

# Spatiotemporal Analysis for Rainfall Prediction Using Extreme Learning Machine Cluster

Renaldy Fredyan <sup>a,\*</sup>, Muhammad Rizki Nur Majiid <sup>a</sup>, Gede Putra Kusuma <sup>b</sup>

<sup>a</sup> Computer Science Department, School of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia

<sup>b</sup> Computer Science Department, Graduate Program, Bina Nusantara University, Jakarta, 11480, Indonesia

Corresponding author: \*renaldy.fredyan@binus.ac.id

**Abstract**— Rainfall prediction is an essential study as a guideline for water resources management to manage disasters. Still, earlier research cares much about temporal information, only considering a single spatial location. The earth's land surface has a large area of spatial location, so to manage spatial information simultaneously as temporal, we use spatiotemporal data to analyze rainfall prediction more accurately. This study uses the spatiotemporal Extreme Learning Machines (ELM) Cluster to forecast rainfall using CHIRPS data from satellites and stations. Data consists of spatial two dimensions and temporal data from 1981 to 2020. The dataset for the experiment contains 480 months. We use focal operation for data preprocessing to the nearest neighbor value. Moreover, the ELM cluster can manage every spatial location by sharing the output weight of ELM, so there is no spatial information left behind. Then, comparing the spatiotemporal Extreme Learning Machines Cluster among SVR, Linear Regression, Gaussian, Ridge, and Lasso are used to predict the data on those timescales. The results indicate that spatiotemporal ELM-Cluster can accurately forecast rainfall. Using ELM-Cluster in hydrological rainfall forecasting is encouraging, and the model can practically be used. Evaluation using MAE with a score of 66.77 and RMSE, 83.77, getting the fastest training with only 28.9 seconds compared to the other methods due to the ELM Cluster does not have backpropagation with spatial improvement.

**Keywords**— Rainfall prediction; spatiotemporal data; CHIRPS data; spatiotemporal ELM cluster.

Manuscript received 8 Aug. 2022; revised 18 May 2023; accepted 3 Oct. 2023. Date of publication 31 Dec. 2023.  
IJASEIT is licensed under a Creative Commons Attribution-Share, similar to the 4.0 International License.



## I. INTRODUCTION

Long-term changes in rainfall, also known as climate, often cause disasters such as floods or droughts and change nature's behavior in agriculture to produce food [1], [2]. Much research has been conducted on physical environment prevention to avoid natural disasters and crop failure [3]. Influenced by various uncertainties involving many factors, rainfall prediction brings significant uncertainty conditions and nonlinearity data. Still, prediction correctness often suffers various conditions [4], [5]. Combining and modifying existing model methods are expected to encourage alternative ways of dealing with water resource management, [7]. Other traditional machine-learning methods have been used by Ridwan *et al.* to forecast rainfall in Malaysia [8]. The method's result was flourishing and reduced the model prediction error rate. An enhanced BP algorithm for rainfall forecasting in short-of-time temporal is reported in [9]. ELM for rainfall forecast is evaluated using another algorithm to ensure the result is the correct traditional algorithms.

Achieving that ELM showed small coefficients of correlation and errors [10]. Anupam and Pani [11] reported that they implemented ELM to build short-term simulations of flood disasters in India to advise the authorities as a consideration of a policy.

A physical, analytical approach has been used in data concentrated based on data-driven with variance performance [12], [13]. Many various based on data-driven approaches have been applied, involving Autoregressive [14], [15], Statistical Downscaling [16], Support Vector Regression (SVR), Random Forest [17], Adaptive Neuro-Fuzzy Inference Systems (ANFIS) [18][19], Autoregressive Integrated Moving Average (ARIMA) model [20], artificial neural network (ANN) models [21]. These models have their strengths and weaknesses. The speed of computation time in the training model and the simple architecture of the model can be the advantages of the autoregressive model. However, this model depends on previous data and cannot predict values that have yet to occur. The SVR model has good

generalizability but does not function well when dealing with noisy data.

Furthermore, ANFIS, as one of the techniques, has linguistics and numerical knowledge. That is extensively applied as a forecast model, particularly in time series [22]. However, its application is limited in cases with large inputs [23]. As a traditional machine, ANN models can usually operate with inadequate data or knowledge. The models have parallel processing abilities, making them more fascinating.

In contrast, the ANN architecture is not constant and needs to be tuned for the hyperparameter to optimize the architecture, for which we need a trial-and-error technique for the hyperparameter to optimize the prediction performance [24]. Moreover, Choubin *et al.* [25] investigate the drought model and explore its relation to the SPI method by using a data-driven approach to make a complex model containing Multiple Linear Regression (MLR), Multi-Layer Perceptron (MLP) by adding several hidden layers, and Adaptive Neuro-Fuzzy Inference Systems (ANFIS) algorithm. Another report has been conducted by Dikshit *et al.*, who proposed a model to predict drought with a longer waiting time by utilizing various climatic variables that comprehensively affect weather changes from other data using the stacked LSTM method [26]. Shahdad and Saber [27] investigated how reduced error pruning of a tree in ensemble-based models could be even more efficacious. Nonetheless, exploring rainfall prediction using an Extreme Learning Machine still needs to be investigated more deeply since drought and rainfall have significantly correlated to predict climate in the future.

The purposes of this research are fourfold: (1) investigating the ability of spatiotemporal ELM clusters in rainfall prediction and choosing the suitable model; (2) using meteorological variables, specifically rainfall, as a main candidate for rainfall prediction and assessing the effect of spatiotemporal prediction; (3) evaluating the effect of focal operation as a preprocessing step on the capability of Extreme Learning Machine (ELM) models and other comparison methods for rainfall forecasting; and (4) comparing other several different machine learning models in rainfall prediction.

In this section, the authors give brief reviews of relevant research that can be inspired the authors to construct the Spatiotemporal ELM cluster model, including several fundamental studies in rainfall forecasting and sequential data using ELM-based models. It should be challenging when dealing with rainfall forecasting since it involves complex systems in the hydrological process, including atmospheric conditions called natural phenomena, which cannot be changed. To make predictions increase accuracy, it should consider the spatial and temporal perspectives that contain their characteristic, including nonlinearity data, complexity, and non-stationary. If the model considers that perspective, it will increase the accuracy of the prediction. Rainfall is part of the climate; in this case, Indonesia has two seasons, the first rainy season and the other dry season; this phenomenon is called meteorological activity. Much research utilized the rainfall data to predict rainfall by modifying the existing model or creating a new model to handle the model to become fitter.

