disasters. Meanwhile, Brovey Transformation helps in fusing Red-Green-Blue (RGB) in panchromatic band shaving spatial resolutions. The deep learning approach for detecting tropical cyclones and their precursors[8] in a simulation, which employs a cloud resolving global non-hydrostatic model, uses 50 000 tropical cyclone images to train two deep convolutional models for binary classification. The model shows a 90% probability of prediction with 10%–30% false alarm. Another deep learning model DeepMicroNet[23] estimates tropical cyclone intensity by using a satellite passive microwave image that involves 85 GHz–90 GHz satellite images to provide a probable estimation of a tropical cyclone. Rotation-blended CNN[24] has been proposed using an open dataset for classifying tropical cyclone intensity on a convolutional model. It provides a scope for the model to detect cyclone intensity swiftly with promising time complexity. A convolutional model is also explored for classifying cyclone intensity, which ultimately eradicates the complexity that arises during the feature extraction process, required for determining the strength of the tropical cyclone[25]. In addition, a series of Artificial Neural Network (ANN) layers[17] have been used to predict cyclone occurrences with 98% detection accuracy by using NOAA-AVHRR satellite images. Tropical cyclone intensity detection through geometric features of cyclone images with the help of multilayer perception is proven to be a great predictor with 84% detection accuracy[26]. ANN approach[27] also has a high potential for modeling rainfall due to typhoons in Taiwan, China. The approach considerably helps in controlling the flooding disaster significantly through the prediction result. The model uses 27 years of data to train the neural network, which yields an accuracy of 96%. In a similar guideline, the occurrence of a typhoon in Taiwan, China, has been proposed using ANN with the help of six statistical measures[28], namely, mean absolute error, root mean square error, coefficient of correlation, error of time to peak discharge, error of peak discharge, and coefficient of efficiency. The model contains the hydrologic modeling system with an ANN to predict the presence of typhoons. Moreover, a forecast model for runner storm surge on Tottori Coast, Japan, using ANN has been proposed[29]. It is a real-time forecasting model for surge predictions. The prediction process involves finding an optimal dataset to train ANN using meteorological and hydrodynamic parameters from Tottori Coast, Japan. Similarly, BackPropagation Neural Network (BPNN) for tropical cyclone track forecasting, the classic backpropagation algorithm, helps track and forecast cyclones with the help of the historical data of tropical cyclones (longitude and latitude)[30]. Due to the complicated nonlinear physical mechanism of the problem, the backpropagation gives a degree of ease for solving the problem. It checks the climatology persistence of the past motion of a storm. A graph convolutional network, a deep neural network known as graph convolutional Long Short-Term Memory (LSTM) Network that works on a picture taken by satellite Himawari-8 between years 2010 and 2019[31], processes irregularities in images where LSTM learns different features from the images taken by the satellite over a period. A Doppler Weather Radar (DWR) based cyclone forecasting model shows new information on cyclone origination[32]. The work also describes the application of many influenced algorithms in the Indian radar data, which helps derive more information about cyclone structure. Using a radar algorithm, DWR data are implemented on next generation radar. The work suggests improvement in the strategies for India's data collection of cyclones. The detection of aerial images for the classification of cyclone disasters with Convolutional Neural Network (CNN) uses satellite and aerial images of disasters (landslides and floods) to train the network[33]. The model mainly focuses on two countries, namely Thailand and Japan. It uses 10 000 images for the training dataset and classifies the disaster region and disaster category that shows an accuracy in the range of 80%–90%. Neural networks are trained with rainfall and radiosonde data to predict the streamflow and flash floods, where neural networks play a significant role in flood prediction[34].

Apart from deep learning, other machine learning techniques are also dominant in the field of weather forecasting to predict the presence of cyclones. In Ref. [35], a machine learning approach for cyclone prediction uses decision trees, random forests, and support vector machines. The research is based on a linear discriminant analysis for detecting tropical cyclones, proving to be a great option for cyclone prediction. This model shows an accuracy of 90%–94%, although it shows 20%–28% false alarm. In another study, the intensity analysis of satellite cloud images of tropical cyclones is performed to extract the features of edges from cloud images on a day-by-day basis. It helps identify the further intensity of tropical cyclones from the ongoing data[36].

3 Proposed Model

Considering the original dataset containing satellite images to be I, the number of images is N, and the training images are $I_1, I_2, \ldots, I_N$. The processing of image dataset I includes reading and loading of the images from each directory, resizing them as per the input requirement of the deep convolutional network (as shown in Fig. 1) and labeling each instance of images, whether the concerned image is a cyclone or not. The training images $I_1, I_2, \ldots, I_N$ are in the form of RGB (color image); thus, they are now represented by an array where each index holds the intensity of an image pixel. Three 2D arrays are defined (R-Red, B-Blue, and G-Green) to be precise. The intensity value of the respective color is stored in their corresponding indexes. As the data are ready, they must pass through a deep convolutional model to train the network, shown in Fig. 2.

3.1 Dataset

The dataset considered here has 4947 training images, out of which 2885 images belong to the "cyclone"

