Hurricane Damage Detection From Satellite Imagery Using Convolutional Neural Networks

Swapandeep Kaur, Chitkara University Institute of Engineering and Technology, Chitkara University, India*
Sheifali Gupta, Chitkara University Institute of Engineering and Technology, Chitkara University, India
Swati Singh, University Institute of Technology, Himachal Pradesh University, India
Isha Gupta, Chitkara University Institute of Engineering and Technology, Chitkara University, India

ABSTRACT

Hurricanes are one of the most disastrous natural phenomena occurring on Earth that cause loss of human lives and immense damage to property as well. For assessment of this damage, windshield survey is commonly used, which is an error-prone and time-consuming method. For solving this problem, computer vision comes into the picture. In this paper, a convolutional neural network-based architecture has been proposed to classify the post-hurricane satellite imagery into damaged and undamaged building classes accurately. The model consists of five convolutional and five pooling layers followed by a flattening layer and two dense layers. For this, a dataset of Hurricane Harvey has been considered having 23000 satellite images each of size 128 X 128 pixels. With the proposed model, the author has achieved an accuracy of 92.91%, F1-score of 93%, sensitivity of 93.34%, specificity of 92.47%, and precision of 92.65% at a learning rate of 0.0001 and 30 epochs. Also, low false positive rate of 7.53% and false negative rate of 6.66% were obtained.

KEYWORDS
Convolutional Neural Network, Damage Assessment, Hurricane, Satellite Imagery

1. INTRODUCTION

With the change in climatic conditions, there is an increase in the frequency as well as severity of the natural calamities (Pi et al., 2020). Hurricanes are the most catastrophic disasters occurring on Earth. They are called as tropical cyclones as they take place in the tropical areas because of the presence of warm water. The sun heats up the sea waters leading to the formation of huge clouds and thus hurricanes. Hurricanes are accompanied by heavy rain, floods and high-speed winds of about 320 km/hour (Dawood et al., 2020). Hurricane Harvey, a Category 4 hurricane made landfall in the Houston region in the year 2017 with a speed of 210 km/hour killing more than 100 people. It was the most powerful hurricane that struck this region in 56 years and caused flooding of the low-lying coastal regions. It caused a huge damage of $125 billion. Timely and accurate detection of damage caused due to hurricanes becomes imperative for effectively mobilizing relief efforts to the inflicted people.

Satellite imagery is of immense importance in disaster response for identification of the impact of the disaster. Since satellite images cover a very vast area of the ground surface, they form a very...
useful resource for leveraging disaster damage detection. However, analysis of satellite images for detection of disaster impact is extremely challenging. Deep learning (DL), a family of machine learning, could be utilized on these satellite images for providing insight on the damages caused by the hurricane disaster (Dotel et al., 2020). Also, DL shows promising results for automation of the disaster detection tasks. DL exceeds the performance of humans in classification of complex images. It is inspired from the structure and functions of the human brain which is called Artificial Neural Networks (ANN). Human beings learn from experience, similarly DL algorithms learn by performing a task repeatedly. It is called deep learning because the DL network is composed of several layers that help the network to learn. The layers are composed of an artificial neuron which is the primary component of an ANN (Kaur et al., 2021).

Numerous architectures are available for Deep Learning. Successful understanding of images can be done through the Convolutional Neural Networks (CNN)(Pritt & Chern, 2018). A CNN is a DL network particularly useful to extract hierarchical characteristics from the images(Kaur et al., 2021). Figure 1 shows the illustration of a basic CNN. The CNN model is divided into two parts, the first being feature extraction and the second: classification (Cao & Choe, 2020). The first part is composed of the input, convolutional layer and the pooling layer while the second part is composed of the dense and the output. The input layer takes an input image of a fixed size whose resizing could be done if required. The image is then convolved with filters. The pooling layer helps in reduction of the image size. The output of the feature extraction part is known as feature map. The output of the classification part gives the classification result through the fully connected and output layer (Phung & Rhee, 2019). A CNN based architecture can be applied on the satellite images for determining the damage caused by the hurricane disaster.

