Classification of Flood Warnings Applying a Convolutional Neural Network

Oscar-Alejandro García-Trujillo(✉), Luis Carlos Sandoval Herazo, Eddy Sánchez-DelaCruz, and Raymundo González Moreno

National Tech, Misantla Campus, Misantla, Veracruz, Mexico
gtrujillo.oa@gmail.com

Abstract. The effects of climate change create climatic temporal imbalances that favor the development of hydrometeorological phenomena and cause socioeconomic damage when they occur. In the absence of Early Warning Systems and dedicated monitoring stations, the effectiveness of a Convolutional Neural Network model is tested to interpret and label dataset on the climatic conditions of the Misantla's river basin and its surroundings, regarding to the flood hazard level. Aiming to classify rainfall events in the region using dataset collected through 3 weather stations around the region: northern zone of Veracruz, Mexico, specifically the municipality of Misantla. Neural networks can maximize the use of dataset collected by weather stations, providing a safer environment in the event of floods, and having a positive effect on the preservation of human activity. The dataset provided allows to label data as 'GREEN', 'YELLOW' and 'RED' with more than 95% accuracy, performing better when working with a large number of validation data, but also shows a slowdown during the integration of larger training data sets.

Keywords: Convolutional Neural Network · Precipitation · Early warning system · Flood risk determination · Flood labeling

1 Introduction

The effects of climate change have caused an imbalance in climatic temporality, favouring the development of hydrometeorological phenomena, causing socio-economic havoc as they occur [4, 10]. In addition, technological developments make it possible to implement weather classification models to alert the population to possible disasters, such as floods [14].

The Misantla river basin, has been affected by the aforementioned imbalance, an example of which, was its overflow when it was affected by hurricane "Roxane" in October 1995, causing extensive damage to a population of approximately 8,000 inhabitants [4]. The lack of adequate monitoring of the variables that cause the river to overflow made it impossible to mitigate its impact on the city of Misantla. In the absence of an Early Warning System (EWS) and dedicated monitoring stations, the population is left at the mercy of inclement weather.

Therefore, the use of a model based on Convolutional Neural Networks (CNN) is explored,a model that is able to interpret and label the dataset on the climatic conditions

O. Pichardo Lagunas et al. (Eds.): MICAI 2022, LNAI 13612, pp. 192–203, 2022.
https://doi.org/10.1007/978-3-031-19493-1_16

of the Misantla river basin and its surroundings, with respect to the present flood hazard level. This model functions as a EWS system supported by data from climate stations established at key points around the Misantla river basin.

The remainder of the article is divided as follows: in Sect. 2, information on different CNN applications in the field of flood prevention and weather forecasting can be seen. Section 3, contains the description of the dataset used, its processing and architecture 3.3 used to produce the results in Sect. 4 and their discussion 4.2. Finally, Sect. 5 presents conclusions and future work.

2 Related Work

This section describes related Convolutional Neural Network (CNN) work for weather forecasting, water body flow monitoring, flood warning and other elements. Larraondo et al. [13] demonstrate how CNNs can be used to interpret the output of Numerical Weather Prediction (NWP) automatically to generate local forecasts. aiming to prove that these models can capture part of the mental and intuitive process that human forecasters follow when interpreting numerical weather data.

Chen et al. [2] employ 3-Dimensional CNN and 2-Dimensional CNN as a method to better understand the spatial relationships of typhoon formation features. Fu et al. [6], apply a 1-Dimensional CNN in conjunction with Bi-directional Long Short-Term Memory in weather prediction because of the effectiveness of CNNs in extracting features and capturing short period connections between datasets.

Donahue et al [5], implement CNN as the basis of an Long-term Recurrent Convolutional Networks for its rapid progress on visual recognition problems, as well as its application to time-varying inputs and outputs.

Shi et al. [17], integrate convolutional layers in conjunction with an Fully Connected Long Short-Term Memory network to generate short-term precipitation forecasts as if it were a long-term spatio-temporal forecasting problem.

Zhang et al. [22], proposes deep architecture using a bidirectional 3-Dimensional CNN and Convolutional Long Short-Term Memory to transform video into a 2-D mapping of learning and classification features.