It is necessary to find related work regarding rainfall prediction with much better algorithms to reach this study's aim. Several years ago, Huang *et al.* [28] proposed an

alternative approach to solve time series data against the famous ANN algorithm. Based on their proposal, ELM can do training very quickly compared to ANN, SVR, and ANFIS. In addition, the model is also able to accurately generate random weights and biases that are adjusted based on the Probability Distribution Function (PDF) approach [29]. ELM has also been used widely in various disciplines, such as climatology, energy, environment, and medicine [30]. The first study of using ELM in rainfall forecasting case was found by Dash *et al.* [31]. They compare ELM and ANN to predict drought using the Effective Drought Index, also known as EDI; exciting results were obtained that ELM could do significantly more than ANN. One-month lead-time EDI juxtaposed by using a wavelet ELM [32]. Furthermore, Ali *et al.* evaluated ELM, ANFIS, and MLR models in predicting the Standardized Precipitation Index (SPI) when dealing with multi-scalar in Pakistan [18]. Integrating ELM with other methods can also decrease the error rate; Ali *et al.* proposed a multi-stage hybridized online sequential extreme learning machine integrated with the Markov Chain Monte Carlo copula-Bat algorithm [33]. Besides that, wavelet packet decomposition (WPD) decomposes the original precipitation data into several sub-layers before ELM models the data. Hereafter, Mouatadid *et al.* [34] investigated Standard Precipitation Evaporation Index (SPEI) forecasting by comparing SVR, ANN, MLR, and ELM in drought-prone regions.

With the literature review that has been explained, it is clear that ELMs still have an area to explore in spatiotemporal rainfall prediction since spatiotemporal data need to consider spatial location and temporal history simultaneously. Hence, the author proposed the ELM Cluster to learn fast when dealing with actual distribution spatial location. This literature can motivate the authors to study the potential of ELM in rainfall prediction. Researchers have proposed many different ELM models during the past decades. Besides that, previous research used a wavelet to preprocess the data with various models, primarily developing applications [25].

Furthermore, using Indian Summer Monsoon Rainfall, Mallela and Jonnalagadda [35] compared ELM and LSTM to produce the lowest MAE. The newest method is to improve ELM using CEEMDAN non-smooth signal decomposition with other PSOs. That method can be applied by enhancing the weights of input and verges in the ELM, which can successfully enhance the forecasting outcome of ELM [36]. Therefore, in this study, the author uses a spatiotemporal extreme learning machine cluster to handle spatiotemporal data in which the models are considered spatial and temporal data simultaneously and can produce predictions in every data area.

## II. MATERIALS AND METHOD

In this study, the authors chose Kalimantan Timur as the study area to evaluate and compare the performance of several LSTM models in forecasting monthly rainfall. Kalimantan Timur is located at 2°33' North Latitude -2°25' South Latitude, 113°44' - 119°00' East Longitude, with a land area of 127,346.92 km<sup>2</sup> and a sea area of 25,656 km<sup>2</sup>. Climate change causes significant changes in rainfall patterns; this is exacerbated by the event of El Nino, which can bring extreme

rain, or it might be because El Nina causes prolonged droughts.



Fig. 1 Sample Data CHIRPS of December 2020 in Kalimantan Timur

Rainfall data obtained from <https://data.chc.ucsb.edu/products/CHIRPS-2.0/>, known as CHIRPS, as shown in Fig. 1 sampling of December 2020, is still in the form of worldwide raster data, where the research only focuses on the Kalimantan Timur region. Hence, the data needs to be split. First, a printout of the Kalimantan Timur area is required from <https://tanahair.indonesia.go.id/>. Still, combining the data using the ArcGIS application is necessary because the custom is city and district data. Furthermore, splitting the rainfall data worldwide using the SAGA application is needed after the data for the East Kalimantan region is obtained. It should be noted that the Split process requires degrees of longitude and degrees of latitude and a grid size that must be adapted to raster data worldwide, which is  $0.05^\circ \times 0.05^\circ$ . As shown in Fig. 1, data visualization has black and white colors, meaning black has representative sea surface and white island surface.

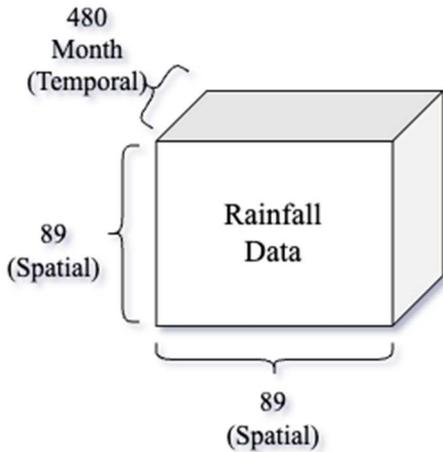


Fig. 2 Illustrated the size of spatiotemporal rainfall data.

Raster data is one of the best formats of data to represent surface area since raster can keep multi-band of data to create complex spatial conditions. CHIRPS contain a single band to interpret monthly precipitation values without additional variables. As shown in Fig. 2, this data includes dimensions  $89 \times 89$  of spatial and 480 of temporal, in this case, monthly data. Having three-dimensional conditions makes this research more complex since it should be done with a specific method, so the spatial and temporal dimensions will not be biased or removed from that dimension.

### A. Extreme Learning Machine

Input samples  $(x_i, t_i)$  is a nonrepetitive case, where  $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$  represents an  $n$ -dimensional input and  $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in R^m$  can express an output of  $m$ -dimensional. The mathematical of the traditional single hidden layer is usually applied in a feedforward neural network (FNN) of the hidden layer with  $\tilde{N}$  nodes and the  $g(x)$ , as shown in Eq. (1):

$$f_L(x) = \sum_{i=1}^{\tilde{N}} \beta_i g_i(x_j) \quad (1)$$

$$t_j, j = 1, 2, \dots, N \quad (2)$$

Where  $w_i = [w_{i1}, w_{i2}, \dots, w_{in}]^T$  denotes the weight of the connection line in the neural network between the input and the  $i$ -th hidden nodes in the hidden layer. The equation between  $i$ -th hidden nodes and the output weight can be expressed mathematically as  $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$ . This equation is not random but for the inverse formula.  $b_i$  can represent the bias value of the  $i$ -th hidden layer node. In contrast,  $x_i \times w_i$  are the inner products of  $x_i \times w_i$  and can be a loop with the exact condition. The symbol of  $\tilde{N}$  hidden layer node and previous explanation and stimulation function  $g(x)$  be able to entirely fit the output value of the input sample in the input layer of  $N$ ; that is, the  $\sum_{i=1}^{\tilde{N}} \|O_j - t_j\| = 0$  of the neural network architecture. There is also  $x_i, w_i$ , and  $b_i$ :

$$f_L(x) = \sum_{i=1}^{\tilde{N}} \beta_i g_i(x_j) = \sum_{i=1}^{\tilde{N}} \beta_i g_i(x_i \times w_i \times b_i) \quad (3)$$

$$t_j, j = 1, 2, \dots, N \quad (4)$$

Where:

- $L$  can be represented as a number of hidden units
- $N$  represents the sample number of training
- $\beta$  is a symbol of the weight vector to connect the hidden layer and output
- $w$  same as  $\beta$  represents a weight vector to calculate a value that comes from input to the hidden layer
- $g$  is a function to handle the data becoming what we need, also known as an activation function
- $b$  can always be a variable together with  $w$  to calculate the value, also known as bias
- $x$  is the input data to be obtained using the model