In this paper, a CNN based architecture has been proposed for classification of images into damaged and undamaged classes using the satellite images of Harvey Hurricane. In this article, the major contribution of the author is as follows:

1. A Convolutional Neural Network based model is proposed having five convolutional layers, five pooling layers, one flattening layer and two dense layers.
2. The proposed CNN network is simulated on the Adam optimizer at a learning rate of 0.0001 and 30 epochs.
3. The dataset considered for simulation has 23000 satellite images of Hurricane Harvey that are classified into damaged and non-damaged classes.
2. LITERATURE SURVEY

This section includes the work done previously on satellite images of hurricanes and the estimation of hurricane intensity and damage detection using deep learning.

Assessing the damage caused by a hurricane is of utmost importance for the emergency responders. But the process is labour intensive and time consuming. For the automatic detection of damage, remote sensing and computer vision play a pivotal role. In this paper, a benchmark dataset of Hurricane Harvey has been developed from the publicly available data. The dataset could be used for object detection models and applied to other hurricanes for the detection of damage (Chen et al., 2018).

Satellite imagery is gaining popularity for monitoring disasters. Immediate steps need to be taken for coordinating relief efforts to the people. In this paper, the author has proposed a CNN model on satellite imagery which is thresholded and clustered to form grids. These grids help in detection of regions severely affected by a disaster. F1-score of 81.2% has been obtained for the Hurricane Harvey dataset (Doshi et al., 2018).

To mitigate the damage caused by tropical cyclones, forecasting techniques need to be developed. In this paper, ANNs have been proposed for interpretation of NOAA-AVHRR satellite images. A multiple layer neural network has been trained for forecasting the movement of cyclones. The model provided the correct forecast for 98% of the testing images (Kovordányi & Roy, 2009).

For identification of the Hurricane Harvey impacted regions, a bitemporal image classification approach was used for comparison of pre and post disaster regions captured by Digital Globe satellite images. An accuracy of 84.5% and an F1-score of 0.675 was obtained (Dotel et al., 2020).

The author has assessed the combination of satellite imagery and different resolutions of CNN for classification of damaged buildings. The combination of various resolutions used for training of the CNN model resulted in an improvement of accuracy by 4% (Duarte et al., 2018).

Intensity estimation of hurricanes obtained from the HURDAT2 satellite dataset has been performed in this paper using deep convolutional neural networks. Accuracy of 80.66% and root mean square error of 10.18 knots were obtained (Pradhan et al., 2018).

Single Shot Multibox Detector technique has been used to determine the damage caused by Hurricane Sandy that occurred in 2012. A Convolutional Auto-Encoder consisting of VGG16 has been used for determining the damage from post-disaster images. mF1 and mAP of 56.025 and 44.83 were obtained respectively (Li et al., 2019).

A semi-supervised classification method has been proposed. This method was particularly useful in case of few labelled samples and large unlabelled samples. The method consisted of three steps: segmentation, convolutional auto-encoder and fine-tuning using CNN. The approach was applied to Hurricane Sandy and an accuracy of 88.3% was achieved (Li et al., 2018).

Three different neural networks have been used for detection of damage to buildings. The first neural network is used for pre-processing while the second and third neural networks have been used for extraction of deep features from the images. Several combinations of these three networks were considered and it was found that taking colour masks of relevant objects performed better for determining the damage (Nia & Mori, 2018).

A stacked CNN has been applied for determination of damage caused by the hurricane Dorian. The severity of the disaster has also been examined. Images have been taken from the unmanned aerial vehicles and a precision of 65.6% and accuracy of 61% has been obtained. This method is also useful when the information from the ground truth is not available (Cheng et al., 2021).

The earlier studies were conducted in order to forecast and estimate tropical cyclones and hurricane intensity. The benchmark dataset was made for hurricane damage detection and also transfer learning techniques were used in order to determine the damage caused to roads due to the hurricane. The summary of the literature survey has been shown in the table 1.