Han et al. [7], employs a CNN, converts the short-term forecasting problem into a binary classification problem and exploits the advantage of CNNs in its global pattern learning to demonstrate its superiority compared to other Machine Learning methods in short-term forecasting.

Pally, Jaku Rabinder Rakshit [16], developed a new python package called "FloodImageClassifier" including various CNNs architectures to classify and detect objects within the collected flood images, embeded in a smartly designed pipeline to train a large number of images and calculate flood water levels and inundation areas which can be used to identify flood depth, severity, and risk.

Nobuaki Kimura et al. [12], combine knowledge transfer and CNNs for time series flood predictions. Its application has a margin of error of less than 10 percent with respect to variation with water level.

Syed Kabir et al. [11] create a CNN model for fast real-time flood estimation and prediction, contrasting it with an Support Vector Machine model and showing the superiority of the former. Supported by data collected over a 10-year period (2005–2015).

Cho et al. [3] propose a model based on CNN classification to determine flood risk. This model is suggested as an initial study to determine time-optimal evacuation actions.

S. Smys et al. [18] through the use of Internet of Things (IoT) sensors that collect data and store it in the cloud, applies a CNN-based model that predicts the inflow of water into dams and lakes in order to prevent flash floods during rainfall events.

ChenChen et al. [1] offer a CNN deep learning-supported model for flood prediction in a watershed, with data collected over 10 years, via IoT devices.

Dostdar Hussain et al. [9] study a time series approach with CNN in the prediction of water flow in a river, comparing it with an Extreme Learning Machines (ELM) model.

Kou-Lin Hsu et al. [8] describe a system, which utilizes the computational strength and flexibility of an adaptive Artificial Neural Network model to estimate rainfall rates using infrared satellite imagery and ground-surface information.

Mhara [15], presents a two-layer CNN-based model, using 10 years of history for flood prediction in a catchment, considering temporal, geographical and trend characteristics. The model is intended to be used for water level monitoring through IoT and flood disaster prevention.

Zhang, C. et al. [21], offer the"Tiny-RainNet" model for short-term rainfall forecasting (nowcasting) in a period between 1 and 2 h, combining CNN and Bi-directional Long Short-Term Memory. The model takes into account the influence of spatio-temporal meteorological conditions, thus avoiding the cumulative error presented in conventional models.

WWang, Y. et al [19], apply two CNN-based models for flood susceptibility mapping that consider 13 flood triggers and demonstrates a higher accuracy of both models compared to an Support Vector Machine model. These models are intended to be used to assist in flood damage mitigation and management.

3 Methodology

3.1 Weather Stations

To collect weather data of Misantla municipality , in Veracruz, Mexico, three weather stations were deployed each in a specific location around Misantla. The deployment areas were selected considering principal points of rain runoff and the hydrological region "RH 27-Ae", where Misantla belongs.

The installed stations are three "ambient weather ws-2902a" liked to a wireless internet connection and powered by two AA batteries respectively, capable to gather indoor and outdoor data, creating each an online dataset available in https://ambientweather.net/ using the respective user and password to which the equipment is linked (Table 1).

3.2 Dataset

The dataset used, contains 10907 different inputs collected by the 3 weather stations over the past 2 years.

Table 1. Weather stations location table.

Station	Location	Coordinates
1	ITSM, Misantla, Veracruz, Mexico	725699N, 2207381E
2	Tenochtitlan, Veracruz, Mexico	718167N, 2191478E
3	Salvador Diaz Mirón, Veracruz, Mexico	723452N, 2187739E

Each input registered in the dataset is associated with a specific label regarding the flood warning that kind of record represents, "GREEN" labeled records mean "No risk of flood", "YELLOW" labeled records mean "Possible risk of flood" and "RED" labeled records mean "Imminent flood in the area". The dataset contains 3909 records labeled as "GREEN", 3501 records labeled as "YELLOW" and 3498 records labeled as "RED". Values in the dataset are used as listed in 2:

Table 2. Dataset data description

#	Value	Description
1	Simple date	Format "DD/MM/YYYY" is converted into integer values calculating the number of days between a given date and 1/1/1900.
2	Hour	Counts the whole hours passed in the current day.
3	Day	Registers the current day number.
4	Month	Registers the current month number.
5	Outdoor temperature	Logs air temperature in exteriors, expressed in degrees Celsius.
6	Hourly rain	Records millimeters of rain per hour, in the current hour.
7	Event rain	Tracks millimeters of rain between intervals lower than an hour. For instance, if 71 mm of rain are registered first and followed by 100 mm of rain in less than an hour, both are part of the same "Event Rain" interval.
8	Hourly rain	Records millimeters of rain per hour, in the current hour.
9	Daily rain	Counts millimeters of rain per day, in the current day, since midnight (00:00).
10	Weekly rain	Shows the amount of precipitation that has accumulated in the calendar week total, and resets on Sunday morning at midnight.
11	Monthly rain	Records millimeters of precipitation in the calendar month total, and resets on the first day of the Month.
12	Total rain	Is defined as the running total since station was powered up.
13	Outdoor humidity	Is the ratio of the current absolute humidity to the highest possible absolute humidity.
14	Solar radiation	Measures in Watts per square meter(W/m^2), radiant energy emitted by the sun from a nuclear fusion reaction that creates electromagnetic energy.
15	Label	Replaces titles like "GREEN", "YELLOW" and "RED" for numerical values 2, 0 and 1, respectively

The data cleansing process consists of the conversion of collected information into actionable numerical formats, converting values such as date, day, time and labels into standardized numerical values available for experimentation, in the dataset cleansing, incomplete records are erased, records that report "null" values due to the lack of data recorded by the weather stations, or periods in which the system reported data out of place because it was undergoing maintenance in an air-conditioned area.

For example, if the weather station required a battery change, or if a bird struck the weather station so hard it required fixing a sensor. The data collected during the maintenance hour reports null values in "Outdoor Temperature", "Outdoor Humidity" and "Solar Radiation", due to the lack of any listed field like the previous mentioned. The whole row is discarded.

Table 3. Dataset Sample

Simple date	Hour	Day	Month	Outdoor temperature (°C)	Hourly rain (mm/hr)	Event rain (mm)	Daily rain (mm)	Weekly rain (mm)	Monthly rain (mm)	Total rain (mm)	Outdoor humidity (%)	Solar radiation (W/m^2)	Label
44706	23	24	5	24	0	0	0	1.8	60.7	278.2	81	0	GREEN
44705.81	19	24	5	25.5	0	0	0	1.8	60.7	278.2	84	21.6	GREEN
44705.8	19	24	5	25.8	0	0	0	1.8	60.7	278.2	84	30.3	GREEN
44705.8	19	24	5	25.8	0	0	0	1.8	60.7	278.2	84	38.2	GREEN
44690.63	15	9	5	33.7	2175	171	171	172.8	231.7	449.2	63	655.5	YELLOW
44690.63	15	9	5	33.5	2176	196	196	197.8	256.7	474.2	62	672.4	YELLOW
44690.63	15	9	5	33.5	2177	220	220	221.8	280.7	498.2	64	666.2	YELLOW
44650.75	18	30	3	28.3	0	265	265	266.8	325.7	543.2	54	22.9	RED
44650.75	17	30	3	28.4	0	266	266	267.8	326.7	544.2	52	26.4	RED
44650.74	17	30	3	28.5	0	257	257	258.8	317.7	535.2	52	29	RED

From the values registered in Table 3, flood predictions are heavily based on "Event Rain", "Outdoor Humidity" and "Outdoor Temperature" since the millimeters of water registered keep increasing in a rain event, the "Outdoor Temperature" marks how fast water can evaporate in the area and "Outdoor Humidity" establishes the water evaporation ratio. The other values in the dataset are provided to the machine in order to find a correlation between all the different values without evaluate them isolated.

