ISSN: 2583-9292 (ONLINE)
DOI: 10.xx/ijdsml.2023.xx

ESTIMATION OF TROPICAL CYCLONE INTENSITY FROM SATELLITE IMAGERY USING A DENSE CONVOLUTIONAL NEURAL NETWORK FROM CHANNEL DATA

D. Bhuvaneswari[1] and V. Shoba[2]
[1]*Department of Computer Science, SRM Arts and Science College, India*
[2]*Department of Computer Applications and Technology, SRM Arts and Science College, India*

Abstract

Predicting the intensity of tropical cyclones is an exciting task since it involves a lot of pre-processing, the removal of characteristics, the extraction of numerous sets of parameters from satellite data, human analysis, and significant human involvement. This research suggested utilizing a war strategy optimization algorithm in conjunction with a channel attentive dense convolutional neural network to calculate the TC intensity. We use satellite imaging data for intensity values and the HURDAT2 database for wind speed input in this work. The image cropping approach is used in this suggested work to preprocess the cyclone input image. This technique is used to reduce the computational complexity and improve classification accuracy by cropping and eliminating undesired image sections. In order to precisely determine the intensities of TCs, the clipped cyclone image is then sent to the suggested CAD-CNN. In this case, the DenseNet-121 CNN model provides additional training parameters to improve the training procedure overall. To achieve useful results, this model is further improved using a channel attention and spatial attention layer. Rather than treating every portion of the image the same, the SA layer attempts to provide special attention to the semantically linked regions. The visual feature to determine the weight is presented by the CA layer because the SA layer lacks it. To improve the overall estimation accuracy, the war strategy optimization technique is used to fine-tune the model's parameters. Attack and defense tactics for metaheuristic optimization procedures, rank and weight update tactics, and poor search agent replacement are a few examples of war strategies. This model uses infrared satellite photos more effectively.

Keywords:

Channel Attentive Dense, Cyclone Intensity, Convolutional Neural Network, DenseNet-121 CNN, War Strategy Optimization

1. INTRODUCTION

Tropical Cyclone (TC) intensity is the wind of highest sustained surface closer to TC centre and it is considered as an essential parameter which requires to be precisely estimated in disaster warning and TC predicting management [1]. In general, the TC can generate higher threats and also makes varied impacts to people's life. To measure the effects of TC, it is very important to determine the stages of hurricanes or TC [2]. In a single event, the TC can cause more than 1000 deaths and also it has the feasibility to enhance the death rate higher than 100,000 in worldwide [3]. From the past few years, it is estimated that, some preceding storms generates more impacts and nearly $100 billion in damage [4]. For disaster preparedness, it is more necessary to precisely detect the TC intensity.

In TC, the direct measurements of winds are sparse, especially over open Ocean. Hence, detecting the TC intensity is enabled initially through satellite measurements [5]. However, accessing the intensity of TC accurately is a challenging issue as the National Hurricane Centre (NHC) predicts 10 to 20% uncertainty in its post analysis intensity predicts when only satellite measurements are presented [6]. As declared by NHC, the average 24 hours intensity error for any provided storm is on the range of 10 to 20% [7]. Therefore, enhancing initial intensity forecasts from satellite imagery can provides suitable enhancements in short term intensity predicting and thereby it improves the disaster readiness in our nation [8]-[12].

To make better precautions for TC disasters, the rapid detection of TC intensity is highly required and precise prediction is more essential [13]. Nevertheless, it is hard to observe TCs through ground-based measurements. This is because, TCs mainly happen and extends its severity in the centre most portion of ocean [14]. An advancement of meteorological satellite sensor systems has established a new era in climate predicting and meteorological measurements. To notice TC in Real time, large temporal resolution geostationary satellite data are assumed to be the flexible one. Because, it gathers information on several properties of TCs like centre location and intensity. Dvorak Technique is one of the major scheme utilized to analyse the TCs. This traditional technique is a manual method which analyse the TC intensity depending an empirical satellite analysis [15, 16]. For fetching the TCs scale, the Dvorak technique has employed in several works and it establishes suitable damage recovery policies. Nevertheless, the feasibility of real-time intensity readings through this traditional technique is inevitably minimal. Thus, to avoid such limitations, the recent studies used deep learning mechanisms for estimating TC intensity through satellite imageries [17].

Deep learning techniques are more popular, efficient and effective computational mechanism for several fields. Using multiple layers, the deep learning models can make effective predictions. The layers presented in the deep learning models can efficiently learns the required characteristics generates suitable decisions. Thus, this deep learning models are highly suitable for estimating TC intensity through satellite imagery measurements. Some of the existing deep learning models used to estimate TC intensity are Convolutional Neural Network (CNN) [18], Recurrent Neural Networks (RNN) [19], Bidirectional Recurrent Gated Unit (BiGRU) [20], etc. Due to an increasing advancement in deep learning models, the proposed study developed an efficient deep learning technique for estimating TC intensity via satellite imagery.

1.1 MOTIVATION

TC is identified to be one of the most serious natural hazards originating from the tropical ocean basins. Studying about the intensity of TC is of high significance as it helps in estimating the losses it can cause and appropriate preventive measures can be taken on time. The research community has shown interests in predicting the motion and landfall of TC for the past few decades whereas, the prediction of intensity of TC still requires more

focus. Deep learning models help to accurately forecast the storm intensity at various levels, which helps the government to prevent hazards. Several methodologies are developed within the past few years to deal with this problem among which the neural network based systems are highly proficient. All these researches are motivated to propose a new and effective deep learning based TC estimation methodology that can provide highly accurate results than the existing techniques.

The main contributions of this paper are as follows:

- A deep learning inspired TS intensity estimation framework is built in this work to accurately estimate the intensities of TCs considering various parameters.
- A novel and proficient deep learning model named the channel attentive dense convolutional neural network (CAD-CNN) model is proposed to accurately estimate the intensities of TCs.
- The proposed network model is then fine-tuned using the meta-heuristic optimization named war strategy optimization (WSO) algorithm. This algorithm iteratively updates the weight values of each layers of the proposed network.
- Extensive evaluations are conducted in terms of various metrics to prove the performance effectiveness of the proposed TC intensity estimation approach.