In this paper, the author has determined from the satellite images that whether there has been a damage or not to the buildings after the occurrence of Hurricane Harvey with a better accuracy than the earlier methods. Very less research has been done on damage to buildings due to the hurricanes. Also, greater number of images that is 23000 has been used in the damage detection.
3. MATERIALS AND METHODS

This segment presents the proposed methodology for hurricane damage detection and the dataset that has been used to validate the proposed model.

3.1 Dataset

For validation of the proposed method, the dataset for hurricane damage detection has been collected from Kaggle (IEEE Dataport). The examples of hurricanes of both damaged and non-damaged classes have been shown in figure 2. The dataset comprises 23000 satellite images which is further divided into training set, validation set and test set. The training set comprises 5000 images of each class that is damaged and undamaged. Validation set consists of 1000 images of each class. The division of the test set is done into the balanced and the unbalanced set. The balanced test set contains 1000 images of each class whereas the unbalanced test set contains 9000 out of which 8000 images of damaged and 1000 images of undamaged classes respectively.

3.2 Proposed Methodology

The proposed methodology for automatic detection of damage caused due to hurricanes is shown in figure 3. The model classifies the images into damaged and non-damaged using three important stages: pre-processing, processing and prediction. The pre-processing stage is further divided into normalization and augmentation. Each step is described in detail in the following sections. The CNN based models have been designed that is CNN Model A and CNN Model B which have been elaborated in sections 3.2.2 and 3.2.3. The models have been trained and simulated with different learning rates and number of epochs. The models are then tested and the performance parameters are evaluated.
3.2.1 Pre-Processing

The pre-processing part in image processing is the most important part that improves the features in satellite images and suppresses unnecessary information in image (Scannell et al., 2020) (Zheng et al., 2018). The pre-processing step is further divided into normalization and data augmentation.

**Normalization:** - The normalization process is of scaling up or scaling down the range of pixel value in an image before it becomes useful for further stages. Its function is to normalize every feature in an image so as to maintain the contribution of each feature since some features have greater pixel value than others. In this way, the network becomes unbiased in nature to the higher value features or pixel value. The normalization of pixel values has been done in between 0 and 1.

**Data augmentation:** - The data augmentation step is done to increase the diversity of input data for the proposed model without actual collection of new data (Wang & Perez, 2017). Rotation and flip operation have been performed to augment the data (Huynh et al., 2020) (Shin et al., 2016). The images were rotated randomly and flipping operation was performed with 50% probability for left-right shift (horizontal flip) and up-down shift (vertical flip).

3.2.2 Design of Layer 14 CNN Model A

CNN model is designed for the detection of damage caused due to hurricane Harvey (Albawi et al., 2017). The model consists of fourteen layers which are: five convolutional layers, five pooling layers, one flattening layer and two dense layers as shown in figure 4. An input image of 128*128*3 is applied to a convolutional layer consisting of 32 filters of size 3*3 that generates an output image of 126*126*32 size with 896 parameters. This is followed by max pooling layer with pool size of 2*2 that generates an output of 63*63*32 size. The second convolutional layer consists of 64 filters that gives an output image size of 61*61*64. Convolutional layer of 128 filters follows the second pooling layer. Another max pooling layer follows this convolutional layer and gives an output of 14*14*128.

It is further followed by two convolutional layers of 128 and 256 filters and a max pooling layer. Finally, the model consists of a flattening layer and two dense layers.
Table 2 describes the size of filter, number of filters, input image size, output image size and total number of parameters used at each layer for the proposed model.