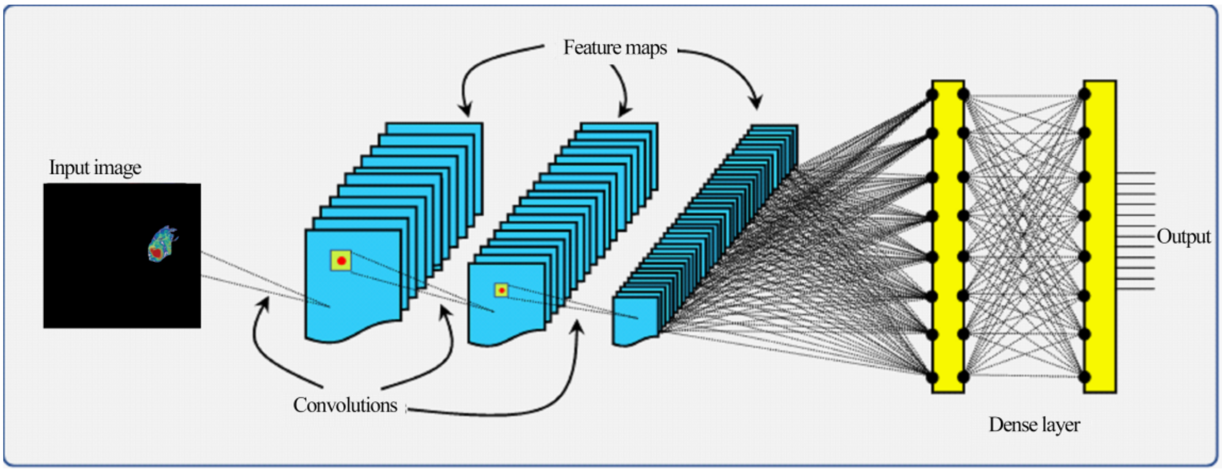

Fig. 1 CNN model.

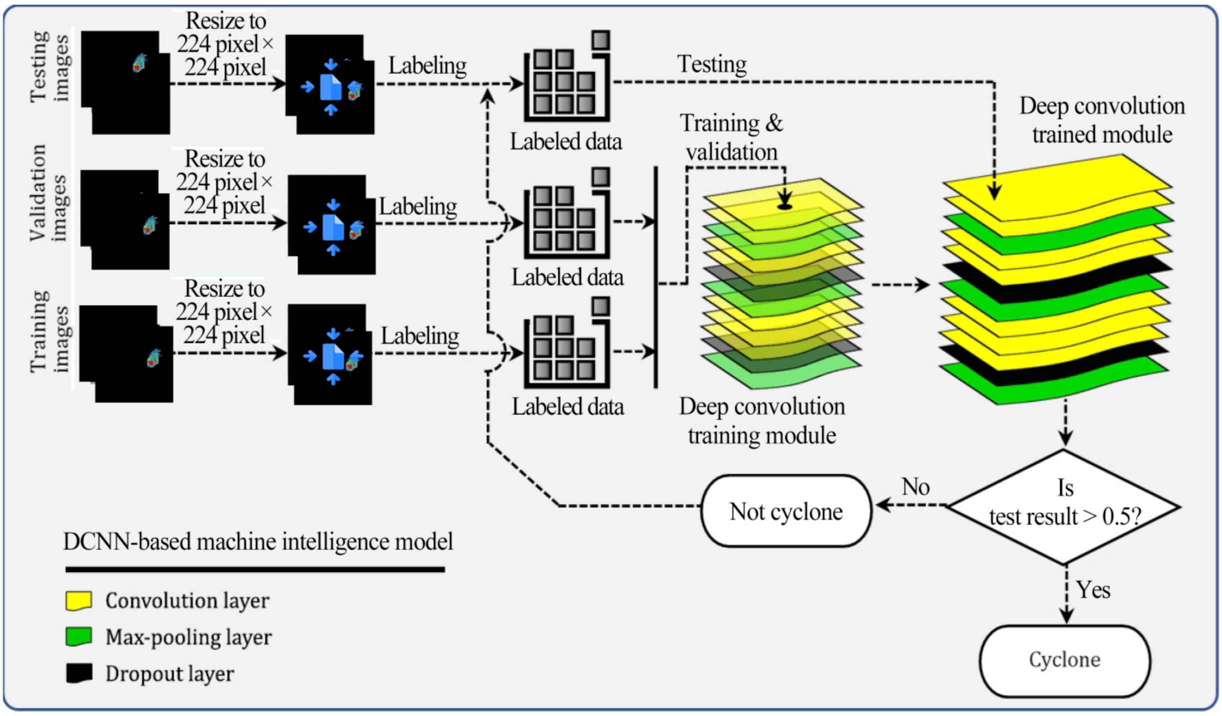

Fig. 2 DCNN-based machine intelligence model for cyclone detection.

category, whereas 2062 belong to the "non-cyclone" category. The dataset also has 1008 testing images (408 cyclone and 600 non-cyclone) and 2481 validation images (1549 cyclone and 932 non-cyclone). All images are retrieved from the Meteorological & Oceanographic Satellite Data Archival Centre[20,21].

3.2 Deep convolutional network

Solving the problem of image classification through a CNN (ConvNet/CNN) gives high accuracy in the prediction of images[37,38].

An accurate prediction of cyclones reduces the chances of casualties of human lives. A CNN contains many hidden layers. It is an algorithm, which takes an image of a certain class as input and trains the hyperparameters, or to be more precise, optimizes the biases and weights of the respective nodes of the network. This factor uniquely defines each aspect of the image, such as features and curves. This process helps differentiate an image from another. Therefore, CNN is a collection of a convolutional layer, an activation layer (which is followed by a convolutional layer), and a max-pooling layer (which extracts the features). The model uses seven convolutional layers broken into three segments, each separated by a max-pooling layer, which executes a down-sampling process. This process extracts the features and reduces the vectors of the neural network it is dealing with. As described above, a convolutional layer is followed by an activation layer, which performs with a Rectified Linear Unit (ReLU) and SoftMax. The transformation by ReLU is to convolve features, implying the entwinement or binding up of the features. It also provides nonlinearity to the model[39, 40]. Thus, the ReLU function can be defined as follows:

$$F(t) = \max(0, t) \tag{1}$$

Function $F(t)$ returns t for all values where $t > 0$, otherwise 0. The model converts the image into a down-sampled feature matrix for the multi-perception level, flattening the feature matrix in a vector where the vector is fed to a feed-forward neural network. Last, the perceptron classifies the testing image by using an activation function called SoftMax,

$$f_j(z) = \frac{\theta^{z_j}}{\sum_{k=1}^{M} \theta^{z_k}} \tag{2}$$

$f_j(z)$ is the output of SoftMax, z is the input vector, θ^{z_j} is the standard exponential function for input vectors, M is the number of classes in the multi-class classification scenario, and θ^{z_k} represents the standard exponential function for the output vector.

