BAREKENG: Journal of Mathematics and Its Applications
December 2025 Volume 19 Issue 4 Page 2921–2936
P-ISSN: 1978-7227 E-ISSN: 2615-3017
https://doi.org/10.30598/barekengvol19iss4pp2921-2936

IMPUTATION OF MISSING DAILY RAINFALL DATA USING CONVOLUTIONAL NEURAL NETWORKS (CNN) WITH SPATIAL INTERPOLATION

Lilis Sriwahyuni[1], Sri Nurdiati[2*], Endar Hasafah Nugrahani[3], Ihwan Sukmana[4], Mohamad Khoirun Najib[5]

[1,4]*Master's Student of Applied Mathematics, Cluster of Mathematics, IPB University*
[2,3,5]*Cluster of Mathematics, IPB University*
Jln. Meranti Kampus Dramaga, Bogor, 16680, Indonesia

*Corresponding author's e-mail: *nurdiati@apps.ipb.ac.id*

Article History:

Received: 3rd April 2025
Revised: 2nd May 2025
Accepted: 10th June 2025
Available online: 1st September 2025

Keywords:

Convolutional Neural Network;
Imputation Missing Data;
Interpolation Spline;
Machine Learning;
Mean Absolute Error.

ABSTRACT

Accurate rainfall estimation is crucial in climate analysis and water resource planning. Observational data from weather stations play a vital role in climatological analysis as they represent actual conditions at specific locations. However, many observation stations in Indonesia need more complete data, hindering analysis and data-driven decision-making. To address this issue, this study aims to impute missing rainfall data for BMKG stations in East Java using the Convolutional Neural Network (CNN) method. Satellite data used in this study include ERA5 without interpolation and ERA5 with interpolation. The study employs a spatial interpolation approach. Data were split into training and testing datasets with various ratios: 95:5%, 90:10%, 80:20%, 70:30%, and 50:50%. The results show that the CNN method with spatially interpolated satellite data yields better results, with a Mean Absolute Error (MAE) of 7.50 on the training data and 7.05 on the testing data, indicating better generalization capability than the method without interpolation. The combination of CNN and ERA5 with interpolation was chosen for imputing missing rainfall data at BMKG stations in East Java due to its lower MAE.

How to cite this article:

L. Sriwahyuni, S. Nurdiati, E. H. Nugrahani, I. Sukmana and M. K. Najib., "IMPUTATION OF MISSING DAILY RAINFALL DATA USING CONVOLUTIONAL NEURAL NETWORKS (CNN) WITH SPATIAL INTERPOLATION," *BAREKENG: J. Math. & App.*, vol. 19, iss. 4, pp. 2921-2936, December, 2025.

Journal homepage: *https://ojs3.unpatti.ac.id/index.php/barekeng/*
Journal e-mail: *barekeng.math@yahoo.com; barekeng.journal@mail.unpatti.ac.id*

Research Article · Open Access

1. INTRODUCTION

Rainfall is one of the natural water sources that play a significant role in replenishing Earth's water resources and supporting the hydrological cycle. In tropical Indonesia, rainfall is a key component of the water cycle, mainly since precipitation in many regions occurs seasonally. Adequate rainfall is crucial for the agricultural sector [1], energy [2], urban spatial planning [3], and maintaining the balance of the ecosystem on land and at sea [4]. Therefore, an accurate understanding of rainfall and systematic monitoring are essential to support various aspects of human life, sustain natural ecosystems, and mitigate the risks of disasters such as droughts or floods.

Rainfall data is obtained through pluviometers or rain gauges placed at weather stations in various locations. Pluviometers measure the amount of rain that falls on a surface over a specific period. The data collected from these instruments is crucial as it is a primary reference for various climatological and hydrological studies [5], [6], [7]. However, rain gauges have limitations in capturing spatial variations in rainfall, especially over large areas or regions with complex topography, such as mountainous terrains. Additionally, these instruments are prone to technical and human errors, including equipment malfunctions, inadequate maintenance, and inaccurate manual recording. These issues can result in missing data, making it challenging for researchers to comprehensively explain the stochastic processes of rainfall [8].

In climatological data analysis, observation data from stations is crucial, representing measurements taken directly from the stations [9][10]. However, due to the significant amount of incomplete station data in Indonesia, analyzing rainfall data using station data alone becomes challenging. Consequently, many researchers turn to satellite data for their analyses [11], [12], [13]. However, generated data from a specific model inevitably contains errors. As a result, the analysis outcomes will also carry these errors. Bias correction is a method researchers use to reduce the mistakes in satellite data [14]. However, this method only reduces errors, particularly systematic errors.

As a result, analysis with satellite data will not necessarily be better than analysis with station data. Therefore, missing data imputation is a method to handle incomplete station data to make it analyzable and produce accurate results. According to [15], imputation replaces missing data with estimated values based on other available information. One approach that can be used for missing data imputation is machine learning [16]. The study by [17], which compared machine learning-based methods with statistical methods for missing data imputation, showed that machine learning methods were more accurate in imputing missing data than statistical methods.

A similar study was conducted by [13], where they researched missing data imputation using Beidou satellite data in China, applying the Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) method. [18] developed an efficient method to impute missing data in satellite imagery using CNN for Aerosol Optical Depth (AOD) data as well [19]. They integrated spatial, temporal, and spectral dimensions using deep CNNs to address the issue of missing data in satellite imagery. [20] studied missing data imputation using several satellite datasets, including ERA5, ERA5 Land, CMORPH CRT, CMORPH BLD, and CHIRPS. The results showed that ERA5 data provided the best performance. In this study, we propose using relatively more complete satellite data to fill in the missing station data. The method to be employed is a Convolutional Neural Network (CNN).

The main objective of this study is to compare the performance of two approaches for imputing missing data, namely using ERA5 satellite data and ERA5 satellite data with an interpolation spline. This comparison assesses the extent to which spline interpolation contributes to reducing rainfall estimation errors. The analysis will involve testing several evaluation metrics, such as Mean Absolute Error (MAE) and correlation, to ensure objective and measurable results. This research's novelty is the application of interpolation to ERA5 satellite data to compare its performance with ERA5 satellite data without interpolation. The findings from this study are expected to provide deeper insights into a more effective approach for imputing missing rainfall data using ERA5 data.

2. RESEARCH METHODS

2.1 Study Area and Datasets

This study focuses on the East Java Province, covering an area of approximately 47.799,75 km². Geographically, East Java is located between 5,37° - 8,48° South Latitude and 111,0° - 114,4° East Longitude. East Java experiences a varied climate due to its diverse topography, ranging from mountains to coastal areas. As one of Indonesia's major agricultural producers, the availability of accurate rainfall data is crucial for supporting water resource management and agricultural sector planning. However, the rainfall data collected by BMKG observation stations in East Java often suffers from missing data due to various factors, such as technical and human errors. The specifications of the station data are shown in **Table 1**. The ERA5 satellite data is used as an alternative to address the issue of missing data. This study implements two main approaches to solve the missing rainfall data problem at BMKG stations in East Java. The first approach uses ERA5 satellite data directly. ERA5 is a global weather reanalysis dataset developed by the European Centre for Medium-Range Weather Forecasts (ECMWF) through the Copernicus Climate Change Service (C3S) project. This dataset provides historical weather and climate information based on weather observations (such as satellite observations, weather station data, and ocean data) and complex atmospheric models. ERA5 offers high spatial resolution data (approximately 31 km) and detailed temporal resolution (hourly).