### Table 2. Parameters of CNN Model A with 14 Layers

| S. no. | Layers            | Input Image Size              | Filter Size | No. of Filter | Activation function | Output Size              | Parameters |
|--------|-------------------|------------------------------|-------------|---------------|---------------------|--------------------------|------------|
| 1      | Input Image       | 128*128*3                   | --          | --            | --                  | --                       | --         |
| 2      | Convolutional     | 128*128*3                   | 3*3         | 32            | ReLU                | 126*126*32               | 896        |
| 3      | Convolutional     | 126*126*32                  | 3*3         | 64            | ReLU                | 124*124*64               | 18496      |
| 4      | Convolutional     | 124*124*64                  | 3*3         | 128           | ReLU                | 122*122*128              | 73856      |
| 5      | Max Pooling       | 122*122*128                 | Pool size   | --            | --                  | 40*40*128                | 0          |
| 6      | Convolutional     | 40*40*128                   | 3*3         | 128           | ReLU                | 38*38*128                | 147584     |
| 7      | Max Pooling       | 38*38*128                   | Pool size   | --            | --                  | 19*19*128                | 0          |
| 8      | Convolutional     | 19*19*128                   | 3*3         | 256           | ReLU                | 17*17*256                | 295168     |
| 9      | Max Pooling       | 17*17*256                   | Pool size   | --            | --                  | 5*5*256                  | 0          |
| 10     | Flatten           | 5*5*256                     | --          | --            | --                  | 6400                     | 0          |
| 11     | Dense             | 6400                        | --          | --            | ReLU                | 512                      | 3277312    |
| 12     | Dense             | 512                         | --          | --            | sigmoid             | 1                        | 513        |

### 3.2.3 Design of Layer 12 CNN Model B

This CNN model consists of twelve layers which are: five convolutional layers, three max pooling layers, one flattening layer and two dense layers as shown in figure 5. Initially the input is provided to a block of three convolutional layers with 32, 64 and 128 filters respectively. These layers are succeeded by a max pooling layer with pool size of 3*3. Further, the output is given to another convolutional layer of 128 filters followed by a max pooling layer. The fifth and the last convolutional layer consists of 256 filters and gives an output of 17*17*256. Lastly, max pooling layer, flattening and two dense layers are applied to the output of the convolutional layer.

Table 3 describes the size of filter, number of filters, input image size, output image size and total number of parameters used at each layer for the 12-layer CNN-B model which has been compared with the proposed model.

### 4. RESULTS AND DISCUSSION

In this segment, the results obtained from two CNN models designed from scratch have been presented. The two CNN models comprised 14 layers and 12 layers. Satellite images of 128*128 sizes were given as input. The batch size was set as 64. Both the models were trained for 15 and 30 epochs.
Adam optimizer was chosen as the optimization technique with a learning rate of 0.0001 and 0.001 which was also decided empirically. Evaluation of the performance parameters of both the models was done by evaluation metrics (Betz et al., 2011; Denil et al., 2013; M & M.N, 2015; Ng et al., 2020) like accuracy, F1-score.

Firstly, the performance parameters of CNN A model have been analysed at different learning rates and number of epochs. Then the performance of CNN B model has been analysed on the basis of number of epochs and learning rate. A comparative analysis of CNN A and CNN B has also been presented. Lastly, the best proposed model i.e. the CNN A model has been compared with the state of the art models.
4.1 Hyperparameter Tuning

The models have been simulated with different learning rates and number of epochs.

Learning rate is a parameter that helps in controlling how much the model should change. An optimal learning rate which is neither too small nor too large needs to be chosen since a very small learning rate can take too long to train whereas a very large learning rate will help in learning faster but at the cost of optimal weights (Liu et al, 2019).

The number of epochs too need to be optimally chosen so that there is no overfitting or underfitting.

4.2 Performance Parameter Analysis of CNN A Model

The performance parameters for the 14 layer CNN A model are analysed on the basis of learning rate and number of epochs in this section. The performance parameters analysed include the accuracy, F1-score, precision, false positive rate (FPR), sensitivity, specificity and false negative rate (FNR). Since, classification of images into damaged and undamaged classes is being performed in this paper, these popular classification parameters have been used for the analysis of the proposed CNN models.

