Journal of Computer Science and Cybernetics, V.39, N.1 (2023), 33–49
DOI no. 10.15625/1813-9663/18028

A NOVEL METHOD FOR WEATHER NOWCASTING BASED ON SPATIAL COMPLEX FUZZY INFERENCE WITH MULTIPLE BAND INPUT DATA

NGUYEN TRUNG TUAN[1], LE TRUONG GIANG[2,3], PHAM HUY THONG[4,*]
NGUYEN VAN LUONG[2], LE MINH TUAN[5], NGUYEN QUOC UY[6], LE MINH HOANG[7]

[1]*School of Information Technology and Digital Economics, National Economics University, 207 Giai Phong Street, Hai Ba Trung District, Ha Noi, Viet Nam*

[2]*Quality Assurance Center, Hanoi University of Industry (HaUI), 298 Cau Dien Street, Bac Tu Liem District, Ha Noi, Viet Nam*

[3]*Graduate University of Science and Technology, Vietnam Academy of Science and Technology, 18 Hoang Quoc Viet Street, Cau Giay District, Ha Noi, Viet Nam*

[4]*Information Technology Institute, Vietnam National University, 144 Xuan Thuy Street, Cau Giay District, Ha Noi, Viet Nam*

[5]*National Academy of Public Administration, 77 Nguyen Chi Thanh Street, Dong Da District, Ha Noi, Viet Nam*

[6]*Post and Telecommunications Institute of Technology (PTIT), Km10, Nguyen Trai Street, Ha Dong District, Ha Noi, Viet Nam*

[7]*Testing and Assessment Centre, Hanoi University of Industry (HaUI), 298 Cau Dien Street, Bac Tu Liem District, Ha Noi, Viet Nam*

Abstract. The prediction of weather changes, such as rainfall, clouds, floods, and storms, is critical in weather forecasting. There are several sources of input data for this purpose, including radar and observational data, but satellite remote sensing images are the most commonly used due to their ease of collection. In this paper, we present a novel method for weather nowcasting based on Mamdani complex fuzzy inference with multiple band input data. The proposed approach splits the process into two parts: the first part converts the multiple band satellite images into real and imaginary parts to facilitate the rule process, and the second part uses the Spatial CFIS+ algorithm to generate the predicted weather state, taking into account factors such as cloud, wind, and temperature. The use of MapReduce helps to speed up the algorithm's performance. Our experimental results show that this new method outperforms other relevant methods and demonstrates improved prediction accuracy.

Keywords. Weather forecast, complex fuzzy inference system, remote sensing images, multiple band satellite images.

*Corresponding author.
E-mail addresses: tuannt@neu.edu.vn (N.T.Tuan); letruonggiang@haui.edu.vn (L.T.Giang); thongph@vnu.edu.vn (P.H. Thong); luong.nguyen@haui.edu.vn (N.V.Luong), letuan104@gmail.com (L.M.Tuan); uynq@ptit.edu.vn (N.Q.Uy); hoanglm@haui.edu.vn (L.M.Hoang).

1. INTRODUCTION

Today's society is planning heavily relies on weather forecasting, which uses science and technology to make predictions about clouds, temperature, and rainfall,... in a specific area. Lower hazards for humans result from more accurate weather predictions. A detailed picture of the current weather conditions in a particular area needs to be prepared to forecast the weather. This requires regular and accurate monitoring of the atmospheric layers from low to high by surface and high-altitude observation stations and remote sensing systems such as satellite images and meteorological radars. With the advantages of science and technology, especially space science, the collection of satellite images is now easier and cheaper. Various satellite images are stored in [1], where users can get them free for weather prediction or other relevant research. Bands separate the satellite images after collecting to distinguish the factor that the band results in, such as cloud, temperature, humidity, etc.

There are some typical widely used approaches to forecast weather from satellite images in [2–8]. The first ones [3–5] used existing models with small changes for adapting with data to forecast the weather. Although these methods resulted in positive values in the experiment, they still depended on these models and were not flexible with other data. The second ones [2, 6–8] employed many scientists' attention relating to fuzzy inference systems and complex fuzzy inference systems. However, the limitations of these approaches were using single-band satellite image series for input data. This may predict overfitting to the training data, while the real weather may change rapidly due to other relevant factors like humidity, temperature, etc.

In this paper, a novel method for weather nowcasting based on Co-Spatial complex fuzzy inference with multiple band input data is introduced. This innovative algorithm has two phases to do the training and testing phases. The first one with split multiple band images to extract the real and imaginary parts to build the rules. The last one uses the rule to export the forecast image. Experimental evaluation on multiple band satellite image sequences will be performed to compare with others on prediction accuracy. The results of the experiments show that the proposed method is better than the relevant ones.

The rest of the paper is organized as follows. Section 2 and 3 presents the related works and backgrounds for weather forecasting. Section 4 indicates the proposed method, and Section 5 shows the experimental results. Finally, conclusions and further works are covered in Section 6.

2. RELATED WORKS

With the vigorous development of recording/imaging equipment and satellite imaging equipment, the quality and accuracy of satellite images is getting higher and higher. Along with that, a variety of applications related to these images are also known. Some of them may include the following:

Gao [3] et al. researched the problem of retrieving water leaving reflectances. In this study, with the database obtained from GOES 16 17, the authors proposed many spectrum-matching methods to improve the efficiency of the prediction process. The results of this study help open up a new direction that can be complementary to the current operational ocean.

In addition, the database from NOAA, GOES 16 17, there have also been many studies related to the problem of predicting variation based on data, which can be mentioned as:

The study of Bao [4] and colleagues on the problem of detecting the direction of storm movement using the HAFS model. The forecast results from this method are highly appreciated in terms of image quality. The characteristics of the storm are also shown more clearly, offering a solution that is more feasible in the future. In the study of Meng [5] et al., in building an operator to calculate the path of clouds/ice on the database GOES-17, the results were obtained from real datasets. Other studies show the powerful effects of the operators built by the author.

To solve problems related to time series forecasting in general or weather forecasting in particular, a recent method attracting many scientists' attention is using fuzzy inference systems and complex fuzzy inference systems. Some research can be mentioned, such as the study of Phong [7] and the research team on the time series prediction problem using the PSO swarm optimization algorithm and simulated annealing technique. The study has shown many essential fuzzy and defuzzifying formulas for time series forecasting. The experimental results of the model on several datasets show very positive effects of this study. Another study that can be mentioned is the study of time series prediction based on the semantic rules of Phong [8]; by including semantic information in the rule system, the author has shown another potential direction to solve this problem based on some experimental results obtained from the proposed model. However, these methods often need help processing time due to many rules and easily cause overfitting. To solve this problem, many solutions to reduce the rule have been proposed, one of which can be mentioned is the study of Liu [6] and collaborators. In this study, the author gave a solution based on the firing strength rule values to evaluate the rule's effectiveness, thereby giving a more suitable set of rules. To enhance the algorithm performance, MapReduce [9], a model designed by Google, was used to process the data more effectively. The model, when tested in practice, has obtained very positive results.