When the values come from the network, they pass through the SoftMax function or called the generalization of a logistic function, which squashes the input vector into a range of [0, 1]; thus, an image is given a real-value probability of prediction[40, 41]. The model states that taking inputs as training data (which are previously classified) and assigning them to a deep convolutional network signal the start of learning the weights for the feature maps (map obtained from each 2D convolutional network). Backpropagation helps the network to learn all the feature maps or hyperparameters, such as the number of nodes per layer of the neural network[42]. Considering the training model into a function $S(\cdot)$ and inputting "T_S", where $s = |T|$ represents the total number of testing images, which ultimately classify whether the concern instance has sufficient probability of a cyclone or not. Algorithm 1 presents the algorithm for the model.

Algorithm 1 Proposed classification algorithm for satellite image classification

Input: $I = I_1, I_2, \ldots, I_r$, set of training images
  $V = V_1, V_2, \ldots, V_q$, set of validation images
  $T = T_1, T_2, \ldots, T_s$, set of testing images
Output: Prediction labels, L (cyclone or not_cyclone)
Begin
Step 1: Resize training, testing, and validation images
  for $i = 0$ //training images to r **do**
    $T_i = Resize\ (224 \times 224, T_i)$
    for $t = 0$ //validation images to q **do**
      $V_i = Resize(224 \times 224, V_i)$
      for $t = 0$ //testing images to s **do**
        $T_i = Resize(224 \times 224, T_i)$
      end for
    end for
  end for
Step 2: Convert all resized images to feature matrix using ReLU

$$\begin{cases} F(\alpha_I) = \max(0, \alpha); \\ F(\alpha_V) = \max(0, \alpha); \quad \forall I, \forall V, \forall T \\ F(\alpha_T) = \max(0, \alpha) \end{cases}$$

Step 3: Train and validate the neural network model using $F(\alpha)_I$ and $F(\alpha)_V$
Step 4: Save the model as $S(\cdot)$ for final testing
Step 5: Classify the testing instances
  for $i = 0$ //testing images to s **do**
    if $S(T_i) \geqslant 0.5$ **then**
      L = cyclone
    else
      L = not_cyclone
    end if
  end for
  return L
End

4 Experimental Evaluation

The deep convolutional network has been implemented on Python with the help of keras API as the front end and TensorFlow as the backend. The keras and TensorFlow provide ease of implementing the deep convolutional network, as it contains predefined layers, models, and optimizers[19]. The implementation platform is GPU NVIDIA MX250, type GDDR5, and has 1518 MHz and a RAM of 8 GB, which enhances the speed of training the model on a 64-bit Windows platform. A sequential model, which contains layers (Conv2D, Max-pooling, Flattening, Dropout) with

the input of 224 pixel× 224 pixel image, is illustrated in Fig. 3. All training and validation images after processing (loading, resizing, and labeling) pass through it. The models are trained and validated in every epoch (once a total dataset passes through forward and backward in the network or sequential model). Some sample images for cyclone and non-cyclone are displayed in Figs. 4 and 5, respectively. An image of 224 pixel×224 pixel is given as the input to the sequential model where two convolutional 2D networks are applied, taking 64 filters, defining the kernel of 3×3, and keeping

