Flood Prediction in the Lower Cape Fear River Using SAR Based Water Extraction

D. McMoran*, A. Langevin, A. Whittaker, P. Poosapati, N. Pricope, and G. Dogan

***Abstract*—The effects of climate change, including severe droughts, fires, and extreme weather events, are increasing every year. One of the most dangerous natural disasters around the world is flooding. Remote sensing and machine learning offer opportunities to utilize remotely collected data for analysis and predictive modeling purposes. Using Python programming methods, Sentinel-1 Synthetic Aperture Radar (SAR) data was collected for water pixel detection from Google Earth Engine and combined with National Oceanic and Atmospheric Administration (NOAA) precipitation data to understand seasonal flood events between 2015 and early 2022. Models were developed to predict flooding in the Lower Cape Fear and greater Wilmington, North Carolina area. One of the challenges in this method is the lack of imagery data recorded within the Region of Interest and time frame as well as the interplay between different data types, programming methods, and outputs. Outputs include data tables and multiple methods for data validation. Overall, accuracy for resulting models was high, with an artificial neural network for binary classification returning an accuracy value greater than 90%.**

***Index Terms*—Flood detection, machine learning, predictive modeling, SAR**

I. Introduction

The effects of climate change, including severe droughts, fires, and extreme weather events, are increasing every year. One of the most dangerous natural disasters around the world is flooding, which causes an average of 4.8 billion USD per event in the United States alone [1]. The risk for flood events is not distributed evenly and is additionally complicated by desirable population centers and their critical infrastructure being located along coasts and waterways [1]. Wilmington, located along the lower Cape Fear River in North Carolina, is such a population center, and has been impacted by severe flooding. Understanding the risk of flooding is critical for city planning, insurance, recovery efforts, infrastructure, and reducing costs during extreme weather and flooding events [2].

Manuscript received November 23, 2022; revised January 10, 2023; accepted February 16, 2023; published October 26, 2023.

D. McMoran currently works as a Data Scientist for Bowman Consulting based out of Virginia, USA

A. Langevin works as an Associate Software Engineer at nCino, inc in Wilmington North Carolina, USA

A. Whittaker works as a Software Engineer at CGI Federal in Durham, North Carolina, USA

P. Poosapati worked at Capgemini Technologies Limited as a Cloud Developer in Azure with 6 Years of Experience, USA

N. Pricope currently works as a Professor of Geography and Geospatial Science at the University of North Carolina Wilmington, Wilmington, North Carolina, USA.

G. Dogan is currently an Assistant Professor in the Computer Science department at the University of North Carolina Wilmington (UNCW), where she founded the Applied AI Lab.

*Correspondence: dogangulus@gmail.com (D.M.)

Since the first LandSat satellite was launched 50 years ago, remote sensing has been used to collect data and monitor environmental conditions [3]. As time has progressed, satellites have been continuously constructed in a more contemporary style, built with state-of-the-art sensors for recording information. With this progression, satellite missions have greatly expanded due to the usefulness of the data they provide to scientists globally. An impediment that satellites forgo during severe weather scenarios, is the inability to penetrate cloud cover to the events transpiring on the ground. This is deeply problematic and restricts the understanding of the disaster at ground level [3, 4]. This impacts emergency response within the immediate aftermath, and also reduces the amount of data on the event for future research and analysis. Copernicus Sentinel-1, a satellite launched by the European Space Agency (ESA), offers a solution with its Synthetic Aperture Radar (SAR), which allows for imagery acquisition from satellites regardless of the weather conditions [4].