3. BACKGROUND

3.1. Mamdani complex fuzzy inference system (M-CFIS)

The Mamdani Complex Fuzzy Inference System (M-CFIS) [10] integrates the concept of a complex fuzzy inference system (CFIS) and the classical Mamdani FIS (M-FIS). The general structure of Mamdani CFIS consists of six steps: Create a set of complex fuzzy rules, fuzzify the inputs, find the rule's firing strength, define the consequence of the complex fuzzy rules, aggregation and defuzzification are converting complex fuzzy output values to obtain crisp values. The steps of the Mamdani complex fuzzy inference system algorithm are described in Figure 1 below.

3.2. Co-Spatial complex fuzzy inference system (Co-Spatial CFIS+)

The Co-Spatial complex fuzzy inference system (Co-Spatial CFIS) [2] is a developing version of Mamdani-CFIS based on CFS theory for change detection from satellite images contains the spatial triangles of the fuzzy rules with the 6 boundary point value of the real part, imaginary part (a, b, c, a', b', c') and the parameter to regulate the boundary point value

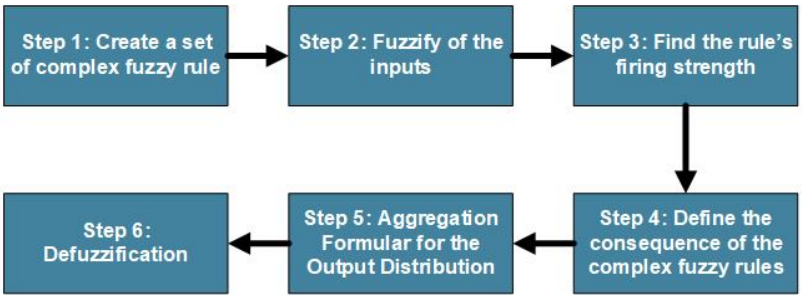


Figure 1: Diagram of M-CFIS algorithm

as the following formulas (1-6)

$$b_{ij} = \alpha_j^b \times V_j^{rel}, \tag{1}$$

$$b'_{ij} = \alpha_j^{b'} \times V_j^{img}, \tag{2}$$

$$a_{ij} = \alpha_j^a \times \left(\frac{\sum\limits_{i=1,2,\ \ldots n \text{ and } X_i^{(k)} \le b_{ij}} U_{\mathrm{i,j}} \times X_i^{(k)}}{\sum\limits_{i=1,2,\ \ldots n \text{ and } X_i^{(k)} \le b_{ij}} U_{\mathrm{i,j}}} \right), \tag{3}$$

$$a'_{ij} = \alpha_j^{a'} \times \left(\frac{\sum\limits_{i=1,2,\ \ldots n \text{ and } HoD_i^{(k)} \le b_{ij}} U_{\mathrm{i,j}} \times HoD_i^{(k)}}{\sum\limits_{i=1,2,\ \ldots n \text{ and } HoD_i^{(k)} \le b_{ij}} U_{\mathrm{i,j}}} \right), \tag{4}$$

$$c_{ij} = \alpha_j^c \times \left(\frac{\sum\limits_{i=1,2,\ \ldots n \text{ and } X_i^{(k)} \ge b_{ij}} U_{\mathrm{i,j}} \times X_i^{(k)}}{\sum\limits_{i=1,2,\ \ldots n \text{ and } X_i^{(k)} \ge b_{ij}} U_{\mathrm{i,j}}} \right), \tag{5}$$

$$c'_{ij} = \alpha_j^{c'} \times \left(\frac{\sum\limits_{i=1,2,\ \ldots n \text{ and } HoD_i^{(k)} \ge b_{ij}} U_{\mathrm{i,j}} \times HoD_i^{(k)}}{\sum\limits_{i=1,2,\ \ldots n \text{ and } HoD_i^{(k)} \ge b_{ij}} U_{\mathrm{i,j}}} \right), \tag{6}$$

where X the set of the input parameters, HoD the value of the imaginary part of the input image, $U_{i,j}$ degree of membership, $a, b, c \in [0, 1]$ the boundary point value of the real part, $a', b', c' \in [0, 1]$ the boundary point value of the imaginary part, V_j^{rel} the real part cluster center value of the jth rule, V_j^{img} the imaginary part cluster center value of the jth rule.

3.3. MapReduce

MapReduce is a model exclusively designed by Google [9] with the ability to programmatically process large amounts of data in parallel while distributing algorithms on the same computer. Recently, MapReduce is becoming one of the more generalizing terms.

MapReduce will include two main stages, Map and Reduce.

- Map stage performs "filtering" and "classifying" input data.

- The Reduce stage will perform the process of aggregating all data based on the Map stage results.

With the above design, the MapReduce model has some advantages as follows:

- MapReduce can easily handle all problems with vast amounts of data thanks to its computational power and complex analytical tasks. In just a short period, it can quickly process and give easy results.

- MapReduce can run in parallel on different distributed computers with the ability to operate independently combined with the distribution and handling of technical errors to bring high efficiency to the entire system.

- MapReduce can execute on various programming languages with corresponding support libraries. Details of the implementation stages of MapReduce are implemented as follows:

- Map function: This function is responsible for processing a key pair (key, value) to create a new key pair (keyl, valuel). At this time, the key-pair (keyl, valuel) will act as an intermediary. After that, the user only needs to write data to the hard disk and quickly notify the Reduce function to let the data enter the Reduce input.

- Reduce function: This function has the task of receiving the pair of intermediate keywords and the value corresponding to that amount of keywords (keyl, valuel) to form a different set of keys by concatenating them. These key/value pairs will be entered into the Reduce functions through a position pointer. This process will make it easier for programmers to manage a large number of lists as well as allocate values that are suitable for system memory.

In addition, in between Map and Reduce, there is another intermediate step called Shuffle. After the Map completes its task, Shuffle will continue to collect and synthesize the keyword/intermediate value pair created by the previous Map and pass it to Reduce for further processing.