(a) Analysis with Different Learning Rate with 15 Epochs

Here, the performance parameter analysis on the basis of learning rate i.e. 0.001 and 0.0001 at 15 epochs as shown in table 4. From the table, it is clear that there is an increase in the accuracy to 90.82%, F1-score to 91.89%, sensitivity to 91.60%, specificity to 89.80%, precision to 92.18% and a decrease in false positive rate to 10.20% and false negative rate to 8.40% due to the decrease in
learning rate from 0.001 to 0.0001. From the results, it can be inferred that an optimal learning rate which is neither too small nor too large gives optimal results.

(b) Analysis with Different Learning Rate with 30 Epochs

Table 5 shows the analysis of the performance parameters for the CNN A model on the basis of learning rate for 30 epochs. With an increase in epochs from 15 to 30 and decrease in learning rate, there is an increase in the percentage of accuracy, F1-score, sensitivity, specificity, precision and a decrease in false positive and negative rate. The accuracy increases from 90.43% to 92.91%, F1-score increases from 90.16% to 93%, sensitivity increases from 90.23% to 93.34%, specificity increases from 90.61% to 92.47% and precision increases from 90.09% to 92.65%. A decrease is obtained in the false positive rate and false negative rate. The false positive rate decreases from 9.39% to 7.53% and false negative rate decreases from 9.77% to 6.66%. The best results have been obtained for the CNN A model for learning rate of 0.0001 and 30 number of epochs.

(c) Analysis with Different Learning Rates and different Epochs

Table 4. Performance parameters of CNN A on the basis of Learning Rate (15 Epochs)

| Metric    | Learning Rate |     |     |
|-----------|---------------|-----|-----|
|           | 0.001         | 0.0001 |
| Accuracy  | 86.04         | 90.82 |
| F1-score  | 85.36         | 91.89 |
| Sensitivity | 84.49       | 91.60 |
| Specificity | 87.48       | 89.80 |
| Precision | 86.25         | 92.18 |
| FPR       | 12.52         | 10.20 |
| FNR       | 15.51         | 8.40  |

Table 5. Performance parameters of CNN A on the basis of Learning Rate (30 Epochs)

| Metric    | Learning Rate |     |     |
|-----------|---------------|-----|-----|
|           | 0.001         | 0.0001 |
| Accuracy  | 90.43         | 92.91 |
| F1-score  | 90.16         | 93   |
| Sensitivity | 90.23       | 93.34 |
| Specificity | 90.61       | 92.47 |
| Precision | 90.09         | 92.65 |
| FPR       | 9.39          | 7.53  |
| FNR       | 9.77          | 6.66  |
From the figure 6, it can be analysed that the highest accuracy, F1-score, precision, specificity, sensitivity and lowest false negative rate and false positive rate are obtained for 30 epochs at a learning rate of 0.0001. Greater number of epochs help to solve the under fitting problem and an optimal learning rate decides how fast or slow to arrive at the results.

4.3 Performance Parameter Analysis of CNN B Model

CNN B model comprises 12 layers including five convolutional layers, three max pooling layers, one flattening layer and two dense layers. This section includes the analysis of the classification performance parameters such as accuracy, F1-score, precision, false negative and positive rate on the basis of learning rate and the number of epochs.

(a) Analysis with Different Learning Rate with 15 Epochs

Learning rate is a parameter that helps in controlling how much the model should change. An accuracy of 89.44% is obtained for learning rate 0.0001 whereas an accuracy of 84.53% is obtained for 0.001 learning rate at 15 epochs. Similarly, there is an increase in F1-score, sensitivity, specificity, precision and decrease in false positive and negative rate with a decrease in learning rate.

(b) Analysis with Different Learning Rate with 30 Epochs

Table 7 presents the values of performance parameters of the model CNN B on the basis of number of epochs. There is not a significant increase in the accuracy, F1-score, specificity, precision with decrease in the learning rate at 30 epochs. Also, there is not a significant decrease in the values of false positive and negative rate with the learning rate. An optimal learning rate which is neither too small nor too large needs to be chosen since a very small learning rate can take too long to train whereas a very large learning rate will helping in learning faster but at the cost of optimal weights. The number of epochs too need to be optimally chosen so that there is no overfitting or underfitting.