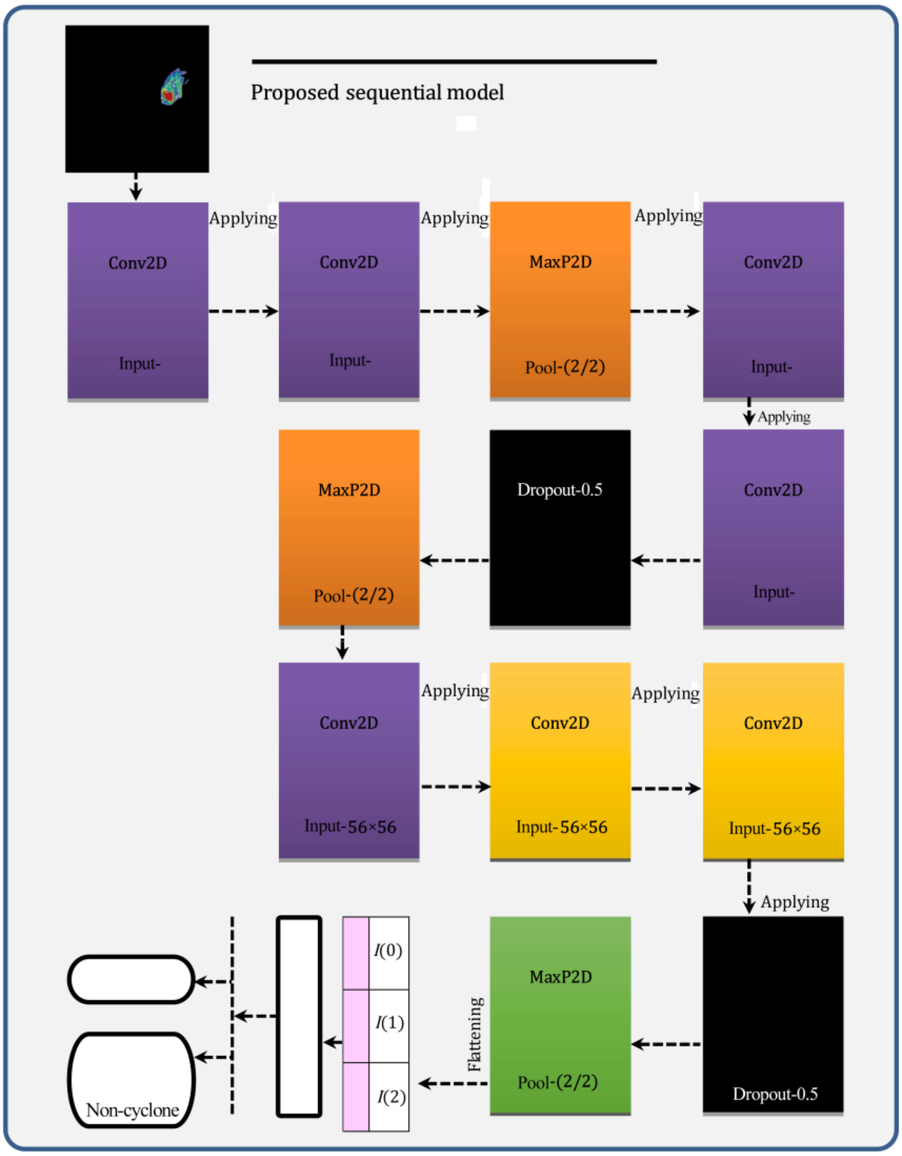

Fig. 3 Proposed sequential model using ReLU.

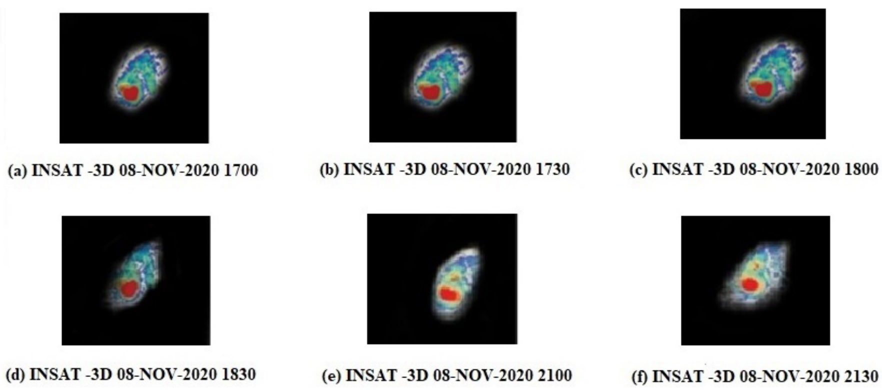

Fig. 4 Bulbul cyclone images in the dataset (sample images)[20,21].

the padding the "same". Each convolutional feature extraction takes place in a feature matrix, and the matrix is followed to the next layer for further processing.

As previously mentioned, convolution is applied to three segments, and the filters double in each segment. Therefore, feature detection gives high accuracy to

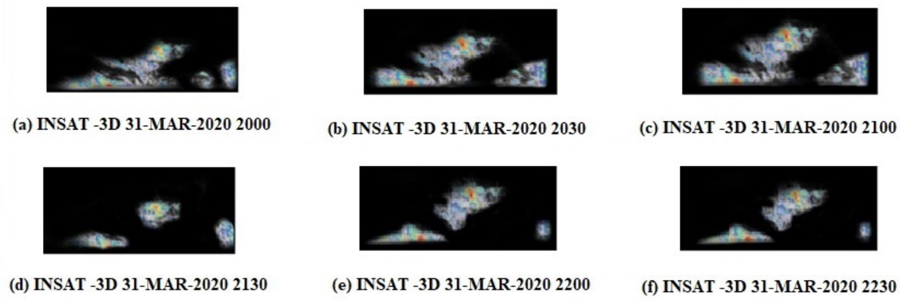

Fig. 5 Normal images in the dataset (sample images)[20,21].

predict the testing images. Dropout is mainly introduced for avoiding overfitting. The model performance is evaluated using a confusion matrix[43,44].

As mentioned above, the model starts with an input image where the first convolutional layer extracts 64 feature matrices taking a kernel size of 3×3. The input of the second convolutional layer is the output of the first convolutional layer. Here the convolution again extracts 64 features from the feature matrices of the previous layer, taking a kernel of 3×3. Down-sampling images is the main objective of the CNN model, and therefore, a MaxPool2D, which is also called a max-pooling layer, reduces images without reducing their features. The max-pooling layer retrieves all the features into a feature map/matrix. The pool size of 2×2 reduces the feature matrices to half of the original size. The output of the max-pooling layer is the input of the third convolutional layer. In the second segment of the model, which contains the third and fourth convolutional layers, 128 feature matrices are extracted, followed by a dropout of 0.5, which prevents the model from overfitting. The second max-pooling layer again down-samples the feature matrices into half of the previous size by using a pool size of 2×2. The third segment of the model, which contains three convolutional layers with 256 feature matrices each and a kernel size of 3×3, again extracts the appropriate features for classifying the problem. The max-pooling layer again comes into the role of down-sampling. The input feature matrices, which are the outputs of the previous convolutional layer, are finally reduced into many feature matrices. The model at this phase deals with the matrices of size 28×28 and 256 feature matrices or filters. Thus, converting these 2D arrays or matrices into a single vector is carried by the flattening layer whose output is 28×28×256, which is equal to 200 704. The model now deals with 200 704 parameters, which are being distinguished by a dense layer of two units, as the model is a binary classifier. The SoftMax classifies the parameters into a probability of a cyclone or not. The epochs "E_p" defined in this article are 10, and the batch size is 10. The percentages of accuracy, loss, validation loss, and validation accuracy are presented in Table 1.