3.3 CNN Architecture

A CNN usually takes a tensor of order 3 as input [20], the input dataset goes through a set of processing, called layers, which may be grouping, convolution or normalization, as well as fully connected or loss layers.

$$\boldsymbol{x}^1 \longrightarrow \boxed{\boldsymbol{w}^1} \longrightarrow \boldsymbol{x}^2 \longrightarrow \cdots \longrightarrow \boldsymbol{x}^{L-1} \longrightarrow \boxed{\boldsymbol{w}^{L-1}} \longrightarrow \boldsymbol{x}^L \longrightarrow \boxed{\boldsymbol{w}^L} \longrightarrow z$$

Fig. 1. Abstract description of CNN structure

Figure 1 illustrates layer by layer how the dataset passes through a CNN, starting with the input layer $x^1$for processing in the tensor w^1. The output of the first layer is x^2, which also acts as input for the processing of the second layer. This processing continues until all the layers of the CNN have been completed, providing an output in x^L. The last layer is a loss layer. Let us assume that t is the corresponding target value for the input x^1, then a cost or loss function can be used to measure the discrepancy between the forecast x^L of CNN and the target t.

In the conducted experiment, a Google Collab Instance was used with the following specs:

- GPU: Tesla P100-PCIE-16 GB
- CPU: Intel(R) Xeon(R) CPU @ 2.00 GHz
- Socket(s): 1
- Core(s) per socket: 1
- Thread(s) per core: 2
- L3 cache: 39424K
- RAM: 13 GB

– DISK SPACE: 100 GB Free Space

The algorithm was run using Python 3.6 including the following libraries:

– tensorflow
– numpy
– pandas
– sklearn

The CNN was fully connected, implementing a Sequential model, "sparse_categorical_crossentropy" as loss function, "Nadam" as Optimizer, initializing with an input layer of 13 values, 10 layers of linear rectification (ReLU) and an output layer of 3 values, training in sets of 600 entries at a time and 20 epochs. The number of convolutions is specified in Table 4:

Table 4. CNN layer data

Layer #	Convolutions	Activation
1	169	ReLU
2	169	ReLU
3	36	ReLU
4	49	ReLU
5	64	ReLU
6	64	ReLU
7	49	ReLU
8	36	ReLU
9	169	ReLU
10	169	ReLU

The experiment was conducted 3 times, each one dividing the current dataset in different proportions (Table 5):

Table 5. Dataset segmentation values

Experiment #	Train data %	Test data %
1	70	30
2	60	40
3	80	20

4 Experiments and Results

The aim of this experiment is to test different distributions of a dataset for a better classification of rainfall events in Misantla, Veracruz, using a dataset of weather data from strategically installed weather stations in the surrounding area. Assessing the effectiveness of CNN models taking part in the process of interpreting numerical climate data, similar to human interpretation.

4.1 Validation

The following are the results of experiments carried out with a data rate of: 60% training data and 40% test data (Figs. 2 and 3).