Additionally, ERA5 provides complete rainfall data, which can be used to fill missing data at BMKG stations in East Java. The second approach involves using ERA5 satellite data that has undergone spline interpolation. This process is applied to refine the spatial resolution of the ERA5 data, hoping to improve the accuracy of the missing data imputation.

Table 1. East Java BMKG Station Data Specifications

Station	Station Name	Longitude	Latitude	Altitude	Available data	Missing data 2010-2023
1	Geofisika Malang	112.45000	-8.15000	285	2010-2024	34.18%
2	Geofisika Nganjuk	111.76682	-7.73486	723	1982-2024	49.53%
3	Geofisika Pasuruan	112.63533	-7.70456	832	1983-2024	31.59%
4	Klimatologi Jawa Timur	112.59790	-7.90080	590	1988-2024	7.18%
5	Meteorologi Banyuwangi	114.35530	-8.21500	52	1980-2024	34.04%
6	Meteorologi Juanda	112.78330	-7.38460	3	1981-2024	11.15%
7	Meteorologi Maritim Tanjung Perak	112.73530	-7.20530	3	1981-2024	10.44%
8	Meteorologi Perak 1	112.72390	-7.22360	3	1972-2024	15.94%
9	Meteorologi Sangkapura	112.65780	-5.85110	3	1971-2024	19.00%
10	Meteorologi Trunojoyo	113.91400	-7.03976	3	1980-2024	15.32%

Table 1 presents information about ten weather stations in East Java operating over various periods. Each station is characterized by its geographical coordinates (longitude and latitude), elevation (altitude), and the range of years for which data is available, with some stations having data records dating back to the 1970s through 2024. In this study, rainfall data from 2010 to 2023 is used. The missing data rate varies across the stations. The station with the highest missing data rate is Geofisika Nganjuk, with a percentage of 49.53%, while the Klimatologi Jawa Timur station records the lowest missing data rate at 7.18%.

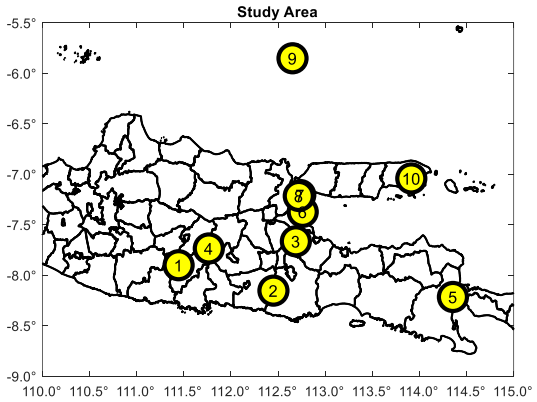


Figure 1. Location of East Java BMKG Station

Figure 1 displays a map of East Java with the locations of weather stations identified by number according to the data in the table. Geographically, the stations are distributed across strategic areas for observation. This distribution allows for relatively even coverage of rainfall data and other weather parameters throughout East Java despite variations in each station's elevation and geographical positioning. The well-distributed stations across mountainous regions, lowlands, and coastal areas are crucial to ensuring the accuracy of climate data, as topography significantly influences rainfall patterns. However, the challenges associated with varying missing data across each station, as shown in **Table 1**, impact the completeness of historical data and weather analysis in several regions of the area.

2.2 Spline Interpolation

Interpolation is a method for estimating the value of a function. $f(x)$ at an unknown point, based on the values $f(x)$, which has been known at specific points $(x_0, x_1, \dots, x_{N-1})$, by drawing a smooth curve through the data points, interpolation is performed if the point being searched for is within a known range of points, and extrapolation is performed outside that range [21]. A cubic polynomial between each pair of data points defines a cubic spline (x_i, y_i). If we have n data point $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$, then the cubic spline function $S(x)$ can be written in **Equation (1)**.

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + c_i(x - x_i)^3 \quad (1)$$

for $x_i \le x \le x_{i+1}$, $S_i(x)$: a cubic spline for the interval $[x_i, x_{i+1}]$ and a_i, b_i, c_i, d_i: the coefficient that must be determined for each interval.

Spline interpolation is an alternative method of interpolation that is more interesting from a theoretical point of view than radial function-based interpolation [22]. In spline interpolation, we look for a prediction function $\hat{y}(s)$ that meets the exact interpolation requirements; the function must pass through the given data points. This interpolation is expressed in **Equation (2)**.

$$\hat{y}(s) = y_i \quad (2)$$

$i = 1, \dots, n$, although many smooth functions can satisfy this requirement, spline interpolation has the unique characteristic of selecting the smoothest function. To measure this "smoothness," we refer to the concept of curvature. The curvature of a function $f(s)$ in one dimension, measured using the second derivative $f''(s)$, where the linear function $f(s) = a + bs$ has zero curvature. To compare the curvature of a function at a specific interval, the total curvature can be calculated using the formula in **Equation (3)**.

$$C(f) = \int_a^b [f''(s)]^2 ds \quad (3)$$

The smaller the total curvature, the smoother the function [23]. The concept of curvature is extended using the Hessian matrix in two-dimensional problems, which consists of each variable's second partial derivatives of a function. The curvature of a two-dimensional function at the point (s_1, s_2) counted as **Equation (4)**.

$$H(s) = \begin{bmatrix} \frac{\partial^2 f}{\partial s_1^2} & \frac{\partial^2 f}{\partial s_1 \partial s_2} \\ \frac{\partial^2 f}{\partial s_2 \partial s_1} & \frac{\partial^2 f}{\partial s_2^2} \end{bmatrix} \quad (4)$$

Curvature is measured by calculating the size of the Hessian, which is formulated as **Equation (5)**.

$$\|H(s)\|^2 = \left(\frac{\partial^2 f}{\partial s_1^2}\right)^2 + 2\left(\frac{\partial^2 f}{\partial s_1 \partial s_2}\right)^2 + \left(\frac{\partial^2 f}{\partial s_2^2}\right)^2 \quad (5)$$

The total curvature in two dimensions is calculated by integrating the curvature values over the entire region **Equation (6)**.

$$C(f) = \int_{R^2} \|H(s)\|^2 ds_1 ds_2 \quad (6)$$

The problem of spline interpolation is finding a function $\hat{y}(s)$ that minimizes this total curvature while still satisfying the interpolation conditions above.

2.3 Convolutional Neural Network (CNN)

A Convolutional Neural Network (CNN) is an Artificial Neural Network (ANN) designed to process grid-like structured data, such as images. ANN is a machine-learning algorithm inspired by the function and structure of the human brain [24]. An ANN consists of interconnected artificial neurons that make predictions. It comprises three layers: the input, hidden, and output layers. Each layer contains a certain number of neurons responsible for processing input data, transforming it through weights and biases, and generating an output. Backpropagation is an algorithm used to train an ANN by adjusting the weights and biases of neurons to minimize the difference between the actual and predicted output. The concept of backpropagation is based on learning from mistakes. The network "learns" how to adjust its weights to improve its predictions when they are incorrect [25].