Google Earth Engine offers multiple remote sensing data sets from a collection of satellites free of cost. Using JavaScript, code can be created to filter Sentinel-1 imagery from specific date ranges and optimize a selection of specific features, or values, using algorithms. SAR is notable for being able to detect environmental change, particularly of water, through cloud cover [4]. By creating an algorithm to highlight water pixels within a selected Region of Interest (ROI) within the greater Wilmington, NC area, the numbers of pixels will change between each image capture date. This data can then be exported as a CSV file for integration into Python scripts for data visualization, deep learning, and predictive flood modeling efforts for the lower Cape Fear River. This methodology has the potential to be applied to other areas that are prone to flooding or experiencing droughts, as well as to analyze trends in water levels of large bodies of water and create a predictive model from the data retrieved. This would become useful to other researchers looking for a low cost and open-source solution to sourcing data for flood predictive modeling. Using machine and deep learning to analyze the output data allows for the development of models, both classification and regression, towards the goal of flood predictive modeling

II. Related Works

The United Nations Office for Outer Space Affairs has produced documentation for using SAR for flood mapping and damage assessments as part of a standardized process to allow countries to avail themselves of the technology for disaster relief, which was used to develop the JavaScript code to extract data from Google Earth Engine [5]. Additionally, Bao, Lv, and Yao and Manavalan discuss the advantages of

DOI: 10.7763/IJMO.2023.V13.838

SAR for flood monitoring and disaster management in their respective works but worry about the difficulties of capturing water in complex areas such as urban environments when compared to simpler locations such as rivers, lakes, and oceans [6, 7]. This also was a concern for our project and our efforts alongside the Cape Fear River.

Examining the interplay in socio-economic characteristics and flood modeling using remote sensing imagery is another methodology for determining and quantifying flood vulnerability, particularly as it pertains to understanding flood plains and mitigation planning measures [8]. All of these models, however, do not involve an element of deep learning or prediction to allow for an enhanced understanding of changing environmental conditions beyond the current flood plain models, which our project attempts for remote sensing imagery data.

In 2012, Skakun of the Space Research Institute NASUNASU researched flood mapping by applying artificial neural networks in his article "A Neural Network Approach to Flood Mapping Using Satellite Imagery" [9]. Skakun specifically utilized self-organizing Kohonen's maps (SOMs), for SAR image segmentation and classification. He applied this research to flood events around the Tisza River, Huaihe River, Mekong River, and the Koshi river. The research was focused on mapping flood areas based on past SAR imaging; however, it did not explore predictive modeling. The tests were conducted on three different sensors and classifications rates were 85.40% or higher.

In 2015, Elkhrachy published his research "Flash Flood Hazard Mapping Using Satellite Images and GIS Tools: A case study of Najran City, Kingdom of Saudi Arabia (KSA)" [10]. As the title suggests, Elkhrachy used SAR technology to map areas of flash flooding within Najran City. The goal of Elkhrachy's research was to identify known areas of flash flooding within a given region to prevent the building of infrastructure within a hazardous zone. Similar to Skakun's research, Elkhrachy's goal was to use past events to inform present decisions rather than develop predictive models.

III. Data Sets

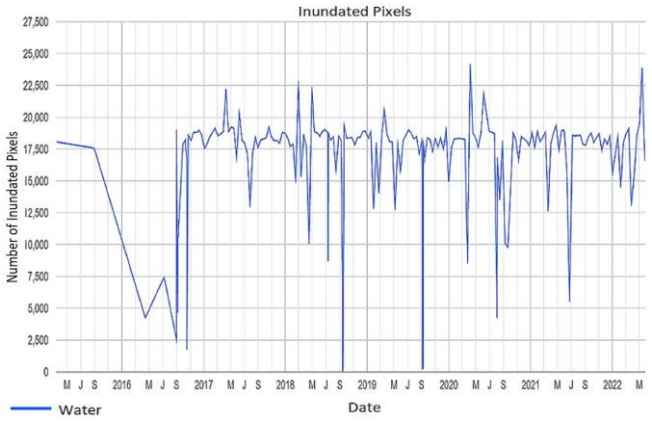

Fig. 1. Lower Cape Fear River SAR chart of water pixel values from January 2015 until June 2022.