Mathematically expressed  $N$  samples can be as follows:

$$H\beta = T \quad (5)$$

Explaining  $H = H(w_1, \dots, w_{\tilde{N}}, b_1, \dots, b_{\tilde{N}}, x_1, \dots, x_N)$  is illustrative of the output of hidden layer nodes.

$$H = H(w_1, \dots, w_{\tilde{N}}, b_1, \dots, b_{\tilde{N}}, x_1, \dots, x_N) \quad (6)$$

$$\begin{bmatrix} g(x_1 \times w_1 \times b_1) & \dots & g(x_{\tilde{N}} \times w_1 \times b_{\tilde{N}}) \\ \dots & \dots & \dots \\ g(x_1 \times w_N \times b_1) & \dots & g(x_{\tilde{N}} \times w_N \times b_{\tilde{N}}) \end{bmatrix}_{N \times \tilde{N}} \quad (7)$$

Where:

- $m$  is the number of the output layer
- $H$  is hidden layer output matrix
- $T$  is a target matrix of training data

Huang *et al.* [28] explain ELM theory with the specific condition that the excitation function is essentially highly differentiated so that the weights of the formula on each layer

at each node and the hidden layer balance can be assigned randomly. However, to control the random number, it is necessary to initialize the number itself [28], [37]. Following the installed input weight  $w_i$  and the hidden layer value  $b_i$  with randomizing the training utilizing a single hidden layer in FNN can take the linear approach by getting minimum squares  $\hat{\beta}$  of the  $H\beta = T$ ; it also can be expressed as:

$$\beta = \begin{bmatrix} \beta_1^T \\ \dots \\ \beta_{\tilde{N}}^T \end{bmatrix}, T = \begin{bmatrix} t_1^T \\ \dots \\ t \end{bmatrix}_{N \times m} \quad (8)$$

$$\hat{\beta} = \min_{\beta} \|H\beta - T\| \quad (9)$$

Another condition can determine the equation inside the architecture,  $\tilde{N} = N$  is equal to both of them, so the hidden layer and training sample must be the same in theory, which is not repeated, that is, but in this research, the authors will change the number of hidden layers to become a hyperparameter that should be tuned to find an optimum model.  $\hat{\beta} = H^{-1}T$  can be found easily when a positive definite invertible is matrix  $H$ . Then, the error rate in the output of the hidden layer is zero. Still, in many cases, the number of  $\tilde{N}$  is lower than  $N$ . That is,  $\tilde{N} \ll N$  at this point,  $H$  is not a positive definite matrix. There is no  $w_i$ , and  $b_i$  and  $\beta_i (i = 1, 2, \dots, \tilde{N})$  makes the  $H\beta = T$ . Currently, it can solve the minimum  $\beta$  of the loss function  $H\beta - T$ , it can show as:

$$\tilde{\beta} = \min_{\beta} \|H\beta - T\| \quad (10)$$

Based on the concept generalized inverse of matrices, the minimum norm least-square solution (meeting the  $\min \|H\beta - T\|$  and  $\min \|\beta\|$  at the same time) can express as:

$$\hat{\beta} = H^+T \quad (11)$$

Where  $H^+$  is the Moore–Penrose augmented inverse matrix conducted of the hidden layer matrix  $H$ . To achieve a better generalization performance, standard items can be combined, as stated in Eq. (12):

$$\hat{\beta} = \left( \frac{1}{c} + H^+H \right)^{-1} H^+T \quad (12)$$

### B. Spatiotemporal Extreme Learning Machine Cluster

Dealing with the rainfall, the features of the neighbor are at a spatial point that must be considered to produce an accurate model. Appropriately, it must be used as a typical input model to verify a single pixel around the data target. Nevertheless, finding the correlation between spatial and temporal perspectives is quite tricky. To resolve this problem, a model should be provided with specific spatial and temporal conditions. The neural network system can learn complex nonlinear systems such as rainfall.

Nonetheless, the artificial neural network has weaknesses with complexity and slow training. Moreover, the problem when dealing with evenly distributed data on spatiotemporal makes the model efficient to train, even though the model should consider spatial information. Following this background, this study investigated a novel ELM Cluster algorithm to manage spatiotemporal information. This

proposed model can solve the problems by building a multi-model of ELM.

Previous findings that compared the effectiveness of neural networks to that of ELM models found that this finding is consistent with those findings. Contrary to common implementations, previous research has conclusively shown that not all weights and biases in a feedforward neural network must be optimized. Theoretically, it can be shown that a single-hidden-layer feedforward neural network with randomly assigned input weights and biases performs better in terms of generalization than a typical feedforward neural network using a backpropagation method. ELMs improve generalization performance by achieving the minimum training error and the smallest weight norm. Applying the same model to several geographical areas is useful for rain forecasting.

We consider that the spatiotemporal ELM Cluster can effectively solve the issue of multi-model training. The perspective of many models has a large enough hand to make ELM models able to compete with ANNs. The ELM theory confirms that no repeated adjustment enables ELM to do better with other methods and short training time. In addition, the hidden layer uses the Fourier series to process data. Hence, ELM can randomly determine the weights and bias, for instance, in the ELM network model (Fig. 3).

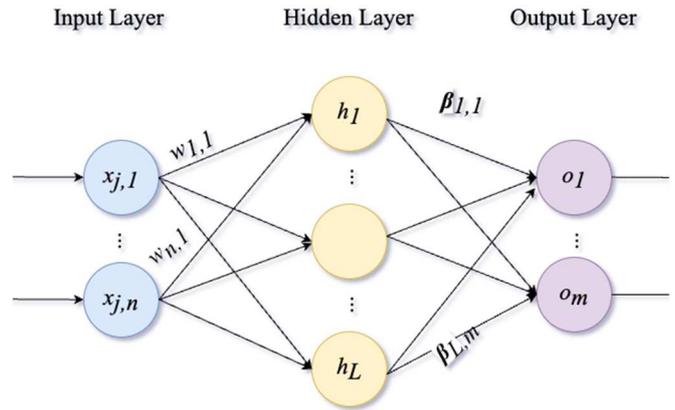


Fig. 3 Extreme Learning Machine (ELM) based

The weights value between the input and the hidden nodes of the ELM are randomized to get an initialized number. In practical implementation, multiple ELM models can share the same random input weight, which we call cluster. Fig. 4 illustrates that multiple sub-models communicate a similar structure and share input weight simultaneously. This shows that the input, hidden layer, and output cells are equal. The ELM algorithm involves randomizing between the input layer weights and the hidden layer form between the input  $x$ , in which the output of the hidden layer weight can be shared with other inputs.

Changes in the ELM network can be modified into ELM-Cluster, where each part of the structure of the hidden nodes can be expanded many times to achieve compatibility with the processed data. This stage is revolutionary since each data sample can be allocated to each ELM block.