4. THE PROPOSED METHOD

4.1. Main system

With input data as satellite images according to different channels, the study was conducted to determine each input image channel's real and imaginary parts. To connect the data of image channels to evaluate better the influence of different channels on the prediction results, the study uses a mapping mechanism to help determine the imaginary part values of each input image channel based on the imaginary part values of the other input image channels.

Next, conduct the rule generation process based on the Co-Spatial CFIS+ [2] method. The rules obtained from this process will be used for the corresponding image prediction process. However, to improve the quality of the rule system, the study also shows some solutions to help reduce the number of rules used while maintaining specific efficiency. In addition to reducing the processing time, the study uses a parallel calculation mechanism between color channels and image bands to reduce prediction time.

Details of the steps are described in Figures 2, 3 as below.

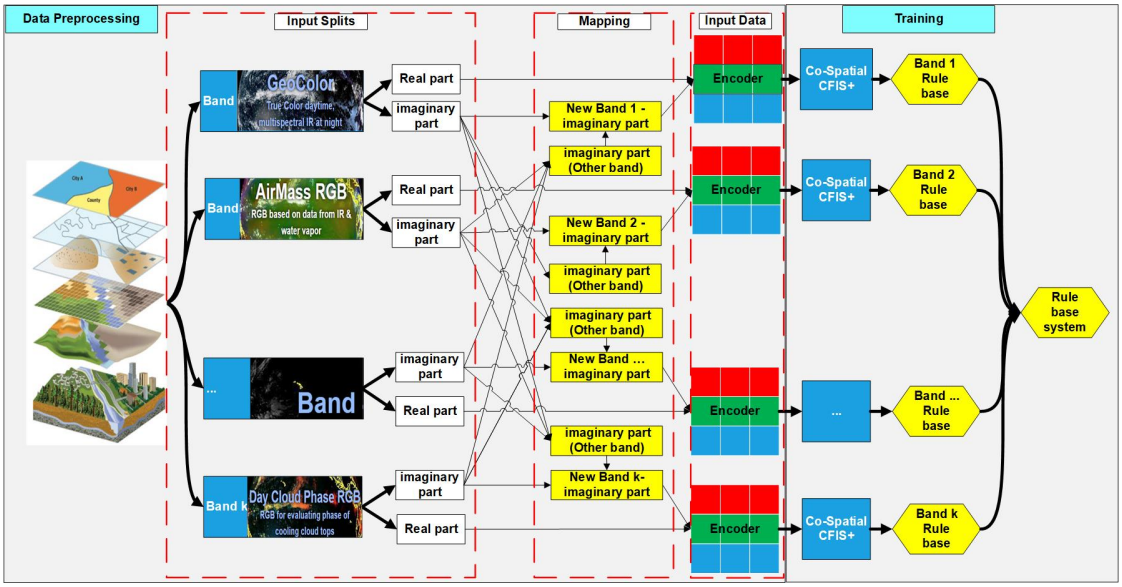


Figure 2: The Training phase of the proposed method

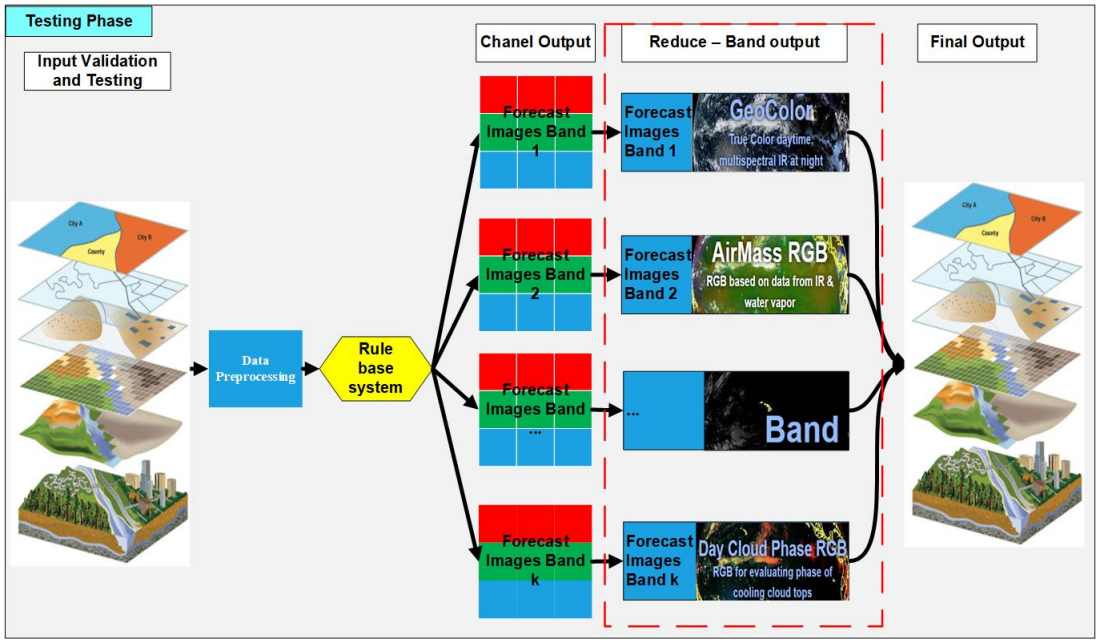


Figure 3: The Testing phase of the proposed method

4.2. Details of the proposed method

- **Step 1.** Data preprocessing

  Step 1.1. Choose input images band

The US Navy database [1] provides 25 bands of images (including single-channel satellite images and composite images from channels); this study used three channels, including:

- GeoColor imagery supplies as close an approximation to daytime True Color imagery as is possible from GOES-16 and thus allows for an intuitive understanding of meteorological and surface-based components.

- The Air Mass RGB analyzes the environment enclosing synoptic systems by improving air masses' temperature and moisture factors. Additionally, this RGB can determine between polar and tropical air masses, especially along upper-level frontal borders, and identify high-level, mid-level, and low-level clouds.

- Day cloud phase RGB is used to estimate the grade of the Cooling cloud: This RGB is used to evaluate the phase of cooling cloud tops to observe convective initiation, storm evolution, and decay. The Day Cloud Phase Distinction RGB takes benefit of cloud reflectance distinctions between the observable and near-infrared channels and temperature variances between ground and clouds in the infrared to supply improved contrast between background textures and grades of clouds (i.e., water vs. ice).

Step 1.2. Input splits

The real part is the actual value of the input image I^t, and the imaginary part is the difference between the nearest pixel and the previous pixel represented by the equation (7) below

$$HoD = I^{(t)} - I^{(t-1)}, \tag{7}$$

where HoD the value of the imaginary part of the input image, $I^{(t)}$ the real value of the input image at time t, $I^{(t-1)}$ the real value of the input image at time $t-1$.

