Flood Detection Using Machine Learning

First Author[#], Second Author[*], Third Author[#]

[#]*Department of Computer Science, Lovely Professional University*

Abstract— Our world has seen many disasters in its life span, some are natural and other are man made. In this research paper we are going to talk about a natural disaster named as Flood. It is consider as one of the most prone natural disaster occur near to water flow especially near to river areas. GIS technology is used top visualize the extent of flood and help to analysis the consequences and risk of this disaster. The purpose of this paper is to analyse and detect the happening of flood using GIS dataset with the help of Machine Learning Model. These model help to know the probability of occurrence of flood in the area by using GIS dataset. The GIS dataset here is of Sentinel 1. The research papers are very helpful method to analysis the occurrence flood in and help authority to take preventive measure for it.

Keywords— Sentinel 1, Machine Learning, Sequnetial Model,

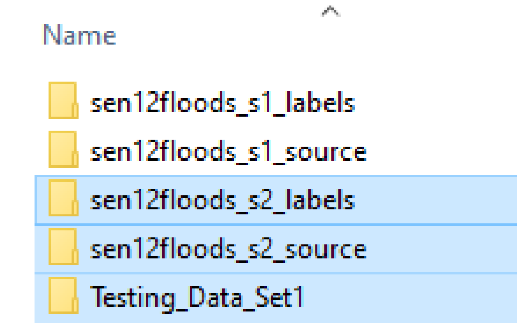
A. *Introduction*

Flood is an overflow of water on the land. It is not only the most powerful natural disaster but also the most effective disaster which cause adverse effect on life of human, animal plant and other organisms as well. In this research paper firstly we had done study on flood prone area by using **Sentinel 2** images and then we design a model for flood detection using Machine Learning. In this model we had use datasets with training sample of **1679** and testing sample of **268** of Sentinel 2 (.tiff) images downloaded from:

https://mlhub.earth/data/sen12floods
(Link Of Datasets)

How the Initial Dataset is Organized

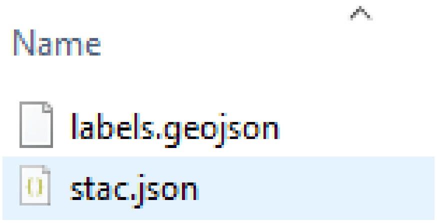
The dataset from Sentinel 2 satellite is splited into 2 main folders, the one with the actual images and another one with the label.Below we illustrates how these two folders look like on a local machine



Name
sen12floods_s1_labels
sen12floods_s1_source
sen12floods_s2_labels
sen12floods_s2_source
Testing_Data_Set1

Fig 1:
![Images and Labels folders](/images/Images_Labels.png)

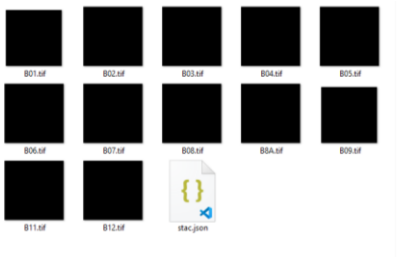
If we then open the folder containing the images the result looks like the following image. Each image is stored in a separate folder.
The name of the folder indicates the different locations with the prefix "source_1" and the date of sensing.



Name
labels.geojson
stac.json

Fig 2: ![Images folder 1](/images/Images_1.png)

The next image demonstrates the contents of one of this folders. As we can see the image is splitted into 13 spectral bands. In the following steps we will have to stack those bands into a single multi-spectral image.



B01.tif B02.tif B03.tif B04.tif B05.tif
B06.tif B07.tif B08.tif B8A.tif B09.tif
B11.tif B12.tif stac.json

Fig 3: ![Images folder 1](/images/Images_2.png)