2. RELATED WORK

Wimmers et al. [21] presented a deep learning-based CNN for exploring the feasibilities of predicting TC intensity through satellite images in the frequency bands of 85 to 92 GHz. In this existing study, DeepMicroNet is developed and it has the characteristics like probabilistic output, capability to perform partial scans and resiliency to inaccurate TC center fixes. The developed model is trained on global best track intensities and it generates accurate optimal track intensity biases. The performance of developed model is determined by analysing the parameters like RMSE. The simulation analysis shows that the developed model is more applicable for forecasting TC intensities.

Maskey et al. [22] introduced a deep learning model for estimating the intensity of TC through infrared satellite imagery. Here, an end-to-end deep learning mechanism was developed for wind speed estimation in real time scenario. A CNN model was established to forecast the TC wind speed through satellite images. The developed model is determined and compared the features with traditional Dvorak T-amount of images. The analysis reveals that the developed deep learning model generates positive results. Also, this existing study suggested a way to observe new storms and develop a workflow to afford wind speed estimates in real time through an awareness portal. The result analysis in terms of RMSE shows that the developed model produced superior outcomes.

Zhang et al. [23] presented estimation and classification of TC intensity through infrared satellite images using deep learning concept. Here, the satellite images were captured from the northwest Pacific Ocean basin by integrating cascaded deep CNN model. The developed model contains dual CNN network modules such as TC intensity classification (TCIC) module and TC intensity estimation (TCIE) module. Initially, the TCIC module is employed to partition TC intensity into 3 varied classes with the help of infrared satellite images. Then, 3 TCIE models depending on CNN regression network is established which integrated varied intensities of infrared satellite images with the TC optimal track data are involved. The efficacy of developed model produced optimal estimation outcomes as compared with existing models.

Wang et al. [24] developed a deep CNN model for estimating TC intensity from geostationary satellite imagery. In this study, the satellite images was captured from Northwest Pacific Ocean. Here, the brightness temperature information was observed through an Advanced Himawari Imager on board the Himawari-8 geostationary satellite. From the year of 2015-2018, the total number f95 TC cases are employed to train the developed deep CNN models. In this existing study, different models with varied parameters and inputs were utilized. The comparative analysis reveals that the chosen criteria of varied infrared channels had a positive effect on the efficacy of TC intensity estimation from the developed CNN models. The simulation analysis shows that the developed model attained the accuracy of 84.8% and reduced RMSE of 5.24 m/s.

Tian et al. [25] designed a multi-dimensional CNN for estimating TC intensity from multi-channel satellite imagery. Estimation of TC intensity is the initial step for performing observing and forecasting threatened TC disasters. This existing study developed a 3DAttentionTCNet model for resolving the issues while forecasting TC intensities. The developed model fetched 3D environmental data corresponded to TC intensity from multi-channel satellite monitory images like water vapour (WV), infrared and passive microwave rain rate (PMW) satellite images via 3D convolution. Here, the convolutional block attention module (CBAM) was utilized to perform visual attention simulation for boosting the efficiency of developed model. The result analysis shows that the developed model provided reduced RMSE as compared with traditional Dvorak technique.

For cyclone estimation Dvorak technique was used to assume the TCs with similar intensity result in similar pattern. This technique is highly subjective and needed to localize the center of TC manually, and also produce large amount of root-mean-square intensity error (RMSE). DeepMicroNet has the characteristics like probabilistic output, capability to perform partial scans and resiliency to inaccurate TC center fixes. Every other existing techniques have issues like complexity in training, fine-tuning complications, inconsistency and uncertainty problems, over fitting issues, etc. The pre-trained CNNs have a stable design and input dimension that may not be same for our needs. Plenty of training data is necessary for CNN process. To overcome these problems we use channel attentive dense convolutional neural network (CAD-CNN) model to accurately identify the intensities of TCs.

3. PROPOSED METHODOLOGIES

The estimation of intensities of TCs is a complex task and requires highly proficient tools to accurately determine it. Current studies focused on developing and applying deep learning based methods to solve this problem and to gain effective results. Since the TC intensity estimation is critical, more studies and analysis

are still required that can properly explore the nature of TCs thereby providing sufficient information to the model for estimation. This work introduces a new deep learning based mechanism that can accurately determine the intensity of TCs using the infrared satellite imagery. The block diagram of the proposed work is shown in Fig.1.

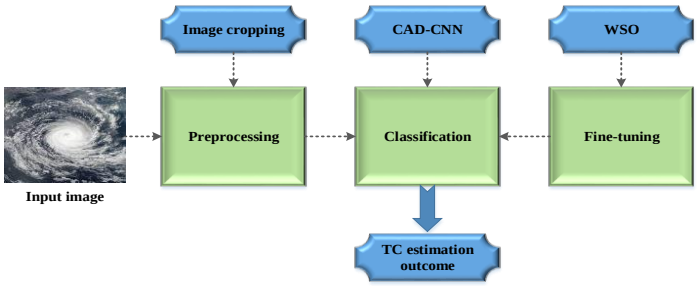


Fig.1. Block Diagram of the Proposed Work

The proposed TC intensity estimation framework includes three major steps such as pre-processing, classification and fine-tuning.

3.1 PRE-PROCESSING

The satellite images are initially subjected to pre-processing where the image cropping is performed to crop and drop the unwanted image portions. The exterior portions of the cyclone are cropped off and the exact part of cyclone image is alone utilized in the model to enhance the classification accuracy and to diminish the computational complexity. For example, if 512 × 512 pixels is the mass of the original satellite image that contains excessive text data. So from the original image, 80×80 image pixel is cropped in the form of longitude and latitude delivered by the best-track records. From this we only get the needed portion of the images.

3.2 CLASSIFICATION USING CAD-CNN

The cropped image is then provided to the proposed CNN model named the channel attentive dense convolutional neural network (CAD-CNN) model to accurately identify the intensities of TCs. The basic CNN model utilized in this work is the DenseNet-121 which is one of the deep CNN architecture providing more training parameters to enhance the overall training process. Further, the model is enhanced with attention criteria such as the channel attention (CA) layer and spatial attention (SA) layer are included to gain valuable results. The DenseNet-121 CNN architecture is shown in Fig.2.