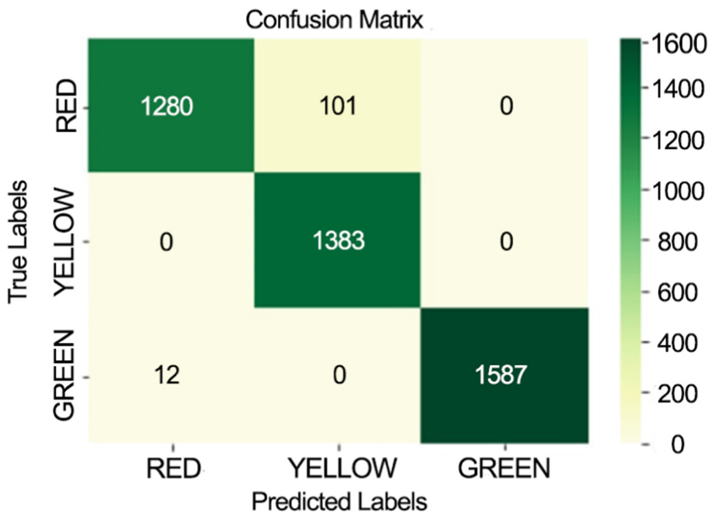

Fig. 2. Confusion matrix 60/40

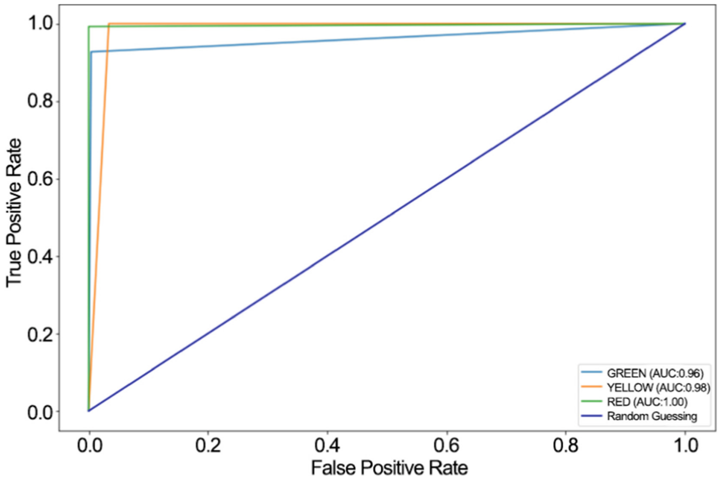

Fig. 3. ROC test 60/40

70% training data and 30% test data (Figs. 4 and 5).

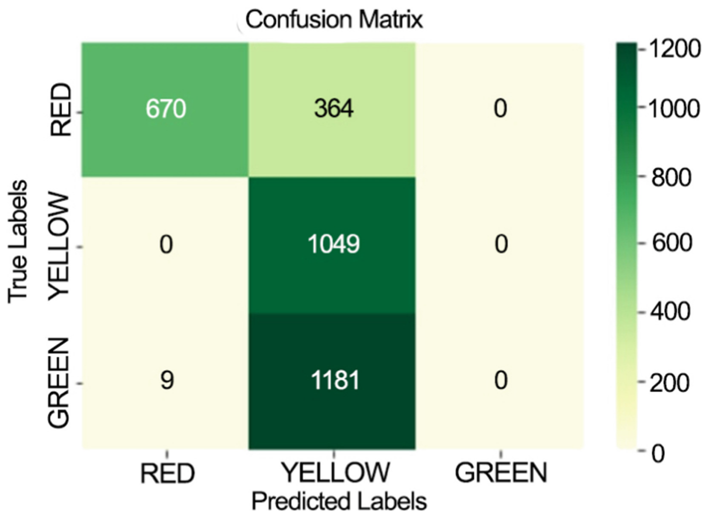

Fig. 4. Confusion matrix 70/30

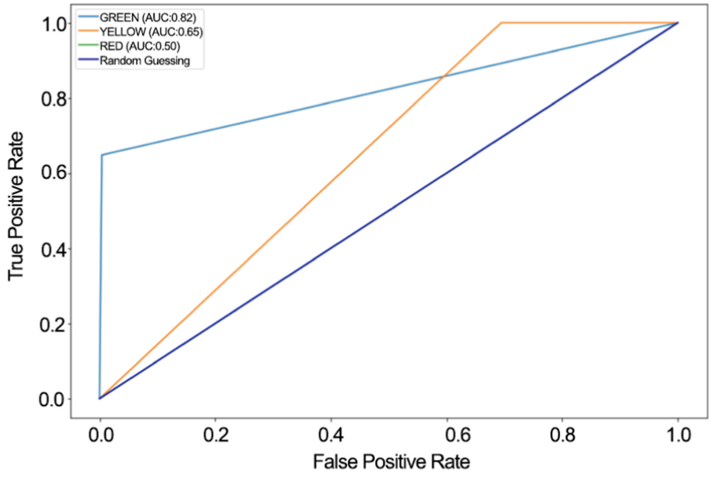

Fig. 5. ROC test 70/30

80% training data and 20% test data (Figs. 6 and 7).