(c) Analysis with Different Learning Rate and different Epochs
From the figure 7, it can be inferred that best results are obtained for a learning rate of 0.0001 at 30 epochs. Highest accuracy of 91.07%, F1-score of 91.51%, sensitivity of 92.52%, specificity of 89.49%, precision of 90.51% and lowest value of false positive rate of 10.51% and false negative rate.

Table 6. Performance parameters of CNN B on the basis of Learning Rate (15 Epochs)

| Metric      | Learning Rate |
|-------------|---------------|
|             | 0.001         | 0.0001        |
| Accuracy    | 84.53         | 89.44         |
| F1-score    | 85.87         | 90.37         |
| Sensitivity | 85.74         | 90.68         |
| Specificity | 83.07         | 87.95         |
| Precision   | 85.99         | 90.06         |
| FPR         | 16.93         | 12.05         |
| FNR         | 14.26         | 9.32          |

Table 7. Performance parameters of CNN B on the basis of Learning Rate (30 Epochs)

| Metric      | Learning Rate |
|-------------|---------------|
|             | 0.001         | 0.0001        |
| Accuracy    | 90.83         | 91.07         |
| F1-score    | 91.18         | 91.51         |
| Sensitivity | 92.84         | 92.52         |
| Specificity | 88.75         | 89.49         |
| Precision   | 89.58         | 90.51         |
| FPR         | 11.25         | 10.51         |
| FNR         | 7.16          | 7.48          |

Figure 7. Analysis with different Learning Rates and different Epochs for CNN B
of 7.48 are obtained. The results achieved were better than the results obtained by other combinations of learning rate and number of epochs.

4.4 Comparative Analysis of performance of CNN A and CNN B Model

From the figure 6 and 7, it can be inferred that both the CNN A model and CNN B model outperformed when the learning rate was 0.0001 and the number of epochs were 30. Hence, in this section, a comparative analysis of both the models has been done for 0.0001 learning rate and 30 epochs. CNN A model outperformed the CNN B model as it has more layers than the CNN B model. CNN A model comprises of 14 layers while CNN B model comprises of 12 layers.

The comparative analysis of both the models has been done on the basis of accuracy, F1-score, precision, specificity and sensitivity. As shown in figure 8, the higher accuracy of 92.91%, F1-score of 93%, sensitivity of 93.34%, specificity of 92.47% and precision of 92.65% is achieved for CNN A model as compared to the results obtained by the CNN B model.

Earlier a lot of work has been done in the field of machine learning in which the features are extracted from the data manually. Currently, deep learning field is gaining importance as there is automatic feature extraction in the deep learning field. CNN based models that is CNN A model and CNN B model have been used for the classification of satellite images in this paper since CNN models provide very good results in terms of accuracy. The models have been chosen in such a way so that they are not too deep in nature but shallow models as the images have to be classified into two classes that is damaged and undamaged. CNN A model achieved the best results as it obtained an accuracy of 92.91%.

There is a scope of improvement in the results that have been achieved by the use of alternate CNN models and appropriate combinations of learning rates and number of epochs. Also, the model could be made more generalized to determine damages caused due to other disasters.