Step 1.3. Mapping

The variation of elements in a band comes not only from the information in that band but can also be influenced by other image bands, so it is necessary to normalize the phase or transform data of the bands to improve high-quality input made according to the formula (8) below

$$MHoD_j^{(i)} = \frac{HoD_j^{(i)} + \alpha \sum_{k=1,k\neq i}^{B} HoD_j^{(k)}}{1 + \alpha\,(B-1)},\ j = 1, 2, ..., N, \tag{8}$$

where $MHoD_j^{(i)}$ the imaginary value of the jth pixel in band i after mapping, $HoD_j^{(i)}$ the imaginary value of pixel j^{th} in band i, $HoD_j^{(k)}$ imaginary value of pixel j^{th} in band k, α the correlation coefficient between the imaginary part of the ith color channel and the imaginary part of the remaining color channels, B number of bands used for forecasting, N number of pixels of the input image.

Step 1.4. Input data

After mapping in Step 1.3, the input data for the training process is synthesized from the real part data of the input data, and the imaginary part value after the mapping has the following form: $X^{(t)}\left(I^{(t)}, MHoD\right)$.

- **Step 2.** Clustering

  In this section, the study proposes the MapReduce_ComplexFuzzy model based on the fundamental improvement of the MapReduce_Fuzzy [11] algorithm to conduct real and imaginary parts clustering. The MapReduce_ComplexFuzzy proposed model is presented in Figure (4), where:

  - The input image is converted into a list for MapReduce processing.

  - The cluster centers of the real and imaginary parts are randomly initialized.

  - Perform data division into multiple windows (windows have different sizes and can be bounded by each other so that the ability to preserve information is the best). Each window is processed in parallel by MapTask to obtain new data.

  - MapTask processes all windows; new data from previous steps will be sorted, merged, and grouped in clusters.

  - ReduceTask will process clustered data to recalculate cluster centers.

  - The system checks the convergence of cluster centers. If not converged, go back and continue to implement MapTask and ReduceTask. If it has converged, then aggregate the clustering results as cluster centers V and degree matrices belonging to U to serve the rule generation process of the next step.

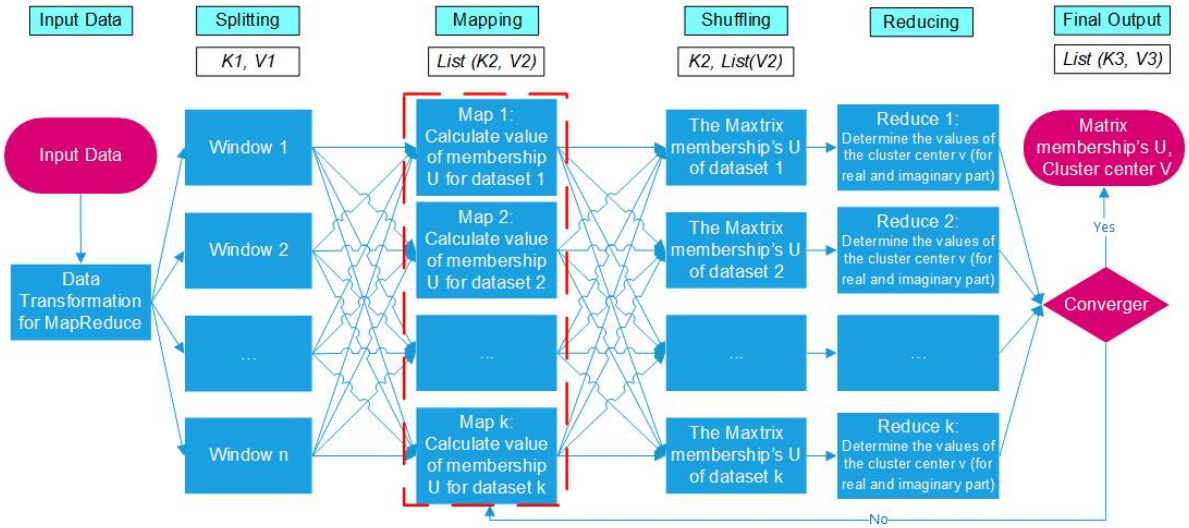


Figure 4: MapReduce complex fuzzy algorithm diagram

Details of the steps to implement the MapReduce_ComplexFuzzy algorithm are as follows.

Step 2.1. Data transformation for MapReduce complex fuzzy

Transform input data into lists containing information such as the relative position, and the real and imaginary part value of pixels. This process will later transform the input data to fit the MapReduce_ComplexFuzzy model and retrieve the relative pixel position later.

Step 2.2. Splitting the input image (Splitting)

From the data obtained from step 2.1, the study divides the input lists into regions of size $c \times c$ based on sliding windows. This process aims to split the data for distributed processing at the nodes but does not ignore the features that can be confused due to the image segmentation process.

Step 2.3. Mapping

Algorithm 1 Mapping Complex Fuzzy algorithm details

Input:

Each X_{ij} data element is a tuple consisting of row and column position information, real part value, imaginary part value (i, j, x_{ij}, HoD_{ij}), the fuzzifier m.

Output:

The result after convergence includes a set: The value of the center of cluster V and the list of elements belonging to the corresponding cluster V is the tuple (i, j, x_{ij}, HoD_{ij}).

Initialize:

Randomly initialize the value of the cluster center V of the real part and the imaginary part.

1: Iterate the elements in the data set X_{ij} one by one

2: Consider each value of cluster center μ_{ij} according to the following formula

$$\mu_{ij} = \left[\left[\sum\nolimits_{k=1}^{C} \sum\nolimits_{l=1}^{C} \left(\frac{d\left(\left(x_{ij}, HoD_{ij}\right), V_k\right)}{d\left(\left(x_{ij}, HoD_{ij}\right), V_l\right)}\right)^{\frac{2}{m-1}}\right]\right]^{-1}$$

where, $d\left(\left(x_{ij}, HoD_{ij}\right), V_l\right)$ is the euclidean distance of the pixel under consideration x_{ij}, HoD_{ij}, and the cluster center V_k

3: Update the parameter set of cluster center V and the value corresponding to cluster center V, including $(i, jx_{ij}, HoD_{ij}$ và $\mu_{ij})$

4: Update cluster centers V and $(i, j, x_{ij}, HoD_{ij}$, and $\mu_{ij})$ to set $lstK2V2$

5: The output is the set of $lstK2V2$

Step 2.4. Shuffling

Based on the values of cluster center V, this research will proceed to group the cluster centers with similar values into the same group. These groups will be saved to $K2lstV2$.