A CNN model generally consists of two stages: feature learning and classification. Feature learning includes convolutional layers and sub-sampling, while classification consists of fully connected layers [26]. In a CNN model, the first step is feature learning. During this stage, convolution is performed, resulting in a feature map output. The convolutional layer consists of four components: padding, stride, kernel, and activation function. Padding is essential for enlarging the input by adding zeros to each side, ensuring that information at the boundaries is preserved when the convolution kernel is adjusted to a specific size. Moreover, stride is used to control the density of the convolution. Stride refers to the length of the shifting step—a more significant stride results in a lower density of the convolution. In a two-dimensional CNN, the kernel is a matrix of size $n \times n$ and the kernel is a matrix of values containing a set of parameters. Each submatrix of the input is element-wise multiplied by the kernel, then summed up and passed through an activation function. After convolution, the feature map consists of several features prone to overfitting [27]. Therefore, a sub-sampling layer is proposed to avoid redundancy [28]. Various pooling techniques are used, such as max pooling, min pooling, average pooling, gated pooling, tree pooling, etc. Max pooling is the most popular and widely used pooling technique. After the convolution and sub-sampling processes are completed, the final feature map is input to the fully connected layer. The fully connected layer is used for regression, where each neuron is connected to every neuron from the previous layer. The final layer of the fully connected layer serves as the output (regression) layer of the CNN architecture. Figure 2 illustrates the procedure of a two-dimensional CNN.

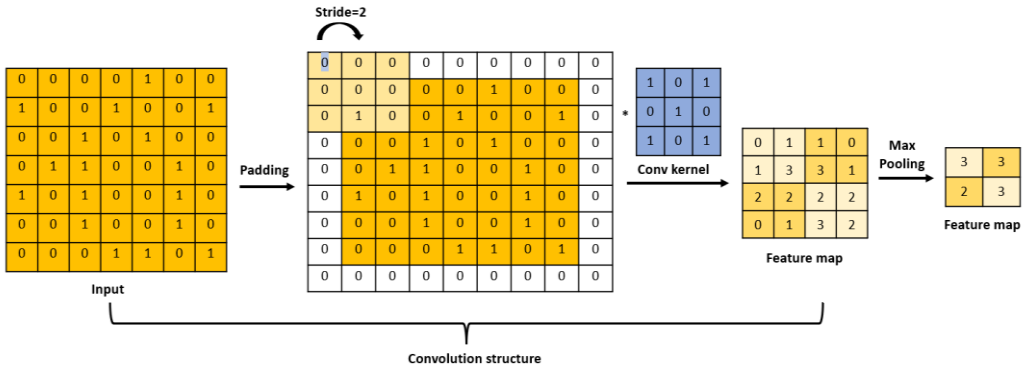


Figure 2. The Procedure of a Two-Dimensional CNN. Source [29]

2.4 Activation Function

In neural network-based models, the main task of the activation function is to map the input to the output. The input value is obtained by calculating the weighted sum of the inputs to the neuron and then adding the bias (if there is a bias). In other words, the activation function determines whether a neuron will activate for a specific input by producing the corresponding output. The activation functions used in this study are as follows.

2.4.1 Sigmoid

The sigmoid activation function is suitable for non-linear data because it captures complex relationships between input and output. This helps the model learn patterns that simple linear functions cannot handle. The curve of the sigmoid activation function has an "S" shape, as shown in **Figure 3** (a).

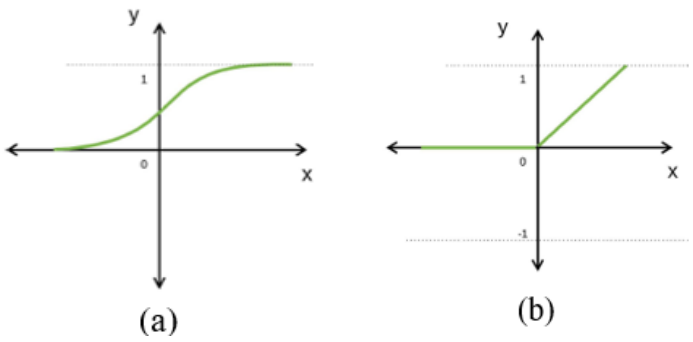


Figure 3. (a) Sigmoid Curve, (b) ReLU Curve

The sigmoid function takes real numbers as input and maps the output to an interval [0,1]. A small value of x will approach 0, while a significant value of x will approach 1. The mathematical representation of the Sigmoid in **Equation (7)**.

$$f(x)_{sigm} = \frac{1}{1 + e^{-x}} \tag{7}$$

x: the input or net of the neuron and e: an exponential number.

2.4.2 ReLu

Rectifier Linear Unit (ReLU) ReLU is the most commonly used activation function in Convolutional Neural Networks **[30]**. It is used to convert all input values into positive numbers. The advantage of this activation function is that it requires minimal computational cost compared to other functions. The mathematical representation of ReLU is as follows in **Equation (8)**.

$$f(x)_{ReLU} = \max(0, x) \tag{8}$$

x: the input or net of the neuron.

2.5 Optimizer

Before using the CNN model, it must be trained using training data to minimize the loss function. The loss function is a metric used in machine learning to assess how well or poorly the predictive model performs. This function calculates the difference between the model's predictions and the actual values. The loss value is used to guide the model training process: the smaller the loss value, the better the model's performance. This study uses Mean Absolute Error (MAE) to calculate the loss value, as described by **[31]**.

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |O_i - P_i| \tag{9}$$

n: the amount of data, O_i: the original or actual value, and P_i: the predicted value.

In the CNN model, an optimizer is used to update the weights and parameters of the model during training to minimize the loss function. Each parameter update is based on the learning rate, which determines the step size in the parameter update. An epoch refers to one complete iteration using all of the training data. Since the learning rate is a vital hyperparameter, its selection must be done carefully to avoid hindering the learning process. This study uses three optimizers, considered the best by **[26]**, and will be explained as follows.

2.5.1 Adaptive Moment Estimation (Adam)

Adam is an advanced gradient descent method that computes an adaptive learning rate for each parameter in the network and combines the advantages of Momentum and RMSprop. It maintains the exponentially decaying average of past gradients, like Momentum, and the exponentially decaying average of past squared gradients, like RMSprop. Therefore, the Adam formula is as follows:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \tag{10}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \tag{11}$$

$$\widehat{m_t} = \frac{m_t}{(1-\beta_1^t)} \tag{12}$$

$$\widehat{v_t} = \frac{v_t}{(1-\beta_2^t)} \tag{13}$$

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\widehat{v_t}} + \epsilon} \widehat{m_t} \tag{14}$$

where m_t is the moving average of the gradients at time t, v_t is the exponentially weighted average of squared gradients, g_t is gradient, η is learning rate, $\widehat{m_t}$ and $\widehat{v_t}$ are bias corrections, β_1 and β_2 are the exponential decay rate. The default values of β_1, β_2, and ϵ, the recommended ones to be set in succession, are 0.9, 0.999, and 10^{-8} [32].

2.5.2 Nesterov Adaptive Moment Estimation (NAdam)

Nadam [33] is a modified version of the Adam optimizer that combines the advantages of both Adam and Nesterov Momentum. Like Adam, Nadam adapts the learning rate for each parameter based on the gradients' first moment (Momentum) and second moment (squared gradients). However, Nadam also incorporates a component of Nesterov Momentum to improve convergence speed. The updated formula for Nadam is as follows in **Equation (15)**.