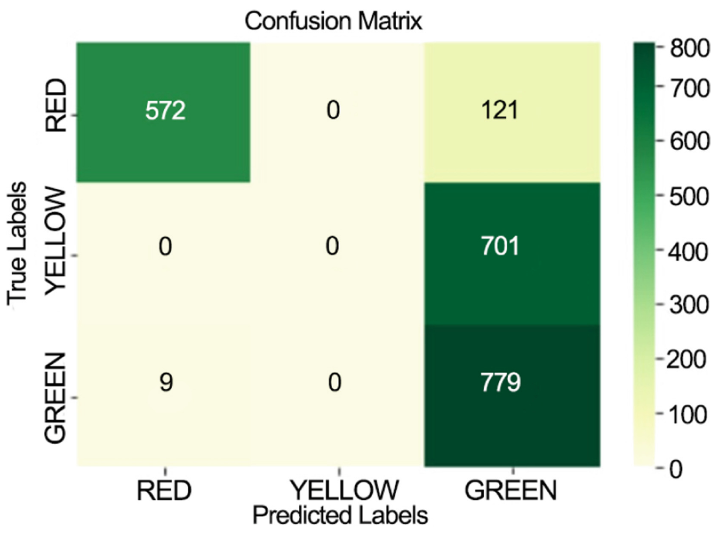

Fig. 6. Confusion matrix 80/20

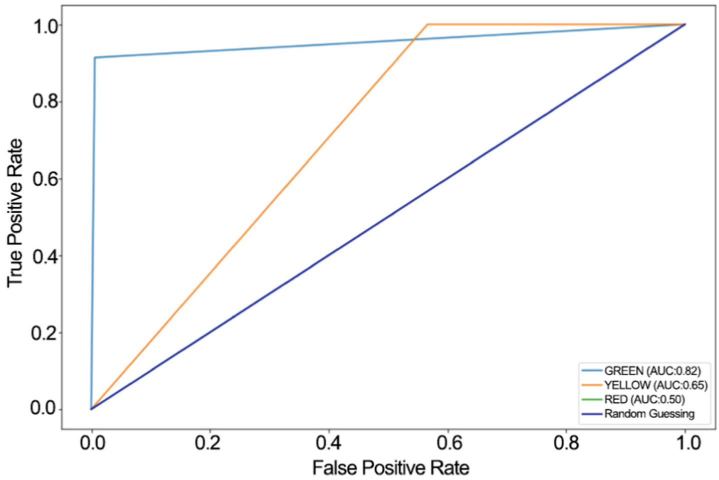

Fig. 7. ROC test 80/20

4.2 Discussion

The dataset provided performs better when working with a large number of validation data, but also shows a slowdown during the integration of larger training data sets.

It is understood that the system achieves adequate training and testing with a dataset ratio of 60/40 without disconnection between cells preparing it for a better integration with a validation subset, in contrast the number of labels mismatched increases in case of changing training and testing ratios. Furthermore, there is a difference in data accuracy between the three labels "GREEN", "YELLOW" and "RED", which seems to be related to the number of elements provided for each label, with the highest number of data provided by "GREEN" labeling, followed by "YELLOW" labeling and the lowest number by "RED" labeling.

Using Convolutional Neural Networks in a 'nowcasting' phenomenon as Flood forecasting, that is strongly dependent on recorded rainfall events, and real time monitoring allows to take advantage of variables that have little to negligible influence on its prediction and develop properly trained Early Warning Systems capable to use real-time inputs as a validation subset and keep track of possible incoming floods.

5 Conclusions and Future Work

Convolutional Neural networks are capable of monitor data collected by weather stations and identify inputs as possible flood warnings like a person could, removing the need of continuous human intervention, as well as the fast computing, prediction and response to flood warnings, providing a safe environment in the event of floods and having a positive effect on the preservation of human activity. Implementations of this nature offer opportunities for action in areas far from large cities or those that are in the process of development. Future applications include:

– Estimate duration of rainfall events
– Estimate water levels of recorded rainfall events
– Estimating water levels of predicted rainfall events
– Determining flood action intervals