Data sets utilized for this project include water pixel data downloaded in CSV format from Google Earth Engine, as seen in Fig. 1. Additional weather station precipitation data was downloaded from NOAA in CSV format to aid in imputation of date for dates between image captures. A look at the data set is provided in Table I.

TABLE I: Examination of Original Data Set and Existing Parameters

Column Name	Data Type	Description
Water	float64	Float, numeric, number of water pixels
Month	Int64	Integer, numeric, month of data capture
Day	Int64	Integer, numeric, day of data capture
Year	Int64	Integer, numeric, year of data capture
Station	Object	Categorical, non-numeric, identification of NOAA weather station
Name	Object	Categorical, non-numeric, name of NOAA weather station
Latitude	Float64	Float, numeric, latitude of NOAA weather station
Longitude	Float64	Float, numeric, longitude of NOAA weather station
Total Precipitation	Float64	Float, numeric, combined precipitation in mm

IV. Methodology

Copernicus *Sentinel-1* data from the European Space Agency for the date range 1 January 2015 through 1 June 2022 was captured using a set spatial geometry to capture a Region of Interest (ROI) comprising the greater Wilmington, North Carolina area as seen in Figure 2. Filtering was applied to smooth outputs and ensure correct classification of water pixels. A time series chart was generated to collect data in a CSV format for use in Python scripting for deep learning. Manual removal of records that did not have full imagery coverage of the ROI was performed to ensure total capture of water pixels for the date of capture. Precipitation data from NOAA weather monitoring stations for the greater Wilmington region was brought in and joined with the water pixel data prior to pre-processing for machine and deep learning. In order to account for water pixel data missing from the join, an algorithm was developed to fill in pixel data based on the mean of the adjacent cell values. The joined data set was then filtered using the bounding geometry to exclude weather stations and precipitation information outside the bounds of the ROI. The data was then explored to determine a proper break point to classify the data as flooded or not flooded using Python.

By examining the statistical information within the data set in the form of tables and using regression analysis, water pixels were able to be separated by percentages within the ROI. Respectively, this created the split between 1 and 0 to be rounded up to 19,000 from the 75% metric of 18,662.68. This was done to ensure flooding was accurately captured, rather than artificially inflating the amount of flooded versus non flooded records in the data set. The categorical information within the data set (Station, Name) were dropped, as were the coordinate data for weather station capture (Latitude, Longitude) as they had no further bearing on modeling the flooding target column. This information was additionally excluded from supervised classification and regression analyses performed. Categorical encoding was performed using OneHotEncoder to generate integer indices

from the integer value columns (Month, Day, Year).

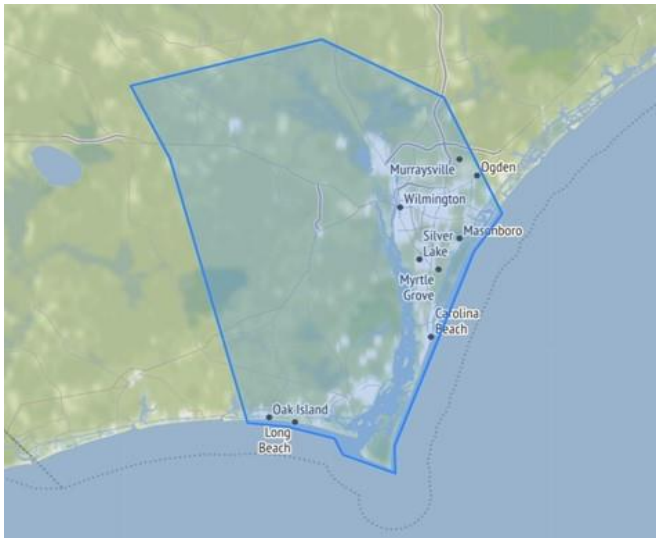

Fig. 2. Region of interest for SAR capture.