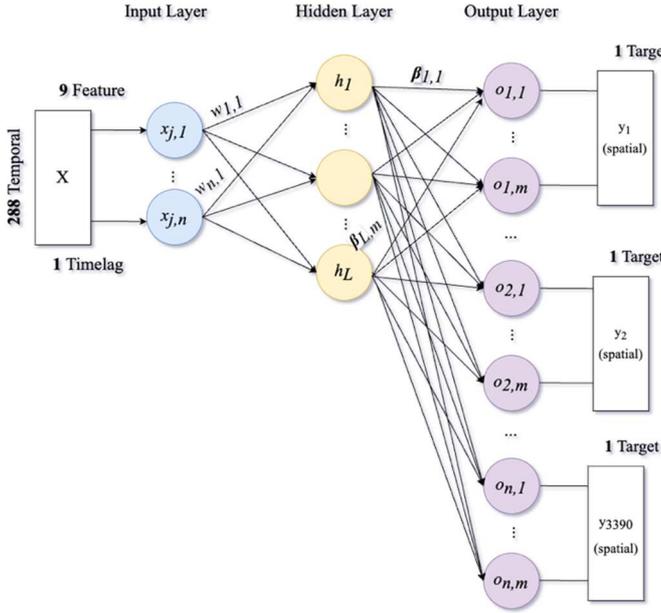


Fig. 4 Spatiotemporal ELM cluster can distribute the same random weight  $w$  but still have different  $\beta$ .

Different inputs enter different ELM sub-models to be trained without affecting each other, which is suitable for parallel processing. In the multi-model training method, the model can express much information, lowering the feature processing load. Meanwhile, the spatiotemporal ELM cluster proposes a smart way to convert some features represented by spatial visualization into multiple models. Features are inputs from machine learning that can be predicted to be as much as the spatial input itself, following data diversification features: Target features and neighboring features. Target features can be the spatial point that predicts 3 x 3 neighbors. In the meantime, the neighbor features are spatial points to consider how they can affect the spatial target. With the subset of spatiotemporal data, the training algorithm is trained as per single spatial data distribution in the feature time to attain the recognizer.

Furthermore, we explained the pseudocode in Table 1 to get more information on how to create the code in Python. To calculate the hidden layer output, the algorithm applies a nonlinear activation function elementwise to  $X$  and  $W$  matrix products and adds the bias vector  $b$ . This hidden layer output denoted as  $H$ , represents the transformed representation of the input data. Next, the algorithm solves for the output weights beta using a linear regression method. This involves taking the pseudo-inverse of the hidden layer output matrix  $H$  and multiplying it with the target matrix  $Y$ . Once the output weights are determined, the trained ELM model consists of the input weights  $W$ , bias  $b$ , and output weights beta. This model can then be used to predict new, unseen data.

TABLE I

ALGORITHM OF SPATIAL ELM CLUSTER BY UTILIZING RANDOM WEIGHT IN THE FIRST ITERATION WITHOUT BACKPROPAGATION

**Algorithm 1.** Spatiotemporal ELM Cluster

**Training**

Input: Training sample  $x$  with temporal axis in every spatial location

Output: Target  $y$  with single-step temporal in every spatial location.

**Algorithm 1.** Spatiotemporal ELM Cluster

1. Randomly allocate weight  $w_i$  and bias, where  $b_i, i = 1, \dots, L$
2. Extract feature as an input candidate
3. Divide the input data training for all spatial location.
4. Calculate output  $H$  of hidden layer.
5. Calculate output weight  $\hat{\beta} = H^+T$
6. Iterate  $\hat{\beta}$  with different spatial location
7. Change input with different spatial location
8. Make predictions by using  $\hat{\beta}$  for all spatial location
9. Evaluate by using MAE and RMSE

*C. Evaluation Metrics*

Postprocessing aims to make better rainfall predictions than “raw” (unprocessed) hydrological simulations. For this aim, it is important to evaluate the models' performance and compare them with each other to conclude which model is the best. Several metrics are used to evaluate predictions for different wait times. Since accurate and reliable predictions are so crucial during rainfall events, the primary accuracy measure for a deterministic forecast is the root-mean-square error (RMSE) in equation (13):

$$RMSE = \sqrt{\frac{\sum(y_i - q_i)^2}{n}} \quad (13)$$

Where  $y_i$  denotes the  $i$  - th time-  $k$  prediction of daily rainfall,  $q_i$  denotes the observed daily, and  $n$  represents the total number of time- $k$  monthly rainfall predictions. Compared with mean absolute error (MAE) metrics, RMSE penalizes significant errors [38], which is desirable for high rainfall forecasts. Unlike RMSE, which gives a relatively high weight to significant errors, Mean Absolute Error (MAE), a linear statistical measure, is more applicable when the overall impact of errors is proportionate to the increase in error. MAE can be formulated as [38] in equation (14).

$$MAE = \frac{1}{n} \sum_{j=1}^n |y_i - q_i| \quad (14)$$

III. RESULT AND DISCUSSION

*A. Data Preprocessing*

The data preprocessing stage is the selection stage, which aims to obtain relevant data. In raw data, missing values are often found, not stored values (mis recording), data sampling needs to be improved, and others. However, because this research does not use raw data but secondary data, preprocessing will be done to process spatial and temporal data. In addition, preprocessing will only focus on the data on cells with value, so the cells with no data will not be used.

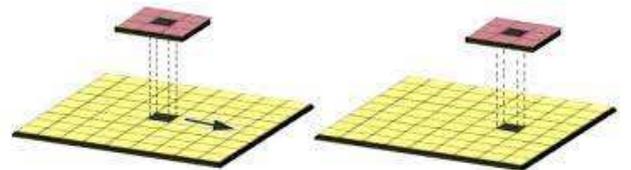


Fig. 5 Focal operation for spatial data to consider the neighbors' value

In this study, focal operation theory is implemented, as shown in Fig. 5, a spatial function to calculate the output value of each cell using neighbor values, like the nearest neighbors' algorithm (K-NN), a machine learning algorithm[39]. In

addition, this theory is also commonly used in convolution, kernel, and moving windows in deep learning algorithms such as CNN or RNN. Moving Window can be imagined as an arrangement of square cells with a specific size, which in this study is 3 x 3, which shifts its position with specific steps. As the operation is applied to each cell of the moving window, the values in the raster tend to be smoother. It was adopted in this study to smooth the predictive value in spatial conditions.

Spatiotemporal data are generally placed in continuous space, while classical data sets such as images or video data are usually in a discrete area. Spatiotemporal data patterns usually present very complex spatial and temporal properties, and correlations between data are challenging to explain with traditional methods. Finally, one of the standard statistical assumptions is that the sample is obtained independently. However, this does not apply in spatiotemporal analysis because spatiotemporal data tend to be highly correlated, so it is impossible to carry out separate studies.