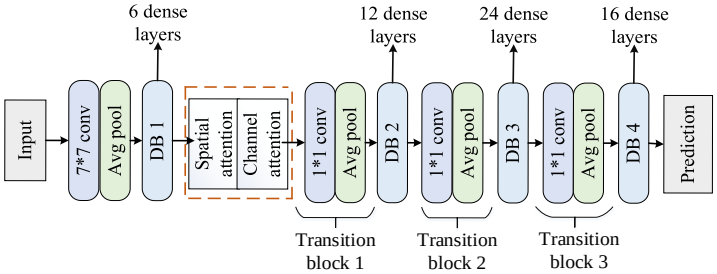


Fig.2. DenseNet-121 CNN architecture

DenseNet-121 contains six dense layers in a dense block. The outcome of each dense block is similar to the development rate of dense block. DenseNet starts with a basic convolution and pooling layer. Then the dense block one follows the classification and progress layer, then the dense block two follows a change layer, dense block three follows a progress layer. Lastly it predicts the outcome. Every dense block contains convolutions, with 1*1 expected pieces. In dense block one and two, it is done multiple times, in dense block three, 24 times finally in dense block four, 16 times. All dense- layer contains convolution processes (1*1 conv) for removing the features. Here we include classification layers namely, Spatial and channel attention are the independent modules of convolutional block attention module (CBAM). It consecutively guesses attention sequences beside the channel and spatial dimensions, and the inner product procedure is used to integrate the input features and attention sequences. The spatial and channel attention CNN is shown in Fig.3.

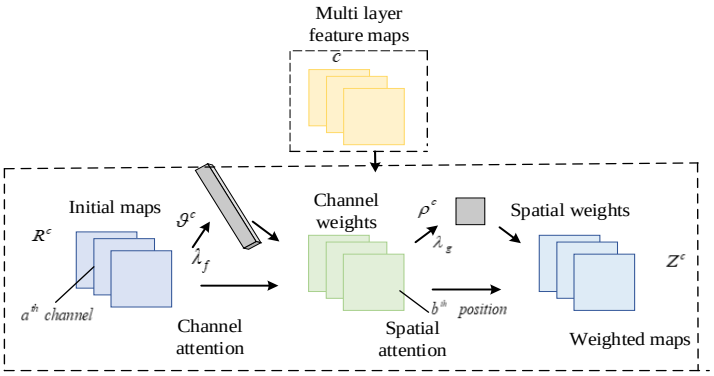


Fig.3: Spatial and Channel attentive CNN

3.2.1 *Spatial Attention:*

When we need to determine cyclone, the only image areas which has cyclone are valuable. Then, an international image feature vector is applied to produce caption can lead to sub-optimal outcomes due to the inappropriate regions. Spatial attention appliance tries to pay extra consideration to the semantic-interrelated regions, instead of considering all image area similarly. Without damage of generalization, we reject the layer-wise superscript c. We reform $R=[R_1,R_2,\cdots,R_n]$ by levelling the length and breadth of the original R, where $R_b \in A^B$ and $n=P\cdot Q$. Consider R_b as the graphical feature of the b^{th} *position*, the unseen state is u_{s-1}, single-layer neural network is used, that are monitored by softmax function to produce the attention distributions ρ^c over the image areas. The spatial attention method λ_g definitions are given below,

$$d=\tan u\left(\left(P_e R+h_e\right)\oplus P_{ue}u_{s-1}\right),$$
$$\rho=soft\max\left(P_b d+h_b\right) \quad (1)$$

where $P_e \in A^{k\times B}, P_{ue} \in A^{k\times l}, P_b \in A^k$ are the change matrices that maps the image visual features and unseen state to equal dimension. Addition of vector and matrix is denoted by $\oplus$ symbol. Among the vector and matrix, addition method is done to add every column. The model biases are $h_e \in A^k, h_b \in k^1$.

3.2.2 *Channel Attention:*

For calculating the weights of spatial attention, it needs visual feature *R*, but in the spatial attention the visual feature *R* is not

attention-based. So we present a channel attention appliance to present the visual feature *R*. Every CNN filter implements as a pattern detector, and feature map of all channel in CNN is a response stimulation of the resultant convolutional filter. Thus, the channel-wise way is applied by an attention mechanism for the process of choosing semantic attributes.

For the channel attention method, first reform $R to T$ and $T=\left[t_1,t_2,\cdots,t_B\right]$ where the b^{th} channel of feature map R is denoted as $t_b \in A^{P\times Q}$, total number of channel is B. For every channel the mean pooling is applied to get the channel feature R:

$$R=\left[\gamma_1,\gamma_2,\cdots,\gamma_B\right], R\in A^B \tag{2}$$

where scalar γ_b is the mean of vector t_b, this denotes the b^{th} channel features. According to the description of spatial attention method, the channel attention method λ_f is defined below:

$$\begin{aligned} h &= \tan u\left(\left(P_B \otimes R + h_b\right) \oplus P_{uB} u_{s-1}\right) \\ \vartheta &= soft\max\left(P_b' h + h' b\right) \end{aligned} \tag{3}$$

The transformation matrices are $P_B \in A^k, P_{uB} \in A^{k\times d}, P_b' A^k$, the external product of vector is represented by $\otimes$ symbol, and the bias terms are $h_B \in A^k, h_b' \in A^1$.

3.3 PARAMETERS TUNING BY WAR STRATEGY OPTIMIZATION ALGORITHM

The parameters in the model are tuned using the war strategy optimization (WSO) algorithm to enhance the overall accuracy in estimation. This algorithm iteratively tunes the model to reduce the loss function so that the error can be minimized and accuracy can be enhanced. At every iteration, all search interface have equivalent probability to get the parameters depends on the fitness value. One search interface will guides all other interface during the searching process. Some of the stages in war strategy are given bellow.