4.5 Comparison of the best Proposed CNN-A Model with State of Art Models

The proposed model has been compared with the state of art models. The results of the proposed model are taken at learning rate 0.0001 and 30 epochs. The proposed CNN model comprising two classes: damage and no damage with 23000 images has obtained the highest accuracy of 92.91% as compared to the other state of the art models. The proposed model has been used for detection of damage caused after hurricane Harvey. Dotel et al [11] proposed a model with 18474 images and used to detect damage caused by hurricane Harvey has achieved an accuracy of 84.5%. Li et al [14] proposed a model that achieved an F1-score of 56.025 which is very less as compared to the F1 score of 93% obtained in this paper. Intensity was estimated by Pradhan et al [13] and accuracy of 80.66% was obtained. Li et al [15] found out damage caused due to hurricane Sandy and obtained an accuracy of 88.3%.
5. CONCLUSION

This paper presents the detection of damage caused due to Hurricane Harvey that struck the Texas region in the year 2017. An automated CNN based model was proposed with 14 layers consisting of five convolutional layers, five pooling layers, a flattening layer and two dense layers which provided better results than the CNN model with 12 layers. Learning rate of 0.0001 and 30 epochs which proved that the proposed model provided better classification of the satellite images of hurricane into damage and no-damage classes. This classification would help emergency crews and responders to provide relief aid to the people affected by the hurricane. The emergency managers would be able to thus allocate necessary resources to the affected people.

The images data used in this study are specific to the buildings of Texas region which was affected by the Hurricane Harvey. So, a more generalized model could be built which would be applicable to images of other hurricanes in other regions.

In this paper, an accuracy of 92.91% has been achieved which could be further improved by use of other alternate models. Also, transfer learning models could also help in improving the results obtained for assessing the damage caused due to the hurricane.
REFERENCES

Albawi, S., Mohammed, T. A. M., & Alzawi, S. (2017). Layers of a Convolutional Neural Network. IEEE.

Betz, J. M., Brown, P. N., & Roman, M. C. (2011). Accuracy, precision, and reliability of chemical measurements in natural products research. Fitoterapia, 82(1), 44–52. doi:10.1016/j.fitote.2010.09.011 PMID:20884340

Cao, Q. D., & Choe, Y. (2020). Building damage annotation on post-hurricane satellite imagery based on convolutional neural networks. Natural Hazards, 103(3), 3357–3376. doi:10.1007/s11069-020-04133-2

Chen, S. E., Escay, A., Haberland, C., Schneider, T., Staneva, V., & Choe, Y. (2018). Benchmark Dataset for Automatic Damaged Building Detection from Post-Hurricane Remotely Sensed Imagery. https://arxiv.org/abs/1812.05581

Dawood, M., Asif, A., & Minhas, F. (2020). Deep-PHURIE: Deep learning based hurricane intensity estimation from infrared satellite imagery. Neural Computing & Applications, 32(13), 9009–9017. doi:10.1007/s00521-019-04410-7

Denil, M., Shakibi, B., Dinh, L., Ranzato, M., & De Freitas, N. (2013). Predicting parameters in deep learning. Advances in Neural Information Processing Systems, 1–9.

Doshi, J., Basu, S., & Pang, G. (2018). From Satellite Imagery to Disaster Insights. Nips. https://arxiv.org/abs/1812.07033

Dotel, S., Shrestha, A., Bhusal, A., Pathak, R., Shaky, A., & Panday, S. P. (2020). Disaster Assessment from Satellite Imagery by Analysing Topographical Features Using Deep Learning. PervasiveHealth: Pervasive Computing Technologies for Healthcare, 86–92. 10.1145/3388818.3389160

Duarte, D., Nex, F., Kerle, N., & Vosselman, G. (2018). Satellite image classification of building damages using airborne and satellite image samples in a deep learning approach. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 4(2), 89–96. doi:10.5194/isprs-annals-IV-2-89-2018

Huynh, E., Hosny, A., Guthier, C., Bitterman, D. S., Petit, S. F., Haas-Kogan, D. A., Kann, B., Aerts, H. J. W. L., & Mak, R. H. (2020). Artificial intelligence in radiation oncology. Nature Reviews. Clinical Oncology, 17(12), 771–781. doi:10.1038/s41571-020-0417-8 PMID:32843739

Kaur, S., Gupta, S., Singh, S., & Gupta, I. (2021). Detection of Alzheimer’s Disease Using Deep Convolutional Neural Network. International Journal of Image and Graphics. Advance online publication. doi:10.1142/S021946782140012X