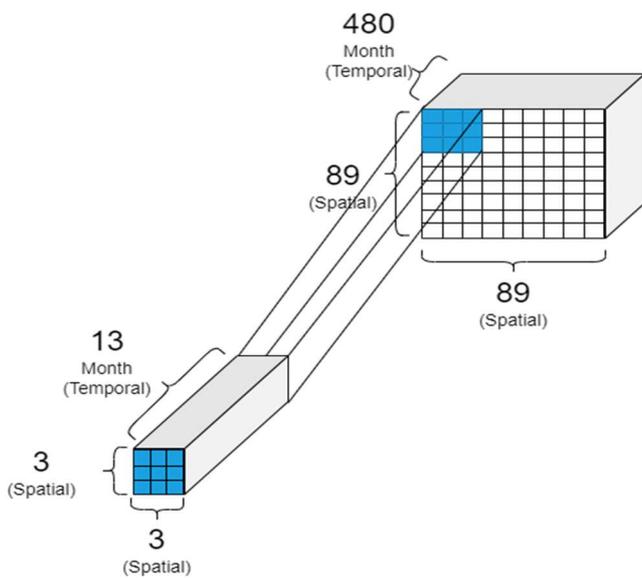


Fig. 6 Illustrated spatiotemporal data using the sliding window in spatial perspective

As explained earlier, the data used in each time unit (temporal) is 89x89 with a length of 480 temporal, as shown in Fig. 5. Hence for modeling, the data is taken spatially with a size of 3x3 for 13 months (temporal); if we use the timestep 12 scenario, this data will slice the sliding window along the temporal axis. Moving to the right side with a single step will be implemented in the data, so after the last window on the right area, it will continue by a sliding window in the next row, from left to right. It can be seen in the blue area in Fig. 6 until the end of the spatial data, which is the right bottom side. The window is initially placed at the beginning of the time series, and computations or analysis are performed on the data within that window. the window moves forward by a predefined step size (often called the stride) and repeats the analysis on the next set of data points within the new window position.

### B. Spatiotemporal ELM Cluster Model Tuning Test

#### 1) Validation Result Based on Length of Timestep

Timestep in rainfall forecasting in much research has different based on data. However, since the data has

seasonality, it should be 12-time steps because the rainfall seasonality in Indonesia is 12 months, with six months in the rainy season and the rest in the dry season. Fig. 7 visualizes the length of the timesteps for each point toward the averages MAE. The increase in timestep length seems to be a small improvement to the model's performance. It is even better with only one month as the shortest timestep, with an average MAE of 68.55. The worst result was using four four-month timestep, resulting in 93.02 average MAE. The graph in Fig. 7 shows that the correlation between the timestep length and the average MAE at each point is less significant.

Moreover, since the study aims to find the fastest training of the model, it needs to explore which timestep is the fastest training. Fig. 8 illustrates the correlation between timestep length, and the average time required for the machine to train the model. Fig. 8 proves that the longer the timestep value, the longer the processing time is needed. This result is reasonable, considering that the data processed is directly proportional to the defined timestep length. For more information, the increasing time training significantly happens in 6 and 12 timesteps. It is influenced by seasonality since the single season has six months.

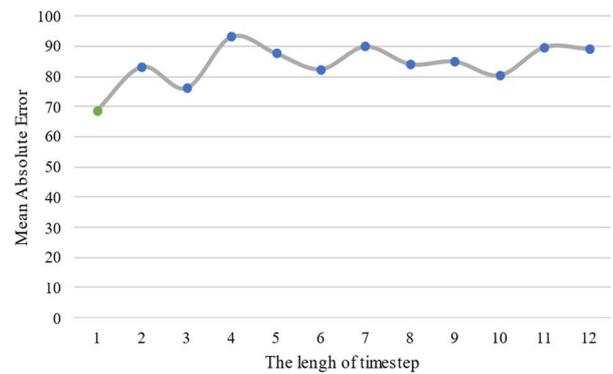


Fig. 7 Time series size of temporal sliding window and Mean Absolute Error

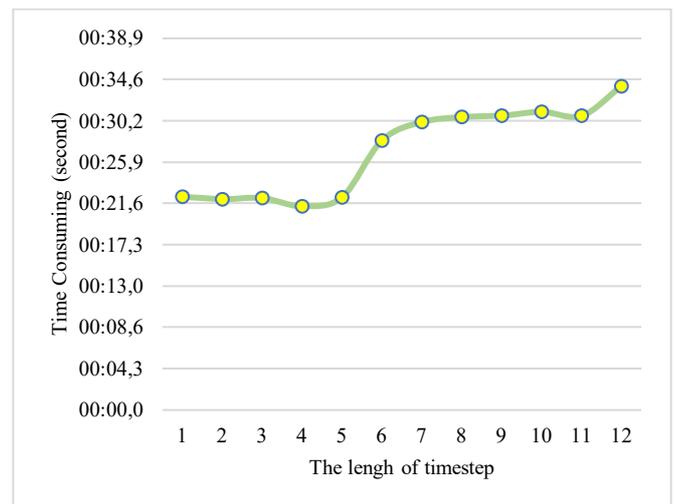


Fig. 8 Training Time of different sizes of temporal sliding window

In addition, we observed variability in the test and validation results in every model. Since we shifted the experiment range by 12 months, we selected the validation and test scores from the rainy season to the dry season. Since more weather events happen during rain time, the movements

of the weather, mainly rainfall, are less predictable. Thus, the error rate fluctuates depending on the odd or time step. The variability in the results highlights the challenges of modeling and predicting weather patterns, especially during periods of high variability, like the rainy season. It is essential to consider the inherent unpredictability of certain weather phenomena when evaluating the performance of your models. Additionally, taking into account the specific characteristics and patterns of different seasons can provide valuable insights into the behavior and accuracy of the models in different weather conditions.

### 2) Validation Result Based on Activation Function in Hidden Layer

This section summarizes the results of comparing activation functions in Table 2. Three activation functions have been compared: ReLU, Sigmoid, and Tanh. The best results were obtained with Sigmoid as the activation function, followed by ReLU and Tanh, where the MAE and RMSE values did not differ much. Based on the sigmoid formula, it can be known that the graph is between zero and one, so it controls the value only at those distances. However, ReLU is quite different. It restricts the value from zero to infinity. Because of that, the value with a significant number will go through the activation function to the next layer. For the tanh, the diagram limits the value between minus one and one. In this condition, the value with a number minus might make the model's worst and produce an accurate prediction.

TABLE III  
HYPERPARAMETER TUNING BASED ON ACTIVATION FUNCTION

Activation	MAE	RMSE
ReLu	68.55	86.19
Sigmoid	<b>66.77</b>	<b>83.77</b>
Tanh	68.46	85.88

### 3) Validation Result Based on Number of Hidden Layer Nodes

Table 3 details the results of comparing the number of units of the model when predicting the testing dataset in MAE and RMSE units. The number of units tested was 16, 32, and 48. The most petite MAE and RMSE were obtained with a model with 16 units. The model with 48 units produces the highest error value. Besides, the model with 32 units produces an error slightly lower than 48 units. The more units used in a model, the higher the error value.