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\widehat{v_t}} + \epsilon} \left(\beta_1 \widehat{m_t} + \frac{1-\beta_1}{1-\beta_1^t} g_t \right) \tag{15}$$

2.5.3 Adaptive Moment Estimation with Weight Decay (AdamW)

AdamW is a variation of the Adam optimizer that adds weight decay directly into the parameter update, providing better control over regularization. This optimizer was introduced to address the issue of ineffective regularization in the Adam algorithm. The updated formula for AdamW is as follows in **Equation (16)**.

$$\theta_{t+1} = \theta_t - \zeta_t \left(\frac{\eta}{\sqrt{\widehat{v_t}} + \epsilon} \widehat{m_t} + \lambda \theta_t \right) \tag{16}$$

Where λ: the weight decay value, AdamW uses l_2 regularization in the calculation of the gradient of the parameter θ_t, which is written as $g_t = \frac{\partial L(\theta_t)}{\partial(\theta)} + \lambda\theta_t$. To adjust the scheduling learning rate η and weight decay λ, AdamW introduces a scaling factor ζ_t, which the user can set through procedures SetScheduleMultiplier(t). At each iteration, the value ζ_t will decrease gradually following the cosine annealing method, where the learning rate decreases slowly in each batch during training [34].

3. RESULTS AND DISCUSSION

3.1 Preprocessing of Datasets

3.1.1 Station Data

Daily rainfall (RR) data were collected in Excel from the official BMKG East Java website, covering January 2010 to December 2023. The downloaded data were compiled into a tabular format to facilitate data analysis. Based on the analysis, the average percentage of missing data from January 2010 to December 2023 is 22.87%. This value was obtained by calculating the percentage of missing data for each station, summing the results, and then dividing by the total number of stations used, which is ten. The tabular data was preprocessed using MATLAB software to prepare the data for imputation. During this stage, the index value "8888" used to indicate unmeasured data and empty cells was converted to Not a Number (NaN).

Table 2. Missing Data Proportion in East Java BMKG Station

Station	Station Name	Missing data 2010-2023
1	Geofisika Malang	34.18%
2	Geofisika Nganjuk	49.53%
3	Geofisika Pasuruan	31.59%
4	Klimatologi Jawa Timur	7.18%
5	Meteorologi Banyuwangi	34.04%
6	Meteorologi Juanda	11.15%
7	Meteorologi Maritim Tanjung Perak	10.44%
8	Meteorologi Perak 1	15.94%
9	Meteorologi Sangkapura	19.00%
10	Meteorologi Trunojoyo	15.32%

Table 2 shows the proportion of missing data for each station. This study focuses on determining the satellite data with the best performance for imputing missing rainfall data at BMKG East Java stations.

3.1.2 Satellite ERA5 Dataset

The satellite rainfall data used in this study is ERA5 data in a grid format with a spatial resolution of 0.25°×0.25°. The dataset contains four variables: longitude, latitude, time, and total precipitation (tp). The data is stored in Network Common Data File (NetCDF) format. The dataset spans hourly data from January 2010 to December 2023. ERA5 satellite data was preprocessed using MATLAB software, and the specifications of the preprocessing results are presented in Table 3.

Table 3. ERA5 Satellite Data Specifications After Interpolation

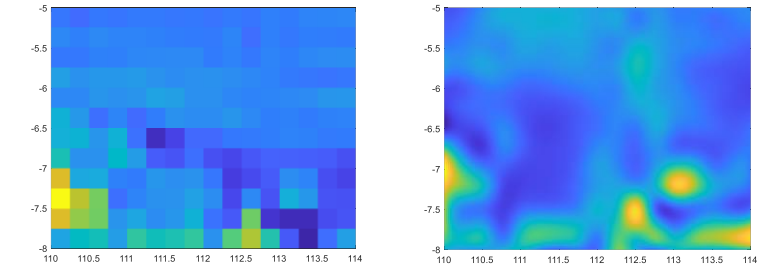
Variable Name	Information	ERA5 Size	ERA5 Size With Interpolation
Longitude	$110°BT - 114°BT$	17×1	161×1
Latitude	$5°LS - 8°LS$	13×1	121×1
Time	day	5113×1	5113×1
Total precipitation (tp)	mm	$17 \times 13 \times 5113$	$161 \times 12 \times 5113$

Table 3 provides information on rainfall data variables from ERA5 spatial. Before interpolation, the size of the ERA5 data matrix for longitude is $\mathbf{17 \times 1}$, and for latitude is $\mathbf{13 \times 1}$, reflecting the initial spatial resolution. After interpolation, the spatial resolution increases, with longitude becoming $\mathbf{161 \times 1}$. Moreover, the latitude becomes $\mathbf{121 \times 1}$; this means more regional observation points, enabling more accurate weather and rainfall analysis. For the time dimension, data is available for 5113 days. This interpolation is crucial for enhancing climate monitoring and prediction precision, especially when rainfall patterns need to be analyzed with higher spatial resolution.

3.1.3 Satellite ERA5 Dataset with Interpolation Spline

The preprocessed ERA5 data will be interpolated using the spline interpolation method. After interpolation, the resolution increases to $0.025° \times 0.025°$, changing the matrix size. The specifications of the ERA5 data after interpolation are presented in Table 3. With this higher resolution, the rainfall analysis is

expected to be more detailed and accurate, supporting a more effective missing data imputation process. A visualization of the ERA5 satellite rainfall data can be seen in Figure 4.



(a) **(b)**

Figure 4. Visualization Satellite Data
(a) ERA5 Without Interpolation, (b) ERA5 With Interpolation

Figure 4 illustrates the visualization of rainfall data from ERA5 satellite data. In Figure 4 (a), the spatial resolution appears lower, depicted by large blocky grids. This indicates that each data point covers a vast area, resulting in less visible spatial detail of the rainfall patterns. Rainfall variations are displayed in distinct blocks but lack precision in capturing subtle changes between regions. Conversely, Figure 4 (b) on the right demonstrates increased spatial resolution. The map's colors are smoother than Figure 4 (a), and transitions between regions appear more seamless. This improvement suggests that interpolation helps estimate rainfall values between the original observation points, producing a more detailed and realistic rainfall distribution map. Overall, interpolation enhances the precision of visualization and analysis. The interpolated map is more representative of climate studies, especially when high-resolution patterns of weather or rainfall are crucial. However, it is important to note that interpolation does not generate new data but estimates values between existing points.

After preparation, the data was divided into training and testing sets using five data split scenarios: 95:5%, 90:10%, 80:20%, 70:30%, and 50:50%. The training data consisted of station and satellite data, while the testing data used complete station data. In the testing set, a portion of the values was randomly removed and then predicted using the model. Mean Absolute Error (MAE) was calculated to evaluate the magnitude of the difference between the original data and the predicted values.

3.2 Parameters Initialization

This section discusses setting up hyperparameters before constructing the model to ensure optimal performance. One important aspect of this setup is the selection of an optimization algorithm that plays a role in updating the model's weights during the training process. Based on the study conducted by [27], three optimization algorithms identified as having the best performance were used to evaluate the model's performance: Adam, NAdam, and AdamW. Three learning rate values were also tested: 0.1, 0.01, and 0.001.

Before training the model across all 10 BMKG stations in East Java, an initial trial was conducted on a single station as a preliminary step. This trial aimed to determine the best combination of optimizer type and learning rate values, ensuring that the selected parameters deliver optimal performance before being applied to other datasets. The trial was conducted at the East Java Climatology Station, and the results are presented in Table 4.