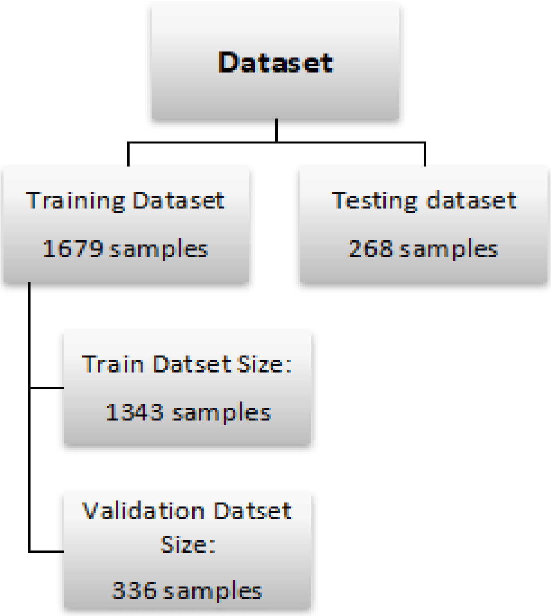


Fig 4: Dateset Hierarchy

B. *Author Name and Details*

S.No	TOPIC	Link	Author	Review
1.	FLOOD PREDICTION USING MACHINE LEARNING MODELS: LITERATURE REVIEW.	https://www.mdpi.com/2073-4441/10/11/1536	Amir Mosavi Pinar Ozturk: Norwegian University of Science and Tech Kwok Wing Chau: The Hong Kong Polytechnic University	Describe the brief information about how the Flood analysis helps us to overcome the damage and reduce the chances of loss cause by flood.
2	FLOOD PREDICTION USING MACHINE LEARNING MODELS:	https://www.researchgate.net/publication/335094911_Flood_Prediction_Using_Machine_Learning_Models_Literature_Review	Amir Mosavi Pinar Ozturk: Norwegian University of Science and Tech Kwok Wing Chau: The Hong Kong Polytechnic University	Describe the information about the use of Machine Learning in Flood Analysis and use of best model of Ml to predict the probability of occurrence of flood.
3	A FLOOD PREDICTION SYSTEM DEVELOPED USING VARIOUS MACHINE LEARNING ALGORITHMS	https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3866524	Kwok Wing Chau: The Hong Kong Polytechnic University	Describe the methodology used in flood analysis. This paper clearly suggest to use SVM and Random Forest Classification for ML models
4	FLOOD FORECASTING USING MACHINE LEARNING METHODS	https://www.researchgate.net/publication/331479872_Flood_Forecasting_Using_Machine_Learning_Methods	Fi-John Chang National Taiwan University Kuolin Hsu University of California, Irvine Li-Chiu Chang Tamkang University	

C. *Methodology*

All hypertext links and section bookmarks will be removed from papers during the processing of papers for publication. If you need to refer to an Internet email address or URL in your paper, you must type out the address or URL fully in Regular font.

Dataset's Name: SEN12-FLOOD : A SAR and Multi-spectral Datasets for Flood Detection

The observed areas correspond to 337 locations (cities and their surroundings) where a flood event occurred during the considered period.

Specifications of the Sentinel-1 satellites:

* 7 year lifetime (12 years for consumables)
* Launcher: Soyuz
* Near-polar (98.18°) Sun-synchronous orbit
* 693 km (431 mi) altitude
* 12-day repeat cycle
* 175 revolutions per cycle
* 98.6 minute orbital period
* 3-axis altitude stabilization
* 2,300 kg (5,100 lb) launch mass
* 3.9 m × 2.6 m × 2.5 m (12.8 ft × 8.5 ft × 8.2 ft) dimensions.

For machine learning purpose first we need to divide the image data into 2 parts : testing_dataset and training_dataset. Then we dign a function to read the json file from labels data set(training_dataset). function read the json_data from .json.

If image(training_data) contains flood then it returns the number 1 otherwise it return 0. Here 1=flood and 0 = no flood. This 0 and 1 label iterate through folders and search for the image alone with their corresponding label. The load_data unction is responsible for the constructing the dataset in a way compatible. Then we get the number of samples i.e=1679 and these samples play a role to train our machine learning model. Same with testing dataset and get the model tested on the sample we had. Now we explore that how many samples had flood and how many are cleaned in both training_dataset and testing_dateset

Model Creation:

Using keras_model Sequential and keras_layers Dense, Conv2D, MaxPool2D, Flatten and Dropout we had designed a deep learning model and create a confusion_matrix. By exporting the model's architecture onto the image. Compile and fit the model on the training_dataset to get the accuracy of model and repeat the same on testing_dataset.

The accuracy of each model and calculation of loss function is a result we get . We use Epoch to get more accuracy and pot the same on graph.