V. Experimental Design

In structuring a machine learning model, multiple algorithms may be chosen to determine overall fitness for the data set including through measures such as accuracy. Due to data volume, or bias, over-fitting may occur. Deep learning can be employed to fine tune parameters to achieve optimal performance. Deep learning, a subset of machine learning, can group, or cluster, components of a data set to reveal correlated relationships [12]. Unlike machine learning, this requires a "more is less" approach, meaning more data points leading to improved outcomes. Tensorflow and Keras provide methods for performing binary classification on data sets which fit our data series. By creating a binary classification for flooded (1), or not flooded (0), based on a break point in the data, the model can be compiled to search through the 30,000 plus records. This compilation allows the model to accurately place records into the appropriate category. Selecting the proper density of features, optimizers, and metrics will allow for optimal performance of the model. Comprised of node layers, artificial neural networks (ANNs) utilize multiple layers to simulate the effects of neurons in the human brain. These layers include input, hidden, and output, which are interconnected and bear a weight and threshold for each specific node layer. Outputs above the threshold are activated, while those below remain inactive and do not pass information to the next layer [13]. Binary classification, classifying a target feature into one of two possible categories, is a fairly common method for machine learning and deep learning. We implemented both machine learning and deep learning experiments to observe the best options for classifying water pixel data as flooded or not flooded, with additional efforts directed towards regression analysis. The first experiment was classification using multiple algorithms to predict flooding, the second was binary classification, and the final was regression using multiple algorithms. Evaluation metrics for classification included accuracy, precision, recall, and F1 scores, while our regression metrics included R2 scores, root mean square error (RMSE), and mean absolute error (MAE). Fig. 3 illustrates our binary classification model structure. Batch normalization was performed to standardize data distribution, which was applied to numerical data columns. This process had the additional effect of creating 6 non trainable parameters out of a total of 615, with 609 being trainable. The model uses the input shape of 17, with an output set to 32 and a dropout rate of 0.5 to aid in the prevention of overfitting. The Keras activation was set to the rectified linear unit (relu) function, applying the maximum of 0 and the input tensor. Once the model was compiled, the Adam algorithm was chosen for optimization due to its efficiency and suitability for large data sets [14]. Binary cross-entropy was used to determine loss between true and predicted labels, and accuracy was used as a metric to determine the fitness of the overall model.

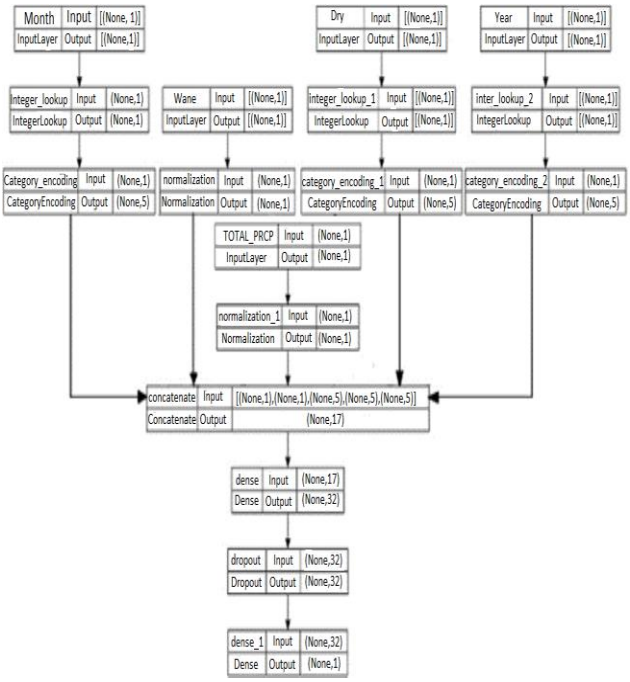

Fig. 3. Binary Classification Model Architecture.

