Univ.AI

# Hurricane Damage Classifier

Team MANDALORIANS: Akshay, Raushan, Rahul and Sai

## Motivation and About the Project

After a hurricane, damage assessment is critical to the emergency response team and first responders. To improve the efficiency and accuracy of damage assessment, instead of driving around the affected area and noting down manually, we propose to detect damaged buildings using image classification algorithms automatically. This technique could give the stakeholders useful information about the severity of the damage to plan and organize necessary resources. In this project, we propose to use a convolutional neural network (CNN) to automatically detect 'Damaged' vs. 'Undamaged' buildings in the area during Hurricane using satellite imagery (Figure 1). With our custom network, we can achieve 97.51% accuracy (Figure 2) on the test set.

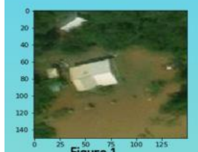

Figure 1

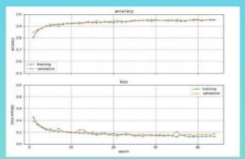

Figure 2

## Data and Labels

Hurricane dataset was provided by UnivAI and has the following subfolders: train_another, validation_another, test_another and test. We performed EDA on the dataset to check the distribution of images(Fig 3), colors(Fig 4) and labels(Fig 5) . From fig 3 we can see the data in train and validation is balanced but images in damage and no damage is unbalanced. We can also see (Fig 4) the RGB colors are balanced.

Figure 3

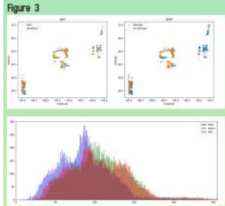

Figure 5

## References

* UnivAi Dataset.
* Exercise problems
* Kaggle

## Model

### Challenges :

Hurricane damage detection problem has multiple challenges. The image dataset provided is of very low resolution, and objects aren't much clearer. Images are of the only top view and have some inconsistencies since its satellite image. For NGOs and volunteers, the manual task of going through the satellite dataset can be very tiresome and may end up with errors.

### Methodology :

We decided to make sure the shape of all the images is consistent. We used the data generator and image augmentation , dropouts, L2 regularization etc to avoid overfitting problem.We agreed to experiment with multiple architectures, including SOTA as a convolution base, some hyperparameter tuning, and present the outcomes in tabular fashion.

### Implementation :

We trained the models using the Keras library tensorflow as backend. We have used the google collab for training the model. We tried the different models including SOTA models as base models(VGG16, InceptionV3 and MobileNet). We also looked at image at layer by layer to see how models are interpreting the images. We used data generator and image augmentation to increase the number of images and to introduce variance. At last we tested the accuracy on both balanced and unbalanced test data.

Figure 6: Bad image quality

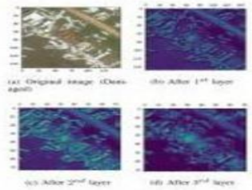

Figure 7: Image from 1 CNN filter

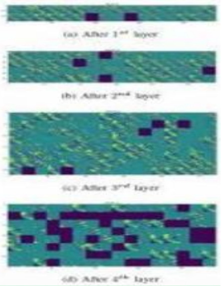

Figure 8: across all filters

## Result

The table below shows the performance of various models. The best test accuracy performance was observed with transfer learning models as base using Adam optimizer. We achieved test accuracy of ~98%

| Model | Validation Acc | Test Acc(Balanced) | Test Acc(Unbalanced) |
|---|---|---|---|
| CNN | 96.90% | 96.90% | 96.80% |
| CNN+DataAug+Dropout | 93.62% | 94.00% | 97.50% |
| VGG16+DataAug+Dropout | 91.50% | 91.70% | 92.30% |
| VGG16+DataAug+Dropout,Adam | 96.88% | 97.20% | 97.30% |
| Mobilenet+DataAug+Dropout | 98.12% | 98.20% | 98.41% |
| InceptionV3+DataAug+Dropout | 94.88% | 94.20% | 94.12% |

## Conclusion and Future Work

Using Deep Learning, we can automatically classify the damaged buildings correctly. The models can be further generalized for other disasters as well or can be used as a base model for transfer learning. Balanced and clear image quality can help improve accuracy. This solution can also be enhanced to detect the severity level and guide the First Responders toward which area to cover first. Some problems like flooded areas detection, food distribution in floods, roof detection, and others can also be done similarly and help save people's lives during calamities.

### Graphs and Plots of Best Model