Step 2.5. Reduce

- **Step 3.** Training parallel

  Using the Co-Spatial CFIS+ [2] method and parallel computing technique to conduct parallel training of each band and each color channel in a band.

- **Step 4.** Synthesize rule

  Step 4.1. Rule pruning

  During the inference process, rules with a small number of pixel values can be inferred within the bounds of these rules. The results of this rule can lead to noisy predictions because only a tiny portion of the data is represented. Therefore, we will remove these rules and keep the ones with better representation.

Algorithm 2 Reduce Complex Fuzzy algorithm details

Input: The result obtained in step 2.4 (Shuffling) $K2lstV2$, the fuzzifier m

Output:

New cluster center value (real part, imaginary part) after being updated

1: Initialize the array c_{new} with the same number of elements as the original divisor cluster. Where values $c_{new} = 0$.

2: Initialize variable $totalU = 0$

3: For each key V_{ij} consider each pair of data points $(i, j, x_{ij}, HoD_{ij}, \mu_{ij})$, respectively. we have $c_{new} += (x_{ij}, HoD_{ij}) \times (\mu_{ij})^m$

4: $totalU += (\mu_{ij})^m$

5: Update $V_{ij} = \dfrac{c_{new}}{totalU}$

6: If V_{ij} is not converged, it will return to the Mapping step to recalculate the U value of membership.
Otherwise, return the value $K2lstV2$

We define the rule to be discarded according to the following expression (9)

$$\begin{cases} \frac{P(i)_{outside}}{N} \leq \varepsilon & \text{remove the } i^{th} \text{ rule} \\ \text{else wise} & \text{keep using the } i^{th} \text{ rule,} \end{cases} \tag{9}$$

where $P(i)_{outside}$ the total number of pixels to infer is outside the bound, ε rule rejection threshold value.

Step 4.2. Rule optimizing

To continue to improve the quality of the rule system, we will remove the duplicate rules in the rule system. Two rules i, j are considered duplicates if the following expression (10) is satisfied

$$\begin{cases} \frac{a_i}{a_j} + \frac{b_i}{b_j} + \frac{c_i}{c_j} + \frac{a'_i}{a'_j} + \frac{b'_i}{b'_j} + \frac{c'_i}{c'_j} \leq \theta & \text{merge } p^{th} \text{ and } q^{th} \text{ rule} \\ \text{elsewise} & \text{keep using } p^{th} \text{ and } q^{th} \text{ rule.} \end{cases} \tag{10}$$

The rule r is combined from two rules i, j satisfying the following expression (11)

$$a_r = \frac{a_i + a_j}{2};\ b_r = \frac{b_i + b_j}{2};\ c_r = \frac{c_i + c_j}{2}; a'_r = \frac{a'_i + a'_j}{2};\ b'_r = \frac{b'_i + b'_j}{2};\ c'_r = \frac{c'_i + c'_j}{2}. \tag{11}$$

- **Step 5.** Predict output image

  From the rule system obtained after training, conduct parallel processing between bands and each color channel to predict the output image of each color channel on each band. The forecast results of each channel and each band will then be aggregated into the final forecast image.

5. EXPERIMENTAL RESULTS

This section presents performance comparisons of the proposed method with Co-Spatial CFIS+ [2] in which the proposed approach demonstrates effectiveness and merit in comparison with the existing approaches.

5.1. Environmental setup

To ensure the quality and objectivity of experimental situations about data, algorithms, and matching measures. The research conducts experiments on the US Navy's [1] dataset in the same location at different time intervals. The dataset contains images collected in Hawaii (Data 1), U.S. Pacific Coast (Data 2), and Gulf of Mexico (Data 3), each dataset includes 3 bands GeoColor, Air Mass RGB, and Day cloud phase RGB, and using the Spatial CFIS+ algorithm and on two measures, $RMSE$ and R^2, to compare the predictive quality and reliability of the model with the number of Clusters of 10 and then to evaluate the efficiency of the model in terms of processing time, the research adds comparisons on 14, 17, and 20 nodes. Experimental results were aggregated and averaged over ten experimental runs. The result is performed on VXRAIL S470 system server with eight physical server nodes; each physical server node has an Intel E5-2660 V414C 2.0 GHz processor, 256 Gb RAM, and 512 Gb Hard Drive.

5.2. Evaluation measures

To evaluate the effectiveness of the proposed model, we use two performance indexes: The R squared (R^2) [12] measures, and the root mean square error (RMSE) [13]. These indexes are defined as fomulas (12), (13) as follows

$$RMSE\ (X^{db},\ X^{(t+1)}) = \sqrt{\sum_{i=1}^{n} \left(X_i^{db} - X_i^{(t+1)}\right)^2}, \tag{12}$$

$$R^2 = 1 - \frac{RSS}{TSS}, \tag{13}$$

where $X^{(t+1)}$ the observed value, X^{db} the predicted value, n the number of samples, RSS sum of squared residuals, TSS total sum of squares.

5.3. Results

The results of comparing the proposed model and the Spatial CFIS+ model on the RMSE measure are shown in Table 1 and Figure 5.

From the above results, the proposed model has an average RMSE result for ten experimental runs, showing that the model's accuracy has been improved through the average value of the total RMSE of ten runs over three times better. The dataset of the proposed model has been reduced by 10% compared to the Spatial CFIS model. The proposed model still uses the spatial-temporal complex fuzzy inference system as the basis for the rule generation process, but the prediction quality is significantly improved because the imaginary

Table 1: Average RMSE results of models on three datasets

Data	Spatial CFIS+					Proposed method				
	Forecast image 1	Forecast image 2	Forecast image 3	Forecast image 4	Forecast image 5	Forecast image 1	Forecast image 2	Forecast image 3	Forecast image 4	Forecast image 5
Data 1	3.237	6.436	6.903	7.078	8.388	2.825	5.876	6.515	6.442	7.432
Data 2	5.582	5.840	6.328	6.493	8.210	5.105	5.287	5.795	5.909	7.601
Data 3	6.544	7.280	7.814	7.967	8.932	6.051	6.640	7.193	7.247	8.020