3.3.1 Attack Strategy:

At the starting of search process all search agent have equal weight and rank. The rank of agent will be increased if the search agent tunes the parameters. The rank and weight of all agents updated as strategy achievement in the searching progress. As the process reaches to conclusion the search agent stay very close to approach the parameter.

$$Z_j\left(l+1\right) = Z_j\left(l\right) + 2\times\lambda\times\left(A-B\right) + randZ\left(T_j\times B - Z_j\left(l\right)\right) \tag{4}$$

where T_j is the weight, $Z_j\left(l+1\right)$ is the new search process, Z_j is previous process. A is the position of search agent. If T_j becomes zero, the updated position of the search agent travels very close to the parameters which denotes the ending stage of the search process.

3.3.2 Rank and Weight Update:

The rank of all search agent depends on achievement account in the search field governed by Eq.(7) which will successively control the weighting factor T_j. If the fitness value in new search process $\left(S_n\right)$ is less than the previous process $\left(S_p\right)$, the search agent takes previous process. The weight T_j differs exponentially as a factor of ρ in the WSO algorithm.

$$Z_j\left(l+1\right) = \left(Z_j\left(l+1\right)\right)\times\left(S_n \succeq S_p\right) + \left(Z_j\left(l\right)\right)\times\left(S_n \prec S_p\right) \tag{5}$$

If the search agent updates the location successfully, the agents rank $\left(K_j\right)$ will be upgraded.

$$K_j = \left(k_j+1\right)\times\left(S_n \succeq S_p\right) + \left(K_j\right)\times\left(S_n \prec S_p\right) \tag{6}$$

New weight is calculated based on the rank:

$$T_j = T_j \times \left(1 - \frac{K_j}{Max\_iter}\right)^{\gamma} \tag{7}$$

3.3.3 Defense Strategy:

The second strategy location update based on the location of search agent. But the weight and rank update is same.

$$\begin{aligned} Z_j\left(l+1\right) = Z_j\left(l\right) &+ 2\times\lambda\times\left(M - Y_{rand}\left(l\right)\right) \\ &+ rand\times T_j\times\left(A - Z_j\left(l\right)\right) \end{aligned} \tag{8}$$

This search process needs additional search space when related to previous process and includes the location of random search agent. If the T_j has high value, the search agent takes more steps for update the location. If the T_j has low value, the agent takes less steps to update the position.

3.3.4 Replacement of Weak Search Agent:

Identify the weak search agent having less fitness value for every iterations. Here we replacing the weak search agent with the random search agent in Eq.(9).

$$Z_T\left(l+1\right) = Q_a + rand\times\left(V_a - Q_a\right) \tag{9}$$

The second approach is repositioning the weak search interface closer to the median of whole search agent in a search field as given in Eq. (10). This method increases the convergence performance of algorithm.

$$Z_T\left(l+1\right) = -\left(1-rand_m\right)\times\left(Z_t\left(l\right) - median\left(Z\right)\right) + M \tag{10}$$

3.3.5 Exploitation and Exploration

The exploitation and exploration are the two chief principles for metaheuristic optimization processes. An upright interchange between these principles will create the processes stronger and more effective. Attack strategy signifies the exploitation however defense strategy signifies the exploration. The pseudocode for WSO is given below:

Initialize the search agent, dimension, upper and lower limits of searching field, position of search agent $\left(B\right)$, attack forces of search agent, parameter size $\left(W\right)$ are as follows:

$$\left(W\right) = 30;\ B = zeros\left(1,\dim\right);\ Max-iteration = 1000;\ \lambda_r = 0\cdot 5$$

Algorithm

Initialize the parameters; $R = zeros\left(1, parameter\ size\right)$;

$T = 2\times ones\left(1, parameter\ size\right)$ Uniformly and randomly allocate the parameter in search field (Random attack)

For 1: Parameter_size

Obtain attack force for each parameter
End of for loop
Sort the fitness (attack force) of each search agent
Select the search agent with best fitness
While $l \prec L\max(Max-iterations)$
For 1: Parameter_size
$\lambda = rand$
If percentage signal assumed to follow the strategy
Update the position of each search agent using Equation (8)
(Exploration)
Else update the position of each search agent using Eq.(4)
(Exploitation)
End of if condition
Calculate the attack force for each search agent
Sort the fitness of each search agent
Update the position of each search agent based on attack force of new and previous
positions using Eq.(5)
Update the weight and rank of each search agent based on the success using Eq.(6)
End of for loop
Identify the weak search agent with low fitness
Relocate the weak search agent by selecting a strong search agent
Update the positions of search agent and parameter
$t = t+1$
End of while loop
Display the attack force and position of parameters.

4. RESULTS AND DISCUSSION

This section presents the results and analysis of the proposed technique. Here, simulation is performed by using Python tool. Training and testing of the model is carried out by utilizing HURDAT2 data. The proposed method experimentally analysed by validating the performance parameters that are discussed below.

4.1 PERFORMANCE METRICS

The proposed technique considers the following performance metrics namely, kappa, F_1 score, RMSE and MAE and accuracy.

- *Kappa*: Kappa is defined as a form of relationship to measure the accord on two or many detection types by two or many methods. It is mathematically expressed in the following equation

$$K = \frac{Q_{observed} - Q_{chance}}{1 - P_{chance}} \quad (11)$$

- F_1 Score: F1-Score is defined as the variant of accuracy which is not influenced by the negatives. It is illustrated in the following equation. It is mathematically expressed in the following equation.

$$2\frac{\text{Re}\,call \times \text{Pr}\,ecision}{\text{Re}\,call + \text{Pr}\,ecision} \quad (12)$$

- RMSE: Root Mean Square Error (RMSE) can be defined as the measure of difference among the scores detected by a model and the exact values. RMSE is expressed mathematically in the following equation.

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left|b-\hat{b}\right|}^{2} \quad (13)$$

where, the exact value is denoted as b and $\hat{b}$ gives the attained value and m gives the amount of samples for evaluation.