TABLE III  
HYPERPARAMETER TUNING BASED ON NUMBER OF HIDDEN NODES

Evaluation Metrics	Number of Nodes		
	16	32	48
MAE	<b>66.77</b>	70.21	73.79
RMSE	<b>83.77</b>	88.24	93.03

### C. Comparison of the Spatiotemporal ELM Cluster and Other Algorithms

Our proposed model has been compared with several other machine learning methods, such as SVR, Linear Regression, Gaussian, Ridge, and Lasso.

- ELM Cluster: Extreme Learning Machine to cluster every model by iterating every piece of information in spatial location, as shown in Fig. 4.

- SVR: Support Vector Regression is a part of Support Vector Machine, a supervised learning algorithm that forecasts discrete values. Support Vector Regression uses the same rule as the SVMs.
- LR: Linear regression is a linear model, e.g., a model that pretends a linear correlation between the input variables (x) and the single value of the output variable (y).
- Gaussian: Gaussian linear model is a unique case of the simplified linear model that just so happens to be conventional least squares
- Ridge: A method of assessing the coefficients of multiple-regression models in circumstances where the independent variables are extremely related.
- Lasso: An alteration of linear regression when the model is penalized for misconduct for the sum of the absolute values of the weights. Thus, the absolute value of the weights will (in general) decrease, and the lot tends to be zero.

The results of the evaluation and training duration of these models are summarized in Table 4. The proposed model occupies the first position with the best results, namely MAE of 66.77 and RMSE of 83.77. The other models, from best to worst, namely Support Vector Regression (SVR), Linear Regression, Gaussian Regression, Ridge Regression, and Lasso Regression, followed the Spatiotemporal ELM in model performance. The other models applied for every spatial point. All tested models are ensured through a training process for each point in the dataset.

Regarding time efficiency, Spatiotemporal ELM occupies the first position as the model with the fastest training duration of 28.9 seconds. It was then followed by other machine learnings with training duration by Linear Regression for 30.5 seconds, Gaussian for 35.8 seconds, Ridge for 40.1 seconds, SVR for 46.4 seconds, and Lasso for 56.2 seconds.

TABLE IV  
RESULT EVALUATIONS BETWEEN ELM CLUSTER AND OTHER METHODS USING MAE AND RMSE, COMPARING WITH A TRAINING TIME OF THE MODEL TO LEARN FROM THE DATA

Model	MAE	RMSE	Training Time
ELM Cluster	66.77	83.77	00:28.9
SVR [40]	67.02	84.74	00:46.4
Linear Regression [41]	67.15	84.91	00:30.5
Gaussian Regression [42]	67.36	86.05	00:35.8
Ridge Regression [43]	69.64	87.1	00:40.1
Lasso Regression [43]	72.45	98.07	00:56.2

### D. Prediction Result Evaluation

The error rate using MAE or RMSE has been explained before. In this section, the authors interpret the prediction result using the ELM cluster to predict monthly rainfall by spatiotemporal data. As shown in Fig 9 and Fig. 10, it explained the ground truth of the data. Fig. 11 shows the prediction using ELM-Cluster that is similar in spatial area. Therefore, this study uses only single data rainfall as input and qualitatively evaluates the prediction results by data itself. The spatial distributions of the prediction have the same pattern as the ground truth. As can be observed, the forecast results of all the spatial models present significant spatial areas with an intensity of more than 300 mm but show similar spatial characteristics of the model's forecast ability with

west-to-east distribution. Forests dominate the western region of Kalimantan Timur. There would be more heavy rain than on the east coast, while areas with rain intensity less than 150mm are mainly near the beach.

The uncertainty quantification is vital in rainfall prediction. Estimating the incoming rain season prescriptive analytics can be challenging if the temporal quantities of available surface land heights at specified locations area. In this study, the scenarios are set up with induced rainfall. Our experimental framework was constructed to show the enhanced value of using the predictors' sequence information from former time steps in increasing the models' predictive abilities. In addition, we tested the value in the model by adding more predictors to the model's performance.

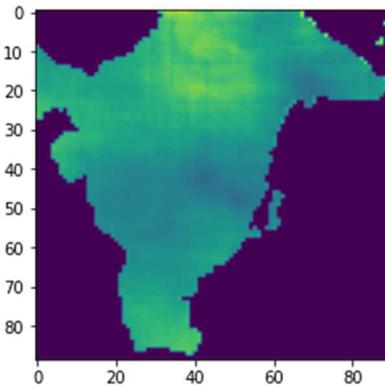


Fig. 9 Ground truth data before preprocessing

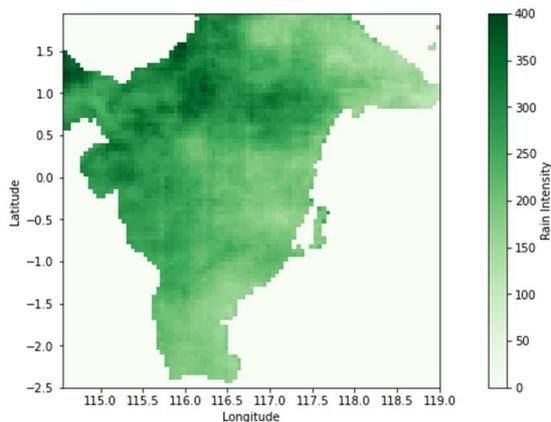


Fig. 10 Ground truth from the CHIRPS data. In one month, rain density ranges from 0 mm to almost 400 mm

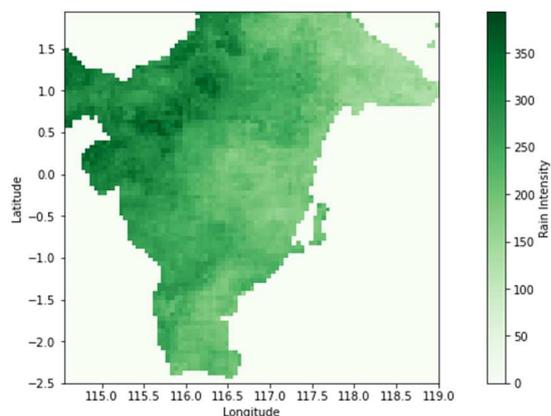


Fig. 11 Result of rainfall prediction using ELM cluster

#### IV. CONCLUSION

This study uses spatiotemporal ELM cluster and several machine learning models to forecast hydrological rainfall captured by satellite and station, combining into data CHIRPS. For this purpose, the authors have used 1 to 12 timescales to find the minimum error. Five different machine learning models and one spatiotemporal ELM Cluster model are used for modeling and using several timestep to ensure the better size timestep. Applied focal operation with the nearest neighbor for the preprocessing data can make the model relateable with spatial dan temporal perspectives. Based on the results, the spatiotemporal ELM Cluster is the best model for rainfall prediction modeling on that timestep and with different tuning hyperparameters, finding the best hyperparameter to create a fit model for the ELM cluster, while Lasso is the worst compared to others. Besides, the spatiotemporal technique positively decreases the error rate and training time around the model learning. The established spatiotemporal ELM Cluster model has a high generalization ability and prediction error rate, which can support the formulation of appropriate environmental management policies.