We can also define model loss and accuracy with reference to the image shown in Fig. 6. Here, accuracy and loss are optimal with respect to per epoch. From the first epoch, the training data show a condition called underfitting. However, while the epoch increases, the training data tend to fit the model. In Ep-1, the accuracy is nearly 93%, and the loss is considered nearly 8%. It suggests that during the first batch of the dataset with 494 images, 93% are trained accurately; meanwhile, 7% are not trained accurately. In Ep-2, the accuracy is approximately 98%, and the loss percentage is approximately 2.0. The model has significantly trained many images accurately from the batch size. As soon as the model is showing a good result at a point exactly at Ep-8, the model starts to overfit and thus signifies that the training accuracy is more, whereas the validation accuracy is less. However, in Ep-9, the accuracy reduces to 1%, and the model perfectly fits. In the last epoch, the accuracy level of model training is nearly 98%, and the

Table 1 Epochs and corresponding results while training and validating the proposed model.

(%)

Epochs E_p	Training loss	Training accuracy	Validation loss	Validation accuracy
Ep-1/10	86.64	93.72	11.72	99.15
Ep-2/10	3.86	98.80	10.64	98.95
Ep-3/10	3.13	99.19	4.15	99.84
Ep-4/10	2.47	99.39	4.81	99.96
Ep-5/10	2.58	99.43	12.56	98.43
Ep-6/10	1.66	99.51	2.37	100.00
Ep-7/10	0.35	99.90	1.93	100.00
Ep-8/10	0.00	100.00	1.55	100.00
Ep-9/10	0.01	100.00	1.09	100.00
Ep-10/10	13.68	98.74	7.07	99.80
Average	11.44	98.87	5.79	99.61
Standard deviation	26.72	1.86	4.43	0.56

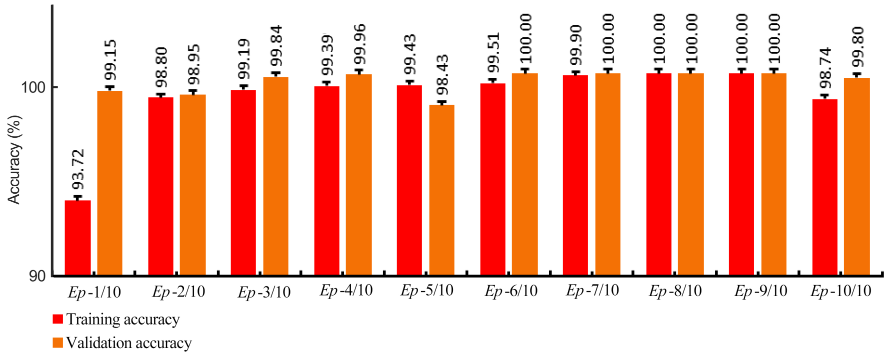

Fig. 6 Training and validation accuracy with respect to per epoch.

accuracy level of validation is 99%. With these results, we can easily confirm that the model fits the data. The underfit problem is also eradicated.

Figure 6 displays the graph between validation accuracy and training accuracy. The deviation between the two lines is evident at the beginning of epochs. However, the deep neural network during the training increases its accuracy, and the deviation between the two lines decreases. The lines shown in Fig. 6 tend to merge at the ninth epoch, justifying the fitting of the proposed model to the training and validating datasets.

Now, the confusion matrix helps us determine the accuracy of the total model. The formula of the confusion matrix is determined in Eq. (3). It can be defined by the ratio of the addition of true positive and false positive with respect to the total number of instances. The total number of testing images is 1008, out of which 408 are random cyclone images of AMPHAN and OCKHI. These images are not considered before in the training or validation dataset, and the model has predicted 352 images to be cyclones[20,21]. Thus, it has also predicted 587 images to be non-cyclones from 600 images. The accuracy can be defined by the following formula:

$$\text{Accuracy} = \frac{\alpha + \beta}{\alpha + \beta + \alpha' + \beta'} \tag{3}$$

where α is denoted as a predicted "cyclone" of a cyclone class, α' is denoted as predicted "not a cyclone" but belongs to the cyclone class, β is denoted as predicted "not a cyclone" of the no_cyclone class, β' is denoted as a predicted "cyclone" which belongs to the no_cyclone class. Putting the values of $\alpha = 352$, $\beta = 597$, $\alpha = 56$, and $\beta' = 3$, the model obtains an accuracy of 94.14%, and an error rate of less than 6%. The confusion matrix of the proposed model is illustrated in Fig. 7. The corresponding experimental results are outlined in Table 2.

From the summarized results in Table 2, the proposed model shows an appealing detection accuracy of 94.2%. The model reveals not only detection accuracy but also a sensitivity of 86.3%, which appears to be unique in the field of cyclone detection. Another point of observation is that the false positive rate is quite impressively low, with only 0.005. It shows the real power of our proposed system, as it seems to be stable in all the performance measures.

In the final stage of analysis, we compare our proposed cyclone detection model with other state-of-the-art cyclone detection models. Given that our model is based on a deep learning technique, we shortlist a

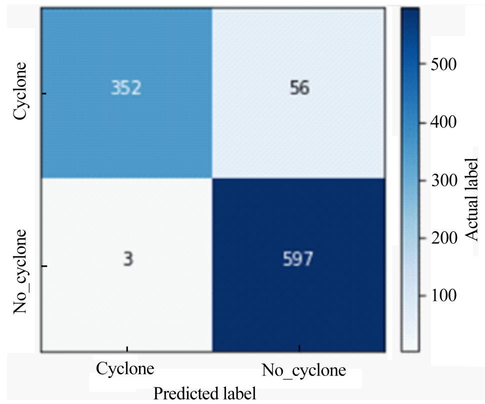

Fig. 7 Confusion matrix of the proposed model.

Table 2 Testing results obtained through the proposed model.