- MAE: Mean Absolute Error (MAE) is obtained by finding the mean of the sum of total variation among the detected value and exact value.

$$MAE = \frac{1}{m}\sum_{i-1}^{m}\left|b-\hat{b}\right| \quad (14)$$

- Accuracy: Accuracy is defined as the measure of nearness of the observed value to the reference value. It can be written as in the below equation.

$$Accuracy = \frac{TP+PN}{TP+TN+FP+FN} \quad (15)$$

where, TN denotes the True Negative value, TP specifies the True Positive value, FP represents the False Positive value and FN gives the False Negative value.

4.2 PERFORMANCE AND COMPARISON ANALYSIS

In this section, the results and analysis of the proposed method and the comparison of the proposed technique with existing methods is discussed. Also, the comparison of the proposed model with existing techniques like RMS prop (no batch normalization), RMS prop (with batch normalization), Adam (no batch normalization), Adam (without batch normalization), C-TCNN and T-CTNN is also discussed briefly. The below Fig.4 shows the accuracy of the proposed model.

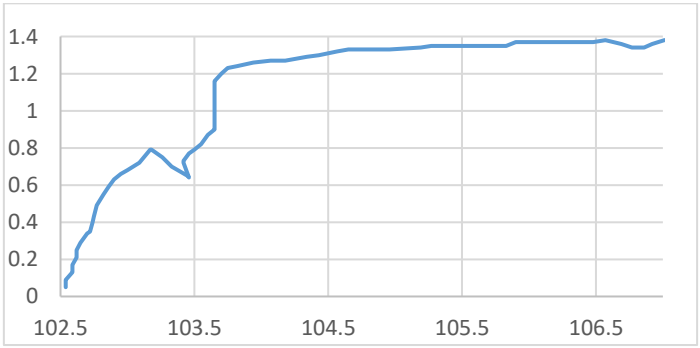
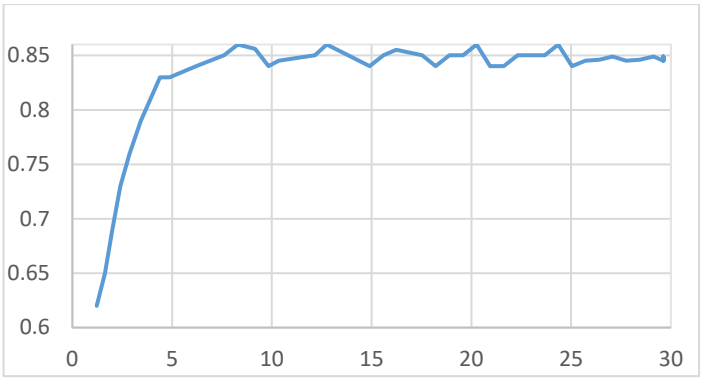
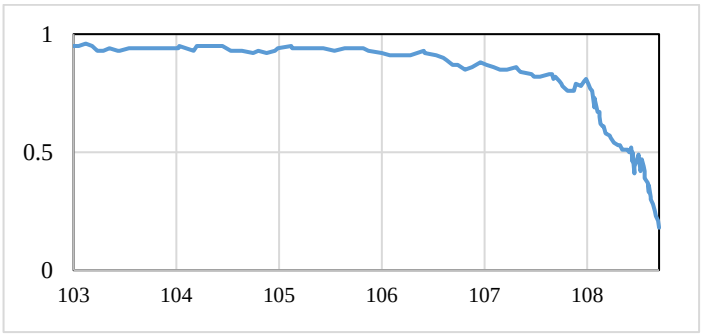


Fig.4. Learning Rate Vs Accuracy

The Fig.4 shows the accuracy in reference to the learning rate. The accuracy is measured for different learning rates by varying the learning rate. When the learning rate is 10^{-9} the accuracy starts increasing and reaches the peak at 10^{-7} and after that remains steady for succeeding learning rates.



(a) Testing Accuracy



(b) Testing Loss

Fig.5. Testing Accuracy and Loss

The Fig.5 shows the accuracy and loss curves of the proposed method by varying the epoch value. From the figure, it is observed that accuracy starts increasing when the epoch value is just greater than zero. When the epoch value is just greater than 5 accuracy reaches its peak and then remains steady for some time and when the epoch is 15 it again starts increasing. Thus it is inferred that the proposed system achieves better accuracy for higher epoch values. In addition to this, the loss value of the proposed method is also calculated for different epoch values. From the above figure, it is clear that the loss is minimum for higher epoch values. Hence, it is observed that for higher epoch values the accuracy will be higher and loss is minimum.

The Fig.6 indicates the accuracy and loss curves of the proposed method while training for different epoch values. It shows that when the epoch value is just after zero accuracy starts increasing and reaches the peak when the epoch value is just greater than 5. After that the accuracy curve remains steady for succeeding epoch values. Moreover, the loss is also computed, by varying the epoch values. From the figure, it is clear that the loss value starts decreasing, when the epoch is just greater than zero and attains a minimum when it is just greater than 5. For higher epoch values, the proposed method shows minimum loss.

The Fig.7 shows the comparison of Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) of the proposed method with existing methods. From the Fig.it is inferred that the MAE (kts) value of RMS prop (no batch normalization) as 7.21, RMS prop (with batch normalization) as 6.68, Adam (no batch normalization) as 8.68, Adam (without normalization) as 8.52 and for the proposed method (WSO) the MAE is 5.64. Thus, it is proved that the proposed system has a minimum mean absolute error when compared with existing techniques. Also, the RMSE (kts) value of RMS prop (no batch normalization) is observed as 9.47, RMS prop (without batch normalization) as 7.6, Adam (no batch normalization) as 10.18, Adam (without batch normalization) as 10.04, and for the proposed model (WSO) the RMSE value is 5.71. This shows that the proposed model has a minimum root mean square error when compared with the existing methods.