Moreover, many complex factors relating to rainfall prediction will lead to the effect of the algorithm in terms of prediction. Exploring other predictors such as wind, temperature, and meteorological effects should consider raising the effectiveness of rainfall prediction with temporal and spatial data. This idea can control different areas with different factors of rain density.

#### ACKNOWLEDGMENT

A research grant from the Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi supports this work.

#### REFERENCES

- [1] G. C. Wang *et al.*, "Mechanics Monthly and seasonal hydrological drought forecasting using multiple extreme learning machine models," 2022, doi:10.1080/19942060.2022.2089732.
- [2] R. He, L. Zhang, and A. W. Z. Chew, "Modeling and predicting rainfall time series using seasonal-trend decomposition and machine learning," *Knowl Based Syst.*, vol. 251, p. 109125, 2022, doi:10.1016/j.knosys.2022.109125.
- [3] N. Nabipour, M. Dehghani, A. Mosavi, and S. Shamshirband, "Short-Term Hydrological Drought Forecasting Based on Different Nature-Inspired Optimization Algorithms Hybridized with Artificial Neural Networks," *IEEE Access*, vol. 8, no. 1993, pp. 15210–15222, 2020, doi:10.1109/ACCESS.2020.2964584.
- [4] X. Zhang, X. Wu, S. He, and D. Zhao, "Precipitation forecast based on CEEMD–LSTM coupled model," *Water Supply*, vol. 21, no. 8, pp. 4641–4657, 2021, doi:10.2166/ws.2021.237.
- [5] S. Wang, H. Peng, Q. Hu, and M. Jiang, "Analysis of runoff generation driving factors based on hydrological model and interpretable machine learning method," *J Hydrol Reg Stud.*, vol. 42, p. 101139, 2022, doi:10.1016/j.ejrh.2022.101139.
- [6] A. Shirvani, W. A. Landman, M. Barlow, and A. Hoell, "Evaluation of the forecast skill of North American Multi-Model Ensemble for monthly and seasonal precipitation forecasts over Iran," *International Journal of Climatology*, vol. 43, no. 2, pp. 1141–1166, 2023.
- [7] A. Y. Barrera-Animas, L. O. Oyedele, M. Bilal, T. D. Akinosho, J. M. D. Delgado, and L. A. Akanbi, "Rainfall prediction: A comparative analysis of modern machine learning algorithms for time-series forecasting," *Machine Learning with Applications*, vol. 7, p. 100204, 2022, doi:10.1016/j.mlwa.2021.100204.
- [8] W. M. Ridwan, M. Sapitang, A. Aziz, K. F. Kushiar, A. N. Ahmed, and A. El-Shafie, "Rainfall forecasting model using machine learning methods: Case study Terengganu, Malaysia," *Ain Shams Engineering*