Result	Value
Sensitivity	86.3%
Specificity	0.995
Precision	0.992
Negative predictive value	0.914
False positive rate	0.005
False discovery rate	0.009
False negative rate	0.137
Accuracy	94.2%
F1 score	0.923
Matthews correlation coefficient	0.881

few deep learning based models for comparison in Table 3. In this regard, DLR–FH, Deep CNN, and DAV–T cyclone detection models implemented by Rajesh et al.[7], Deep CNN (2D) model proposed by Matsuoka et al.[8], Multilayer CNN proposed by Kovordányi and Roy[17], and PCA-based CNN model proposed by Rai et al.[22] are considered. The proposed model exhibits at par performance with other models in terms of detection accuracy. The proposed multilayer CNN model shows 12.20% and 5.1% better detection accuracy than the deep learning approach proposed by Matsuoka et al.[8] and the DAV-T approach implemented by Rajesh et al.[7], respectively. Our system also slightly lacks less than DLR-FH, Multilayer CNN, and PCA-CNN approaches.

5 Conclusion

The classification of cloud images is important for the prediction of the presence of cyclones. Satellite cloud images are proven to be an ideal option for analyzing irregular cyclone movements. These images are used for training and testing sets of the deep convolutional network. Satellite remote sensing can be achieved easily by using the deep convolutional network to provide complete observation and monitoring of the global cyclone activity, which gives a real-time measurement of the accuracy of cyclone conditions. A deep convolutional network not only classifies satellite cloud images but also predicts the presence of cyclones. The confusion matrix with an accuracy of the prediction model shows a detection accuracy of 94% and an error of less than 6%. The degree to which accuracy improves is determined by the machine and the model on which it is used. Using a high-performance computing machine results in a better image resolution, which may improve accuracy even further. The enormous size of the training images causes the training time to be delayed. To improve the processing speed, CNN demands tiny image sizes. To cope with the CNN environment, images of 997 pixel $\times$ 969 pixel size are reduced to 224 pixel $\times$ 224 pixel size as a common practice. It works well in the case of ordinary photographs where important image features remain preserved after size reduction. However, the standard practice of reducing images for CNN may not work well for satellite images in predicting the presence of cyclones. The reason is that the cyclone periphery is usually not limited to few meters; instead, the periphery covers many kilometers of geographical area. In such a case, reducing the image size for CNN may destroy the tiny details about a cyclone in the cloud. Hence, prediction about the presence of cyclones may not always be attained precisely, remaining as the limitation of this article. Numerous current CNN-based techniques, which work directly on satellite images without compromising image attributes or processing speed, may be investigated in the future.

Table 3 Comparison of the proposed model with other state-of-the-art models.

Author	Method	Accuracy (%)
Rajesh et al.[7]	DLR–FH	96.00
	Deep CNN	90.00
	DAV–T	82.00
Matsuoka et al.[8]	Deep CNN (2D)	89.10
Kovordányi and Roy[17]	Multilayer CNN	98.70
Rai et al.[22]	Principal Component Analysis-CNN (PCA-CNN)	94.50
Proposed model	Multilayer CNN	94.20

Acknowledgment

We want to thank Parala Maharaja Engineering College (Govt.), Berhampur, India for providing adequate facility and infrastructure for conducting this major project research work. For this work, we also want to thank Soumya Ranjan Jena, Major Project Group Leader, for supporting this work under our guidance (Kalyan Kumar Jena- Guide and Sourav Kumar Bhoi- Co-Guide).

References

[1] M. Mizutori and D. Guha-Sapir, *Economic Losses*, *Poverty and Disasters 1998-2017*. United Nations Office for Disaster Risk Reduction, United Nations, New York, USA, 2017.

[2] P. Wallemacq and H. Rowena, *Economic Losses*, *Poverty & Disasters*: *1998–2017*. Brussels, Belgium: Centre for Research on the Epidemiology of Disasters, 2018.

[3] F. Thomalla and H. Schmuck, 'We all knew that a cyclone was coming': Disaster preparedness and the cyclone of 1999 in Orissa, India, *Disasters*, vol. 28, no. 4, pp. 373–387, 2004.

[4] L. Z. Jiang, S. M. Fu, and J. H. Sun, New method for detecting extratropical cyclones: The eight-section slope detecting method, *Atmos. Ocean. Sci. Lett.*, vol. 13, no. 5, pp. 436–442, 2020.

[5] U. Neu, M. G. Akperov, N. Bellenbaum, R. Benestad, R. Blender, R. Caballero, A. Cocozza, H. F. Dacre, Y. Feng, K. Fraedrich, et al., IMILAST: A community effort to intercompare extratropical cyclone detection and tracking algorithms, *Bull. Am. Meteorol. Soc.*, vol. 94, no. 4, pp. 529–547, 2013.

[6] A. Murata, S. I. I. Watanabe, H. Sasaki, H. Kawase, and M. Nosaka, The development of a resolution-independent

tropical cyclone detection scheme for high-resolution climate model simulations, *J. Meteorol. Soc. Japan. Ser. II*, vol. 97, no. 2, pp. 519–531, 2019.

[7] K. Rajesh, V. Ramaswamy, K. Kannan, and N. Arunkumar, Satellite cloud image classification for cyclone prediction using dichotomous logistic regression based fuzzy hypergraph model, *Futur. Gener. Comput. Syst.*, vol. 98, pp. 688–696, 2019.

[8] D. Matsuoka, M. Nakano, D. Sugiyama, and S. Uchida, Deep learning approach for detecting tropical cyclones and their precursors in the simulation by a cloud-resolving global nonhydrostatic atmospheric model, *Prog. Earth Planet. Sci.*, vol. 5, no. 1, p. 80, 2018.

[9] S. Shakya, S. Kumar, and M. Goswami, Deep learning algorithm for satellite imaging based cyclone detection, *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 13, pp. 827–839, 2020.

[10] J. G. Powers, J. B. Klemp, W. C. Skamarock, C. A. Davis, J. Dudhia, D. O. Gill, J. L. Coen, D. J. Gochis, R. Ahmadov, S. E. Peckham, et al., The weather research and forecasting model: Overview, system efforts, and future directions, *Bull. Am. Meteorol. Soc.*, vol. 98, no. 8, pp. 1717–1737, 2017.