References

1. Chen, C., Hui, Q., Xie, W., Wan, S., Zhou, Y., Pei, Q.: Convolutional neural networks for forecasting flood process in internet-of-things enabled smart city. Comput. Netw. **186**, 107744 (2021)
2. Chen, R., Wang, X., Zhang, W., Zhu, X., Li, A., Yang, C.: A hybrid cnn-lstm model for typhoon formation forecasting. GeoInformatica **23**(3), 375–396 (2019)
3. Cho, M., Kim, D., Jung, H.: Implementation of cnn-based classification model for flood risk determination. J. Korea Inst. Inf. Commun. Eng. **26**(3), 341–346 (2022)
4. CONAGUA: Análisis de las temporadas de huracanes de los años 2009, 2010 y 2011 en méxico. http://www.conagua.gob.mx/conagua07/publicaciones/publicaciones/cgsmn-2-12.pdf. Accessed June 2022
5. Donahue, J., et al.: Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
6. Fu, Q., Niu, D., Zang, Z., Huang, J., Diao, L.: Multi-stations' weather prediction based on hybrid model using 1d cnn and bi-lstm. In: 2019 Chinese control conference (CCC), pp. 3771–3775. IEEE (2019)
7. Han, L., Sun, J., Zhang, W.: Convolutional neural network for convective storm nowcasting using 3-d doppler weather radar data. IEEE Trans. Geosci. Remote Sens. **58**(2), 1487–1495 (2019)
8. Hsu, K.L., Gao, X., Sorooshian, S., Gupta, H.V.: Precipitation estimation from remotely sensed information using artificial neural networks. J. Appl. Meteorol. **36**(9), 1176–1190 (1997)
9. Hussain, D., Hussain, T., Khan, A.A., Naqvi, S.A.A., Jamil, A.: A deep learning approach for hydrological time-series prediction: a case study of gilgit river basin. Earth Sci. Inf. **13**(3), 915–927 (2020)
10. INEGI: Características hidrográficas. méxico. https://www.inegi.org.mx/temas/hidrografia/. Accessed June 2022
11. Kabir, S., Patidar, S., Xia, X., Liang, Q., Neal, J., Pender, G.: A deep convolutional neural network model for rapid prediction of fluvial flood inundation. J. Hydrol. **590**, 125481 (2020)
12. Kimura, N., Yoshinaga, I., Sekijima, K., Azechi, I., Baba, D.: Convolutional neural network coupled with a transfer-learning approach for time-series flood predictions. Water **12**(1), 96 (2019)
13. Larraondo, P.R., Inza, I., Lozano, J.A.: Automating weather forecasts based on convolutional networks. In: Proceedings of the ICML Workshop on Deep Structured Prediction, PMLR, vol. 70 (2017)
14. Marín Vilca, D.G., Pineda Torres, I.A.: Modelo predictivo machine learning aplicado a análisis de datos hidrometeorológicos para un sat en represas (2019)
15. Mhara, M.A.O.A.: Complexity neural networks for estimating flood process in internet-of-things empowered smart city. Available at SSRN 3775433 (2021)

16. Pally, J.R.R.: Application of image processing and convolutional neural networks for flood image classification and semantic segmentation (2021)
17. Shi, X., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.C.: Convolutional lstm network: a machine learning approach for precipitation nowcasting. Adv. Neural Inf. Process. Syst. **28**, 1–9 (2015)
18. Smys, S., Basar, A., Wang, H., et al.: CNN based flood management system with IoT sensors and cloud data. J. Artif. Intell. **2**(04), 194–200 (2020)
19. Wang, Y., Fang, Z., Hong, H., Peng, L.: Flood susceptibility mapping using convolutional neural network frameworks. J. Hydrol. **582**, 124482 (2020)
20. Wu, J.: Introduction to convolutional neural networks, vol. 5, no. 23, p. 495. National Key Lab for Novel Software Technology. Nanjing University, China (2017)
21. zhang, c., Wang, H., Zeng, J., Ma, L., Guan, L.: Tiny-rainnet: a deep cnn-bilstm model for short-term rainfall prediction (2019)
22. Zhang, L., Zhu, G., Shen, P., Song, J., Afaq Shah, S., Bennamoun, M.: Learning spatiotemporal features using 3d cnn and convolutional lstm for gesture recognition. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 3120–3128 (2017)