VI. Experimental Results

One of the primary goals of this study was to develop predictive modeling for flooding within a given data set. This was achieved using both binary classification and regression analysis, with the results from the regression analysis presented in Table II. Supervised classification outputs, available in Table III, illustrate the results of an overfitted model, which leads to the results of our binary classification model in the form of plots, Figs. 4 and Figs. 5 respectively. Accuracy percentage for this model was 99.96%

VII. Discussion

Deep learning is the optimal solution when machine learning models result in overfitting. This can happen for multiple reasons, one of which being the complexity of the model relative to the training data noisiness. Simplifying the model, either by reduction of attributes or parameters, are potential solutions to resolve overfitting. Gathering additional supporting data can also aid in resolving the issue by improving the training data, as can noise reduction [12].

As is evident in the supervised classification table (Table III), overfitting by consistently reaching 1 as a percentage indicates the model is in effect too simple, and therefore easily predicted by the algorithms utilized. Using binary classification and an ANN, the accuracy value of 99.96% is within the tolerance of 0 and 1 and is viewed as a high measure of accuracy for the output result. It is typical to expect single column or binary classification predictions to be a simpler effort for training and prediction when compared against predicting multiple columns or values. This is evident in the results from our efforts with Fig. 5. Loss plot for binary classification. regression analysis, where while R2 scores for K Neighbors exceed 0.70, RMSE values are nowhere near the 0.2 to 0.5 range indicative of an effective prediction, despite the accuracy achievement. In this scenario, a stronger RMSE score would suggest a better predictive model for hidden layers. While a simple model should have a greater likelihood of falling within acceptable metrics, a model can in effect be too simple, or lack enough features to allow for accurate prediction [12]. Despite the information presented for target columns, it is possible that additional information, or feature engineering to improve the currently available data, would improve the output measures of accuracy and precision. Another point to consider is the method for filling the missing water pixel information. Radhika and Shashi noted in their 2009 paper that real world databases are susceptible to missing data, which must be managed in such a way through cleaning or transformation to be usable for programmatic efforts [15]. The current method of using a mean of the values in the preceding and succeeding cells does not account for the amount of precipitation in a gain or loss, which may impact the amount of water pixels present for modeling. Additionally, this method may skew the data set towards lower or higher values, depending on the cells that need to be filled, and would thus affect the resulting model metrics.

TABLE II: REGRESSION MODELS (TRAIN AND TEST SET) FOR WATER PIXEL PREDICTIONS

Train Set Algorithms	R2 Score	RMSE	MAE
Decision Tree	0.894493	1402.516754	381.836544
GradientBoosting	0.894272	1403.979016	382.719715
KNeighbors	0.803813	1912.502560	667.870422
XGradientBoosting	0.592199	2757.344103	1632.200653
Linear	0.085509	4129.108142	2852.012299
Ridge	0.085509	4129.108142	2852.011792
Lasso	0.085509	4129.108573	2851.968489
ElasticNet	0.084684	4130.971525	2844.496592

Test Set Algorithm	R2 Score	RMSE	MAE
Decision Tree	0.734665	2250.347831	813.453782
GradientBoosting	0.661935	2540.108547	800.347733
KNeighbors	0.647154	2595.042602	822.579290
XGradientBoosting	0.609615	2729.599336	1623.389689
Linear	0.093018	4160.557077	2873.781205
Ridge	0.093015	4160.562714	2873.867227
Lasso	0.093015	4160.562808	2873.866682
ElasticNet	0.091355	4164.368621	2866.355405

TABLE III: CLASSIFICATION MODELS (TRAIN AND TEST SETS) FOR FLOOD PREDICTIONS

Train Set Algorithms	Accuracy	Precision	Recall	F1Score
K Nearest	1	1	1	1
Logistic	1	1	1	1
GradientBoosting	1	1	1	1
Decision Tree	1	1	1	1
AdaBostClassifier	1	1	1	1
SVC	0.972523	1	0.682986	0.81163
GaussianNB	0.906645	0.234927	0.034149	0.05963
BernoulliNB	0.913325	0	0	0