Table 4. Trial for Determining the Best Optimizer and Learning Rate (LR)

LR/Optimizer	0.001			0.01			0.1		
	MAE train	MAE test	ε	MAE train	MAE test	ε	MAE train	MAE test	ε
Adam	7.661	7.555	127	7.641	7.594	40	7.717	7.306	24
NAdam	7.605	7.473	283	7.656	7.337	63	7.762	7.322	28
AdamW	7.967	7.673	70	7.929	7.631	30	7.944	7.245	13

Table 4 shows that the NAdam optimizer with a learning rate of 0.001 demonstrates the best performance, indicated by the low MAE values on both training and testing data, and optimal generalization capability. Therefore, this combination is selected for training the model on other stations, as it balances accuracy, learning stability, and generalization. The results of this testing will be applied to data from ten different stations.

3.3 Performance Loss Function and Metric Evaluation

This section discusses the performance loss function and metric evaluation for two types of ERA5 datasets: ERA5 without interpolation and ERA5 with interpolation spline. These two data types have different sizes, so the models were adjusted accordingly.

3.3.1 ERA5 Satellite Data

The CNN method has evolved with various architectural variations, such as LeNet, AlexNet, VGGNet, GoogLeNet, and ResNet. Among these architectures, LeNet and AlexNet have demonstrated high success rates in various applications. LeNet was designed and introduced by [35] as the first CNN architecture. In this study, the LeNet architecture was modified to adapt to the data size of the ERA5 satellite. The modified model was trained using observation station data, where missing values were predicted based on data from the ERA5 satellite. The design of the CNN architecture with the modified LeNet can be seen in Figure 5.

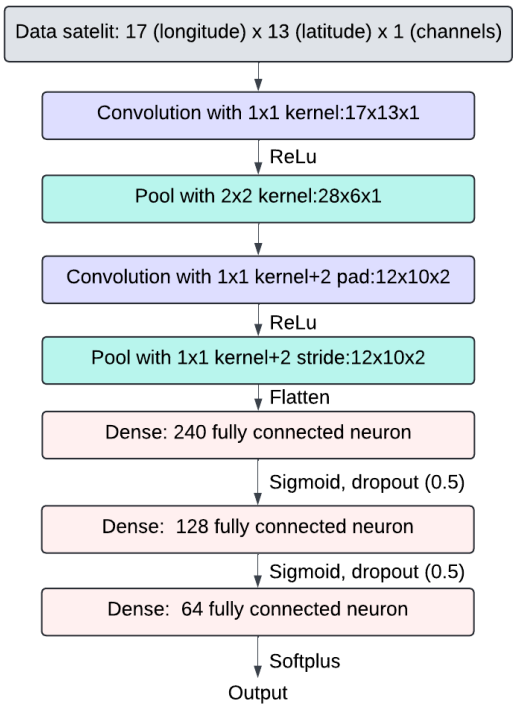


Figure 5. Modified LeNet Architecture

Following the construction of the model, it was trained on data from ten BMKG stations in East Java, conducted over 300 epoch (ε). Testing was performed using five scenarios for splitting training and testing data: 95:5%, 90:10%, 80:20%, 70:30%, and 50:50%. The procedure was repeated ten times to ensure consistent results and avoid computational bias. The average testing results for all stations are presented in Table 5.

Table 5. Training and Testing Model Using ERA5 Satellite Data

Station	95%	5%		90%	10%		80%	20%	
	MAE *train*	MAE *test*	ε	MAE *train*	MAE *test*	ε	MAE *train*	MAE *test*	ε
1	7.605	6.982	169	7.487	7.079	300	7.564	7.243	300
2	13.666	13.525	163	13.654	13.376	199	13.666	13.383	260
3	11.578	11.312	300	11.496	11.658	300	11.519	11.588	300
4	5.907	5.603	300	5.880	6.307	300	5.886	6.110	300
5	7.497	5.819	300	7.381	7.870	300	7.460	7.231	300
6	6.745	6.093	300	6.794	6.569	300	6.748	6.671	300
7	5.352	5.282	300	5.171	5.114	300	5.155	5.185	300
8	6.198	5.858	300	6.194	6.031	300	6.209	6.026	300
9	7.529	6.299	300	7.494	7.279	300	7.510	7.622	300
10	4.720	4.623	300	4.689	4.275	300	4.774	4.446	300
Mean	7.679	7.139		7.624	7.555		7.649	7.550	

Station	70%	30%		50%	50%	
	MAE *train*	MAE *test*	ε	MAE *train*	MAE *test*	ε
1	7.418	7.428	163	7.335	7.397	145
2	13.149	13.024	238	13.196	13.025	318
3	11.639	11.681	303	10.709	10.856	215
4	5.905	5.873	152	5.836	5.981	126
5	7.374	7.479	109	7.168	7.609	175
6	6.529	6.639	195	6.496	6.738	114
7	5.139	5.125	195	5.181	5.106	194
8	6.269	6.024	130	6.300	6.098	120
9	7.124	7.405	503	7.008	7.653	157
10	4.732	4.482	263	4.689	4.653	139
Mean	7.528	7.516		7.319	7.511	

Table 5 presents the MAE values for training and testing data and the number of iterations at BMKG stations in East Java, tested using ERA5 satellite data without interpolation. The gray color in the table indicates the risk of overfitting, which occurs when the model performs well on the training data but fails to maintain that performance on the testing data. In this context, overfitting is indicated by the significantly higher Mean Absolute Error (MAE) value on the testing data than the MAE on the training data. Stations two and three have relatively high MAE values compared to other stations. One of the main causes is the large amount of missing data at these stations, which hinders the model's ability to learn rainfall patterns optimally. In the 95:5% scenario, there was no indication of overfitting, demonstrating that the model could learn from the training data without losing its ability to generalize well on the testing data. This is evident from the MAE value on the testing data, which remained low and was not significantly different from the MAE value on the training data.

On the other hand, in scenarios with a larger proportion of training data, some stations experienced overfitting. Overfitting occurs when the model becomes too aligned with specific patterns in the training data, causing its performance to decline when tested on new data. In this study, overfitting was indicated by a lower MAE on the training data compared to the MAE on the testing data. The larger the proportion of training data, the higher the risk of overfitting, as the model tends to focus more on specific details in the training data and becomes less flexible in recognizing more general patterns. Therefore, maintaining a balance between training and testing data is crucial to ensure that the model achieves high accuracy on the training data and retains strong generalization capabilities when applied to new data. Overall, the 95:5% scenario provided the best results, with the lowest MAE value on the testing data, reflecting the model's ability to handle missing data effectively. This scenario minimized overfitting and resulted in better prediction generalization compared to other scenarios, which tended to show potential overfitting and accuracy reduction.

3.3.2 ERA5 Satellite Dataset Using Interpolation

Inspired by LeNet, **[36]** developed AlexNet, the first large-scale CNN model. In this study, the AlexNet architecture was modified to accommodate the size of ERA5 satellite data that has undergone interpolation. The modified model was trained using data from observation stations to predict missing values based on the interpolated ERA5 satellite data. The structure of the CNN architecture with the modified AlexNet is shown in **Figure 6**.