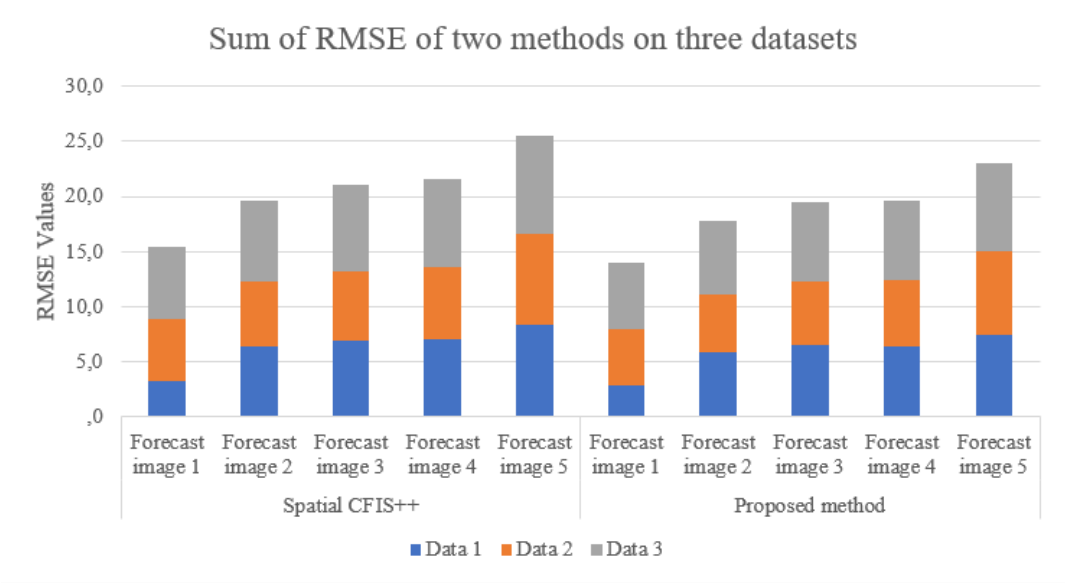


Figure 5: Sum of RMSE of two methods on three datasets

preprocessing of the model is synthesized with additional factors. Impact factors from other channels help enhance the quality of the forecast.

In addition, the proposed model and Spatial CFIS+, both models have problems related to cumulative errors. The more predictive images, the greater the error. However, with the proposed model, the cumulative error between consecutive images has been significantly reduced compared to the Spatial CFIS+ model, which plays an essential role in improving the model's efficiency.

The results of comparing the proposed model and the Spatial CFIS+ model on the R^2 measure are shown in Table 2 and Figure 6.

Table 2: Results R^2 of the proposed method and Spatial CFIS+

Data	Spatial CFIS+					Proposed method				
	Forecast image 1	Forecast image 2	Forecast image 3	Forecast image 4	Forecast image 5	Forecast image 1	Forecast image 2	Forecast image 3	Forecast image 4	Forecast image 5
Data 1	0.964	0.961	0.966	0.967	0.965	0.970	0.965	0.971	0.974	0.970
Data 2	0.966	0.961	0.961	0.959	0.963	0.972	0.966	0.967	0.965	0.967
Data 3	0.960	0.959	0.964	0.967	0.966	0.966	0.966	0.969	0.972	0.972

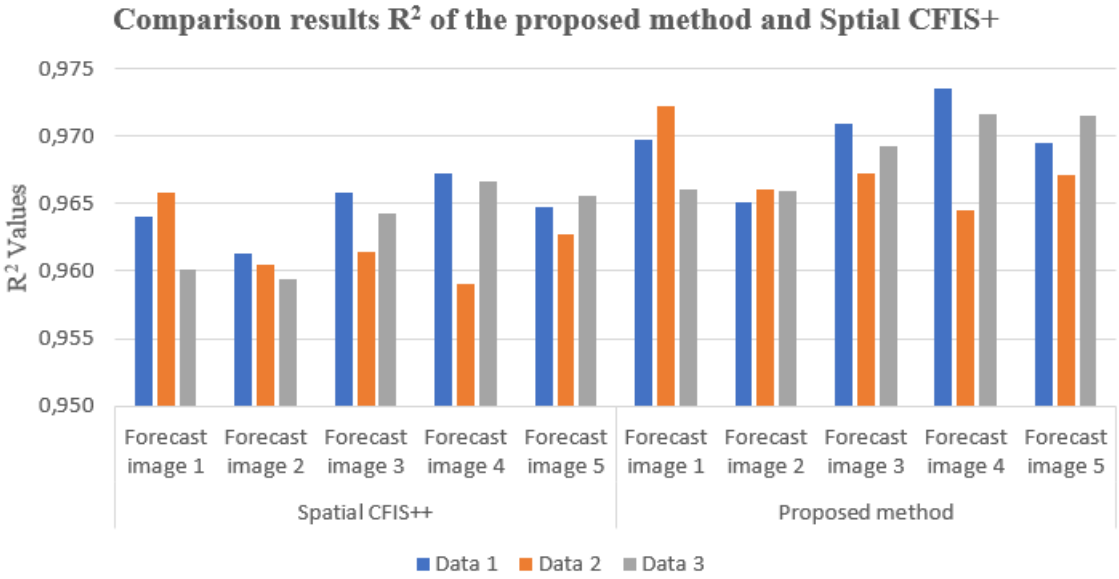


Figure 6: Comparison results R^2 of the proposed method and Spatial CFIS+

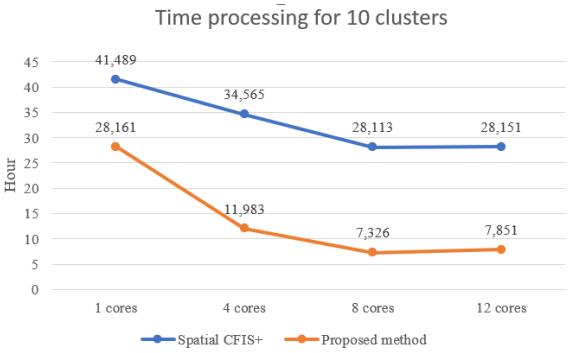
From the results of comparing R^2 of the two models on the three datasets shown in Table 2 and Figure 6, we can see that the correlation of the proposed model and the Spatial CFIS+ model is relatively similar to the total difference value of 0.6%.

Especially with Data 1 and Data 3, there is a relatively straightforward difference in the corresponding forecast images compared with the Spatial CFIS+ model. This shows that the forecast results from the proposed model have high confidence in these datasets. The experimental results of the total processing time of the two models on the three datasets are shown in Table 3 and Figure 7 below.