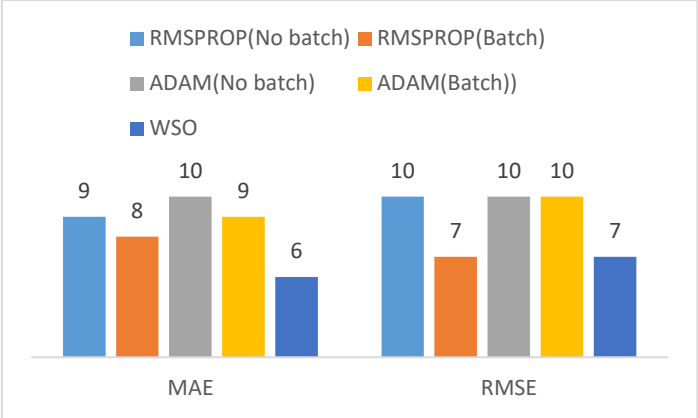


Fig.6. Comparison of MAE and RMSE

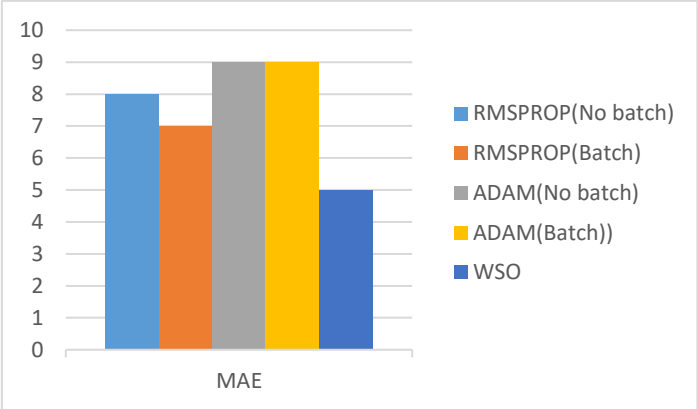


Fig.7. Comparison of Relative RMSE

The Fig.7 indicates the comparison of Relative RMSE of proposed method with existing methods. The Fig.7 shows the Relative RMSE of RMS prop (no batch normalization) as 0.19, RMS prop (with batch normalization) as 0.17, Adam (no batch normalization) as 0.22, Adam (without batch normalization) as 0.2 and for the proposed method the Relative RMSE is 0.151, thereby showing that the proposed model has a low Relative RMSE.

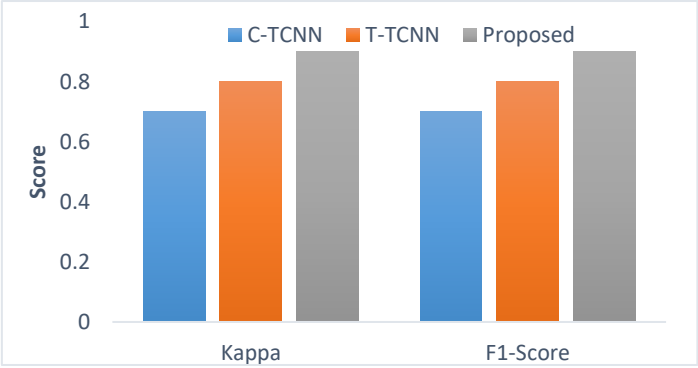


Fig.8. Comparison with Kappa and F1 Score

The Fig.8 indicates the comparison of the proposed method with two other existing methods namely C-TCNN and T-TCNN. The Fig.8 demonstrates the kappa and F1 score of the proposed technique and the existing (C-TCNN and T-TCNN) methods. It shows the Kappa of C-TCNN as 0.85 and for the T-TCNN it is 0.92 and the proposed model has a Kappa of 0.95. The Fig.8 also indicates the F1 score of C-TCNN as 0.85, T-TCNN as 0.91 and for the proposed method it is 0.945. Thus, it can be said that the proposed system has an increased Kappa and F1 score values while comparing with C-TCNN and T-TCNN, thus enhancing the performance of the proposed model.

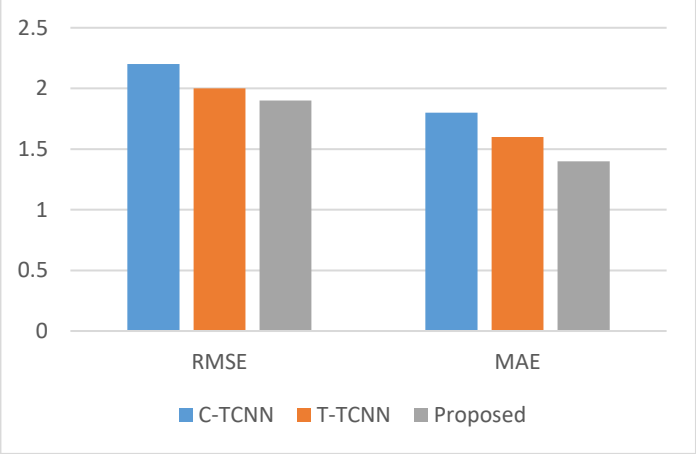


Fig.9. Comparison of RMSE and MAE

The Fig.9 illustrates the comparison of the RMSE (m/s) and MAE (m/s) value of the proposed method with C-TCNN and T-TCNN. This Fig.10 clearly shows that the RMSE of C-TCNN is 2.13, for T-TCNN it is 2 and for the proposed method it is 1.95. From this it is observed that the proposed method has a low RMSE value when compared to C-TCNN and T-TCNN. Also, the Fig.10 shows that the MAE of C-TCNN is 1.85, for T-TCNN it is 1.67 and the proposed method has an MAE value of 1.42. This also reveals that the proposed model has minimum MAE value while comparing with C-TCNN and T-TCNN, thereby proving that the proposed system is very effective.