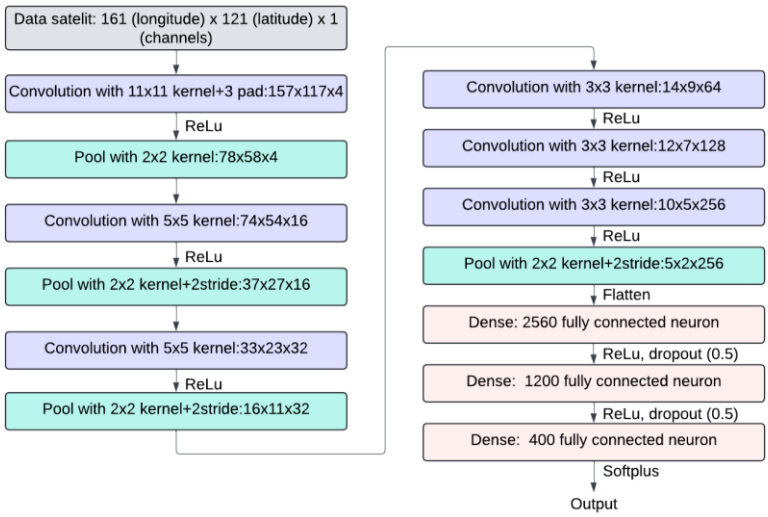


Figure 6. Modified AlexNet Architecture

The modified AlexNet architecture has more processing layers than the modified LeNet architecture. This architecture comprises 13 processing layers, including ten convolutional layers and three fully connected layers. Sub-sampling is performed using max pooling, and the activation function used is the rectified linear unit (ReLU). In deep learning, using limited data for parameter estimation can result in high variance and overfitting. The dropout technique is applied to the fully connected layers to address this issue and improve the network's generalization ability. Dropout helps prevent overfitting by randomly turning off neurons during training [37]. Softplus activation function then follows the layer to produce the final prediction.

After the model is successfully constructed, the next step is training and evaluating the model using data from 10 BMKG stations in East Java. The training is carried out until 30 epochs (ε) are reached. The number of epochs for the interpolated data is fewer because the computation time is significantly longer than that of the data without interpolation. This longer computation time is due to the much larger data size. The scenarios used are the same as when training the CNN model with ERA5 data. The results of the training and testing model are shown in Table 6.

Table 6. Training and Testing Model Using ERA5 Satellite Data with Interpolation

Station	MAE								
	95% *train*	5% *test*	ε	90% *train*	90% *test*	ε	80% *train*	20% *test*	ε
1	7.50	6.78	25	7.33	7.89	24	7.44	7.38	30
2	12.91	11.83	3	12.76	11.04	28	12.63	12.17	28
3	11.08	10.75	19	11.04	12.68	29	11.19	12.89	3
4	5.82	5.67	29	5.78	6.12	29	5.80	5.90	29
5	7.48	4.99	30	7.59	5.26	30	7.32	7.52	17
6	6.82	7.71	10	6.85	6.98	9	6.86	6.89	9
7	5.35	5.74	9	5.13	4.92	30	5.09	5.20	29
8	6.14	5.02	30	6.47	6.57	10	5.99	6.21	25
9	7.15	6.33	24	7.32	6.85	29	7.24	7.43	30
10	4.77	5.67	8	4.82	4.71	9	4.86	4.61	8
Mean	7.50	7.05		7.51	7.30		7.44	7.62	

Station	MAE					
	70% *train*	30% *test*	ε	50% *train*	50% *test*	ε
1	7.50	7.26	30	7.256	7.198	23
2	12.79	12.06	23	12.626	12.458	30
3	10.92	12.12	25	10.947	11.167	29
4	6.22	5.89	1	6.097	6.269	10
5	7.62	6.77	19	7.653	7.314	7
6	6.89	6.81	9	6.540	6.557	29
7	5.13	5.20	8	5.417	5.327	8
8	6.43	6.62	9	6.529	6.444	9
9	7.16	7.54	30	6.874	7.832	29
10	4.82	4.82	8	4.725	4.910	7
Mean	7.55	7.51		7.466	7.548	

Table 6 presents the MAE values for training and testing data and the number of iterations at BMKG stations in East Java, tested using ERA5 satellite data with interpolation. The gray color in the table indicates the risk of overfitting, which occurs when the model performs well on the training data but fails to maintain that performance on the testing data. In this context, overfitting is indicated by the significantly higher Mean Absolute Error (MAE) value on the testing data than the MAE on the training data. In the 95:5% scenario, the average MAE on the test data is the lowest compared to other scenarios. In the 90:10% scenario, this value increases to 7.30, with several stations experiencing overfitting, indicating the model's limited generalization ability. In the 80:20% scenario, the average MAE on the test data rises to 7.62, with most stations experiencing overfitting. In the 70:30% scenario, the MAE decreases to 7.51, but increases again in the 50:50% scenario to 7.55. Overall, the 95:5% scenario provides the best results, reflecting the model's superior ability to predict rainfall and handle missing data. This scenario also demonstrates better generalization than the other scenarios, which tend to suffer from overfitting and reduced prediction accuracy.

3.4 Imputation Missing Data

This section discusses imputation of missing data using Convolutional Neural Network (CNN) with ERA5 satellite data interpolation. The visualization of missing data imputation is presented as a graph. The green bars represent the imputed missing data results, while the red line represents the actual rainfall data values from BMKG stations in East Java.

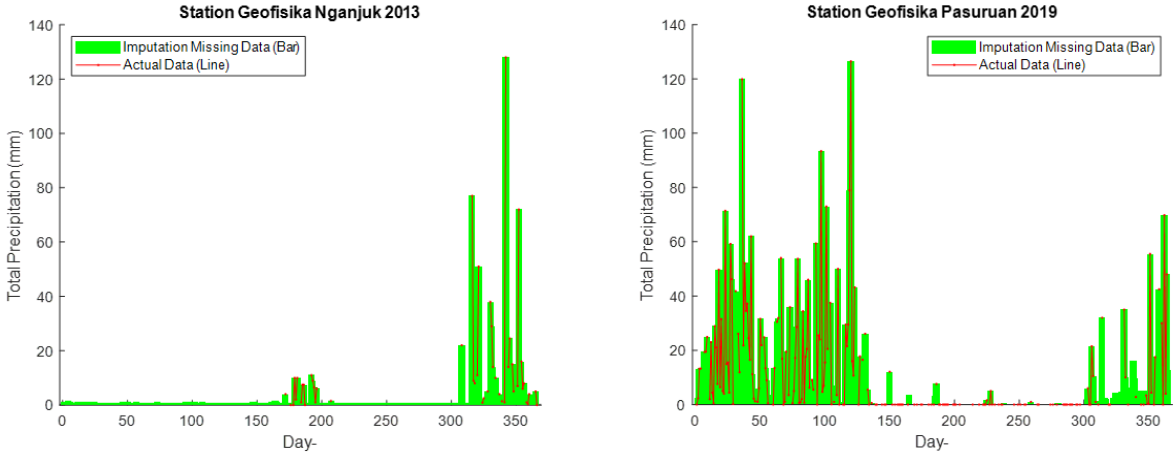


Figure 7. Compared Imputation Missing Data with Actual Data (a) Station Geofisika Nganjuk 2013, (b) Station Geofisika Pasuruan 2019

This section presents two stations as samples to demonstrate the model's ability to predict missing data, namely the Nganjuk Geophysics Station in 2013 and the Pasuruan Geophysics Station in 2019. These stations were selected due to the significant amount of missing data, indicated by dashed red lines in the rainfall data visualization. In both cases, the CNN-based model with interpolated ERA5 data successfully imputed the missing data. The green bars in the graph represent the model's imputation results, filling in the gaps caused by missing observations. This approach allows the rainfall patterns to be reconstructed accurately, closely approximating the actual available values. These findings highlight the effectiveness of combining the CNN method and interpolated ERA5 data in handling missing data, providing a potential solution for enhancing the completeness and quality of rainfall data at BMKG stations in East Java.