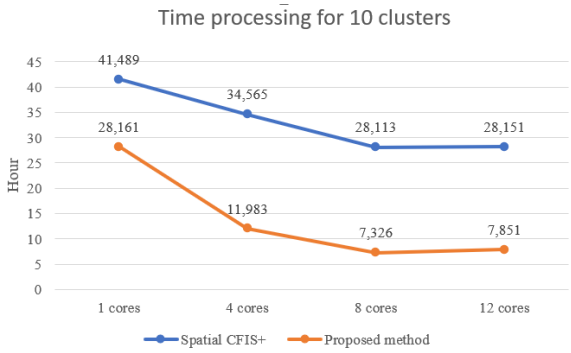
The experimental results of the total processing time of two models on three datasets are shown in Table 3 and Figure 7, indicating that the processing time of the proposed model is significantly reduced. The proposed model has a total processing time on all three datasets from 3 to 5 times faster than the Spatial CFIS+ model. When the number of clusters is increased, the proposed model shows its superiority.

Table 3: Comparing the total processing time of the proposed model and Spatial CFIS+ on different Cores.

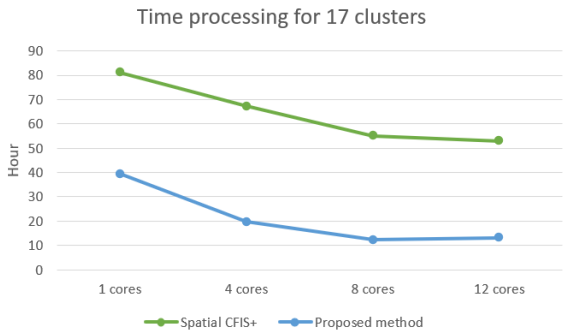
Data	Spatial CFIS+				Proposed method			
	10 Cluster	14 Cluster	17 Cluster	20 Cluster	10 Cluster	14 Cluster	17 Cluster	20 Cluster
1 core	41.489	73.416	81.098	104.345	28.161	33.651	39.381	46.519
4 cores	34.565	61.561	67.203	86.731	11.983	17.416	19.712	29.943
8 cores	28.113	50.394	55.010	71.382	7.326	9.376	12.333	19.447
12 cores	28.151	49.841	53.026	70.793	7.851	9.703	13.095	19.456



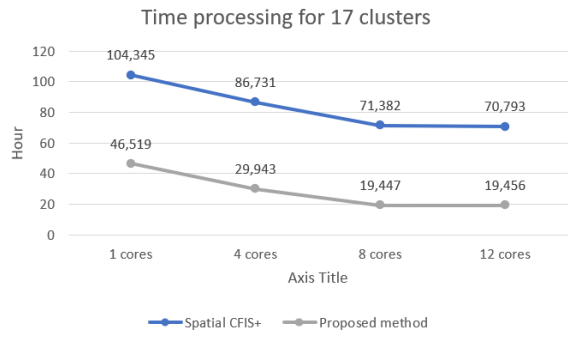
(a) Time processing for 10 clusters



(b) Time processing for 14 clusters



(c) Time processing for 17 clusters

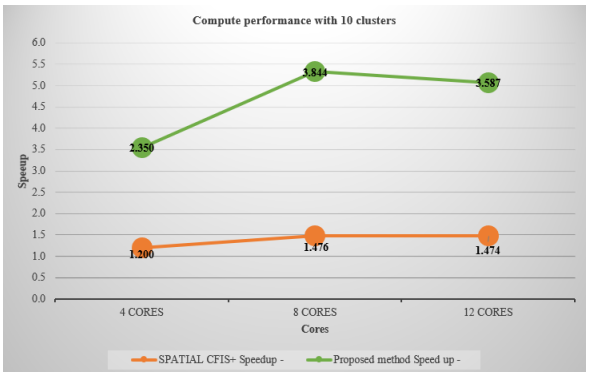


(d) Time processing for 20 clusters

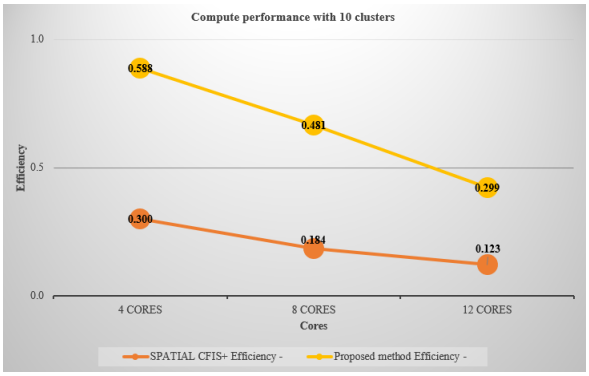
Figure 7: Computational time of methods on three datasets

Table 4: The Summary table of performance evaluation results on 10 clusters

No Core CPU	SPATIAL CFIS+		Proposed method	
	Speedup	Efficiency	Speedup	Efficiency
1 cores	-	-	-	-
4 cores	1.200	0.300	2.350	0.588
8 cores	1.476	0.184	3.844	0.481
12 cores	1.474	0.123	3.587	0.299



(a) Speedup

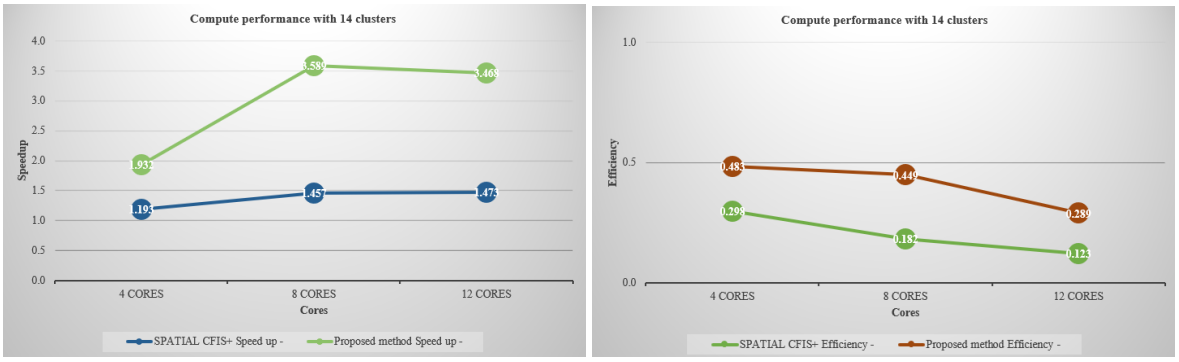


(b) Efficiency

Figure 8: The total speedup and efficiency factor between three datasets of 10 clusters

Table 5: The Summary table of performance evaluation results on 14 clusters

No Core CPU	SPATIAL CFIS+		Proposed method	
	Speedup	Efficiency	Speedup	Efficiency
1 cores	-	-	-	-
4 cores	1.193	0.298	1.932	0.483
8 cores	1.457	0.182	3.589	0.449
12 cores	1.473	0.123	3.468	0.289