Test Set Algorithms	Accuracy	Precision	Recall	F1Score
K Nearest	1	1	1	1
Logistic	1	1	1	1
GradientBoosting	1	1	1	1
Decision Tree	1	1	1	1
AdaBostClassifier	1	1	1	1
SVC	0.974018	1	0.69000	0.81656
GaussianNB	0.906862	0.17037	0.02875	0.04919
BernoulliNB	0.916186	0	0	0

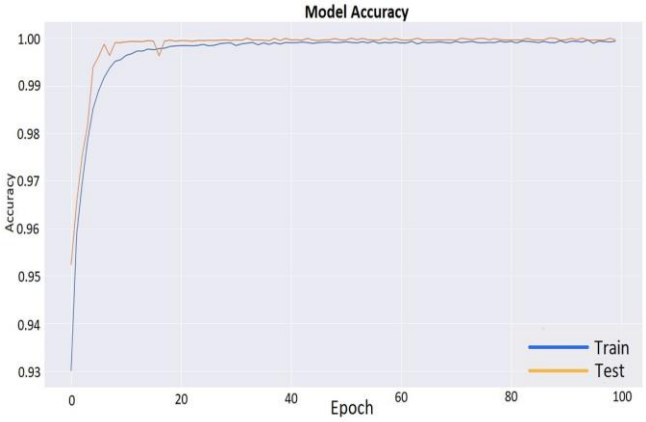

Fig. 4. Accuracy plot for binary classification.

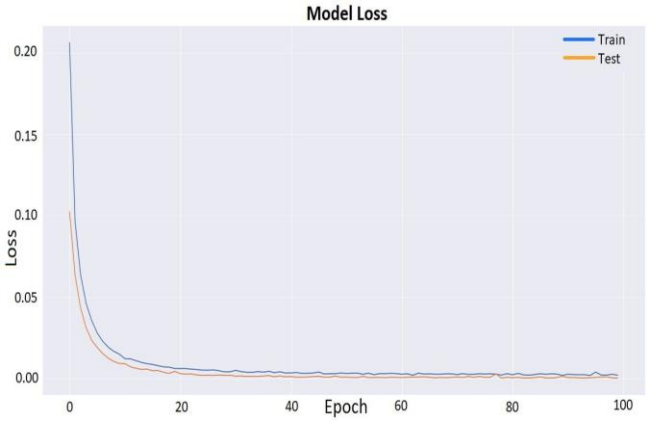

Fig. 5. Loss plot for binary classification

VIII. CONCLUSIONS

Building low-cost predictive models for flooding has multiple potential uses. During the course of this research effort, methods were developed for collecting data from SAR imagery through Google Earth Engine to be refined for machine and deep learning models. Ultimately, supervised

and binary classification provided overfitted and accurate results respectively, with an accuracy value greater than 90%. Regression analysis was less satisfactory, with an R2 value greater than 0.70 for the K Neighbors algorithm and RMSE values far exceeding the optimal 0.2 to 0.5 range. Additional efforts may be undertaken with feature engineering, additional data inclusion, or improvements for filling missing water pixel information. A limitation for this study design is the inclusion of a portion of the coastline adjacent to the greater Wilmington area which sees frequent change to water levels as a result of flood events. Potential future works could focus on a Region of Interest solely on the water body or a condensed area of concern, rather than on a greater regional area of concern. Additional limitations encountered are the filtering parameters which are specific to an area. Focusing on a greater regional area means that some areas will not show all flooding or may show too much flooding due to the filtering value chosen. This too could be remedied by focusing on a smaller ROI to fine tune a model, which may improve accuracy results for both classification and regression modeling efforts.