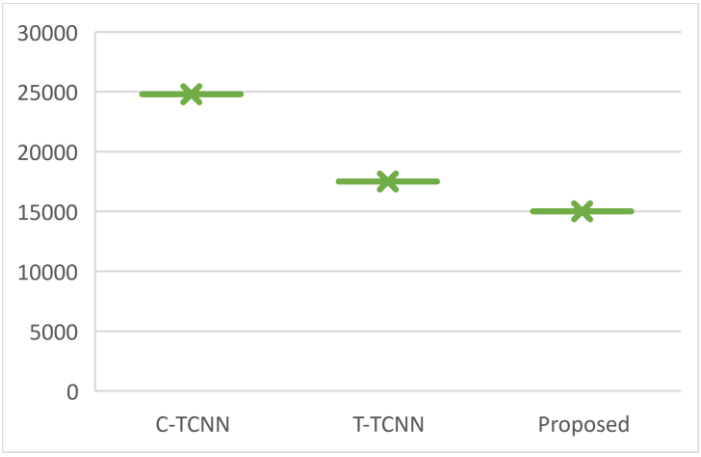


Fig.10. Comparing training time with C-TCNN and T-TCNN

In Fig.10, the training time of the proposed technique is compared with the existing (C-TCNN and T-TCNN) techniques. It indicates that the training time of C-TCNN is 23580 seconds and for T-TCNN it is 17885 seconds. The Fig.10 also shows the training time of proposed model as 15221 seconds. Thus for the proposed system the training time is less when compared to existing techniques. Hence the proposed system saves the computational time.

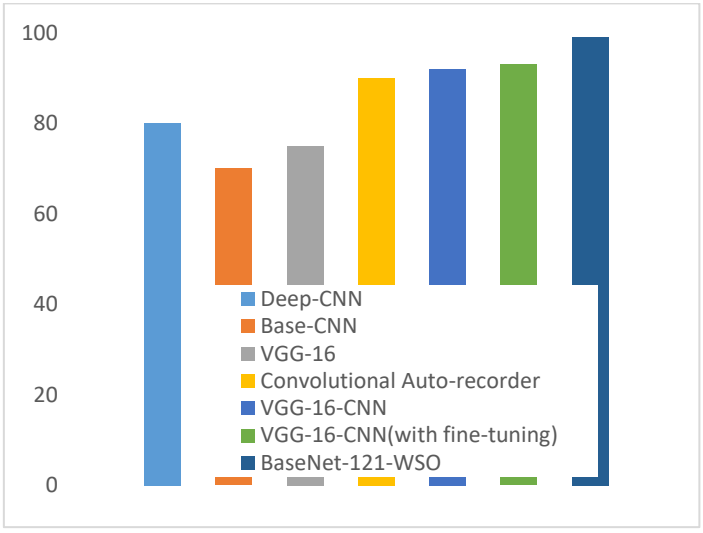


Fig.11. Comparing the accuracy of proposed method with existing techniques

The Fig.11 indicates the comparison for the accuracy of proposed technique with various existing techniques. From the Fig.it is inferred that the Deep-CNN obtained an accuracy of 80.66%, Base-CNN has an accuracy of 70%, VGG has an accuracy of 74%, Convolutional Auto-encoder has an accuracy of 88.4%, VGG-16-CNN obtained an accuracy of 89.5% and VGG-16-CNN attained an accuracy of 98%. But the proposed method has an accuracy of 99.2% which is far better than the existing methods.

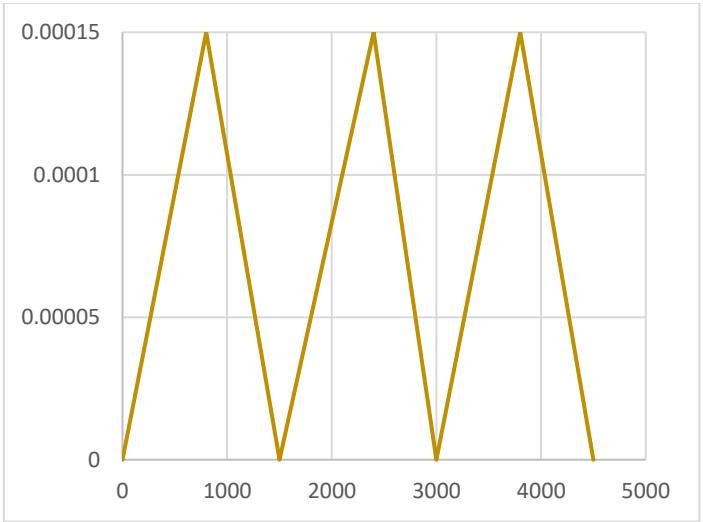


Fig.12. Learning Rate for varied training iterations.

The Fig.12 indicates the Learning Rate for different training iterations. When the training iteration is zero, the learning rate is also zero and when the training iteration is 750, the learning rate reaches the peak and is 0.00015, when the training iteration is 1500, the learning rate is again zero. Also, if the training iteration is 2250, the learning rate is 0.00015. Again if the training iteration

is 3000, the learning rate is 0.00015. From this it is inferred that when the training iteration is zero the learning rate is also zero and for other iterations the learning rate is 0.00015.

5. CONCLUSION

To predict the strength of a tropical storm from satellite data, we developed a new channel attentive dense convolutional neural network in this research. To improve classification accuracy, the image is cropped in the pre-processing approach. The clipped image is then sent into the suggested CAD-CNN technique to precisely determine the TC intensities. Here, the DenseNet-121 CNN model provides additional training parameters with a spatial attention layer and channel attention layer to improve the entire training process and produce useful results. The location of a beneficial portion is the center of spatial attention, which is equivalent to the channel attention estimation.

REFERENCES

[1] Q. Jin and Hongdeng Jian, "Estimating Tropical Cyclone Intensity in the South China Sea using the XGBoost Model and Feng Yun Satellite Images", *Atmosphere*, Vol. 11, No. 4, pp. 423-432, 2020.

[2] S.C. Velden and Derrick Herndon. "A Consensus Approach for Estimating Tropical Cyclone Intensity from Meteorological Satellites: SATCON", *Weather and Forecasting*, Vol. 35, No. 4, pp. 1645-1662, 2020.

[3] Jing Yi and Zhe-Min Tan, "Physics-Augmented Deep Learning to Improve Tropical Cyclone Intensity and Size Estimation from Satellite Imagery", *Monthly Weather Review*, Vol. 149, No. 7, pp. 2097-2113, 2021.

[4] Rui Chen, Weimin Zhang, and Xiang Wang, "Machine Learning in Tropical Cyclone Forecast Modeling: A Review", *Atmosphere*, Vol. 11, No. 7, pp. 676-687, 2020.