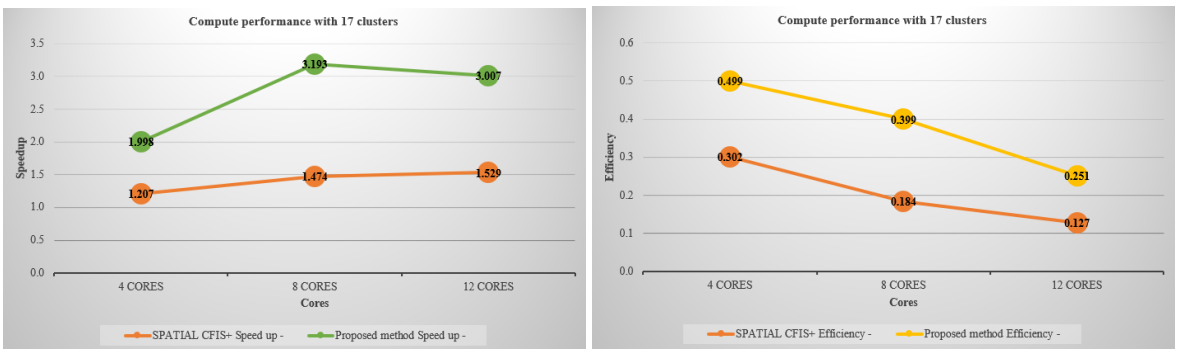


(a) Speedup (b) Efficiency

Figure 9: The total speedup and efficiency factor between three datasets of 14 clusters

Table 6: The Summary table of performance evaluation results on 17 clusters

No Core CPU	SPATIAL CFIS+		Proposed method	
	Speedup	Efficiency	Speedup	Efficiency
1 core	-	-	-	-
4 cores	1.207	0.302	1.998	0.499
8 cores	1.474	0.184	3.193	0.399
12 cores	1.529	0.127	3.007	0.251



(a) Speedup (b) Efficiency

Figure 10: The total speedup and efficiency factor between three datasets of 17 clusters

The results in the Table 4, 5, 6 shows that the speedup value will be most effective when the number of CPU cores is 8, and the efficiency value will be the most effective when the number of CPU cores is 4. This is consistent with the assessment in experiments. When performing parallel computation on multiple CPUs, the increase in the amount of the core is not proportional to the efficiency of the model's processing time.

The Speedup value effectively describes the computation time of the model with one and multiple CPU cores. For both models, this value tends to decrease as the number of cores increases. However, the proposed model gives much better results in all cases.

The efficiency value determined by the ratio of computational efficiency over time and the number of cores, tends to decrease as the number of cores increases. However, the proposed model gives better results, up to 100%, than the Spatial CFIS+ model.

When the number of clusters changes these values, the proposed model still maintains speedup and efficiency. Still, the model tends to slow down significantly when the number of clusters becomes very large.

Although the proposed model for the efficiency and speedup values of the proposed model has improved, the new efficiency values are only at a relative threshold between approximately 0.4 and 0.5.

6. CONCLUSIONS

This research has focused on presenting the application model of a spatial-temporal complex fuzzy inference system in predicting the change of satellite image series with the following main characteristics:

- Propose a way to synthesize the new imaginary part based on the synthesis of its imaginary part and the imaginary part synthesized from the other channels.

- Propose an algorithm that uses MapReduce in the data clustering process called MapReduce_ComplexFuzzy based on the improvement of the MapReduce_Fuzzy algorithm.

- Propose a mechanism for synthesizing rules and optimizing the rule system.

Although the model has had initial outcomes, as mentioned above, the model still has some limitations:

- The MapReduce method and distributed processing in the proposed model are only processed at the clustering step to replace the FCM algorithm, so the model's processing speed has remained relatively high.

- The rule reduction and optimization mechanism of the rule system in the model are still relatively simple, so the rule has yet to be improved much.

REFERENCES

[1] N. Oceanic and A. Administration, "Mtsat west color infrared loop," https://www.star.nesdis.noaa.gov/GOES/index.php.

[2] L. Giang, L. Son, N. Giang, and et al., "A new co-learning method in spatial complex fuzzy inference systems for change detection from satellite images," *Neural Comput & Applic*, vol. 35, pp. 4519–4548, 2023.

[3] B. Gao and R. Li, "Improving water leaving reflectance retrievals from abi and ahi data acquired over case 2 waters from present geostationary weather satellite platforms," *Remote Sensing*, vol. 12, no. 19, p. 3257, 2020.

[4] S. Bao, Z. Zhang, E. Kalina, and B. Liu, "The use of composite goes-r satellite imagery to evaluate a tc intensity and vortex structure forecast by an fv3gfs-based hurricane forecast model," *Atmosphere*, vol. 13, no. 1, p. 126, 2022.

[5] Y. Zhao, G. Wang, G. Etiope, and et al., "New observation operators for cloud liquid/ice water path from abi and their impact on assimilation and hurricane forecasts," *Journal of Geophysical Research: Atmospheres*, vol. 126, no. 10, 2021.

[6] Y. Liu, X. Lu, W. Peng, and et al., "Compression and regularized optimization of modules stacked residual deep fuzzy system with application to time series prediction," *Information Sciences*, vol. 608, pp. 551–577, 2022.

[7] P. D. Phong, N. D. Du, P. H. Hiep, and T. X. Thanh, "A hybrid pso-sa scheme for improving accuracy of fuzzy time series forecasting models," *Journal of Computer Science and Cybernetics*, vol. 38, no. 3, pp. 275–293, 2022.

[8] P. D. Phong, "A time series forecasting model based on linguistic forecasting rules," *Journal of Computer Science and Cybernetics*, vol. 37, no. 1, pp. 23–42, 2021.

[9] R. Lämmel, "Google's mapreduce programming model—revisited," *Science of Computer Programming*, vol. 70, no. 1, pp. 1–30, 2008.

[10] G. S. et al., "A new design of mamdani complex fuzzy inference system for multiattribute decision making problems," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 4, pp. 716–730, 2021.

[11] T. N. Tu, "A fuzzy approach of large size remote sensing image clustering," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 11, no. 4, pp. 187–198, 2020.

[12] T. O. Kvålseth, "Cautionary note about r-squared," *The American Statistician*, vol. 39, no. 4, pp. 279–285, 1985.

[13] D. Christie and S. Neill, *Measuring and Observing the Ocean Renewable Energy Resource.* Elsevier: Amsterdam, The Netherlands, 2021.

Received on January 14, 2023
Accepted on February 28, 2023