IX. Future Work

Predictive modeling for disaster events such as flooding has multiple uses, and the ability to achieve accurate predictive and eventually forecasting models is critical to developing mitigation strategies. Improvements to the current modeling efforts presented here, including the previously mentioned filling of missing water pixel data as well as the inclusion of additional features, will produce stronger and more accurate models. Additionally, these methodologies may be able to be applied with Convolutional Neural Networks (CNN) to retrieve imagery for analysis and prediction. By focusing on areas hit frequently by flooding, drought, or natural disasters, strong predictive models can begin to be used to develop forecasting models for continuing to understand our changing world and better prepare for these events.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

D. McMoran assumed the principal role in the investigation presented in this paper, spearheading data analysis and incorporating graphical and tabular representations for enhanced data visualization; A. Langevin was accountable for the paper's structural organization and composition, as well as the development of a tailored algorithm for the data structure employed in the study; A. Whittaker meticulously assessed and amended the document to ensure grammatical accuracy, content relevance, and adherence to required formatting standards; P. Poosapati contributed essential resources pertinent to the conducted research; N. Pricope offered valuable guidance grounded in her domain expertise and profound understanding of the project's research scope; G. Dogan established strategic milestones and a coherent timeline to facilitate the organized completion of the research, drawing on her own expertise and knowledge to contribute to the investigation. All authors have reviewed and endorsed the final version of this paper.

References

[1] N. Oceanic and A. Administration. “Billion-Dollar Disasters: Calculating the Costs.” (Jul. 2022), [Online]. Available: https://www.ncei.noaa.gov/access/monitoring/.

[2] P. Narcisa, C. Hidalgo, J. Pippin, and J. Evans, “Shifting landscapes of risk: Quantifying pluvial flood vulnerability beyond the regulated floodplain,” Journal of Environmental Management, vol. 304, p. 114 221, Feb. 2022. DOI: 10.1016/j.jenvman.2021.114221.

[3] L. Missions. “Landstat 1.” () [Online]. Available: https://www.usgs.gov/landset-missions/.

[4] E. S. Agency. “Sentinel-1.” (), [Online]. Available: https://sentinel.esa.int/web/sentinel/missions/sentinel-1.

[5] “United Nations Platform for Space-based Information for Disaster Management and Emergency Response (UN-SPIDER).” (), [Online] Available: https://www.unoosa.org/oosa/en/ourwork/un-spider/index.html

[6] L. Bao, X. Lv, and J. Yao, “Water extraction in sar images using features analysis and dual-threshold graph cut model,” Remote Sensing, vol. 13, no. 17, p. 3465, 2021.

[7] R. Manavalan, “SAR Image Analysis Techniques for Flood Area Mapping - Literature Survey,” Earth Science Informatics, vol. 10, Mar. 2017. DOI: 10.1007/s12145- 016-0274-2.

[8] P. Narcisa, H. Joanne, and R. Lauren, “Modeling residential coastal flood vulnerability using finished-floor elevations and socio-economic characteristics,” Journal of environmental management, vol. 237, 2019.

[9] S. Skakun, “A Neural Network Approach to Flood Mapping Using Satellite Imagery,” Computing and Informatics, vol. 29, no. 6, pp. 1013–1024, Jan.

[10] I. Elkhrachy, “Flash Flood Hazard Mapping Using Satellite Images and GIS Tools: A case study of Najran City, Kingdom of Saudi Arabia (KSA),” The Egyptian Journal of Remote Sensing and Space Science, vol. 18, no. 2, pp. 261–278, 2015, ISSN: 1110-9823. DOI: https://doi.org/10.1016/j.ejrs.2015.06.007.

[11] “Sentinel-1.” (), [Online]. Available: https://sentinel.esa.int/web/sentinel/missions/sentinel-1.

[12] A. Geron. “Hands-on machine learning with scikit-learn ´ & tensorflow, concepts, tools, and techinques to build intelligent systems.”(2017), [Online]. Available: https://ebookcentral.proquest.com/lib/uncw/detail.action?docID=4822582.

[13] I. C. Education, What are neural networks? Aug. 2020...

[14] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, 2014. DOI: 10.48550/ARXIV.1412.6980.

[15] Y. Radhika and M. Shashi, “Atmospheric temperature prediction using support vector machines,” International journal of computer theory and engineering, vol. 1, no. 1, p. 55, 2009.