[11] A. C. Lorenc, Analysis methods for numerical weather prediction, *Q.J.R. Meteorol. Soc.*, vol. 112, no. 474, pp. 1177–1194, 1986.

[12] G. E. Liston and R. A. Pielke, A climate version of the regional atmospheric modeling system, *Theor. Appl. Climatol.*, vol. 66, nos. 1&2, pp. 29–47, 2000.

[13] R. A. Pielke, W. R. Cotton, R. L. Walko, C. J. Tremback, W. A. Lyons, L. D. Grasso, M. E. Nicholls, M. D. Moran, D. A. Wesley, T. J. Lee, et al., A comprehensive meteorological modeling system-RAMS, *Meteorol. Atmos. Phys.*, vol. 49, nos. 1–4, pp. 69–91, 1992.

[14] M. Mu, W. S. Duan, and B. Wang, Conditional nonlinear optimal perturbation and its applications, *Nonlin. Process. Geophys.*, vol. 10, no. 6, pp. 493–501, 2003.

[15] J. J. Baik and H. S. Hwang, Tropical cyclone intensity prediction using regression method and neural network, *J. Meteorol. Soc. Japan. Ser. II*, vol. 76, no. 5, pp. 711–717, 1998.

[16] M. Rütgers, S. Lee, and D. You, Typhoon track prediction using satellite images in a generative adversarial network, arXiv preprint arXiv: 1808.05382, 2018.

[17] R. Kovordányi and C. Roy, Cyclone track forecasting based on satellite images using artificial neural networks, *ISPRS J. Photogramm. Remote Sens.*, vol. 64, no. 6, pp. 513–521, 2009.

[18] T. J. Sejnowski, *The Deep Learning Revolution*. Cambridge, MA, USA: MIT Press, 2018, pp. 1–10.

[19] H. J. Yoo, Deep convolution neural networks in computer vision-a review, *IEIE Trans. Smart Process. Comput.*, vol. 4, no. 1, pp. 35–43, 2015.

[20] Meteorological & Oceanographic Satellite Data Archival Centre, Goverment of India, MOSDAC Data, https://www.mosdac.gov.in/satellite-catalog, 2013.

[21] J. Salat, J. Pascual, M. Flexas, T. M. Chin, and J. Vazquez-Cuervo, Forty-five years of oceanographic and meteorological observations at a coastal station in the NW Mediterranean: A ground truth for satellite observations, Ocean Dynamics, vol. 69, no. 9, pp. 1067–1084, 2019.

[22] A. K. Rai, N. Mandal, A. Singh, and K. K. Singh, Landsat 8 OLI satellite image classification using convolutional neural network, *Procedia Comput. Sci.*, vol. 167, pp. 987–993, 2020.

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

[24] 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. Knowledge Discovery & Data Mining*, London, UK, 2018, pp. 90–99.

[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 *2018 24th Int. Conf. Pattern Recognition* (*ICPR*), Beijing, China, 2018, pp. 1474–1480.

[26] C. Kar, A. Kumar, and S. Banerjee, Tropical cyclone intensity detection by geometric features of cyclone images and multilayer perceptron, *SN Appl. Sci.*, vol. 1, no. 9, p. 1099, 2019.

[27] S. M. Chen, Y. M. Wang, and I. Tsou, Using artificial neural network approach for modelling rainfall-runoff due to typhoon, *J. Earth Syst. Sci.*, vol. 122, no. 2, pp. 399–405, 2013.

[28] C. C. Young and W. C. Liu, Prediction and modelling of rainfall-runoff during typhoon events using a physically-based and artificial neural network hybrid model, *Hydrol. Sci.J.*, vol. 60, no. 12, pp. 2102–2116, 2015.

[29] S. Kim, Y. Matsumi, S. Q. Pan, and H. Mase, A real-time forecast model using artificial neural network for after-runner storm surges on the Tottori coast, Japan, *Ocean Eng.*, vol. 122, pp. 44–53, 2016.

[30] Y. F. Wang, W. Zhang, and W. Fu, Back Propogation(BP)-neural network for tropical cyclone track forecast, in *Proc. 2011 19th Int. Conf. Geoinformatics*, Shanghai, China, 2011, pp. 1–4.

[31] J. Y. Zhou, J. Xiang, and S. X. Huang, Classification and prediction of typhoon levels by satellite cloud pictures through GC-LSTM deep learning model, *Sensors*, vol. 20, no. 18, p. 5132, 2020.

[32] S. S. Roy, V. Lakshmanan, S. K. R. Bhowmik, and S. B. Thampi, Doppler weather radar based nowcasting of cyclone Ogni, *J. Earth Syst. Sci.*, vol. 119, no. 2, pp. 183–199, 2010.

[33] S. N. K. B. Amit and Y. Aoki, Disaster detection from aerial imagery with convolutional neural network, in *Proc. 2017 Int. Electronics Symp. Knowledge Creation and Intelligent Computing* (*IES-KCIC*), Surabaya, Indonesia, 2017, pp. 239–245.

[34] G. Kim and A. P. Barros, Quantitative flood forecasting using multisensor data and neural networks, *J. Hydrol.*, vol. 246, nos. 1–4, pp. 45–62, 2001.

[35] M. Kim, M. S. Park, J. Im, S. Park, and M. I. Lee, Machine learning approaches for detecting tropical cyclone formation using satellite data, *Remote Sens.*, vol. 11, no. 10, p. 1195, 2019.