D. Result:

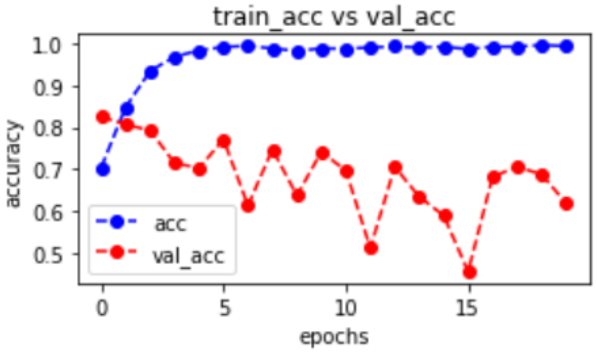


Fig 5: Model's Accuracy Graph (before epoch)

This plot shows the result of train_acc (blue color) and val_acc (red color). The train_accuracy (train_acc) is increasing as the number of epoch increases , on the other hand the validation_data_accuracy (val_acc) is decreasing as the epoch increases.

The sudden fluctuation in val_acc is due to quality of training_data in that batch which ocntain some noise and cause sudden raise or fall in val_acc.

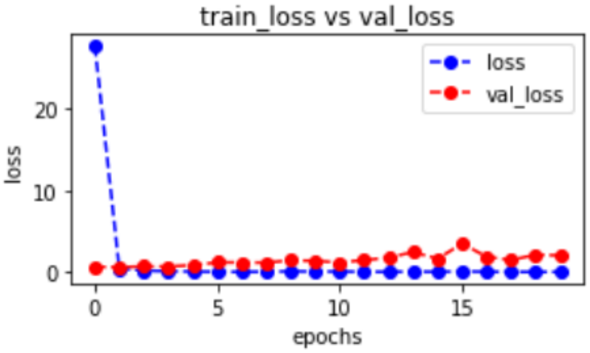
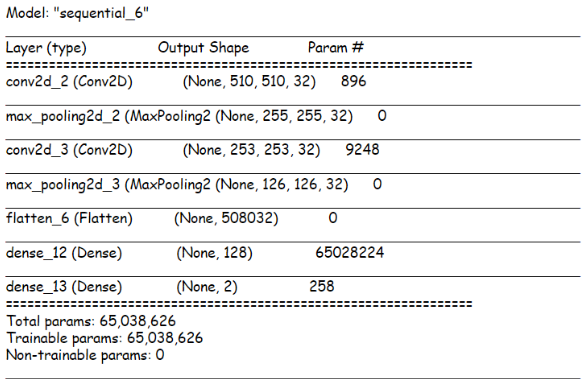


Fig 5: Model's Loss Plot (before epoch)

As the accuracy of the model increase the loss cause in it decreases. This loss here shows that the times when model predict the bad result.

The val_loss increases with more epoch , shows the model is struggling a little bit on validation_data.

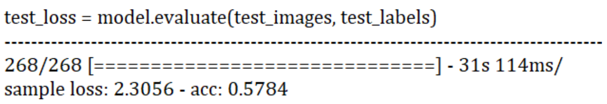


```
Model: "sequential_6"

Layer (type)              Output Shape            Param #
=================================================================
conv2d_2 (Conv2D)          (None, 510, 510, 32)     896

max_pooling2d_2 (MaxPooling2 (None, 255, 255, 32)     0

conv2d_3 (Conv2D)          (None, 253, 253, 32)     9248

max_pooling2d_3 (MaxPooling2 (None, 126, 126, 32)     0

flatten_6 (Flatten)        (None, 508032)           0

dense_12 (Dense)           (None, 128)             65028224

dense_13 (Dense)           (None, 2)               258
=================================================================
Total params: 65,038,626
Trainable params: 65,038,626
Non-trainable params: 0
```

Fig 6: Layer Summary and Param

The layer summary gives the detailed output shape of every layer type we had used in this model and also shows the parameter involve in that particular layer.

At last it shows the total number of params (parameters) involve in model I.e. 65,038,626. and 0 non-trainable params show the dataset is training_dataset.



```
test_loss = model.evaluate(test_images, test_labels)
------------------------------------------------------------------------------------
268/268 [==============================] - 31s 114ms/
sample loss: 2.3056 - acc: 0.5784
```

Fig 7: Test_loss using test_images and test_labels (before feature extraction)