Kovordányi, R., & Roy, C. (2009). Cyclone track forecasting based on satellite images using artificial neural networks. ISPRS Journal of Photogrammetry and Remote Sensing, 64(6), 513–521. doi:10.1016/j.isprsjprs.2009.03.002

Li, Y., Hu, W., Dong, H., & Zhang, X. (2019). Building damage detection from post-event aerial imagery using single shot multibox detector. Applied Sciences (Switzerland), 9(6), 1128. Advance online publication. doi:10.3390/app9061128

Li, Y., Ye, S., & Bartoli, I. (2018). Semisupervised classification of hurricane damage from postevent aerial imagery using deep learning. Journal of Applied Remote Sensing, 12(04), 1. doi:10.1117/1.JRS.12.045017

Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., & Han, J. (2019). On the variance of the adaptive learning rate and beyond. arXiv preprint arXiv:1908.03265.

M, H., & M.N, S. (2015). A Review on Evaluation Metrics for Data Classification Evaluations. International Journal of Data Mining & Knowledge Management Process, 5(2), 1–11. 10.5121/ijdkp.2015.5201

Ng, B., Quinete, N., & Gardinali, P. R. (2020). Assessing accuracy, precision and selectivity using quality controls for non-targeted analysis. The Science of the Total Environment, 713, 136568. doi:10.1016/j.scitotenv.2020.136568 PMID:31955085

Nia, K. R., & Mori, G. (2018). Building damage assessment using deep learning and ground-level image data. Proceedings - 2017 14th Conference on Computer and Robot Vision, 95–102. doi:10.1109/CRV.2017.54

Phung, V. H., & Rhee, E. J. (2019). A High-accuracy model average ensemble of convolutional neural networks for classification of cloud image patches on small datasets. Applied Sciences (Switzerland), 9(21), 4500. Advance online publication. doi:10.3390/app9214500
Pi, Y., Nath, N. D., & Behzadan, A. H. (2020). Convolutional neural networks for object detection in aerial imagery for disaster response and recovery. *Advanced Engineering Informatics, 43*(September), 101009. 10.1016/j.aei.2019.101009

Pradhan, R., Aygun, R. S., Maskey, M., Ramachandran, R., & Cecil, D. J. (2018). Tropical Cyclone Intensity Estimation Using a Deep Convolutional Neural Network. *IEEE Transactions on Image Processing, 27*(2), 692–702. doi:10.1109/TIP.2017.2766358 PMID:29185987

Pritt, M., & Chern, G. (2018). Satellite image classification with deep learning. *Proceedings - Applied Imagery Pattern Recognition Workshop*, 1–7. doi:10.1109/AIPR.2017.8457969

Scannell, C. M., Veta, M., Villa, A. D. M., Sammut, E. C., Lee, J., Breeuwer, M., & Chiribiri, A. (2020). Deep-Learning-Based Preprocessing for Quantitative Myocardial Perfusion MRI. *Journal of Magnetic Resonance Imaging, 51*(6), 1689–1696. doi:10.1002/jmri.26983 PMID:31710769

Shin, H. C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., & Summers, R. M. (2016). Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning. *IEEE Transactions on Medical Imaging, 35*(5), 1285–1298. doi:10.1109/TMI.2016.2528162 PMID:26886976

Wang, J., & Perez, L. (2017). The effectiveness of data augmentation in image classification using deep learning. *ArXiv.*

Zheng, X., Wang, M., & Ordieres-Meré, J. (2018). Comparison of data preprocessing approaches for applying deep learning to human activity recognition in the context of industry 4.0. *Sensors (Switzerland), 18*(7), 2146. Advance online publication. doi:10.3390/s18072146 PMID:29970873

Cheng, C. S., Behzadan, A. H., & Noshadravan, A. (2021). Deep learning for post-hurricane aerial damage assessment of buildings. *Computer-Aided Civil and Infrastructure Engineering.*