[36] V. F. Dvorak, Tropical cyclone intensity analysis and forecasting from satellite imagery, *Mon. Wea. Rev.*, vol. 103, no. 5, pp. 420–430, 1975.
[37] B. B. Traore, B. Kamsu-Foguem, and F. Tangara, Deep convolution neural network for image recognition, *Ecol. Inform.*, vol. 48, pp. 257–268, 2018.
[38] Y. LeCun, LeNet-5, convolutional neural networks, http://yann.lecun.com/exdb/lenet, 2015.
[39] A. F. Agarap, Deep learning using rectified linear units (ReLU), arXiv Preprint arXiv: 1803.08375, 2018.
[40] S. Sharma, S. Sharma, and A. Athaiya, Activation functions in neural networks, *Int.J . Eng. Appl. Sci. Technol.*, vol. 4, no. 12, pp. 310–316, 2020.
[41] I. Kouretas and V. Paliouras, Simplified hardware implementation of the softmax activation function, in *Proc. 2019 8^{th} Int. Conf. Modern Circuits and Systems Technologies* (*MOCAST*), Thessaloniki, Greece, 2019, pp. 1–4.
[42] R. Rojas, *Neural Networks-A Systematic Introduction*. Berlin, Germany: Springer, 1996, pp. 149–182.
[43] X. Y. Deng, Q. Liu, Y. Deng, and S. Mahadevan, An improved method to construct basic probability assignment based on the confusion matrix for classification problem, *Inf. Sci.*, vols. 340&341, pp. 250–261, 2016.
[44] R. Susmaga, Confusion matrix visualization, in *Intelligent Information Processing and Web Mining*, M. A. Kłpotek, S. T. Wierzchoń K. Trojanowski, eds. Berlin, Germany: Springer, 2004, pp. 107–116.

Soumya Ranjan Nayak is an assistant professor at Amity School of Engineering and Technology, Amity University, Noida, India. He received the PhD degree in computer science and engineering under MHRD Govt. of India fellowship from CET, BPUT Rourkela, India. He has published over 80 articles in peer-reviewed journals and conferences of international repute, such as Elsevier, Springer, World Scientific, IOS Press, Taylor & Francis, Inderscience, and IGI Global. Apart from that, he published 12 book chapters, 6 books, and 4 Indian patents (2 patents granted) and two International patents (two patents granted) under his credit. His current research interests include medical image analysis and classification, machine learning, deep learning, pattern recognition, fractal graphics, and computer vision. His publications have more than 500 citations with h-index of 15, and i10 index of 19 (Google Scholar). He serves as a reviewer of many peer-reviewed journals, such as *IEEE Access*, *Applied Mathematics and Computation*, *Journal of Applied Remote Sensing*, *Mathematical Problems in Engineering*, *International Journal of Light and Electron optics*, *Journal of Intelligent and Fuzzy Systems*, *Future Generation Computer Systems*, and *Pattern Recognition Letters*. He has also served as Technical Program Committee Member of several conferences of international repute.

Sourav Kumar Bhoi received the PhD and MEng degrees from the National Institute of Technology (NIT), Rourkela, India in 2017 and 2013, respectively. He is currently an assistant professor at the Department of CSE, PMEC (Govt.), Berhampur, India. His research interests include IoTs, edge and fog computing, ML/DL, VANETs, SDNs, and information security. He received the IET Premium Award-2016 from IET Networks. He has published more than 100 research papers in reputed international journals and conferences, such as IEEE, Elsevier, Springer, Hindawi, and Wiley.

Kalyan Kumar Jena received the PhD degree in computer science engineering from Utkal University, Odisha, India (Nodal Centre -IGIT, Sarang), and the BEng and MEng degrees in computer science & engineering from BPUT, Odisha, India. He is working as an assistant professor at the Department of Computer Science and Engineering (CSE), Parala Maharaja Engineering College (PMEC, Govt.), Berhampur, India. He has more than 7 years of research, as well as teaching experience. His research interests include image and video processing, ML/DL, IoT, WSN, UAV, and fractal and parallel algorithms. His publications comprise more than 50 papers in national/international journals and conferences. He acts as an editorial board member, technical program committee member, and reviewer of various national/international journals and conferences. He acts as a member of several professional bodies and invited speaker to deliver several talks. He received Sadananda Memorial Award (2020) in research, Bhubananda Das Award (Gold Medal recipient 2012–2013), Bidyalaya Gaurav Award (2004), and several prizes due to his greater achievements.

Ranjit Panigrahi received the MEng degree in computer sciences and engineering from Sikkim Manipal Institute of Technology, and the PhD degree in computer applications from Sikkim Manipal University, India. His research interests include machine learning, pattern recognition, and wireless sensor networks. He actively involves in various national and international conferences of repute. He serves as a member of technical review committee for various international journals of Inderscience and Springer Nature. At present he is deputed as an assistant professor at the Department of Computer Applications, Sikkim Manipal University.

Akash Kumar Bhoi has been working at the Directorate of Research, Sikkim Manipal Institute of Technology (SMIT), India since 2012. He is appointed as the honorary title of "Adjunct Fellow" Institute for Sustainable Industries & Liveable Cities (ISILC), Victoria University, Melbourne, Australia for the period from 1 August 2021 to 31 July 2022. He is also working (20th Jan 2021–19th Jan 2022) as a research associate at Wireless Networks (WN) Research Laboratory, Institute of Information Science and Technologies, National Research Council (ISTI-CRN) Pisa, Italy. He is a university PhD course coordinator for "Research & Publication Ethics (RPE)". He is a member of IEEE, ISEIS, and IAENG, an associate member of IEI and UACEE, and an editorial board member reviewer of Indian and international journals. He is also a regular reviewer of repute journals, namely IEEE, Springer, Elsevier, Taylor and Francis, Inderscience, etc. His research areas are biomedical technologies, IoT, computational intelligence, and antenna renewable energy. He has published several papers in national and international journals and conferences. He has 100+ documents registered in the Scopus database by the year 2020. He has also served on numerous organizing panels for international conferences and workshops. He is currently editing several books with Springer Nature, Elsevier, and Routledge & CRC Press. He is also serving as guest editor for special issues of the journal, like *Springer Nature and Inderscience*.