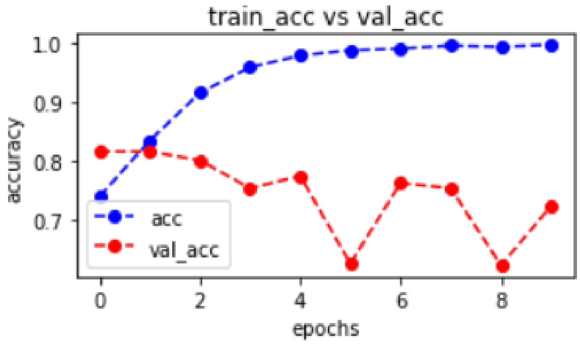
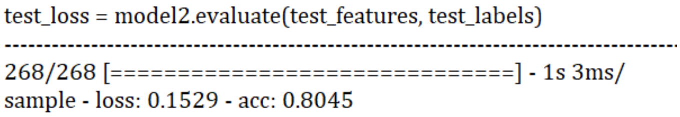


Fig 8t: Model's Accuracy Graph (after epoch)



```
test_loss = model2.evaluate(test_features, test_labels)
------------------------------------------------------------------------------------------
268/268 [==============================] - 1s 3ms/
sample - loss: 0.1529 - acc: 0.8045
```

Fig 8: Test_loss using test_features and test_labels (after feature extraction)

References

The heading of the References section must not be numbered. All reference items must be in 8 pt font. Please use Regular and Italic styles to distinguish different fields as shown in the References section. Number the reference items consecutively in square brackets (e.g. [1]).

When referring to a reference item, please simply use the reference number, as in [2]. Do not use "Ref. [3]" or "Reference [3]" except at the beginning of a sentence, e.g. "Reference [3] shows …". Multiple references are each numbered with separate brackets (e.g. [2], [3], [4]–[6]).

Examples of reference items of different categories shown in the References section include:

- example of a book in [1]
- example of a book in a series in [2]
- example of a journal article in [3]
- example of a conference paper in [4]
- example of a patent in [5]
- example of a website in [6]
- example of a web page in [7]
- example of a databook as a manual in [8]
- example of a datasheet in [9]
- example of a master's thesis in [10]
- example of a technical report in [11]
- example of a standard in [12]

I. Conclusions

The version of this template is V2. Most of the formatting instructions in this document have been compiled by Causal Productions from the IEEE LaTeX style files. Causal Productions offers both A4 templates and US Letter templates for LaTeX and Microsoft Word. The LaTeX templates depend on the official IEEEtran.cls and IEEEtran.bst files, whereas the Microsoft Word templates are self-contained. Causal Productions has used its best efforts to ensure that the templates have the same appearance.

Acknowledgment

The heading of the Acknowledgment section and the References section must not be numbered.

Causal Productions wishes to acknowledge Michael Shell and other contributors for developing and maintaining the IEEE LaTeX style files which have been used in the preparation of this template. To see the list of contributors, please refer to the top of file IEEETran.cls in the IEEE LaTeX distribution.

References

[1] S. M. Metev and V. P. Veiko, *Laser Assisted Microtechnology*, 2nd ed., R. M. Osgood, Jr., Ed. Berlin, Germany: Springer-Verlag, 1998.

[2] J. Breckling, Ed., *The Analysis of Directional Time Series: Applications to Wind Speed and Direction*, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61.

[3] S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok, "A novel ultrathin elevated channel low-temperature poly-Si TFT," *IEEE Electron Device Lett.*, vol. 20, pp. 569–571, Nov. 1999.

[4] M. Wegmuller, J. P. von der Weid, P. Oberson, and N. Gisin, "High resolution fiber distributed measurements with coherent OFDR," in *Proc. ECOC'00*, 2000, paper 11.3.4, p. 109.

[5] R. E. Sorace, V. S. Reinhardt, and S. A. Vaughn, "High-speed digital-to-RF converter," U.S. Patent 5 668 842, Sept. 16, 1997.

[6] (2002) The IEEE website. [Online]. Available: http://www.ieee.org/

[7] M. Shell. (2002) IEEEtran homepage on CTAN. [Online]. Available: http://www.ctan.org/tex-archive/macros/latex/contrib/supported/IEEEtran/

[8] *FLEXChip Signal Processor (MC68175/D)*, Motorola, 1996.

[9] "PDCA12-70 data sheet," Opto Speed SA, Mezzovico, Switzerland.

[10] A. Karnik, "Performance of TCP congestion control with rate feedback: TCP/ABR and rate adaptive TCP/IP," M. Eng. thesis, Indian Institute of Science, Bangalore, India, Jan. 1999.

[11] J. Padhye, V. Firoiu, and D. Towsley, "A stochastic model of TCP Reno congestion avoidance and control," Univ. of Massachusetts, Amherst, MA, CMPSCI Tech. Rep. 99-02, 1999.

[12] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, IEEE Std. 802.11, 1997.