4. CONCLUSION

The conclusion of this study shows that imputing missing rainfall data using a spatial interpolation approach provides better performance than the approach without interpolation. This is evident from the lower Mean Absolute Error (MAE) values at most stations when the ERA5 data is first processed with spline interpolation before input into the CNN model. Based on the model evaluation results, the 95:5% data split scenario demonstrated the best performance in handling missing data. In this scenario, the average MAE on the test data was the lowest among all scenarios, indicating that the model could make predictions with

minimal error. Overall, the findings of this study provide evidence that the combination of spatial interpolation and CNN modeling can be an effective approach for addressing rainfall data completeness issues.

AUTHOR CONTRIBUTIONS

Lilis Sriwahyuni: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Software, Visualization, Writing - Original Draft. Sri Nurdiati: Formal Analysis, Methodology, Project Administration, Supervision, Validation, Writing - Review and Editing. Endar Hasafah Nugrahani: Formal Analysis, Methodology, Project Administration, Supervision, Validation, Writing - Review and Editing. Ihwan Sukmana: Data Curation, Formal Analysis, Software, Visualization, Writing - Original Draft. Mohamad Khoirun Najib: Conceptualization, Data Curation, Software, Validation, Writing - Review and Editing. All authors discussed the results and contributed to the final manuscript.

FUNDING STATEMENT

This research has been funded by the Kemendikbudristek program through the scheme "Penelitian Tesis Magister (PTM)" with contract number 027/E5/PG.02.00.PL/2024 and subcontract number 22306/IT3.D10/PT.01.03/P/B/2024.

ACKNOWLEDGMENT

We thank the Kemendikbudristek program, which has supported this research through the scheme "Penelitian Tesis Magister (PTM)." We also thank the Department of Mathematics, IPB University, for the support provided in this research.

CONFLICT OF INTEREST

The authors declare that no conflicts of interest to report study.

REFERENCES

[1] H. R. Bedane, K. T. Beketie, E. E. Fantahun, G. L. Feyisa, and F. A. Anose, "THE IMPACT OF RAINFALL VARIABILITY AND CROP PRODUCTION ON VERTISOLS IN THE CENTRAL HIGHLANDS OF ETHIOPIA," *Environmental Systems Research*, vol. 11, no. 1, p. 26, Dec. 2022, doi: https://doi.org/10.1186/s40068-022-00275-3 .

[2] B. Hamududu and A. Killingtveit, "ASSESSING CLIMATE CHANGE IMPACTS ON GLOBAL HYDROPOWER," *Energies (Basel)*, vol. 5, no. 2, pp. 305–322, Feb. 2012, doi: https://doi.org/10.3390/en5020305

[3] T. D. Fletcher, H. Andrieu, and P. Hamel, "UNDERSTANDING, MANAGEMENT AND MODELLING OF URBAN HYDROLOGY AND ITS CONSEQUENCES FOR RECEIVING WATERS: A STATE OF THE ART," *Adv Water Resour*, vol. 51, pp. 261–279, Jan. 2013, doi: https://doi.org/10.1016/j.advwatres.2012.09.001

[4] C. J. Walsh, A. H. Roy, J. W. Feminella, P. D. Cottingham, P. M. Groffman, and R. P. Morgan, "THE URBAN STREAM SYNDROME: CURRENT KNOWLEDGE AND THE SEARCH FOR A CURE," *J North Am Benthol Soc*, vol. 24, no. 3, pp. 706–723, Sep. 2005, doi: https://doi.org/10.1899/04-028.1

[5] T. Schnepper, J. Groh, H. H. Gerke, B. Reichert, and T. Pütz, "EVALUATION OF PRECIPITATION MEASUREMENT METHODS USING DATA FROM A PRECISION LYSIMETER NETWORK," *Hydrol Earth Syst Sci*, vol. 27, no. 17, pp. 3265–3292, Sep. 2023, doi: https://doi.org/10.5194/hess-27-3265-2023

[6] F. J. Tapiador *et al.*, "GLOBAL PRECIPITATION MEASUREMENTS FOR VALIDATING CLIMATE MODELS," *Atmos Res*, vol. 197, pp. 1–20, Nov. 2017, doi: https://doi.org/10.1016/j.atmosres.2017.06.021

[7] M. Bernard and C. Gregoretti, "THE USE OF RAIN GAUGE MEASUREMENTS AND RADAR DATA FOR THE MODEL-BASED PREDICTION OF RUNOFF-GENERATED DEBRIS-FLOW OCCURRENCE IN EARLY WARNING SYSTEMS," *Water Resour Res*, vol. 57, no. 3, Mar. 2021, doi: https://doi.org/10.1029/2020WR027893

[8] L. C. Sieck, S. J. Burges, and M. Steiner, "CHALLENGES IN OBTAINING RELIABLE MEASUREMENTS OF POINT RAINFALL," *Water Resour Res*, vol. 43, no. 1, Jan. 2007, doi: https://doi.org/10.1029/2005WR004519