- Journal*, vol. 12, no. 2, pp. 1651–1663, 2021, doi:10.1016/j.asej.2020.09.011.
- [9] D. Pirone, L. Cimorelli, G. Del Giudice, and D. Pianese, “Short-term rainfall forecasting using cumulative precipitation fields from station data: a probabilistic machine learning approach,” *J Hydrol (Amst)*, vol. 617, p. 128949, Feb. 2023, doi:10.1016/j.jhydrol.2022.128949.
- [10] M. Saroughi, E. Mirzania, D. K. Vishwakarma, S. Nivesh, K. C. Panda, and F. A. Daneshvar, “A Novel Hybrid Algorithms for Groundwater Level Prediction,” *Iranian Journal of Science and Technology, Transactions of Civil Engineering*, Mar. 2023, doi:10.1007/s40996-023-01068-z.
- [11] S. Anupam and P. Pani, “Flood forecasting using a hybrid extreme learning machine-particle swarm optimization algorithm (ELM-PSO) model,” *Model Earth Syst Environ*, vol. 6, no. 1, pp. 341–347, 2020, doi:10.1007/s40808-019-00682-z.
- [12] D. Salim *et al.*, “Comparative Study of Different Discrete Wavelet Based Neural Network Models for long term Drought Forecasting,” *Water Resources Management*, vol. 37, no. 3, pp. 1401–1420, Feb. 2023, doi:10.1007/s11269-023-03432-0.
- [13] O. S. Ojo and S. T. Ogunjo, “Machine learning models for prediction of rainfall over Nigeria,” *Sci Afr*, vol. 16, p. e01246, 2022, doi:10.1016/j.sciaf.2022.e01246.
- [14] J. Wu *et al.*, “Application of Time Serial Model in Water Quality Predicting,” *Computers, Materials & Continua*, vol. 74, no. 1, pp. 67–82, 2023, doi:10.32604/cmc.2023.030703.
- [15] M. Hasanuzzaman, A. Islam, B. Bera, and P. K. Shit, “A comparison of performance measures of three machine learning algorithms for flood susceptibility mapping of river Silabati (tropical river, India),” *Physics and Chemistry of the Earth, Parts A/B/C*, vol. 127, p. 103198, 2022, doi:10.1016/j.pce.2022.103198.
- [16] P. Singh, A. Y. Shamseldin, B. W. Melville, and L. Wotherspoon, “Development of statistical downscaling model based on Volterra series realization, principal components and ridge regression,” *Model Earth Syst Environ*, Jan. 2023, doi:10.1007/s40808-022-01649-3.
- [17] S. Lee *et al.*, “Estimation of rainfall erosivity factor in Italy and Switzerland using Bayesian optimization based machine learning models,” *Catena (Amst)*, vol. 211, p. 105957, 2022, doi:10.1016/j.catena.2021.105957.
- [18] A. Chevuturi *et al.*, “Improving global hydrological simulations through bias-correction and multi-model blending,” *J Hydrol (Amst)*, p. 129607, May 2023, doi:10.1016/j.jhydrol.2023.129607.
- [19] O. Rahmati *et al.*, “Capability and robustness of novel hybridized models used for drought hazard modeling in southeast Queensland, Australia,” *Science of the Total Environment*, vol. 718, p. 134656, 2020, doi:10.1016/j.scitotenv.2019.134656.
- [20] A. Ben Abbes, R. Inoubli, M. Rhif, and I. R. Farah, “Combining deep learning methods and multi-resolution analysis for drought forecasting modeling,” *Earth Sci Inform*, Apr. 2023, doi:10.1007/s12145-023-01009-4.
- [21] M. M. H. Khan, N. S. Muhammad, and A. El-Shafie, “Wavelet based hybrid ANN-ARIMA models for meteorological drought forecasting,” *J Hydrol (Amst)*, vol. 590, no. August, 2020, doi:10.1016/j.jhydrol.2020.125380.
- [22] U. Okkan, Z. B. Ersoy, A. Ali Kumanlioglu, and O. Fistikoglu, “Embedding machine learning techniques into a conceptual model to improve monthly runoff simulation: A nested hybrid rainfall-runoff modeling,” *J Hydrol (Amst)*, vol. 598, p. 126433, 2021, doi:10.1016/j.jhydrol.2021.126433.
- [23] M. N. M. Salleh, N. Talpur, and K. Hussain, “Adaptive neuro-fuzzy inference system: Overview, strengths, limitations, and solutions,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10387 LNCS, pp. 527–535, 2017, doi:10.1007/978-3-319-61845-6\_52.
- [24] J. Diez-Sierra and M. del Jesus, “Long-term rainfall prediction using atmospheric synoptic patterns in semi-arid climates with statistical and machine learning methods,” *J Hydrol (Amst)*, vol. 586, p. 124789, 2020, doi:10.1016/j.jhydrol.2020.124789.
- [25] B. Choubin, A. Malekian, and M. Golshan, “Application of several data-driven techniques to predict a standardized precipitation index,” *Atmosfera*, vol. 29, no. 2, pp. 121–128, 2016, doi:10.20937/ATM.2016.29.02.02.
- [26] A. Dikshit, B. Pradhan, and A. M. Alamri, “Long lead time drought forecasting using lagged climate variables and a stacked long short-term memory model,” *Science of the Total Environment*, vol. 755, p. 142638, 2021, doi:10.1016/j.scitotenv.2020.142638.
- [27] M. Shahdad and B. Saber, “Drought forecasting using new advanced ensemble-based models of reduced error pruning tree,” *Acta Geophysica*, vol. 70, no. 2, pp. 697–712, 2022, doi:10.1007/s11600-022-00738-2.
- [28] G. Bin Huang, Q. Y. Zhu, and C. K. Siew, “Extreme learning machine: Theory and applications,” *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006, doi:10.1016/j.neucom.2005.12.126.
- [29] Z. M. Yaseen, S. O. Sulaiman, R. C. Deo, and K. W. Chau, “An enhanced extreme learning machine model for river flow forecasting: State-of-the-art, practical applications in water resource engineering area and future research direction,” *J Hydrol (Amst)*, vol. 569, pp. 387–408, 2019, doi:10.1016/j.jhydrol.2018.11.069.
- [30] S. Shamshirband *et al.*, “Predicting Standardized Streamflow index for hydrological drought using machine learning models,” *Engineering Applications of Computational Fluid Mechanics*, vol. 14, no. 1, pp. 339–350, 2020, doi:10.1080/19942060.2020.1715844.
- [31] Y. Dash, S. K. Mishra, and B. K. Panigrahi, “Rainfall prediction of a maritime state (Kerala), India using SLFN and ELM techniques,” *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies, ICICICT 2017*, vol. 2018-Janua, pp. 1714–1718, 2018, doi:10.1109/ICICICT1.2017.8342829.
- [32] R. C. Deo, O. Kisi, and V. P. Singh, “Drought forecasting in eastern Australia using multivariate adaptive regression spline, least square support vector machine and M5Tree model,” *Atmos Res*, vol. 184, pp. 149–175, 2017, doi:10.1016/j.atmosres.2016.10.004.
- [33] M. Ali, R. C. Deo, N. J. Downs, and T. Maraseni, “Multi-stage hybridized online sequential extreme learning machine integrated with Markov Chain Monte Carlo copula-Bat algorithm for rainfall forecasting,” *Atmos Res*, vol. 213, pp. 450–464, 2018, doi:10.1016/j.atmosres.2018.07.005.
- [34] S. Moutatdid, N. Raj, R. C. Deo, and J. F. Adamowski, “Input selection and data-driven model performance optimization to predict the Standardized Precipitation and Evaporation Index in a drought-prone region,” *Atmos Res*, vol. 212, pp. 130–149, 2018, doi:10.1016/j.atmosres.2018.05.012.
- [35] S. S. Mallela and S. K. Jonnalagadda, “Rainfall Prediction Based on Spatial Attention Layer: A Case Study Analysis,” *International Journal of Intelligent Engineering and Systems*, vol. 15, no. 3, pp. 49–58, 2022, doi:10.22266/ijies2022.0630.05.
- [36] X. Zhang, D. Zhao, T. Wang, X. Wu, and B. Duan, “A novel rainfall prediction model based on CEEMDAN-PSO-ELM coupled model,” *Water Supply*, vol. 22, no. 4, pp. 4531–4543, 2022, doi:10.2166/ws.2022.115.
- [37] Y. Xing, X. Ban, X. Liu, and Q. Shen, “Large-scale traffic congestion prediction based on the symmetric extreme learning machine cluster fast learning method,” *Symmetry (Basel)*, vol. 11, no. 6, pp. 1–19, 2019, doi:10.3390/sym11060730.
- [38] N. Nurhamidah, R. Andari, A. Junaidi, and D. Daoed, “Evaluation of the Compatibility of TRMM Satellite Data with Precipitation Observation Data,” *JOIV: International Journal on Informatics Visualization*, vol. 7, no. 2, pp. 287–294, 2023.
- [39] D. Haynes, P. Mitchell, and E. Shook, “Developing the raster big data benchmark: A comparison of raster analysis on big data platforms,” *ISPRS Int J Geoinf*, vol. 9, no. 11, 2020, doi:10.3390/ijgi9110690.
- [40] A. Danandeh Mehr, V. Nourani, V. Karimi Khosrowshahi, and M. A. Ghorbani, “A hybrid support vector regression–firefly model for monthly rainfall forecasting,” *International Journal of Environmental Science and Technology*, vol. 16, no. 1, pp. 335–346, 2019, doi:10.1007/s13762-018-1674-2.
- [41] P. C. Shaker Reddy and A. Sureshbabu, “An enhanced multiple linear regression model for seasonal rainfall prediction,” *International Journal of Sensors Wireless Communications and Control*, vol. 10, no. 4, pp. 473–483, 2020.
- [42] A. Elbeltagi *et al.*, “Prediction of meteorological drought and standardized precipitation index based on the random forest (RF), random tree (RT), and Gaussian process regression (GPR) models,” *Environmental Science and Pollution Research*, vol. 30, no. 15, pp. 43183–43202, Jan. 2023, doi:10.1007/s11356-023-25221-3.
- [43] F. Palacios-Rodriguez, E. Di Bernardino, and M. Mailhot, “Smooth copula-based generalized extreme value model and spatial interpolation for extreme rainfall in Central Eastern Canada,” *Environmetrics*, vol. 34, no. 3, May 2023, doi:10.1002/env.2795.