[5] Amina Asif, Muhammad Dawood, Bismillah Jan, Javaid Khurshid, Mark DeMaria and Fayyaz Ul Amir Afsar Minhas, "PHURIE: Hurricane Intensity Estimation from Infrared Satellite Imagery using Machine Learning", *Neural Computing and Applications*, Vol. 32, pp. 4821-4834, 2020.

[6] Alberto Bento, Rajchandar Padmanaban, Pedro Cabral, Salomao Bandeira and Maria M. Romeiras, "Impacts of the Tropical Cyclone Idai in Mozambique: A Multi-Temporal Landsat Satellite Imagery Analysis", *Remote Sensing*, Vol. 13, No. 2, pp. 201-211, 2021.

[7] Peng Yu, Johnny A. Johannessen, Xiao-Hai Yan, Xupu Geng, Xiaojing Zhong and Lin Zhu, "A Study of the Intensity of Tropical Cyclone Idai using Dual-Polarization Sentinel-1 Data", *Remote Sensing*, Vol. 11, No. 23, pp. 2837-2842, 2019.

[8] Hui Yu, Xiaoming Yang and Xiaofeng Li, "Estimating Tropical Cyclone Size in the Northwestern Pacific from Geostationary Satellite Infrared Images", *Remote Sensing*, Vol. 9, No. 7, pp. 728-734, 2017.

[9] P. Mohan and Eric Strobl, "The Short-Term Economic Impact of tropical Cyclone Pam: An Analysis using VIIRS Nightlight Satellite Imagery", *International Journal of Remote Sensing*, Vol. 38, No. 21, pp. 5992-6006, 2017.

[10] Muhammad Al-Amin, Stuart Phinn, Chris Roelfsema and Iraphne Child, "Tropical Cyclone Disaster Management using Remote Sensing and Spatial Analysis: A Review", *International Journal of Disaster Risk Reduction*, Vol. 22, pp. 345-354, 2017.

[11] C. Zhang and David Lagomasino, "Modeling Risk of Mangroves to Tropical Cyclones: A Case Study of Hurricane Irma", *Estuarine, Coastal and Shelf Science*, Vol. 224, pp. 108-116, 2019.

[12] M. Kim and Myong-In Lee, "Machine Learning Approaches for Detecting Tropical Cyclone Formation using Satellite Data", *Remote Sensing*, Vol. 11, No. 10, pp. 1195-1199, 2019.

[13] James D. Doyle, Mark Beaubien, Michael M. Bell and Daniel L. Cecil, "A View of Tropical Cyclones from Above: The Tropical Cyclone Intensity Experiment", *Bulletin of the American Meteorological Society*, Vol. 98, No. 10, pp. 2113-2134, 2017.

[14] Hui Su and Mark DeMaria, "Applying Satellite Observations of Tropical Cyclone Internal Structures to Rapid Intensification Forecast with Machine Learning", *Geophysical Research Letters*, Vol. 47, No. 17, pp. 1-12, 2020.

[15] Muhammad Al-Amin, Stuart Phinn, Chris Roelfsema and Iraphne Childs, "Assessing Tropical Cyclone Impacts using Object-based Moderate Spatial Resolution Image Analysis: A Case Study in Bangladesh", *International Journal of Remote Sensing*, Vol. 37, No. 22, pp. 5320-5343, 2016.

[16] T.L. Olander and Christopher S. Velden, "The Advanced Dvorak Technique (ADT) for Estimating Tropical Cyclone Intensity: Update and New Capabilities", *Weather and Forecasting*, Vol. 34, No. 4, pp. 905-922, 2019.

[17] Anthony Wimmers, Christopher Velden and James P. Kossin, "Investigation of Machine Learning using Satellite-Based Advanced Dvorak Technique Analysis Parameters to Estimate Tropical Cyclone Intensity", *Weather and Forecasting*, Vol. 36, No. 6, pp. 2161-2186, 2021.

[18] Buo Fu Chen, Boyo Chen, Hsuan Tien Lin and Russell L. Elsberry, "Estimating Tropical Cyclone Intensity by Satellite Imagery Utilizing Convolutional Neural Networks", *Weather and Forecasting*, Vol. 34, No. 2, pp. 447-465, 2019.

[19] Rohit Jagadale, "A Novel Approach for Predicting the Tropical Storm Trajectories using Gridbased Recurrent Neural Networks", PhD Dissertations, Department of Computer Science, National College of Ireland, pp. 1-189, 2020.

[20] R.S. Ramya, S. Ayushi, R. Bhumika, P. Adhoksh, Keshav Jhawar, Ayush Shah and K.R. Venugopal, "Tropical Cyclone Tracking and Forecasting using BiGRU [TCTFB]", *Soft Computing*, Vol. 78, pp. 1-11, 2022.

[21] Anthony Wimmers, Christopher Velden and Joshua H. Cossuth. "Using Deep Learning to Estimate Tropical Cyclone Intensity from Satellite Passive Microwave Imagery", *Monthly Weather Review*, Vol. 147, No. 6, pp. 2261-2282, 2021.

[22] Manil Maskey and Jeffrey Miller. "Deepti: Deep-Learning-Based Tropical Cyclone Intensity Estimation System", *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, Vol. 13, pp. 4271-4281, 2021.

[23] Chang Jiang Zhang, Xiao-Jie Wang, Lei-Ming Ma, and Xiao-Qin Lu, "Tropical Cyclone Intensity Classification and

Estimation using Infrared Satellite Images with Deep Learning", *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, Vol. 14, pp. 2070-2086, 2021.

[24] Chong Wang, Bin Liu and Jun Zhang, "Tropical Cyclone Intensity Estimation from Geostationary Satellite Imagery using Deep Convolutional Neural Networks", *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 60, pp. 1-16, 2021.

[25] Wei Tian, Wei Huang, Yonghong Zhang, Pengfei Zhang, and Shifeng Hao, "Tropical Cyclone Intensity Estimation using Multidimensional Convolutional Neural Network from Multichannel Satellite Imagery", *IEEE Geoscience and Remote Sensing Letters*, Vol. 19, pp. 1-5, 2021.