[9] S. A. Rafhida, S. Nurdiati, R. Budiarti, and M. K. Najib, "BIAS CORRECTION OF LAKE TOBA RAINFALL DATA USING QUANTILE DELTA MAPPING," *CAUCHY: Jurnal Matematika Murni dan Aplikasi*, vol. 9, no. 2, pp. 297–309, Nov. 2024, doi: https://doi.org/10.18860/ca.v9i2.29124
[10] D. Desmonda, T. Tursina, and M. A. Irwansyah, "PREDIKSI BESARAN CURAH HUJAN MENGGUNAKAN METODE FUZZY TIME SERIES," *Jurnal Sistem dan Teknologi Informasi (JUSTIN)*, vol. 6, no. 4, p. 141, Oct. 2018, doi: https://doi.org/10.26418/justin.v6i4.27036
[11] E. Ardiyani, S. Nurdiati, A. Sopaheluwakan, P. Septiawan, and M. K. Najib, "PROBABILISTIC HOTSPOT PREDICTION MODEL BASED ON BAYESIAN INFERENCE USING PRECIPITATION, RELATIVE DRY SPELLS, ENSO AND IOD," *Atmosphere (Basel)*, vol. 14, no. 2, p. 286, Jan. 2023, doi: https://doi.org/10.3390/atmos14020286
[12] L. Li, "A ROBUST DEEP LEARNING APPROACH FOR SPATIOTEMPORAL ESTIMATION OF SATELLITE AOD AND PM2.5," *Remote Sens (Basel)*, vol. 12, no. 2, p. 264, Jan. 2020, doi: https://doi.org/10.3390/rs12020264
[13] X. Yang, R. Cui, C. Tian, S. Hu, J. Jiang, and P. Xu, "LINEAR SPLINE AND CNN-LSTM FOR MISSING VALUES IMPUTATION OF BEIDOU SATELLITE RADIATION DOSE DATA," *Chinese Journal of Space Science*, vol. 42, no. 1, p. 163, 2022, doi: https://doi.org/10.11728/cjss2022.01.201116100
[14] M. K. Najib and S. Nurdiati, "KOREKSI BIAS STATISTIK PADA DATA PREDIKSI SUHU PERMUKAAN AIR LAUT DI WILAYAH INDIAN OCEAN DIPOLE BARAT DAN TIMUR," *Jambura Geoscience Review*, vol. 3, no. 1, pp. 9–17, Jan. 2021, doi: https://doi.org/10.34312/jgeosrev.v3i1.8259
[15] A. Gelman and J. Hill, *DATA ANALYSIS USING REGRESSION AND MULTILEVEL/HIERARCHICAL MODELS*. Cambridge University Press, 2006. doi: https://doi.org/10.1017/CBO9780511790942
[16] A. Wangwongchai, M. Waqas, P. Dechpichai, P. T. Hlaing, S. Ahmad, and U. W. Humphries, "IMPUTATION OF MISSING DAILY RAINFALL DATA; A COMPARISON BETWEEN ARTIFICIAL INTELLIGENCE AND STATISTICAL TECHNIQUES," *MethodsX*, vol. 11, p. 102459, Dec. 2023, doi: https://doi.org/10.1016/j.mex.2023.102459
[17] J. M. Jerez *et al.*, "MISSING DATA IMPUTATION USING STATISTICAL AND MACHINE LEARNING METHODS IN A REAL BREAST CANCER PROBLEM," *Artif Intell Med*, vol. 50, no. 2, pp. 105–115, Oct. 2010, doi: https://doi.org/10.1016/j.artmed.2010.05.002
[18] Y. Lops, A. Pouyaei, Y. Choi, J. Jung, A. K. Salman, and A. Sayeed, "APPLICATION OF A PARTIAL CONVOLUTIONAL NEURAL NETWORK FOR ESTIMATING GEOSTATIONARY AEROSOL OPTICAL DEPTH DATA," *Geophys Res Lett*, vol. 48, no. 15, Aug. 2021, doi: https://doi.org/10.1029/2021GL093096
[19] Q. Zhang, Q. Yuan, C. Zeng, X. Li, and Y. Wei, "MISSING DATA RECONSTRUCTION IN REMOTE SENSING IMAGE WITH A UNIFIED SPATIAL–TEMPORAL–SPECTRAL DEEP CONVOLUTIONAL NEURAL NETWORK," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 8, pp. 4274–4288, Aug. 2018, doi: https://doi.org/10.1109/TGRS.2018.2810208
[20] L. Sriwahyuni, S. Nurdiati, E. H. Nugrahani, and M. K. Najib, "PERFORMANCE OF MACHINE LEARNING FOR IMPUTING MISSING DAILY RAINFALL DATA IN EAST JAVA UNDER MULTIPLE SATELLITE DATA MODELS," *Geographia Technica*, vol. 20, no. 1/2025, pp. 346–368, Mar. 2025, doi: https://doi.org/10.21163/GT_2025.201.23
[21] T. C. H. Lux, L. T. Watson, T. H. Chang, Y. Hong, and K. Cameron, "INTERPOLATION OF SPARSE HIGH-DIMENSIONAL DATA," *Numer Algorithms*, vol. 88, no. 1, pp. 281–313, Sep. 2021, doi: https://doi.org/10.1007/s11075-020-01040-2
[22] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng, "END-TO-END TEXT RECOGNITION WITH CONVOLUTIONAL NEURAL NETWORKS."
[23] Z. Chen and Y. Li, "FDSPC: FAST AND DIRECT SMOOTH PATH PLANNING VIA CONTINUOUS CURVATURE INTEGRATION," May 2024.
[24] S. Moghtadernejad, Y. Jin, and B. T. Adey, "ESTIMATING THE VALUES OF MISSING DATA RELATED TO INFRASTRUCTURE CONDITION STATES USING THEIR SPATIAL CORRELATION," *Journal of Infrastructure Systems*, vol. 29, no. 1, Mar. 2023, doi: https://doi.org/10.1061/(ASCE)IS.1943-555X.0000726
[25] M. N. Arefin and A. K. M. Masum, "A PROBABILISTIC APPROACH FOR MISSING DATA IMPUTATION," *Complexity*, vol. 2024, pp. 1–15, Jan. 2024, doi: https://doi.org/10.1155/2024/4737963
[26] S. Nurdiati, M. K. Najib, F. Bukhari, R. Revina, and F. N. Salsabila, "PERFORMANCE COMPARISON OF GRADIENT-BASED CONVOLUTIONAL NEURAL NETWORK OPTIMIZERS FOR FACIAL EXPRESSION RECOGNITION," *BAREKENG: Jurnal Ilmu Matematika dan Terapan*, vol. 16, no. 3, pp. 927–938, Sep. 2022, doi: https://doi.org/10.30598/barekengvol16iss3pp927-938
[27] D. M. Hawkins, "THE PROBLEM OF OVERFITTING," *J Chem Inf Comput Sci*, vol. 44, no. 1, pp. 1–12, Jan. 2004, doi: https://doi.org/10.1021/ci0342472
[28] K. Fukushima, "Neocognitron: A SELF-ORGANIZING NEURAL NETWORK MODEL FOR A MECHANISM OF PATTERN RECOGNITION UNAFFECTED BY SHIFT IN POSITION," *Biol Cybern*, vol. 36, no. 4, pp. 193–202, Apr. 1980, doi: https://doi.org/10.1007/BF0034425
[29] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A SURVEY OF CONVOLUTIONAL NEURAL NETWORKS: ANALYSIS, APPLICATIONS, AND PROSPECTS," *IEEE Trans Neural Netw Learn Syst*, vol. 33, no. 12, pp. 6999–7019, Dec. 2022, doi: https://doi.org/10.1109/TNNLS.2021.3084827
[30] V. Nair and G. E. Hinton, "RECTIFIED LINEAR UNITS IMPROVE RESTRICTED BOLTZMANN MACHINES."
[31] H. Junninen, H. Niska, K. Tuppurainen, J. Ruuskanen, and M. Kolehmainen, "METHODS FOR IMPUTATION OF MISSING VALUES IN AIR QUALITY DATA SETS," *Atmos Environ*, vol. 38, no. 18, pp. 2895–2907, Jun. 2004, doi: https://doi.org/10.1016/j.atmosenv.2004.02.026
[32] D. P. Kingma and J. Ba, "ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION," Dec. 2014.
[33] T. Dozat, "WORKSHOP TRACK-ICLR 2016 INCORPORATING NESTEROV MOMENTUM INTO ADAM."
[34] I. Loshchilov and F. Hutter, "SGDR: STOCHASTIC GRADIENT DESCENT WITH WARM RESTARTS," Aug. 2016.
[35] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "GRADIENT-BASED LEARNING APPLIED TO DOCUMENT RECOGNITION," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998, doi: https://doi.org/10.1109/5.726791

[36] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "IMAGENET CLASSIFICATION WITH DEEP CONVOLUTIONAL NEURAL NETWORKS," *Commun ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: https://doi.org/10.1145/3065386

[37] S. Park and N. Kwak, "ANALYSIS ON THE DROPOUT EFFECT IN CONVOLUTIONAL NEURAL NETWORKS," 2017, pp. 189–204. doi: https://doi.org/10.1007/978-3-319-54184-6_12